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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18325t-i-sl

TABLE 1-2: PIC16(L)F18325 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/ANA0/C1IN0+/DAC1OUT/ $\overline{SS2}$(1)/ ICDDAT/ICSPDAT	RA0	TTL/ST	CMOS	General purpose I/O.
	ANA0	AN	—	ADC Channel A0 input.
	C1IN0+	AN	—	Comparator C1 positive input.
	DAC1OUT	—	AN	Digital-to-Analog Converter output.
	$\overline{SS2}$	TTL/ST	—	Slave Select 2 input.
	ICDDAT	TTL/ST	CMOS	In-Circuit Debug Data I/O.
	ICSPDAT	TTL/ST	CMOS	ICSP™ Data I/O.
RA1/ANA1/VREF+/C1IN0-/ C2IN0-/DAC1REF+/ ICDCLK/ ICSPCLK	RA1	TTL/ST	CMOS	General purpose I/O.
	ANA1	AN	—	ADC Channel A1 input.
	VREF+	AN	—	ADC positive voltage reference input.
	C1IN0-	AN	—	Comparator C1 negative input.
	C2IN0-	AN	—	Comparator C2 negative input.
	DAC1REF+	—	AN	Digital-to-Analog Converter positive reference input.
	ICDCLK	TTL/ST	CMOS	In-Circuit Debug Clock I/O.
	ICSPCLK	TTL/ST	CMOS	ICSP™ Clock I/O.
RA2/ANA2/VREF-/ DAC1REF-/ T0CKI(1)/ CCP3(1)/CWG1IN(1)/ CWG2IN(1)/INT(1)	RA2	TTL/ST	CMOS	General purpose I/O.
	ANA2	AN	—	ADC Channel A2 input.
	VREF-	AN	—	ADC negative voltage reference input.
	DAC1REF-	—	AN	Digital-to-Analog Converter negative reference input.
	T0CKI	TTL/ST	—	TMR0 Clock input.
	CCP3	TTL/ST	CMOS	Capture/Compare/PWM 3 input.
	CWG1IN	TTL/ST	—	Complementary Waveform Generator 1 input.
	CWG2IN	TTL/ST	—	Complementary Waveform Generator 2 input.
	INT	TTL/ST	—	External interrupt input.
RA3/$\overline{MCLR}$/VPP	RA3	TTL/ST	CMOS	General purpose I/O.
	$\overline{MCLR}$	TTL/ST	—	Master Clear with internal pull-up.
	VPP	HV	—	Programming voltage.
RA4/ANA4/T1G(1)/ SOSCO/ CLKOUT/OSC2	RA4	TTL/ST	CMOS	General purpose I/O.
	ANA4	AN	—	ADC Channel A4 input.
	T1G	ST	—	TMR1 gate input.
	SOSCO	—	XTAL	Secondary Oscillator connection.
	CLKOUT	—	CMOS	FOSC/4 output.
	OSC2	—	XTAL	Crystal/Resonator (LP, XT, HS modes).

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^2C = Schmitt Trigger input with I^2C
HV = High Voltage XTAL = Crystal levels

Note 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection registers. See Register 13-1.

2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers. See Register 13-2.

3: These I^2C functions are bidirectional. The output pin selections must be the same as the input pin selections.

TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18325	PIC16(L)F18345	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
Bank 30													
CPU CORE REGISTERS; see Table 4-2 for specifics													
F20h	CLC2GLS0			LC2G1D4T	LC2G1D4N	LC2G1D3T	LC2G1D3N	LC2G1D2T	LC2G1D2N	LC2G1D1T	LC2G1D1N	xxxx xxxx	uuuu uuuu
F21h	CLC2GLS1			LC2G2D4T	LC2G2D4N	LC2G2D3T	LC2G2D3N	LC2G2D2T	LC2G2D2N	LC2G2D1T	LC2G2D1N	xxxx xxxx	uuuu uuuu
F22h	CLC2GLS2			LC2G3D4T	LC2G3D4N	LC2G3D3T	LC2G3D3N	LC2G3D2T	LC2G3D2N	LC2G3D1T	LC2G3D1N	xxxx xxxx	uuuu uuuu
F23h	CLC2GLS3			LC2G4D4T	LC2G4D4N	LC2G4D3T	LC2G4D3N	LC2G4D2T	LC2G4D2N	LC2G4D1T	LC2G4D1N	xxxx xxxx	uuuu uuuu
F24h	CLC3CON			LC3EN	—	LC3OUT	LC3INTP	LC3INTN	LC3MODE<2:0>			0-00 0000	0-00 0000
F25h	CLC3POL			LC3POL	—	—	—	LC3G4POL	LC3G3POL	LC3G2POL	LC3G1POL	0--- xxxx	0--- uuuu
F26h	CLC3SEL0			—	—	LC3D1S<5:0>						--xx xxxx	--uu uuuu
F27h	CLC3SEL1			—	—	LC3D2S<5:0>						--xx xxxx	--uu uuuu
F28h	CLC3SEL2			—	—	LC3D3S<5:0>						--xx xxxx	--uu uuuu
F29h	CLC3SEL3			—	—	LC3D4S<5:0>						--xx xxxx	--uu uuuu
F2Ah	CLC3GLS0			LC3G1D4T	LC3G1D4N	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	xxxx xxxx	uuuu uuuu
F2Bh	CLC3GLS1			LC3G2D4T	LC3G2D4N	LC3G2D3T	LC3G2D3N	LC3G2D2T	LC3G2D2N	LC3G2D1T	LC3G2D1N	xxxx xxxx	uuuu uuuu
F2Ch	CLC3GLS2			LC3G3D4T	LC3G3D4N	LC3G3D3T	LC3G3D3N	LC3G3D2T	LC3G3D2N	LC3G3D1T	LC3G3D1N	xxxx xxxx	uuuu uuuu
F2Dh	CLC3GLS3			LC3G4D4T	LC3G4D4N	LC3G4D3T	LC3G4D3N	LC3G4D2T	LC3G4D2N	LC3G4D1T	LC3G4D1N	xxxx xxxx	uuuu uuuu
F2Eh	CLC4CON			LC4EN	—	LC4OUT	LC4INTP	LC4INTN	LC4MODE<2:0>			0-00 0000	0-00 0000
F2Fh	CLC4POL			LC4POL	—	—	—	LC4G4POL	LC4G3POL	LC4G2POL	LC4G1POL	0--- xxxx	0--- uuuu
F30h	CLC4SEL0			—	—	LC4D1S<5:0>						--xx xxxx	--uu uuuu
F31h	CLC4SEL1			—	—	LC4D2S<5:0>						--xx xxxx	--uu uuuu
F32h	CLC4SEL2			—	—	LC4D3S<5:0>						--xx xxxx	--uu uuuu
F33h	CLC4SEL3			—	—	LC4D4S<5:0>						--xx xxxx	--uu uuuu
F34h	CLC4GLS0			LC4G1D4T	LC4G1D4N	LC4G1D3T	LC4G1D3N	LC4G1D2T	LC4G1D2N	LC4G1D1T	LC4G1D1N	xxxx xxxx	uuuu uuuu
F35h	CLC4GLS1			LC4G2D4T	LC4G2D4N	LC4G2D3T	LC4G2D3N	LC4G2D2T	LC4G2D2N	LC4G2D1T	LC4G2D1N	xxxx xxxx	uuuu uuuu
F36h	CLC4GLS2			LC4G3D4T	LC4G3D4N	LC4G3D3T	LC4G3D3N	LC4G3D2T	LC4G3D2N	LC4G3D1T	LC4G3D1N	xxxx xxxx	uuuu uuuu
F37h	CLC4GLS3			LC4G4D4T	LC4G4D4N	LC4G4D3T	LC4G4D3N	LC4G4D2T	LC4G4D2N	LC4G4D1T	LC4G4D1N	xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, q =depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Only on PIC16F18325/18345.

2: Register accessible from both User and ICD Debugger.

REGISTER 6-1: BORCON: BROWN-OUT RESET CONTROL REGISTER

R/W-1/u	R/W-0-0	U-0	U-0	U-0	U-0	U-0	R-q/u
SBOREN(1)	Reserved	—	—	—	—	—	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(1)
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 Reserved. Bit must be maintained as '0'.

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.

6.12 Power Control (PCON0) Register

The Power Control (PCON0) register contains flag bits to differentiate between a:

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

The PCON0 register bits are shown in Register 6-2.

Hardware will change the corresponding register bit during the Reset process; if the Reset was not caused by the condition, the bit remains unchanged (Table 6-4).

Software should reset the bit to the inactive state after the restart (hardware will not reset the bit).

Software may also set any PCON0 bit to the active state, so that user code may be tested, but no Reset action will be generated.

