



#### 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	20MHz
Connectivity	
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	18
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f631-e-so

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
RB6/SCK/SCL	RB6	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on- change. Individually enabled pull-up.
	SCK	ST	CMOS	SPI clock.
	SCL	ST	OD	I <sup>2</sup> C™ clock.
RB7/TX/CK	RB7	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on- change. Individually enabled pull-up.
	ТΧ		CMOS	EUSART asynchronous output.
	CK	ST	CMOS	EUSART synchronous clock.
RC0/AN4/C2IN+	RC0	ST	CMOS	General purpose I/O.
	AN4	AN	_	A/D Channel 4 input.
	C2IN+	AN		Comparator C2 positive input.
RC1/AN5/C12IN1-	RC1	ST	CMOS	General purpose I/O.
	AN5	AN	_	A/D Channel 5 input.
	C12IN1-	AN		Comparator C1 or C2 negative input.
RC2/AN6/C12IN2-	RC2	ST	CMOS	General purpose I/O.
	AN6	AN	—	A/D Channel 6 input.
C12I		AN	_	Comparator C1 or C2 negative input.
RC3/AN7/C12IN3-	RC3	ST	CMOS	General purpose I/O.
	AN7	AN	_	A/D Channel 7 input.
C12		AN	_	Comparator C1 or C2 negative input.
RC4/C2OUT	RC4	ST	CMOS	General purpose I/O.
	C2OUT		CMOS	Comparator C2 output.
RC5	RC5	ST	CMOS	General purpose I/O.
RC6/AN8/SS	RC6	ST	CMOS	General purpose I/O.
	AN8	AN	—	A/D Channel 8 input.
	SS	ST	—	Slave Select input.
RC7/AN9/SDO	RC7	ST	CMOS	General purpose I/O.
	AN9	AN	_	A/D Channel 9 input.
	SDO		CMOS	SPI data output.
Vss	Vss	Power	_	Ground reference.
VDD	Vdd	Power	—	Positive supply.
Legend: AN = Analog input of TTL = TTL compatib HV = High Voltage	or output le input	CMOS= ST= S XTAL= (	CMOS co Schmitt T Crystal	ompatible input or outputOD= Open Drain rigger input with CMOS levels

#### **TABLE 1-4:** PINOUT DESCRIPTION - PIC16F687/PIC16F689 (CONTINUED)

# 2.2 Data Memory Organization

The data memory (see Figures 2-6 through 2-8) is partitioned into four banks which contain the General Purpose Registers (GPR) and the Special Function Registers (SFR). The Special Function Registers are located in the first 32 locations of each bank. The General Purpose Registers, implemented as static RAM, are located in the last 96 locations of each Bank. Register locations F0h-FFh in Bank 1, 170h-17Fh in Bank 2 and 1F0h-1FFh in Bank 3 point to addresses 70h-7Fh in Bank 0. The actual number of General Purpose Resisters (GPR) in each Bank depends on the device. Details are shown in Figures 2-4 through 2-8. All other RAM is unimplemented and returns '0' when read. RP<1:0> of the STATUS register are the bank select bits:

#### <u>RP1</u> <u>RP0</u>

0	0	$\rightarrow$	Bank 0 is selected
0	1	$\rightarrow$	Bank 1 is selected
1	0	$\rightarrow$	Bank 2 is selected

1 1  $\rightarrow$  Bank 3 is selected

#### 2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 128 x 8 in the PIC16F687 and 256 x 8 in the PIC16F685/PIC16F689/ PIC16F690. Each register is accessed, either directly or indirectly, through the File Select Register (FSR) (see **Section 2.4 "Indirect Addressing, INDF and FSR Registers"**).

### 2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Tables 2-1 through 2-4). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function Registers associated with the "core" are described in this section. Registers related to the operation of peripheral features are described in the section of that peripheral feature.

# 2.2.2.2 OPTION Register

The OPTION register, shown in Register 2-2, is a readable and writable register, which contains various control bits to configure:

- Timer0/WDT prescaler
- External RA2/INT interrupt
- Timer0
- Weak pull-ups on PORTA/PORTB

Note: To achieve a 1:1 prescaler assignment for Timer0, assign the prescaler to the WDT by setting PSA bit of the OPTION register to '1'. See Section 6.3 "Timer1 Prescaler".

