



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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	I ² C, SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-MQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc64at-04-pq

Email: info@E-XFL.COM

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

Pin Diagrams (Cont.'d)



1.0 GENERAL DESCRIPTION

The PIC16CXX is a family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers.

All PIC16/17 microcontrollers employ an advanced RISC architecture. The PIC16CXX microcontroller family has enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with separate 8-bit wide data. The two stage instruction pipeline allows all instructions to execute in a single cycle, except for program branches (which require two cycles). A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16CXX microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

The **PIC16C61** device has 36 bytes of RAM and 13 I/O pins. In addition a timer/counter is available.

The **PIC16C62/62A/R62** devices have 128 bytes of RAM and 22 I/O pins. In addition, several peripheral features are available, including: three timer/counters, one Capture/Compare/PWM module and one serial port. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPITM) or the two-wire Inter-Integrated Circuit (I²C) bus.

The **PIC16C63/R63** devices have 192 bytes of RAM, while the **PIC16C66** has 368 bytes. All three devices have 22 I/O pins. In addition, several peripheral features are available, including: three timer/counters, two Capture/Compare/PWM modules and two serial ports. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I^2C) bus. The Universal Synchronous Asynchronous Receiver Transmitter (USART) is also know as a Serial Communications Interface or SCI.

The **PIC16C64/64A/R64** devices have 128 bytes of RAM and 33 I/O pins. In addition, several peripheral features are available, including: three timer/counters, one Capture/Compare/PWM module and one serial port. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I²C) bus. An 8-bit Parallel Slave Port is also provided.

The **PIC16C65/65A/R65** devices have 192 bytes of RAM, while the **PIC16C67** has 368 bytes. All four devices have 33 I/O pins. In addition, several peripheral features are available, including: three timer/counters, two Capture/Compare/PWM modules and two serial ports. The Synchronous Serial Port can be configured as either a 3-wire Serial Peripheral Interface (SPI) or the two-wire Inter-Integrated Circuit (I²C) bus. The Universal Synchronous Asynchronous Receiver Transmit-

ter (USART) is also known as a Serial Communications Interface or SCI. An 8-bit Parallel Slave Port is also provided.

The PIC16C6X device family has special features to reduce external components, thus reducing cost, enhancing system reliability and reducing power consumption. There are four oscillator options, of which the single pin RC oscillator provides a low-cost solution, the LP oscillator minimizes power consumption, XT is a standard crystal, and the HS is for High Speed crystals. The SLEEP (power-down) mode offers a power saving mode. The user can wake the chip from SLEEP through several external and internal interrupts, and resets.

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lockup.

A UV erasable CERDIP packaged version is ideal for code development, while the cost-effective One-Time-Programmable (OTP) version is suitable for production in any volume.

The PIC16C6X family fits perfectly in applications ranging from high-speed automotive and appliance control to low-power remote sensors, keyboards and telecom processors. The EPROM technology makes customization of application programs (transmitter codes, motor speeds, receiver frequencies, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series perfect for all applications with space limitations. Low-cost, low-power, high performance, ease-of-use, and I/O flexibility make the PIC16C6X very versatile even in areas where no microcontroller use has been considered before (e.g. timer functions, serial communication, capture and compare, PWM functions, and co-processor applications).

1.1 Family and Upward Compatibility

Those users familiar with the PIC16C5X family of microcontrollers will realize that this is an enhanced version of the PIC16C5X architecture. Please refer to Appendix A for a detailed list of enhancements. Code written for PIC16C5X can be easily ported to PIC16CXX family of devices (Appendix B).

1.2 Development Support

PIC16C6X devices are supported by the complete line of Microchip Development tools.

Please refer to Section 15.0 for more details about Microchip's development tools.

4.2.2 SPECIAL FUNCTION REGISTERS:

The Special Function Registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. The special function registers can be classified into two sets (core and peripheral). The registers associated with the "core" functions are described in this section and those related to the operation of the peripheral features are described in the section of that peripheral feature.

TABLE 4-1:	SPECIAL	FUNCTION	REGISTERS	FOR	THE	PIC16C61

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR	Value on all other resets ⁽³⁾
Bank 0											
00h ⁽¹⁾	INDF	Addressing	this location	uses conten	ts of FSR to	address data	a memory (n	ot a physical	register)	0000 0000	0000 0000
01h	TMR0	Timer0 mod	lule's registe	r						xxxx xxxx	uuuu uuuu
02h ⁽¹⁾	PCL	Program Co	ounter's (PC)	Least Signif	icant Byte					0000 0000	0000 0000
03h ⁽¹⁾	STATUS	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
04h ⁽¹⁾	FSR	Indirect data	ndirect data memory address pointer								uuuu uuuu
05h	PORTA	—	—	—	PORTA Dat	a Latch whe	n written: PC	RTA pins wh	en read	x xxxx	u uuuu
06h	PORTB	PORTB Dat	ta Latch whe	n written: PC	ORTB pins wh	nen read				xxxx xxxx	uuuu uuuu
07h	—	Unimpleme	Inimplemented —							—	—
08h	-	Unimpleme	nted							—	—
09h	—	Unimpleme	nted							—	—
0Ah ^(1,2)	PCLATH	—	—	—	Write Buffer	for the uppe	er 5 bits of the	e Program C	ounter	0 0000	0 0000
0Bh ⁽¹⁾	INTCON	GIE	—	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u
Bank 1											
80h ⁽¹⁾	INDF	Addressing	this location	uses conten	ts of FSR to	address data	a memory (n	ot a physical	register)	0000 0000	0000 0000
81h	OPTION	RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h ⁽¹⁾	PCL	Program Co	ounter's (PC)	Least Signif	icant Byte					0000 0000	0000 0000
83h ⁽¹⁾	STATUS	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
84h ⁽¹⁾	FSR	Indirect data	a memory ad	dress pointe	r					xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	—	PORTA Dat	a Direction F	Register			1 1111	1 1111
86h	TRISB	PORTB Dat	ta Direction C	Control Regis	ster					1111 1111	1111 1111
87h	-	Unimpleme	nted							—	—
88h	-	Unimpleme	nted							—	—
89h	-	Unimpleme	nted							—	—
8Ah ^(1,2)	PCLATH	—	—	—	Write Buffer	for the uppe	er 5 bits of th	e Program C	ounter	0 0000	0 0000
8Bh ⁽¹⁾	INTCON	GIE	_	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0-00 000x	0-00 000u