TABLE 7-3: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

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>				90
OSCCON2	—	COSC<2:0>			CDIV<3:0>				90
OSCCON3	CWSHOLD	SOSCPWR	SOSCBE	ORDY	NOSCR	—	—	—	91
OSCSTAT1	EXTOR	HFOR	—	LFOR	SOR	ADOR	—	PLLR	92
OSCEN	EXTOEN	HFOEN	—	LFOEN	SOSCEN	ADOEN	—	—	93
OSCFRQ	—	—	—	—	HFFRQ<3:0>				94
OSCTUNE	—	—	HFTUN<5:0>						95

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

TABLE 7-4: 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
CONFIG1	13:8	—	—	FCMEN	—	CSWEN	—	—	$\overline{\text{CLKOUTEN}}$	64
	7:0	—	RSTOSC2	RSTOSC1	RSTOSC0	—	FEXTOSC2	FEXTOSC1	FEXTOSC0	

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

9.2 IDLE Mode

When the IDLE Enable (IDLEN) bit is clear (IDLEN = 0), the SLEEP instruction will put the device into full Sleep mode (see **Section 9.3 "Sleep Mode"**). When IDLEN is set (IDLEN = 1), the SLEEP instruction will put the device into IDLE mode. In IDLE mode, the CPU and memory operations are halted, but the peripheral clocks continue to run. This mode is similar to DOZE mode, except that in IDLE both the CPU and program memory are shut off.

Note:	Peripherals using FOSC will continue running while in IDLE (but not in Sleep). Peripherals using HFINTOSC, LFINTOSC, or SOSC will continue operation in both IDLE and SLEEP.

Note:	If CLKOUT is enabled (CLKOUT = 0, Configuration Word 1), the output will continue operating while in IDLE.

9.2.1 IDLE AND INTERRUPTS

IDLE mode ends when an interrupt occurs (even if GIE = 0), but IDLEN is not changed. The device can re-enter IDLE by executing the SLEEP instruction.

If Recover-on-Interrupt is enabled (ROI = 1), the interrupt that brings the device out of IDLE also restores full-speed CPU execution when DOZE is also enabled.

9.2.2 IDLE AND WDT

When in IDLE, the WDT Reset is blocked and will instead wake the device. The WDT wake-up is not an interrupt, therefore ROI does not apply.

Note:	The WDT can bring the device out of IDLE, in the same way it brings the device out of Sleep. The DOZEN bit is not affected.

9.3 Sleep Mode

Sleep mode is entered by executing the SLEEP instruction, while the Idle Enable (IDLEN) bit of the CPUDOZE register is clear (IDLEN = 0). If the SLEEP instruction is executed while the IDLEN bit is set (IDLEN = 1), the CPU will enter the Idle mode (**Section 9.3.3 "Low-Power Sleep Mode"**).

Upon entering Sleep mode, the following conditions exist:

1. Resets other than WDT are not affected by Sleep mode; WDT will be cleared but keeps running if enabled for operation during Sleep.
2. The $\overline{PD}$ bit of the STATUS register is cleared.
3. The $\overline{TO}$ bit of the STATUS register is set.
4. The CPU and System clocks are disabled.
5. 31 kHz LFINTOSC, HFINTOSC and SOSC will remain enabled if any peripheral has requested them as a clock source or if the HFOEN, LFOEN, or SOSCEN bits of the OSCEN register are set.
6. ADC is unaffected if the dedicated ADCRC oscillator is selected. When the ADC clock source is something other than ADCRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.
7. I/O ports maintain the status they had before Sleep was executed (driving high, low, or high-impedance) only if no peripheral connected to the I/O port is active.

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using any oscillator

I/O pins that are high-impedance inputs should be pulled to VDD or VSS externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules. See **Section 24.0 "5-bit Digital-to-Analog Converter (DAC1) Module"** and **Section 16.0 "Fixed Voltage Reference (FVR)"** for more information on these modules.

REGISTER 12-20: ANSELC: PORTC ANALOG SELECT REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
ANSC7(1)	ANSC6(1)	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0
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 **ANSC<7:6>**: Analog Select between Analog or Digital Function on pins RC<7:6>, respectively(1)
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(2). Digital input buffer disabled.

bit 5-0 **ANSC<5:0>**: Analog Select between Analog or Digital Function on pins RC<5:0>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(2). Digital input buffer disabled.