# REGISTER 2-2: OPTION\_REG: OPTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RABPU	INTEDG	T0CS	TOSE	PSA	PS2	PS1	PS0
bit 7							bit 0

Legend:									
R = Readabl	le bit	W = V	Nritable bit	U	= Unimplemented	bit, read as '0'			
-n = Value at	t POR	'1' = l	Bit is set	ʻ(	)' = Bit is cleared	x = Bit is unknown			
bit 7	RABPU	PORTA/PO	ORTB Pull-up	Enable bit					
	1 = POR 0 = POR	<ul> <li>1 = PORTA/PORTB pull-ups are disabled</li> <li>0 = PORTA/PORTB pull-ups are enabled by individual PORT latch values</li> </ul>							
bit 6	INTEDG	NTEDG: Interrupt Edge Select bit							
	1 = Inter 0 = Inter	1 = Interrupt on rising edge of RA2/INT pin 0 = Interrupt on falling edge of RA2/INT pin							
bit 5	<b>T0CS</b> : ⊺	imer0 Clocł	Source Sele	ct bit					
	1 = Tran	sition on RA	A2/T0CKI pin						
	0 = Inter	nal instructi	on cycle cloc	k (Fosc/4)					
bit 4	<b>T0SE:</b> ⊤	imer0 Sour	ce Edge Sele	ct bit					
	1 = Incre 0 = Incre	ement on high ement on lo	gh-to-low tran w-to-high tran	sition on RA	A2/T0CKI pin A2/T0CKI pin				
bit 3	PSA: Pr	escaler Ass	ignment bit						
	1 = Pres 0 = Pres	<ul> <li>1 = Prescaler is assigned to the WDT</li> <li>0 = Prescaler is assigned to the Timer0 module</li> </ul>							
bit 2-0	PS<2:0>	. Prescaler	Rate Select I	bits					
		Bit Value	Timer0 Rate	WDT Rate					
		000	1:2	1:1					
		010	1:8	1:4					
		011	1:16	1:8					

100

101

110 111 1:32

1 : 64 1 : 128

1:256

1:16 1:32

1:64

1:128

# 3.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

### 3.1 Overview

The Oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 3-1 illustrates a block diagram of the Oscillator module.

Clock sources can be configured from external oscillators, quartz crystal resonators, ceramic resonators and Resistor-Capacitor (RC) circuits. In addition, the system clock source can be configured from one of two internal oscillators, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal via software.
- Two-Speed Start-up mode, which minimizes latency between external oscillator start-up and code execution.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, EC or RC modes) and switch automatically to the internal oscillator.

The Oscillator module can be configured in one of eight clock modes.

- 1. EC External clock with I/O on OSC2/CLKOUT.
- 2. LP 32 kHz Low-Power Crystal mode.
- 3. XT Medium Gain Crystal or Ceramic Resonator Oscillator mode.
- 4. HS High Gain Crystal or Ceramic Resonator mode.
- 5. RC External Resistor-Capacitor (RC) with Fosc/4 output on OSC2/CLKOUT.
- 6. RCIO External Resistor-Capacitor (RC) with I/O on OSC2/CLKOUT.
- 7. INTOSC Internal oscillator with Fosc/4 output on OSC2 and I/O on OSC1/CLKIN.
- 8. INTOSCIO Internal oscillator with I/O on OSC1/CLKIN and OSC2/CLKOUT.

Clock Source modes are configured by the FOSC<2:0> bits in the Configuration Word register (CONFIG). The internal clock can be generated from two internal oscillators. The HFINTOSC is a calibrated highfrequency oscillator. The LFINTOSC is an uncalibrated low-frequency oscillator.