 $\label{eq:logend: condition} \ensuremath{\mathsf{Legend: }} x = \mathsf{unknown}, u = \mathsf{unchanged}, q = \mathsf{value} \ensuremath{\, depends} \ensuremath{\, on \, condition}, \ensuremath{\, - \, = \, unimplemented \, locations \, read \, as \, '0'.$

Shaded locations are unimplemented and read as '0'

Note 1: These registers can be addressed from either bank.

2: The upper byte of the Program Counter (PC) is not directly accessible. PCLATH is a holding register for the PC whose contents are transferred to the upper byte of the program counter. (PC<12:8>)

3: Other (non power-up) resets include external reset through MCLR and the Watchdog Timer Reset.

4: The IRP and RP1 bits are reserved on the PIC16C61, always maintain these bits clear.

Address	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 ⁽³⁾
Bank 1											
80h ⁽¹⁾	INDF	Addressing	this location	uses conter	nts of FSR to	address data	a memory (n	ot a physical	register)	0000 0000	0000 0000
81h	OPTION	RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h ⁽¹⁾	PCL	Program Co	ounter's (PC)	Least Sigr	nificant Byte					0000 0000	0000 0000
83h ⁽¹⁾	STATUS	IRP ⁽⁵⁾	RP1 ⁽⁵⁾	RP0	TO	PD	z	DC	С	0001 1xxx	000q quuu
84h ⁽¹⁾	FSR	Indirect data	ndirect data memory address pointer							xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Da	ta Direction R	egister				11 1111	11 1111
86h	TRISB	PORTB Dat	ta Direction F	Register						1111 1111	1111 1111
87h	TRISC	PORTC Da	ta Direction I	Register						1111 1111	1111 1111
88h	TRISD	PORTD Da	ta Direction I	Register						1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE		PORTE Da	ta Direction I	Bits	0000 -111	0000 -111
8Ah ^(1,2)	PCLATH	—	-	—	Write Buffer	for the uppe	r 5 bits of the	e Program C	ounter	0 0000	0 0000
8Bh ⁽¹⁾	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE	(6)	—	_	SSPIE	CCP1IE	TMR2IE	TMR1IE	00 0000	00 0000
8Dh	_	Unimpleme	nted							—	-
8Eh	PCON	_	—	—	—		—	POR	BOR ⁽⁴⁾	dd	uu
8Fh	_	Unimpleme	nted							—	-
90h	_	Unimpleme	nted							—	
91h	—	Unimpleme	nted							—	-
92h	PR2	Timer2 Peri	od Register							1111 1111	1111 1111
93h	SSPADD	Synchronou	us Serial Por	t (I ² C mode)	Address Reg	ister				0000 0000	0000 0000
94h	SSPSTAT	—	—	D/A	Р	S	R/W	UA	BF	00 0000	00 0000
95h-9Fh	-	Unimpleme	nted							-	_

TABLE 4-4: SPECIAL FUNCTION REGISTERS FOR THE PIC16C64/64A/R64 (Cont.'d)

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

Note 1: These registers can be addressed from either bank.

2: The upper byte of the Program Counter (PC) is not directly accessible. PCLATH is a holding register for the PC whose contents are transferred to the upper byte of the program counter. (PC<12:8>)

3: Other (non power-up) resets include external reset through MCLR and the Watchdog Timer reset.

4: The BOR bit is reserved on the PIC16C64, always maintain this bit set.

5: The IRP and RP1 bits are reserved on the PIC16C64/64A/R64, always maintain these bits clear.

6: PIE1<6> and PIR1<6> are reserved on the PIC16C64/64A/R64, always maintain these bits clear.

FIGURE 4-15: PIE1 REGISTER FOR PIC16C65/65A/R65/67 (ADDRESS 8Ch)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
PSPIE	_	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	R = Readable bit
bit7							bitO	W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset
bit 7:	PSPIE: Part 1 = Enable 0 = Disable	rallel Slave s the PSP es the PSP	Port Read read/write i read/write	/Write Interr nterrupt interrupt	upt Enable b	vit		
bit 6:	Reserved:	Always ma	aintain this	bit clear.				
bit 5:	RCIE: USA 1 = Enable 0 = Disable	ART Receiv s the USAF es the USA	e Interrupt RT receive RT receive	Enable bit interrupt interrupt				
bit 4:	TXIE: USA 1 = Enable 0 = Disable	RT Transm s the USAF s the USA	it Interrupt RT transmit RT transmi	Enable bit interrupt t interrupt				
bit 3:	SSPIE: Syn 1 = Enable 0 = Disable	nchronous s the SSP es the SSP	Serial Port interrupt interrupt	Interrupt Er	nable bit			
bit 2:	CCP1IE : C 1 = Enable 0 = Disable	CP1 Interrors the CCP ⁻ is the CCP ⁻ ies the CCP	upt Enable I interrupt 1 interrupt	bit				
bit 1:	TMR2IE: T 1 = Enable 0 = Disable	MR2 to PR s the TMR2 es the TMR	2 Match In 2 to PR2 m 2 to PR2 m	terrupt Enat atch interrup atch interru	ole bit pt ipt			
bit 0:	TMR1IE: T 1 = Enable 0 = Disable	MR1 Overf s the TMR es the TMR	low Interru 1 overflow i 1 overflow	ot Enable bi nterrupt interrupt	t			