Note 1: PIC16(L)F18345 only; otherwise read as '0'.
2: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

REGISTER 12-21: WPUC: WEAK PULL-UP PORTC REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
WPUC7(1)	WPUC6(1)	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
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 **WPUC<7:6>**(1): Weak Pull-up Register bits(2)
1 = Pull-up enabled
0 = Pull-up disabled

bit 5-0 **WPUC<5:0>**: Weak Pull-up Register bits(2)
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: PIC16(L)F18345 only; otherwise read as '0'.
2: The weak pull-up is disabled if the pin is configured as an output except when the pin is also configured as open-drain. When configured as open-drain, the pull-up is enabled when the output value is high, and disabled when the output value is low.

18.10 CWG Auto-shutdown Source

The output of the comparator module can be used as an auto-shutdown source for the CWG module. When the output of the comparator is active and the corresponding ASxE is enabled, the CWG operation will be suspended immediately (**Section 20.7.1.2 "External Input Source Shutdown"**).

18.11 Operation in Sleep Mode

The comparator module can operate during Sleep. The comparator clock source is based on the Timer1 clock source. If the Timer1 clock source is either the system clock (FOSC) or the instruction clock (FOSC/4), Timer1 will not operate during Sleep, and synchronized comparator outputs will not operate.

A comparator interrupt will wake the device from Sleep. The CxIE bits of the PIE2 register must be set to enable comparator interrupts.

REGISTER 20-2: CWGxCON1: CWGx CONTROL REGISTER 1

U-0	U-0	R-x	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IN	—	POLD	POLC	POLB	POLA
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 7-6 **Unimplemented:** Read as ‘0’

bit 5 **IN:** CWGx Data Input Signal (read-only)

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **POLD:** WGxD Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 2 **POLC:** WGxC Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 1 **POLB:** WGxB Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

bit 0 **POLA:** WGxA Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

REGISTER 20-3: CWGxCLKCON: CWGx CLOCK INPUT SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	CS
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 7-1 **Unimplemented:** Read as ‘0’

bit 0 **CS:** CWG Clock Source Selection Select bits

CS	Clock Source
0	FOSC
1	HFINTOSC (remains operating during Sleep)

REGISTER 21-9: CLCxGLS2: GATE 2 LOGIC SELECT REGISTER

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
LCxG3D4T	LCxG3D4N	LCxG3D3T	LCxG3D3N	LCxG3D2T	LCxG3D2N	LCxG3D1T	LCxG3D1N
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 **LCxG3D4T:** Gate 2 Data 4 True (non-inverted) bit
- 1 = CLCIN3 (true) is gated into CLCx Gate 2
- 0 = CLCIN3 (true) is not gated into CLCx Gate 2

bit 6 **LCxG3D4N:** Gate 2 Data 4 Negated (inverted) bit
- 1 = CLCIN3 (inverted) is gated into CLCx Gate 2
- 0 = CLCIN3 (inverted) is not gated into CLCx Gate 2

bit 5 **LCxG3D3T:** Gate 2 Data 3 True (non-inverted) bit
- 1 = CLCIN2 (true) is gated into CLCx Gate 2
- 0 = CLCIN2 (true) is not gated into CLCx Gate 2

bit 4 **LCxG3D3N:** Gate 2 Data 3 Negated (inverted) bit
- 1 = CLCIN2 (inverted) is gated into CLCx Gate 2
- 0 = CLCIN2 (inverted) is not gated into CLCx Gate 2

bit 3 **LCxG3D2T:** Gate 2 Data 2 True (non-inverted) bit
- 1 = CLCIN1 (true) is gated into CLCx Gate 2
- 0 = CLCIN1 (true) is not gated into CLCx Gate 2

bit 2 **LCxG3D2N:** Gate 2 Data 2 Negated (inverted) bit
- 1 = CLCIN1 (inverted) is gated into CLCx Gate 2
- 0 = CLCIN1 (inverted) is not gated into CLCx Gate 2

bit 1 **LCxG3D1T:** Gate 2 Data 1 True (non-inverted) bit
- 1 = CLCIN0 (true) is gated into CLCx Gate 2
- 0 = CLCIN0 (true) is not gated into CLCx Gate 2

bit 0 **LCxG3D1N:** Gate 2 Data 1 Negated (inverted) bit
- 1 = CLCIN0 (inverted) is gated into CLCx Gate 2
- 0 = CLCIN0 (inverted) is not gated into CLCx Gate 2