### FIGURE 3-1: SIMPLIFIED PIC<sup>®</sup> MCU CLOCK SOURCE BLOCK DIAGRAM

# 3.2 Oscillator Control

The Oscillator Control (OSCCON) register (Figure 3-1) controls the system clock and frequency selection options. The OSCCON register contains the following bits:

- Frequency selection bits (IRCF)
- Frequency Status bits (HTS, LTS)
- System clock control bits (OSTS, SCS)

#### REGISTER 3-1: OSCCON: OSCILLATOR CONTROL REGISTER

U-0	R/W-1	R/W-1	R/W-0	R-1	R-0	R-0	R/W-0
—	IRCF2	IRCF1	IRCF0	OSTS <sup>(1)</sup>	HTS	LTS	SCS
bit 7							bit 0
Legend:							

Logonan			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	d as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7	Unimplemented: Read as '0'
bit 6-4	IRCF<2:0>: Internal Oscillator Frequency Select bits
	111 = 8 MHz
	110 = 4 MHz (default)
	101 <b>= 2 MHz</b>
	100 <b>= 1 MHz</b>
	011 <b>= 500 kHz</b>
	010 = 250 kHz
	001 = <b>125</b> kHz
	000 = 31 kHz (LFINTOSC)
bit 3	OSTS: Oscillator Start-up Time-out Status bit <sup>(1)</sup>
	<ul> <li>1 = Device is running from the clock defined by FOSC&lt;2:0&gt; of the CONFIG register</li> <li>0 = Device is running from the internal oscillator (HFINTOSC or LFINTOSC)</li> </ul>
bit 2	HTS: HFINTOSC Status bit (High Frequency – 8 MHz to 125 kHz)
	1 = HFINTOSC is stable
	0 = HFINTOSC is not stable
bit 1	LTS: LFINTOSC Stable bit (Low Frequency – 31 kHz)
	1 = 1  EINTOSC is stable
	0 = LFINTOSC is not stable
bit 0	SCS: System Clock Select bit
	1 = Internal oscillator is used for system clock
	0 = Clock source defined by FOSC<2:0> of the CONFIG register

**Note 1:** Bit resets to '0' with Two-Speed Start-up and LP, XT or HS selected as the Oscillator mode or Fail-Safe mode is enabled.

# 4.4.3.4 RB7/TX/CK

Figure 4-10 shows the diagram for this pin. The RB7/  $TX/CK^{(1)}$  pin is configurable to function as one of the following:

- a general purpose I/O
- an asynchronous serial output
- a synchronous clock I/O

Note 1: TX and CK are available on PIC16F687/ PIC16F689/PIC16F690 only.

#### FIGURE 4-10: BLOCK DIAGRAM OF RB7



Note: TMR1GE bit of the <u>T1CON</u> register must be set to use either <u>T1G</u> or C2OUT as the Timer1 gate source. See the CM2CON1 register (Register 8-3) for more information on selecting the Timer1 gate source.

Timer1 gate can be inverted using the T1GINV bit of the T1CON register, whether it originates from the T1G pin or Comparator C2 output. This configures Timer1 to measure either the active-high or active-low time between events.

# 6.7 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- TMR10N bit of the T1CON register
- TMR1IE bit of the PIE1 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note: The TMR1H:TTMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

# 6.8 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- PEIE bit of the INTCON register must be set
- T1SYNC bit of the T1CON register must be set
- TMR1CS bit of the T1CON register must be set
- T1OSCEN bit of the T1CON register (can be set)

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine (0004h).

# 6.9 ECCP Capture/Compare Time Base

The ECCP module uses the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPR1H:CCPR1L register pair on a configured event.

In Compare mode, an event is triggered when the value CCPR1H:CCPR1L register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see Section 11.0 "Enhanced Capture/Compare/PWM Module".

# 6.10 ECCP Special Event Trigger

When the ECCP is configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The ECCP module may still be configured to generate a ECCP interrupt.

In this mode of operation, the CCPR1H:CCPR1L register pair becomes the period register for Timer1.

Timer1 should be synchronized to the FOSC to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the ECCP, the write will take precedence.

For more information, see **Section 11.2.4** "Special **Event Trigger**".

# 6.11 Comparator Synchronization

The same clock used to increment Timer1 can also be used to synchronize the comparator output. This feature is enabled in the Comparator module.

When using the comparator for Timer1 gate, the comparator output should be synchronized to Timer1. This ensures Timer1 does not miss an increment if the comparator changes.