7.3 Prescaler

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer (WDT), respectively (Figure 7-6). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet. Note that the prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

The PSA and PS2:PS0 bits (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. CLRF TMR0, MOVWF TMR0, BSF TMR0, bitx) will clear the prescaler count. When assigned to the Watchdog Timer, a CLRWDT instruction will clear the Watchdog Timer and the prescaler count. The prescaler is not readable or writable.

Note: Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count, but will not change the prescaler assignment.



FIGURE 7-6: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER

8.3 <u>Timer1 Operation in Asynchronous</u> <u>Counter Mode</u>

Applicable Devices

61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

If control bit $\overline{T1SYNC}$ (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and generate an interrupt on overflow which will wake the processor. However, special precautions in software are needed to read-from or write-to the Timer1 register pair, TMR1L and TMR1H (Section 8.3.2).

In asynchronous counter mode, Timer1 cannot be used as a time-base for capture or compare operations.

8.3.1 EXTERNAL CLOCK INPUT TIMING WITH UNSYNCHRONIZED CLOCK

If control bit $\overline{T1SYNC}$ is set, the timer will increment completely asynchronously. The input clock must meet certain minimum high time and low time requirements, as specified in timing parameters (45 - 47).

8.3.2 READING AND WRITING TMR1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L, while the timer is running from an external asynchronous clock, will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself poses certain problems since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers while the register is incrementing. This may produce an unpredictable value in the timer register.

Reading the 16-bit value requires some care. Example 8-1 is an example routine to read the 16-bit timer value. This is useful if the timer cannot be stopped.

EXAMPLE 8-1: READING A 16-BIT FREE-RUNNING TIMER

;	All Int	errupts	are	disabled
	MOVF	TMR1H,	W	;Read high byte
	MOVWF	TMPH		;
	MOVF	TMR1L,	W	;Read low byte
	MOVWF	TMPL		;
	MOVF	TMR1H,	W	;Read high byte
	SUBWF	TMPH,	W	;Sub 1st read
				;with 2nd read
	BTFSC	STATUS	Z	;is result = 0
	GOTO	CONTINU	JE	;Good 16-bit read
;	TMR1L ma	y have r	olle	d over between the read
;	of the h	igh and	low	bytes. Reading the high
;	and low	bytes no	w w	ill read a good value.
	MOVF	TMR1H,	W	;Read high byte
	MOVWF	TMPH		;
	MOVF	TMR1L,	W	;Read low byte
	MOVWF	TMPL		;
;	Re-ena	ble Inte	rrup	ot (if required)
CC	ONTINUE			;Continue with
	:			;your code

8.4 <u>Timer1 Oscillator</u>

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

A crystal oscillator circuit is built in-between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 8-1 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must allow a software time delay to ensure proper oscillator start-up.

TABLE 8-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR

Osc Type	Freq	C1	C2
LP	32 kHz 33 pF		33 pF
	100 kHz	15 pF	15 pF
	200 kHz	15 pF	15 pF
These v	alues are for o	design guidan	ce only.
Crystals Tes	sted:		
32.768 kHz	Epson C-00	1R32.768K-A	\pm 20 PPM
100 kHz	Epson C-2 1	00.00 KC-P	\pm 20 PPM
200 kHz	STD XTL 20	0.000 kHz	\pm 20 PPM
Note 1: Hig of o time 2: Sind cha reso ate	her capacitand scillator but al ce each reson racteristics, th phator/crystal values of exte	ce increases t lso increases ator/crystal ha le user should manufacturer ernal compone	he stability the start-up as its own consult the for appropri- ents.

TABLE 9-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valu PC BC	e on:)R,)R	Valu all o res	e on ther ets
0Bh,8Bh 10Bh,18Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000	000x	0000	000u
0Ch	PIR1	PSPIF ⁽²⁾	(3)	RCIF ⁽¹⁾	TXIF ⁽¹⁾	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000	0000	0000	0000
8Ch	PIE1	PSPIE ⁽²⁾	(3)	RCIE ⁽¹⁾	TXIE ⁽¹⁾	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000	0000	0000	0000
11h	TMR2	Timer2 m	odule's reg	ister						0000	0000	0000	0000
12h	T2CON		TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000	0000	-000	0000
92h	PR2	Timer2 P	eriod regist	er						1111	1111	1111	1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer2.

Note 1: The USART is implemented on the PIC16C63/R63/65/65A/R65/66/67 only.

2: Bits PSPIE and PSPIF are reserved on the PIC16C62/62A/R62/63/R63/66, always maintain these bits clear.

3: PIR1<6> and PIE1<6> are reserved, always maintain these bits clear.