REGISTER 22-2: ADCON1: ADC CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
ADFM	ADCS<2:0>			—	ADNREF	ADPREF<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 **ADFM:** ADC Result Format Select bit
1 = Right justified. Six Most Significant bits of ADRESH are set to '0' when the conversion result is loaded.
0 = Left justified. Six Least Significant bits of ADRESL are set to '0' when the conversion result is loaded.

bit 6-4 **ADCS<2:0>:** ADC Conversion Clock Select bits
111 = ADCRC (dedicated RC oscillator)
110 = FOSC/64
101 = FOSC/16
100 = FOSC/4
011 = ADCRC (dedicated RC oscillator)
010 = FOSC/32
001 = FOSC/8
000 = FOSC/2

bit 3 **Unimplemented:** Read as '0'

bit 2 **ADNREF:** A/D Negative Voltage Reference Configuration bit
When ADON = 0, all multiplexer inputs are disconnected.
0 = VREF- is connected to VSS
1 = VREF- is connected to external VREF-

bit 1-0 **ADPREF<1:0>:** ADC Positive Voltage Reference Configuration bits
11 = VREF+ is connected to internal Fixed Voltage Reference (FVR) module(1)
10 = VREF+ is connected to external VREF+ pin(1)
01 = Reserved
00 = VREF+ is connected to VDD

Note 1: When selecting the VREF+ pin as the source of the positive reference, be aware that a minimum voltage specification exists. See Table 35-13 for details.

REGISTER 22-6: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

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<9:8>	
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 Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-2 **Reserved**: Do not use.

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
Upper two bits of 10-bit conversion result

REGISTER 22-7: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

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

FIGURE 25-5: CARRIER LOW SYNCHRONIZATION (MDSHSYNC = 0, MDCLSYNC = 1)

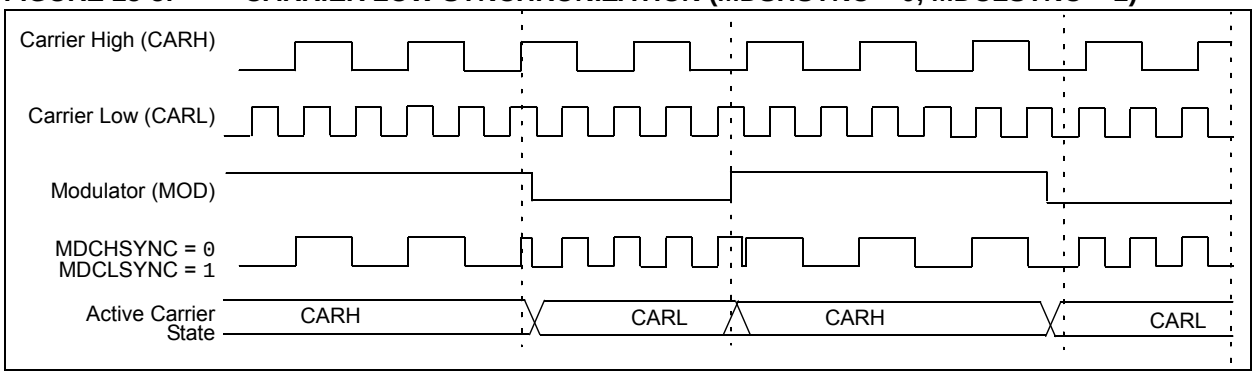
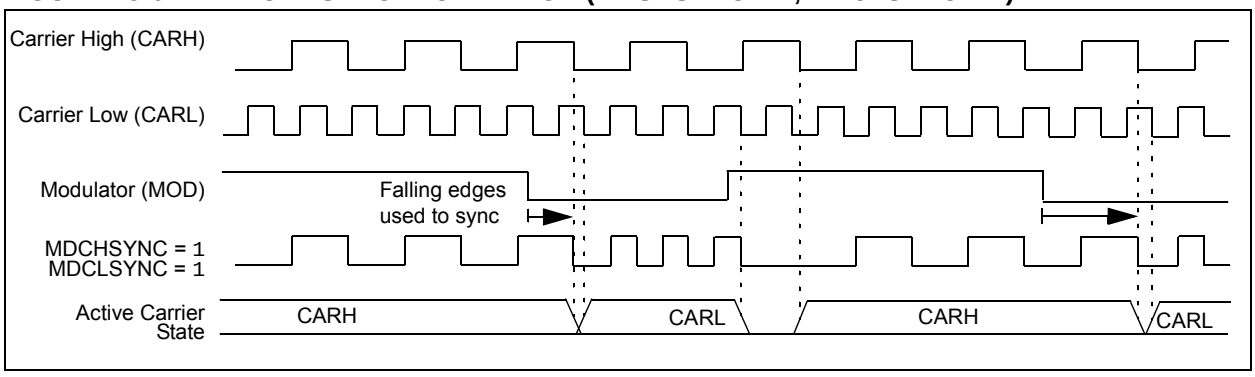