For more information, see Section 8.8.2 "Synchronizing Comparator C2 output to Timer1".

# 9.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- · Port configuration
- · Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Results formatting

#### 9.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. See the corresponding port section for more information.

Note:	Analog voltages on any pin that is defined
	as a digital input may cause the input buf-
	fer to conduct excess current.

#### 9.1.2 CHANNEL SELECTION

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 9.2 "ADC Operation"** for more information.

# 9.1.3 ADC VOLTAGE REFERENCE

The VCFG bit of the ADCON0 register provides control of the positive voltage reference. The positive voltage reference can be either VDD or an external voltage source. The negative voltage reference is always connected to the ground reference.

### 9.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- Fosc/2
- Fosc/4
- Fosc/8
- Fosc/16
- Fosc/32
- Fosc/64
- FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11 TAD periods as shown in Figure 9-2.

For correct conversion, the appropriate TAD specification must be met. See A/D conversion requirements in **Section 17.0 "Electrical Specifications"** for more information. Table 9-1 gives examples of appropriate ADC clock selections.

**Note:** Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0		
—	ADCS2	ADCS1	ADCS0		—	—	—		
bit 7							bit 0		
Legend:									
R = Readable	bit	W = Writable	bit	U = Unimpler	nented bit, read	d as '0'			
-n = Value at POR '1' = Bit is set				'0' = Bit is cle	ared	x = Bit is unkn	nown		
bit 7	Unimplemen	ted: Read as '	0'						
bit 6-4	ADCS<2:0>:	A/D Conversio	on Clock Selec	t bits					
	000 = Fosc/2	2							
	001 = Fosc/8	3							
	010 = Fosc/3	32							
	x11 = FRC (C	lock derived fro	om a dedicated	d internal oscilla	ator = $500 \text{ kHz}$	max)			
	100 = Fosc/4								
	101 = Fosc/1	6							
	110 = Fosc/6	64							
bit 3-0	Unimplemen	ted: Read as '	0'						

# REGISTER 9-2: ADCON1: A/D CONTROL REGISTER 1

### 11.3.1 PWM PERIOD

The PWM period is specified by the PR2 register of Timer2. The PWM period can be calculated using the formula of Equation 11-1.

# EQUATION 11-1: PWM PERIOD

$$PWM Period = [(PR2) + 1] \bullet 4 \bullet TOSC \bullet$$
$$(TMR2 \ Prescale \ Value)$$
Note: TOSC = 1/FOSC

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is latched from CCPR1L into CCPR1H.

#### Note: The Timer2 postscaler (see Section 7.1 "Timer2 Operation") is not used in the determination of the PWM frequency.

# 11.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPR1L register and DC1B<1:0> bits of the CCP1CON register. The CCPR1L contains the eight MSbs and the DC1B<1:0> bits of the CCP1CON register contain the two LSbs. CCPR1L and DC1B<1:0> bits of the CCP1CON register can be written to at any time. The duty cycle value is not latched into CCPR1H until after the period completes (i.e., a match between PR2 and TMR2 registers occurs). While using the PWM, the CCPR1H register is read-only.

Equation 11-2 is used to calculate the PWM pulse width.

Equation 11-3 is used to calculate the PWM duty cycle ratio.

# EQUATION 11-2: PULSE WIDTH

 $Pulse Width = (CCPR1L:CCP1CON < 5:4>) \bullet$ 

TOSC • (TMR2 Prescale Value)

# EQUATION 11-3: DUTY CYCLE RATIO

$$Duty Cycle Ratio = \frac{(CCPR1L:CCP1CON < 5:4>)}{4(PR2 + 1)}$$

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

When the 10-bit time base matches the CCPR1H and 2-bit latch, then the CCP1 pin is cleared (see Figure 11-3).

# 11.3.3 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 11-4.

### EQUATION 11-4: PWM RESOLUTION

Resolution = 
$$\frac{\log[4(PR2 + 1)]}{\log(2)}$$
 bits

Note: If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

### TABLE 11-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6.6

# TABLE 11-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

© 2005-2015 Microchip Technology Inc.