FIGURE 11-27: OPERATION OF THE I²C MODULE IN IDLE_MODE, RCV_MODE OR XMIT_MODE

IDLE_MODE (7-bit): if (Addr_match) { Set interrupt;	
else if (R/₩ = 0) set RCV_MODE; }	
RCV_MODE: if ((SSPBUF=Full) OR (SSPOV = 1)) { Set SSPOV; Do not acknowledge; }	
else { transfer SSPSK \rightarrow SSPBUF; send $\overline{ACK} = 0;$ }	
Receive 8-bits in SSPSR; Set interrupt;	
XMIT_MODE: While ((SSPBUF = Empty) AND (CKP=0)) Hold SCL Low; Send byte; Set interrupt; if (ACK Received = 1) { End of transmission; Go back to IDLE_MODE;	
else if (ACK Received = 0) Go back to XMIT_MODE; IDLE_MODE (10-Bit): If (High_byte_addr_match AND (R/W = 0)) { PRIOR_ADDR_MATCH = FALSE; Set interrupt; if ((SSPBUF = Full) OR ((SSPOV = 1)) { Set SSPOV; Do not acknowledge; } else { Set UA = 1; Send ĀCK = 0; While (SSPADD not updated) Hold SCL low; Clear UA = 0; Receive Low_addr_byte; Set interrupt; Set UA = 1; If (Low_byte_addr_match) { PRIOR_ADDR_MATCH = TRUE; Send ĀCK = 0; while (SSPADD not updated) Hold SCL low; Clear UA = 0; Set UA = 1; If (Low_byte_addr_match) { PRIOR_ADDR_MATCH = TRUE; Send ĀCK = 0; while (SSPADD not updated) Hold SCL low; Clear UA = 0; Set RCV_MODE; }	
} else if (High_byte_addr_match AND (RW = 1) {	
} else PRIOR_ADDR_MATCH = FALSE; }	

Г

FIGURE 12-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER (ADDRESS 18h)