FIGURE 25-6: FULL SYNCHRONIZATION (MDSHSYNC = 1, MDCLSYNC = 1)



REGISTER 25-2: MDSRC: MODULATION SOURCE CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	—	—	MDMS<3:0>			
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 7-4 **Unimplemented:** Read as '0'

bit 3-0 **MDMS<3:0>** Modulation Source Selection bits

1111 = CLC4 output
1110 = CLC3 output
1101 = CLC2 output
1100 = CLC1 output
1011 = NCO1 output
1010 = EUSART1 TX output
1001 = MSSP2 SDO2 output
1000 = MSSP1 SDO1 output
0111 = C2 (Comparator 2) output
0110 = C1 (Comparator 1) output
0101 = PWM6 output
0100 = PWM5 output
0011 = CCP2 output (PWM Output mode only)
0010 = CCP1 output (PWM Output mode only)
0001 = MDMINPPS
0000 = MDBIT bit of MDCON register is modulation source

27.0 TIMER1/3/5 MODULE WITH GATE CONTROL

Timer1/3/5 modules are 16-bit timers/counters, each with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 2-bit prescaler
- Clock source for optional comparator synchronization
- Multiple Timer1 gate (count enable) sources
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function with the CCP modules
- Auto-Conversion Trigger (with CCP)
- Selectable Gate Source Polarity
- Gate Toggle mode
- Gate Single-Pulse mode
- Gate Value Status
- Gate Event Interrupt

Figure 27-1 is a block diagram of the Timer1 module.

Note 1:	In devices with more than one Timer module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the T1CON and T3CON control the same operational aspects of two completely different Timer modules.
2:	Throughout this section, generic references to Timer1 module in any of its operating modes may be interpreted as being equally applicable to Timerx module. Register names, module signals, I/O pins and bit names may use the generic designator ‘x’ to indicate the use of a numeral to distinguish a particular module, when required.

On the last byte of data communicated, the master device may end the transmission by sending a Stop bit. If the master device is in Receive mode, it sends a NACK in place of the last ACK bit. A Stop bit is indicated by a low-to-high transition of the SDA line while the SCL line is held high.

In some cases, the master may want to maintain control of the bus and re-initiate another transmission. If so, the master device may send a Restart condition in place of the Stop bit or last ACK bit when it is in Receive mode.

The I^2C bus specifies three message protocols;

- Single message where a master writes data to a slave.
- Single message where a master reads data from a slave.
- Combined message where a master initiates a minimum of two writes, or two reads, or a combination of writes and reads, to one or more slaves.

30.3.1 CLOCK STRETCHING

When a slave device has not completed processing data, it can delay the transfer of more data through the process of clock stretching. An addressed slave device may hold the SCL clock line low after receiving or sending a bit, indicating that it is not yet ready to continue. The master that is communicating with the slave will attempt to raise the SCL line in order to transfer the next bit, but will detect that the clock line has not yet been released. Because the SCL connection is open-drain, the slave has the ability to hold that line low until it is ready to continue communicating.

Clock stretching allows receivers that cannot keep up with a transmitter to control the flow of incoming data.

30.3.2 ARBITRATION

Each master device must monitor the bus for Start and Stop bits. If the device detects that the bus is busy, it cannot begin a new message until the bus returns to an Idle state.

However, two master devices may try to initiate a transmission on or about the same time. When this occurs, the process of arbitration begins. Each transmitter checks the level of the SDA data line and compares it to the level that it expects to find. The first transmitter to observe that the two levels do not match, loses arbitration, and must stop transmitting on the SDA line.

For example, if one transmitter holds the SDA line to a logical one (lets it float) and a second transmitter holds it to a logical zero (pulls it low), the result is that the SDA line will be low. The first transmitter then observes that the level of the line is different than expected and concludes that another transmitter is communicating.