## 11.3.4 OPERATION IN SLEEP MODE

In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the CCP1 pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.

#### 11.3.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See Section 3.0 "Oscillator Module (With Fail-Safe Clock Monitor)" for additional details.

#### 11.3.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

#### 11.3.7 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

- 1. Disable the PWM pin (CCP1) output driver by setting the associated TRIS bit.
- 2. Set the PWM period by loading the PR2 register.
- Configure the CCP module for the PWM mode by loading the CCP1CON register with the appropriate values.
- Set the PWM duty cycle by loading the CCPR1L register and DC1B<1:0> bits of the CCP1CON register.
- 5. Configure and start Timer2:
  - •Clear the TMR2IF interrupt flag bit of the PIR1 register.

•Set the Timer2 prescale value by loading the T2CKPS bits of the T2CON register.

•Enable Timer2 by setting the TMR2ON bit of the T2CON register.

6. Enable PWM output after a new PWM cycle has started:

•Wait until Timer2 overflows (TMR2IF bit of the PIR1 register is set).

• Enable the CCP1 pin output driver by clearing the associated TRIS bit.

## 11.4 PWM (Enhanced Mode)

The Enhanced PWM Mode can generate a PWM signal on up to four different output pins with up to ten bits of resolution. It can do this through four different PWM Output modes:

- Single PWM
- Half-Bridge PWM
- Full-Bridge PWM, Forward mode
- Full-Bridge PWM, Reverse mode

To select an Enhanced PWM mode, the P1M bits of the CCP1CON register must be set appropriately.

The PWM outputs are multiplexed with I/O pins and are designated P1A, P1B, P1C and P1D. The polarity of the PWM pins is configurable and is selected by setting the CCP1M bits in the CCP1CON register appropriately.

Table 11-4 shows the pin assignments for each Enhanced PWM mode.

Figure 11-5 shows an example of a simplified block diagram of the Enhanced PWM module.

Note: To prevent the generation of an incomplete waveform when the PWM is first enabled, the ECCP module waits until the start of a new PWM period before generating a PWM signal.



Note 1: The TRIS register value for each PWM output must be configured appropriately.

- 2: Clearing the CCP1CON register will relinquish ECCP control of all PWM output pins.
- **3:** Any pin not used by an Enhanced PWM mode is available for alternate pin functions

### TABLE 11-4: EXAMPLE PIN ASSIGNMENTS FOR VARIOUS PWM ENHANCED MODES

ECCP Mode	P1M<1:0>	CCP1/P1A	P1B	P1C	P1D
Single	0 0	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>
Half-Bridge	10	Yes	Yes	No	No
Full-Bridge, Forward	01	Yes	Yes	Yes	Yes
Full-Bridge, Reverse	11	Yes	Yes	Yes	Yes

Note 1: Pulse Steering enables outputs in Single mode.

#### 11.4.7 PULSE STEERING MODE

In Single Output mode, pulse steering allows any of the PWM pins to be the modulated signal. Additionally, the same PWM signal can be simultaneously available on multiple pins.

Once the Single Output mode is selected (CCP1M<3:2> = 11 and P1M<1:0> = 00 of the CCP1CON register), the user firmware can bring out the same PWM signal to one, two, three or four output pins by setting the appropriate STR<D:A> bits of the PSTRCON register, as shown in Figure 11-19.

Note:	The associated TRIS bits must be set to
	output ('0') to enable the pin output driver
	in order to see the PWM signal on the pin.

While the PWM Steering mode is active, CCP1M<1:0> bits of the CCP1CON register select the PWM output polarity for the P1<D:A> pins.

The PWM auto-shutdown operation also applies to PWM Steering mode as described in **Section 11.4.4** "**Enhanced PWM Auto-shutdown mode**". An autoshutdown event will only affect pins that have PWM outputs enabled.