SPEN RX9 SREN CREN — FERR OERR RX9D R = Readable bit bit7 bit0 If = Readable bit If = Writable bit If =	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R-0	R-0	R-x		
bit7 bit8 bit0 W = Writable bit W = Writable bit U = Uunimplemented bit, read as '0' - n = 'value at POR reset x = unknown bit 7: SPEN: Serial Port Enable bit (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7.6> are set) 1 = Serial port disabled bit 6: RX9: 9-bit Receive Enable bit 1 = Selects 9-bit reception 0 = Selects 8-bit reception 0 = Selects 8-bit reception bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care Synchronous mode - master 1 = Enables single receive This bit is cleared after reception is complete. Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERF: Framing Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error bit 0: RX9D: 9th bit of received data (Can be parity bit)	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	R	= Readable bit
bit 7: SPEN: Serial Port Enable bit (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7:6> are set) 1 = Serial port enabled 0 = Serial port disabled 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 1 = Enables single receive 0 = Don't care Synchronous mode Don't care Synchronous mode - master 1 = Enables single receive 0 = Disables single receive This bit is cleared after reception is complete. Synchronous mode Dunts of this mode Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode Disables continuous receive 0 = Disables continuous receive bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 4: CREN: Continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 4: CREN: Craning Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) 0 = No traming error bit 1: OERF: Overrun Error bit 1 = Overrun error bit 1: CREN: Overrun error bit 1: Overrun error bit 1: Received data (Can be parity bit)	bit7							bit0	W	= Writable bit
-n = Value at POR reset x = unknown bit 7: SPEN: Serial Port Enable bit (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7:6> are set) 1 = Serial port enabled 0 = Serial port enabled 0 = Selects 8-bit Receive Enable bit 1 = Selects 9-bit reception 0 = Selects 8-bit reception 0 = Selects 8-bit reception bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care Synchronous mode - master 1 = Enables single receive 0 = Disables single receive 0 = Disables single receive Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 4: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>0</td> <td>bit. read as '0'</td>									0	bit. read as '0'
k = unknown bit 7: SPEN: Serial Port Enable bit (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7:6> are set) 1 = Serial port disabled 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 Don't care 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 bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 0 = Disables continuous receive bit 4: CREN: Continuous receive 0 = Disables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error bit 1 = Framing error bit 1 = Overrun error bit 1: OERR: Overrun Error bit 1 = Overrun error bit 1: OERR: Overrun Error bit 1 = Overrun error									- n	= Value at POR reset
bit 7: SPEN: Serial Port Enable bit (Configures RC7/RX/DT and RC6/TX/CK pins as serial port pins when bits TRISC<7:6> are set) 1 = Serial port enabled 0 = Serial port disabled bit 6: RX9: 9-bit reception 0 = Selects 8-bit reception bit 5: SREN: Single Receive Enable bit <u>Asynchronous mode</u> Don't care <u>Synchronous mode - master</u> 1 = Enables single receive Don't care <u>Synchronous mode - master</u> 1 = Enables single receive This bit is cleared after reception is complete. <u>Synchronous mode - slave</u> Unused in this mode bit 4: CREN: Continuous Receive Enable bit <u>Asynchronous mode</u> 1 = Enables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive 1 = Enables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit)									х	= unknown
bit 6: RX9: 9-bit Receive Enable bit 1 = Selects 9-bit reception 0 0 = Selects 8-bit reception 0 bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care Synchronous mode - master 1 1 = Enables single receive 0 0 = Disables single receive This bit is cleared after reception is complete. Synchronous mode - slave Unused in this mode Unused in this mode Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 0 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Framing Error bit 1 = Framing Error bit 1 = Framing error (Can be updated by reading RCREG reg	bit 7:	SPEN: Ser (Configure 1 = Serial 0 = Serial	rial Port En s RC7/RX/I port enable port disable	able bit DT and RC d ed	6/TX/CK	oins as ser	ial port pin	s when bits	TRIS	C<7:6> are set)
1 = Selects 9-bit reception 0 = Selects 8-bit reception bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care 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 bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 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: Unimplemented: Read as '0' bit 3: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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 1: OERR: Overrun Error bit 1 = Overrun error 1 = No overrun error Deared by clearing bit CREN) 0 = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit)	bit 6:	RX9 : 9-bit	Receive Er	nable bit						
 bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care Synchronous mode - master 1 = Enables single receive 0 = Disables single receive 0 = Disables single receive This bit is cleared after reception is complete. Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 1 = Enables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error bit 1: OERR: Overrun Error bit 1 = Overrun error bit 1: OERR: Overrun Error bit 1 = Overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) 		1 = Selects	s 9-bit rece	ption						
 bit 5: SREN: Single Receive Enable bit Asynchronous mode Don't care 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 bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 4: FERR: Framing Error bit 1 = Framing error bit 1: OERR: Overrun Error bit 1 = Overrun error (Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error bit 0: RX9D: 9th bit of received data (Can be parity bit) 		0 = Selects	s 8-bit rece	ption						
Asynchronous mode Don't care 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 bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 4: FERR: Framing Error bit 1 = Framing error 1 bit 1: OERR: Overrun Error bit 1 = Overrun error 1 bit 1: OERR: Overrun Error bit 1 = Overrun error 1 bit 0: RX9D: 9th bit of received data (Can be parity bit)	bit 5:	SREN: Sin	gle Receiv	e Enable bi	t					
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 bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive bit 3: Unimplemented: Read as '0' bit 4: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error bit 1: OERR: Overrun Error bit 1 = Overrun error Icared by clearing bit CREN) 0 = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) <td></td> <td>Asynchron Don't care</td> <td><u>ous mode</u></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>		Asynchron Don't care	<u>ous mode</u>							
 1 = Enables single receive 0 = Disables single receive This bit is cleared after reception is complete. Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error bit 1: OERR: Overrun Error bit 1 = Overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) 		Synchrono	ous mode -	master						
 bit di is cleared after reception is complete. Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 		1 = Enable	s single real	ceive						
Synchronous mode - slave Unused in this mode bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive Synchronous mode 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error 0 bit 1: OERR: Overrun Error bit 1 = Overrun error (Can be cleared by clearing bit CREN) 0 = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit)		0 = Disable	es single re	ceive er recention	is comple	ote				
bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 0 = Disables continuous receive 0 = Disables continuous receive Synchronous mode 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive 0 bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error 1 = Framing error bit 1: OERR: Overrun Error bit 1 = Overrun error 1 = Overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit)		Synchrono	us mode -	slave	lo compi					
bit 4: CREN: Continuous Receive Enable bit Asynchronous mode 1 = Enables continuous receive 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: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG register and receive next valid byte) 0 = No framing error bit 1: OERR: Overrun Error bit 1 = Overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit)		Unused in	this mode							
Asynchronous mode 1 = Enables continuous receive 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: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit)	bit 4:	CREN: Co	ntinuous R	eceive Ena	ble bit					
 1 = Enables continuous receive 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: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 		Asynchron	ous mode							
 bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 		1 = Enable	s continuo	us receive						
Synchronous mode 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN) 0 = Disables continuous receive bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit)		0 = Disable	es continuo	us receive						
 bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 		Synchronc	ous mode		ntil on ohl		lia alaarad			
 bit 3: Unimplemented: Read as '0' bit 2: FERR: Framing Error bit = Framing error (Can be updated by reading RCREG register and receive next valid byte) = No framing error bit 1: OERR: Overrun Error bit = Overrun error (Can be cleared by clearing bit CREN) = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) 		0 = Disable	es continuo	us receive t			NIS Cleared		ennue	S SHEN)
 bit 2: FERR: Framing Error bit 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 	bit 3:	Unimplem	ented: Rea	ad as '0'						
 1 = Framing error (Can be updated by reading RCREG 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: 9th bit of received data (Can be parity bit) 	bit 2:	FERR: Fra	ming Error	bit						
 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: 9th bit of received data (Can be parity bit) 		1 = Framin	g error (Ca	n be updat	ed by read	ding RCRE	G register	and receive	next	valid byte)
 bit 1: OERR: Overrun Error bit 1 = Overrun error (Can be cleared by clearing bit CREN) 0 = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) 		0 = No frar	ning error							
 0 = No overrun error bit 0: RX9D: 9th bit of received data (Can be parity bit) 	bit 1:	OERR: Ov	errun Error	bit						
bit 0: RX9D : 9th bit of received data (Can be parity bit)		1 = Overru 0 = No over	m error (Ca errun error	n pe cleare	u by clear	ning bit CR	EN)			
bit 0. The strok of received data (Call be party bit)	hit 0.		hit of rocoi	vod data (C	an ha na	rity hit)				
	DIE U.	11730. 901	DIL UI IECEI	veu uaia (C	an be pa					

FIGURE 13-14: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



- Note 1: External Power-on Reset circuit is required only if VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.
 - R < 40 kΩ is recommended to make sure that voltage drop across R does not violate the devices electrical specifications.
 - 3: $R1 = 100\Omega$ to 1 k Ω will limit any current flowing into MCLR from external capacitor C in the event of MCLR/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrostatic Overstress (EOS).

FIGURE 13-15: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1



- Note 1: This circuit will activate reset when VDD goes below (Vz + 0.7V) where Vz = Zener voltage.
 - Internal brown-out detection on the PIC16C62A/R62/63/R63/64A/R64/65A/ R65/66/67 should be disabled when using this circuit.
 - 3: Resistors should be adjusted for the characteristics of the transistors.