The first transmitter to notice this difference is the one that loses arbitration and must stop driving the SDA line. If this transmitter is also a master device, it also must stop driving the SCL line. It then can monitor the lines for a Stop condition before trying to reissue its transmission. In the meantime, the other device that has not noticed any difference between the expected and actual levels on the SDA line continues with its original transmission.

Slave Transmit mode can also be arbitrated, when a master addresses multiple slaves, but this is less common.

30.5.3.3 7-bit Transmission with Address Hold Enabled

Setting the AHEN bit of the SSPxCON3 register enables additional clock stretching and interrupt generation after the eighth falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPxIF interrupt is set.

Figure 30-19 displays a standard waveform of a 7-bit address slave transmission with AHEN enabled.

1. Bus starts Idle.
2. Master sends Start condition; the S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Master sends matching address with R/$\overline{W}$ bit set. After the eighth falling edge of the SCL line the CKP bit is cleared and SSPxIF interrupt is generated.
4. Slave software clears SSPxIF.
5. Slave software reads ACKTIM bit of SSPxCON3 register, and R/$\overline{W}$ and D/$\overline{A}$ of the SSPxSTAT register to determine the source of the interrupt.
6. Slave reads the address value from the SSPxBUF register clearing the BF bit.
7. Slave software decides from this information if it wishes to $\overline{ACK}$ or not $\overline{ACK}$ and sets the ACKDT bit of the SSPxCON2 register accordingly.
8. Slave sets the CKP bit releasing SCL.
9. Master clocks in the $\overline{ACK}$ value from the slave.
10. Slave hardware automatically clears the CKP bit and sets SSPxIF after the $\overline{ACK}$ if the R/$\overline{W}$ bit is set.
11. Slave software clears SSPxIF.
12. Slave loads value to transmit to the master into SSPxBUF setting the BF bit.

Note: SSPxBUF cannot be loaded until after the $\overline{ACK}$.

13. Slave sets the CKP bit releasing the clock.
14. Master clocks out the data from the slave and sends an $\overline{ACK}$ value on the ninth SCL pulse.
15. Slave hardware copies the $\overline{ACK}$ value into the ACKSTAT bit of the SSPxCON2 register.
16. Steps 10-15 are repeated for each byte transmitted to the master from the slave.
17. If the master sends a not $\overline{ACK}$ the slave releases the bus allowing the master to send a Stop and end the communication.

Note: Master must send a not $\overline{ACK}$ on the last byte to ensure that the slave releases the SCL line to receive a Stop.

31.6 Register Definitions: EUSART1 Control

REGISTER 31-1: TX1STA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN(1)	SYNC	SENDB	BRGH	TRMT	TX9D
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 **CSRC:** Clock Source Select bit
Asynchronous mode:
Unused in this mode – value ignored
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit enabled
0 = Transmit disabled

bit 4 **SYNC:** EUSART1 Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **SENDB:** Send Break Character bit
Asynchronous mode:
1 = Send SYNCH BREAK on next transmission – Start bit, followed by 12 '0' bits, followed by Stop bit; cleared by hardware upon completion
0 = SYNCH BREAK transmission disabled or completed
Synchronous mode:
Unused in this mode – value ignored

bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode – value ignored

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 31-2: RC1STA: RECEIVE STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/x
SPEN(1)	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
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 **SPEN:** Serial Port Enable bit(1)
- 1 = Serial port enabled
- 0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-bit Receive Enable bit
- 1 = Selects 9-bit reception
- 0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit

Asynchronous mode:

Unused in this mode – value ignored

Synchronous mode – Master:
- 1 = Enables single receive
- 0 = Disables single receive

This bit is cleared after reception is complete.

Synchronous mode – Slave

Unused in this mode – value ignored

bit 4 **CREN:** Continuous Receive Enable bit

Asynchronous mode:
- 1 = Enables continuous receive until enable bit CREN is cleared
- 0 = Disables continuous receive

Synchronous mode:
- 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
- 0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit

Asynchronous mode 9-bit (RX9 = 1):
- 1 = Enables address detection – enable interrupt and load of the receive buffer when the ninth bit in the receive buffer is set
- 0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit

Asynchronous mode 8-bit (RX9 = 0):

Unused in this mode – value ignored

bit 2 **FERR:** Framing Error bit
- 1 = Framing error (can be updated by reading RC1REG register and receive next valid byte)
- 0 = No framing error

bit 1 **OERR:** Overrun Error bit
- 1 = Overrun error (can be cleared by clearing bit CREN)
- 0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data

This can be address/data bit or a parity bit and must be calculated by user firmware.