# **REGISTER 11-4: PSTRCON: PULSE STEERING CONTROL REGISTER<sup>(1)</sup>**

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1				
		_	STRSYNC	STRD	STRC	STRB	STRA				
bit 7						•	bit 0				
Legend:											
R = Readable	bit	W = Writable	bit	U = Unimple	mented bit, rea	d as '0'					
-n = Value at F	POR	'1' = Bit is set		'0' = Bit is cle	eared	x = Bit is unkr	Iown				
bit 7-5	Unimplemen	ted: Read as '	0'								
bit 4	STRSYNC: S	steering Sync b	it								
	1 = Output ste	eering update	occurs on next	PWM period							
	0 = Output st	eering update	occurs at the b	eginning of the	e instruction cy	cle boundary					
bit 3	STRD: Steeri	ng Enable bit [	)								
	1 = P1D pin h	nas the PWM w	vaveform with p	olarity control	from CCP1M<	:1:0>					
	0 = P1D pin is	s assigned to p	ort pin								
bit 2	STRC: Steeri	ng Enable bit (	)								
	1 = P1C pin h	nas the PWM w	vaveform with p	olarity control	from CCP1M<	:1:0>					
	0 = P1C pin is	s assigned to p	ort pin								
bit 1	STRB: Steeri	ng Enable bit E	3								
	1 = P1B pin has the PWM waveform with polarity control from CCP1M<1:0>										
	0 = P1B pin is	s assigned to p	ort pin								
bit 0	STRA: Steeri	ng Enable bit A	A								
	1 = P1A pin h	as the PWM w	aveform with p	olarity control	from CCP1M<	1:0>					
	0 = P1A pin is	s assigned to p	ort pin								

Note 1: The PWM Steering mode is available only when the CCP1CON register bits CCP1M<3:2> = 11 and P1M<1:0> = 00.

### 12.3 EUSART Baud Rate Generator (BRG)

The Baud Rate Generator (BRG) is an 8-bit or 16-bit timer that is dedicated to the support of both the asynchronous and synchronous EUSART operation. By default, the BRG operates in 8-bit mode. Setting the BRG16 bit of the BAUDCTL register selects 16-bit mode.

The SPBRGH, SPBRG register pair determines the period of the free running baud rate timer. In Asynchronous mode the multiplier of the baud rate period is determined by both the BRGH bit of the TXSTA register and the BRG16 bit of the BAUDCTL register. In Synchronous mode, the BRGH bit is ignored.

Table 12-3 contains the formulas for determining the baud rate. Example 12-1 provides a sample calculation for determining the baud rate and baud rate error.

Typical baud rates and error values for various asynchronous modes have been computed for your convenience and are shown in Table 12-3. It may be advantageous to use the high baud rate (BRGH = 1), or the 16-bit BRG (BRG16 = 1) to reduce the baud rate error. The 16-bit BRG mode is used to achieve slow baud rates for fast oscillator frequencies.

Writing a new value to the SPBRGH, SPBRG register pair causes the BRG timer to be reset (or cleared). This ensures that the BRG does not wait for a timer overflow before outputting the new baud rate. If the system clock is changed during an active receive operation, a receive error or data loss may result. To avoid this problem, check the status of the RCIDL bit to make sure that the receive operation is Idle before changing the system clock.

#### EXAMPLE 12-1: CALCULATING BAUD RATE ERROR



C	Configuration Bits			Baud Rate Formula			
SYNC	BRG16	BRGH	BRG/EUSART Mode	Bauu Kate Formula			
0	0	0	8-bit/Asynchronous	Fosc/[64 (n+1)]			
0	0	1	8-bit/Asynchronous	$E_{0} = c \left[ \frac{1}{2} \left( \frac{1}{2} + \frac{1}{2} \right) \right]$			
0	1	0	16-bit/Asynchronous	FOSC/[16 (n+1)]			
0	1	1	16-bit/Asynchronous				
1	0	x	8-bit/Synchronous	Fosc/[4 (n+1)]			
1	1	x	16-bit/Synchronous				