FIGURE 13-16: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2



Note 1: This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

$$V_{DD} \bullet \frac{R1}{R1 + R2} = 0.7V$$

- 2: Internal brown-out detection on the PIC16C62A/R62/63/R63/64A/R64/65A/ R65/66/67 should be disabled when using this circuit.
- 3: Resistors should be adjusted for the characteristics of the transistors.

CLRF	Clear f				
Syntax:	[<i>label</i>] C	LRF f			
Operands:	$0 \le f \le 12$	7			
Operation:	$00h \rightarrow (f)$ 1 $\rightarrow Z$	1			
Status Affected:	Z				
Encoding:	0.0	0001	lfff	ffff	
Description:	The conte and the Z	nts of regi bit is set.	ister 'f' are	cleared	
Words:	1				
Cycles:	1				
Q Cycle Activity:	Q1	Q2	Q3	Q4	
	Decode	Read register 'f'	Process data	Write register 'f'	
Example	CLRF	FLAC	G_REG		
	Before In	struction	1		
		FLAG_RE	EG =	0x5A	
	Atter instruction				
		Ζ	=	1	

CLRW	Clear W			
Syntax:	[label]	CLRW		
Operands:	None			
Operation:	$00h \rightarrow (W 1 \rightarrow Z$	V)		
Status Affected:	z			
Encoding:	00	0001	0xxx	xxxx
Description:	W register set.	is cleared	. Zero bit ((Z) is
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	No- Operation	Process data	Write to W
Example	CLRW			
	Before In	struction		
	After Inst	W =	0x5A	
		W =	0x00	
		Z =	1	
CLRWDT	Clear Wa	atchdog	Гimer	
A				
Syntax:	[label]	CLRWD	Γ	
Syntax: Operands:	[<i>label</i>] None	CLRWD1	Γ	
Syntax: Operands: Operation:	$\begin{bmatrix} label \end{bmatrix}$ None $00h \rightarrow W$ $0 \rightarrow WD^{-}$ $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$	CLRWD1 /DT F prescale	ər,	
Syntax: Operands: Operation: Status Affected:	$\begin{bmatrix} label \\ None \\ 00h \rightarrow W \\ 0 \rightarrow WD^{-} \\ 1 \rightarrow \overline{TO} \\ 1 \rightarrow \overline{PD} \\ \overline{TO}, \overline{PD} \end{bmatrix}$	CLRWD1 /DT F prescale	ər,	
Syntax: Operands: Operation: Status Affected: Encoding:	$\begin{bmatrix} label \end{bmatrix}$ None $00h \rightarrow WD$ $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$ $\overline{TO}, \overline{PD}$ 00	CLRWD1 /DT F prescale	9 r,	0100
Syntax: Operands: Operation: Status Affected: Encoding: Description:	$\begin{bmatrix} label \end{bmatrix}$ None $00h \rightarrow WD$ $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$ $\overline{TO}, \overline{PD}$ $\boxed{00}$ CLRWDT irr dog Timer of the WD set.	CLRWDT /DT f prescale 0000 nstruction r . It also ree T. Status b	o110 esets the v set <u>s th</u> e pr its TO and	0100 Watch- escaler PD are
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words:	$\begin{bmatrix} label \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ WD^{-1} \\ 0 \\ 1 \\ 0 \\ \hline TO, PD \\ \hline TO, PD \\ \hline 0 \\ CLRWDT \\ inf \\ og Timer \\ of the WD \\ set. \\ 1 \\ \end{bmatrix}$	CLRWD1 /DT f prescale struction r. . It also res T. Status b	0110 esets the prisets the prite TO and	0100 Watch- escaler PD are
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles:	$\begin{bmatrix} label \\ None \\ 00h \rightarrow WD \\ 0 \rightarrow WD^{-} \\ 1 \rightarrow \overline{PD} \\ \hline \overline{TO}, \overline{PD} \\ \hline \overline{TO}, \overline{PD} \\ \hline 00 \\ CLRWDT in \\ dog Timer \\ of the WD \\ set. \\ 1 \\ 1 \end{bmatrix}$	CLRWD1 /DT f prescale oooo istruction r . It also re: T. Status b	0110 esets the V esets the price of the formation of the	0100 Watch- escaler PD are
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity:	$\begin{bmatrix} label \\ None \\ 00h \rightarrow W \\ 0 \rightarrow WD^{-1} 1 \rightarrow \overline{PD} \\ \hline 1 \rightarrow \overline{PD} \\ \hline \overline{TO}, \overline{PD} \\ \hline 0 \\ \hline CLRWDT ir \\ dog Timer \\ of the WD \\ set. \\ 1 \\ 1 \\ Q1 \\ \end{bmatrix}$	CLRWD1 /DT f prescale 0000 struction r . It also rea T. Status b Q2	0110 esets the pr its TO and Q3	0100 Watch- escaler PD are Q4
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity:	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $00h \rightarrow W$ $0 \rightarrow WD^{-1} \rightarrow \overline{TO}$ $1 \rightarrow \overline{TO}$ TO, \overline{PD} 00 CLRWDT if dog Timer of the WD set. 1 1 $Q1$ Decode	CLRWD1 /DT F prescale oooo struction r . It also res T. Status b Q2 No- Operation	0110 esets the pr its TO and Q3 Process data	0100 Watch- escaler PD are Q4 Clear WDT Counter
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity: Example	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $00h \rightarrow W$ $0 \rightarrow WD^{-1} \rightarrow \overline{TO}$ $1 \rightarrow \overline{TO}$ TO, \overline{PD} 00 $CLRWDT ir dog Timer of the WD set.$ 1 1 $Q1$ $Decode$ $CLRWDT$	(DT (DT (prescale oooo istruction r It also res T. Status b Q2 No- Operation	0110 esets the pr its TO and Q3 Process data	0100 Watch- escaler PD are Q4 Clear WDT Counter
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity: Example	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $00h \rightarrow W$ $0 \rightarrow WD^{-1} \rightarrow \overline{TO}$ $1 \rightarrow \overline{TO}$ TO, \overline{PD} 00 CLRWDT if dog Timer of the WD set. 1 1 $Q1$ $CLRWDT$ Before In	CLRWD1 /DT F prescale or one struction r . It also res T. Status b Q2 No- Operation struction	0110 esets the prits TO and Q3 Process data	0100 Watch- escaler PD are Q4 Clear WDT Counter
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity: Example	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $00h \rightarrow WD$ $1 \rightarrow TO$ $1 \rightarrow PD$ TO, PD 00 CLRWDT ir dog Timer of the WD set. 1 1 $Q1$ $CLRWDT$ Before In 1	CLRWD1 /DT F prescale one struction r . It also res T. Status b Q2 No- Operation struction wDT cour	0110 esets the pr sets the pr its TO and Q3 Process data	0100 Watch- escaler PD are Q4 Clear WDT Counter
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity: Example	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $\begin{array}{c} 00h \rightarrow W \\ 0 \rightarrow WD^{-1} \\ 1 \rightarrow \overline{PD} \\ \hline TO, \overline{PD} \\ \hline 00 \\ \hline CLRWDT ir \\ dog Timer \\ of the WD \\ set. \\ 1 \\ 1 \\ \hline Q1 \\ \hline \\ CLRWDT \\ Before In \\ After Inst$	CLRWD1 /DT f prescale 0000 istruction r . It also res T. Status b Q2 No- Operation Struction wDT cour ruction	or, ollo esets the p esets the prite sets the prite and Q3 Process data	0100 Watch- escaler PD are Q4 Clear WDT Counter ?
Syntax: Operands: Operation: Status Affected: Encoding: Description: Words: Cycles: Q Cycle Activity: Example	$\begin{bmatrix} label \\ label \end{bmatrix}$ None $\begin{array}{c} 00h \rightarrow W \\ 0 \rightarrow WD^{-1} \\ 1 \rightarrow \overline{PD} \\ \hline \overline{TO}, \overline{PD} \\ \hline \overline{TO}, \overline{PD} \\ \hline 0 \\ \hline 0 \\ \hline CLRWDT ir dog Timer of the WD set. \\ 1 \\ 1 \\ \hline 2 \\ \hline 0 \\ \hline CLRWDT \\ Before In \\ After Inst$	CLRWD1 (DT F prescale 0000 Istruction r It also res T. Status b Q2 No- Operation WDT cour ruction WDT cour ruction WDT cour	OIIIO esets the prites	0100 Watch- escaler PD are Q4 Clear WDT Counter ? 0x00 0