Note 1: The EUSART1 module automatically changes the pin from tri-state to drive as needed. Configure the associated TRIS bits for TX/CK and RX/DT to 1.

37.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

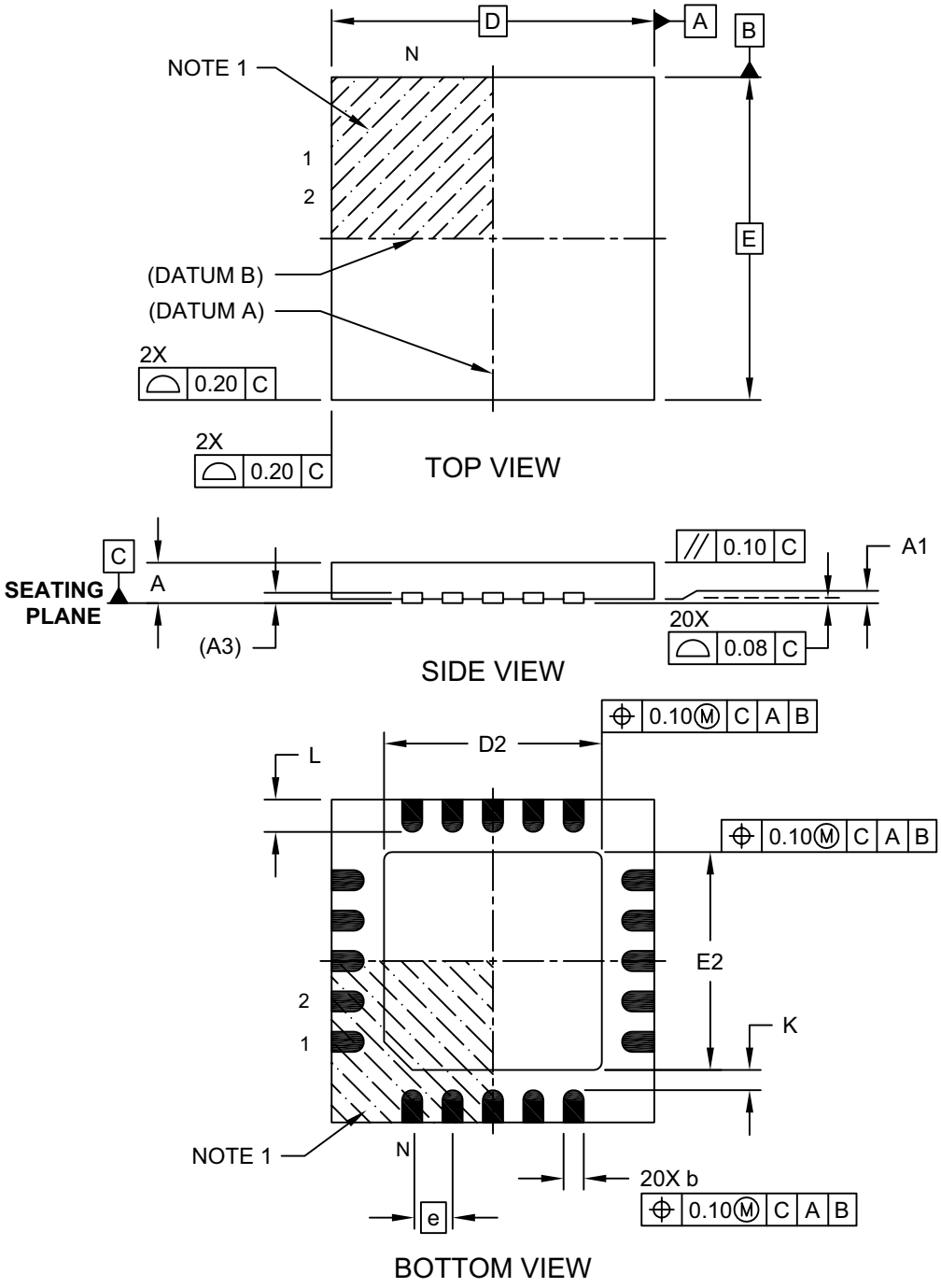
37.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

20-Lead Ultra Thin Plastic Quad Flat, No Lead Package (GZ) - 4x4x0.5 mm Body [UQFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-255A Sheet 1 of 2