### TABLE 12-3: BAUD RATE FORMULAS

**Legend:** x = Don't care, n = value of SPBRGH, SPBRG register pair

#### TABLE 12-4: REGISTERS ASSOCIATED WITH THE BAUD RATE GENERATOR

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	
ABDOVF	RCIDL	_	SCKP	BRG16		WUE	ABDEN	01-0 0-00	01-0 0-00	
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	x000 000x	x000 000x	
BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000	
BRG15	BRG14	BRG13	BRG12	BRG11	BRG10	BRG9	BRG8	0000 0000	0000 0000	
CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	0000 0010	
	Bit 7 ABDOVF SPEN BRG7 BRG15 CSRC	Bit 7Bit 6ABDOVFRCIDLSPENRX9BRG7BRG6BRG15BRG14CSRCTX9	Bit 7Bit 6Bit 5ABDOVFRCIDL—SPENRX9SRENBRG7BRG6BRG5BRG15BRG14BRG13CSRCTX9TXEN	Bit 7Bit 6Bit 5Bit 4ABDOVFRCIDL—SCKPSPENRX9SRENCRENBRG7BRG6BRG5BRG4BRG15BRG14BRG13BRG12CSRCTX9TXENSYNC	Bit 7Bit 6Bit 5Bit 4Bit 3ABDOVFRCIDL—SCKPBRG16SPENRX9SRENCRENADDENBRG7BRG6BRG5BRG4BRG3BRG15BRG14BRG13BRG12BRG11CSRCTX9TXENSYNCSENDB	Bit 7Bit 6Bit 5Bit 4Bit 3Bit 2ABDOVFRCIDL—SCKPBRG16—SPENRX9SRENCRENADDENFERRBRG7BRG6BRG5BRG4BRG3BRG2BRG15BRG14BRG13BRG12BRG11BRG10CSRCTX9TXENSYNCSENDBBRGH	Bit 7Bit 6Bit 5Bit 4Bit 3Bit 2Bit 1ABDOVFRCIDL—SCKPBRG16—WUESPENRX9SRENCRENADDENFERROERRBRG7BRG6BRG5BRG4BRG3BRG2BRG1BRG15BRG14BRG13BRG12BRG11BRG10BRG9CSRCTX9TXENSYNCSENDBBRGHTRMT	Bit 7Bit 6Bit 5Bit 4Bit 3Bit 2Bit 1Bit 0ABDOVFRCIDL—SCKPBRG16—WUEABDENSPENRX9SRENCRENADDENFERROERRRX9DBRG7BRG6BRG5BRG4BRG3BRG2BRG1BRG0BRG15BRG14BRG13BRG12BRG11BRG10BRG9BRG8CSRCTX9TXENSYNCSENDBBRGHTRMTTX9D	Bit 7Bit 6Bit 5Bit 4Bit 3Bit 2Bit 1Bit 0Value on POR BORABDOVFRCIDL-SCKPBRG16-WUEABDEN01-0 0-00SPENRX9SRENCRENADDENFERROERRRX9D0000 000xBRG7BRG6BRG5BRG4BRG3BRG2BRG1BRG00000 000xBRG15BRG14BRG13BRG12BRG11BRG10BRG9BRG80000 000xCSRCTX9TXENSYNCSENDBBRGHTRMTTX9D0000 001	

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for the Baud Rate Generator.

# 12.4 EUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective receive and transmit shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The EUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

#### 12.4.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the EUSART for Synchronous Master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART. If the RX/DT or TX/CK pins are shared with an analog peripheral the analog I/O functions must be disabled by clearing the corresponding ANSEL bits.

#### 12.4.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/CK line. The TX/CK pin output driver is automatically enabled when the EUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

#### 12.4.1.2 Clock Polarity

A clock polarity option is provided for Microwire compatibility. Clock polarity is selected with the SCKP bit of the BAUDCTL register. Setting the SCKP bit sets the clock Idle state as high. When the SCKP bit is set, the data changes on the falling edge of each clock. Clearing the SCKP bit sets the Idle state as low. When the SCKP bit is cleared, the data changes on the rising edge of each clock.

#### 12.4.1.3 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the EUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note:	The TSR register is not mapped in data
	memory, so it is not available to the user.

# 12.4.1.4 Synchronous Master Transmission Set-up:

- 1. Initialize the SPBRGH, SPBRG register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see Section 12.3 "EUSART Baud Rate Generator (BRG)").
- 2. Enable the synchronous master serial port by setting bits SYNC, SPEN, and CSRC.
- 3. Disable Receive mode by clearing bits SREN and CREN.
- 4. Enable Transmit mode by setting the TXEN bit.
- 5. If 9-bit transmission is desired, set the TX9 bit.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 7. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
- 8. Start transmission by loading data to the TXREG register.

### 12.4.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are un-read characters in the receive FIFO.

### 12.4.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/ CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

### 12.4.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RCREG is read to access the FIFO. When this happens the OERR bit of the RCSTA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RCREG. If the overrun occurred when the CREN bit is

set then the error condition is cleared by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

#### 12.4.1.8 Receiving 9-bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the EUSART will shift 9-bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

#### 12.4.1.9 Synchronous Master Reception Setup:

- 1. Initialize the SPBRGH, SPBRG register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
- 2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
- 3. Ensure bits CREN and SREN are clear.
- 4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 5. If 9-bit reception is desired, set bit RX9.
- 6. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
- 7. Interrupt flag bit RCIF will be set when reception of a character is complete. An interrupt will be generated if the enable bit RCIE was set.
- 8. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
- 9. Read the 8-bit received data by reading the RCREG register.
- 10. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.



- is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.
- 3: CLKOUT is available only in INTOSC and RC Oscillator modes.
- 4: For minimum width of INT pulse, refer to AC specifications in Section 17.0 "Electrical Specifications".
- 5: INTF is enabled to be set any time during the Q4-Q1 cycles.

#### TABLE 14-6: SUMMARY OF INTERRUPT REGISTERS

Name	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	
INTCON	GIE	PEIE	TOIE	INTE	RABIE	T0IF	INTF	RABIF	x000 000x	x000 000x	
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000	
PIE2	OSFIE	C2IE	C1IE	EEIE	—	—	—	—	0000	0000	
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000	
PIR2	OSFIF	C2IF	C1IF	EEIF	_	_	_	_	0000	0000	

**Legend:** x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends upon condition.

Shaded cells are not used by the Interrupt module.

Mnemonic,		Description	Cycles	14-Bit Opcode				Status	Neteo
Oper	ands	Description	Cycles	MSb			LSb	Affected	Notes
		BYTE-ORIENTED FILE REGIS	TER OPE	RATIC	ONS				
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	1, 2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1, 2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1, 2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1, 2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2, 3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1, 2
INCFSZ	f, d	Increment f, Skip if 0	1 <b>(2)</b>	00	1111	dfff	ffff		1, 2, 3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1, 2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1, 2
MOVWF	f	Move W to f	1	00	0000	lfff	ffff		
NOP	_	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	С	1, 2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	С	1, 2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	1, 2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1, 2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1, 2
		BIT-ORIENTED FILE REGIST	ER OPER	RATIO	NS				
BCF	f. b	Bit Clear f	1	01	00bb	bfff	ffff		1.2
BSF	f. b	Bit Set f	1	01	01bb	bfff	ffff		1.2
BTESC	f, b	Bit Test f. Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f. b	Bit Test f. Skip if Set	1 (2)	01	11bb	bfff	ffff		3
	, -	LITERAL AND CONTROL	OPERAT	IONS					
ADDI W	k	Add literal and W	1	11	111x	kkkk	kkkk	C. DC. 7	
	k	AND literal with W	1	11	1001	kkkk	kkkk	7	
CALL	k	Call Subroutine	2	10	 0kkk	kkkk	kkkk	_	
CLRWDT	_	Clear Watchdog Timer	1	0.0	0000	0110	0100	TO PD	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk	,.	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	_	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	_	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	_	Go into Standby mode	1	00	0000	0110	0011	TO, PD	
SUBLW	k	Subtract w from literal	1	11	110x	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Ζ	
			L						

#### TABLE 15-2: PIC16F684 INSTRUCTION SET

Note 1: When an I/O register is modified as a function of itself (e.g., MOVF GPIO, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.

**3:** If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

# 16.11 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<sup>™</sup> and dsPICDEM<sup>™</sup> demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ<sup>®</sup> security ICs, CAN, IrDA<sup>®</sup>, PowerSmart battery management, SEEVAL<sup>®</sup> 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.

# 16.12 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<sup>®</sup> and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika<sup>®</sup>