SLEEP

Syntax:	[label]	SLEEP		
Operands:	None			
Operation:	$\begin{array}{l} 00h \rightarrow WD \\ 0 \rightarrow WD \\ 1 \rightarrow \overline{TO}, \\ 0 \rightarrow \overline{PD} \end{array}$	VDT, T presca	ler,	
Status Affected:	TO, PD			
Encoding:	0.0	0000	0110	0011
Description:	The power cleared. T set. Watch caler are The proce mode with Section 1	er-down sta Time-out st hdog Time cleared. essor is pu n the oscilla 3.8 for mo	atus bit, PI tatus bit, T er and its p t into SLE ator stopp re details.	D is TO is res- EP ed. See
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	No- Operation	No- Operation	Go to Sleep
Example:	SLEEP			

SUBLW	Subtract	W from	Literal	
Syntax:	[label]	SUBLW	/ k	
Operands:	$0 \le k \le 25$	5		
Operation:	k - (W) \rightarrow	(W)		
Status Affected:	C, DC, Z			
Encoding:	11	110x	kkkk	kkkk
Description:	The W regiment meth	ister is sub od) from th is placed i	otracted (2's ne eight bit n the W reg	s comple- literal 'k'. jister.
Words:	1			
Cycles:	1			
Q Cycle Activity:	Q1	Q2	Q3	Q4
	Decode	Read literal 'k'	Process data	Write to W
Example 1:	SUBLW	0x02		
	Before Ins	struction		
		W = C = Z =	1 ? ?	
	After Instr	ruction		
		W = C = Z =	1 1; result is 0	positive
Example 2:	Before Ins	struction		
		W = C = Z =	2 ? ?	
	After Instr	ruction		
		W = C = Z =	0 1; result i 1	s zero
Example 3:	Before Ins	struction		
		W = C = Z =	3 ? ?	
	After Instr	ruction		
		W = C = Z =	0xFF 0; result is 0	negative

-

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67

FIGURE 18-8: PARALLEL SLAVE PORT TIMING (PIC16C64A/R64)



TABLE 18-7: PARALLEL SLAVE PORT REQUIREMENTS (PIC16C64A/R64)

Parameter No.	Sym	Characteristic		Min	Тур†	Max	Units	Conditions
62	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (set	up time)	20		—	ns	
				25	_	_	ns	Extended Range Only
63*	TwrH2dtl	\overline{WR}^{\uparrow} or \overline{CS}^{\uparrow} to data–in invalid (hold	PIC16 C 64A/R64	20	I	_	ns	
		time)	PIC16 LC 64A.R64	35	_	—	ns	
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data–out valid		_	I	80	ns	
				_	_	90	ns	Extended Range Only
65*	TrdH2dtl	$\overline{\text{RD}}$ or $\overline{\text{CS}}$ to data-out invalid		10		30	ns	

These parameters are characterized but not tested.

Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not t tested.

Applicable Devices 61 62 62A R62 63 R63 64 64A R64 65 65A R65 66 67



FIGURE 20-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

FIGURE 20-5: BROWN-OUT RESET TIMING



TABLE 20-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Parameter	Sym	Characteristic	Min	Тур†	Max	Units	Conditions
No.							
30	TmcL	MCLR Pulse Width (low)	2	—	—	μs	VDD = 5V, -40°C to +125°C
31*	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +125°C
32	Tost	Oscillation Start-up Timer Period		1024 Tosc	—	—	TOSC = OSC1 period
33*	Tpwrt	Power-up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +125°C
34	Tıoz	I/O Hi-impedance from MCLR Low or WDT reset		_	2.1	μs	
35	TBOR	Brown-out Reset Pulse Width	100	_	_	μs	$VDD \le BVDD$ (D005)

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

24.2 28-Lead Plastic Dual In-line (300 mil) (SP)



		Package Gro	up: Plastic Dual	In-Line (PLA)		
		Millimeters			Inches	
Symbol	Min	Max	Notes	Min	Max	Notes
α	0°	10°		0°	10°	
A	3.632	4.572		0.143	0.180	
A1	0.381	_		0.015	_	
A2	3.175	3.556		0.125	0.140	
В	0.406	0.559		0.016	0.022	
B1	1.016	1.651	Typical	0.040	0.065	Typical
B2	0.762	1.016	4 places	0.030	0.040	4 places
B3	0.203	0.508	4 places	0.008	0.020	4 places
С	0.203	0.331	Typical	0.008	0.013	Typical
D	34.163	35.179		1.385	1.395	
D1	33.020	33.020	Reference	1.300	1.300	Reference
E	7.874	8.382		0.310	0.330	
E1	7.112	7.493		0.280	0.295	
e1	2.540	2.540	Typical	0.100	0.100	Typical
eA	7.874	7.874	Reference	0.310	0.310	Reference
eB	8.128	9.652		0.320	0.380	
L	3.175	3.683		0.125	0.145	
N	28	28		28	28	
S	0.584	1.220		0.023	0.048	



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



		Package	Group: Plastic	SOIC (SO)		
		Millimeters			Inches	
Symbol	Min	Мах	Notes	Min	Max	Notes
α	0°	8°		0°	8°	
Α	2.362	2.642		0.093	0.104	
A1	0.101	0.300		0.004	0.012	
В	0.355	0.483		0.014	0.019	
С	0.241	0.318		0.009	0.013	
D	17.703	18.085		0.697	0.712	
E	7.416	7.595		0.292	0.299	
е	1.270	1.270	Typical	0.050	0.050	Typical
Н	10.007	10.643		0.394	0.419	
h	0.381	0.762		0.015	0.030	
L	0.406	1.143		0.016	0.045	
N	28	28		28	28	
CP	-	0.102		-	0.004	

APPENDIX C: WHAT'S NEW

Added PIC16CR63 and PIC16CR65 devices.

Added PIC16C66 and PIC16C67 devices. The PIC16C66/67 devices have 368 bytes of data memory distributed in 4 banks and 8K of program memory in 4 pages. These two devices have an enhanced SPI that supports both clock phase and polarity. The USART has been enhanced.

When upgrading to the PIC16C66/67 please note that the upper 16 bytes of data memory in banks 1,2, and 3 are mapped into bank 0. This may require relocation of data memory usage in the user application code.

Q-cycles for instruction execution were added to Section 14.0 Instruction Set Summary.

APPENDIX D: WHAT'S CHANGED

Minor changes, spelling and grammatical changes.

Divided SPI section into SPI for the PIC16C66/67 (Section 11.3) and SPI for all other devices (Section 11.2).

Added the following note for the USART. This applies to all devices except the PIC16C66 and PIC16C67.

For the PIC16C63/R63/65/65A/R65 the asynchronous high speed mode (BRGH = 1) may experience a high rate of receive errors. It is recommended that BRGH = 0. If you desire a higher baud rate than BRGH = 0 can support, refer to the device errata for additional information or use the PIC16C66/67.

APPENDIX E: REVISION E

January 2013 - Added a note to each package drawing.

negisiers	•				
CCF	P1CON				
	Diagram				78
	Section	24	26 28	30	78
CCF	2CON		20, 20	, 00,	52
	Diagram				78
	Section				78
	Summary			, 30,	32
CCF	'R1H	04	00.00	~~	~~
CCE	Summary PB1I		26, 28	, 30,	32
001	Summary		26. 28	. 30.	32
CCF	°R2H	,	-, -	, ,	
	Summary			, 30,	32
CCF	PR2L				
E0D	Summary			, 30,	32
FSH	Indirect Addressing				<u>1</u> 0
	Summary	.24.26.	28.30	. 32.	34
IND	F	,,	,	,,	
	Indirect Addressing				49
	Summary	. 24, 26,	28, 30	, 32,	34
INTO	CON				07
	Section				37
	Summary	24 26	28 30	32	34
OPT	TION		20, 00	, 02,	• •
	Diagram				36
	Section				36
	Summary	. 25, 27,	29, 31	, 33,	34
PCL	Castian				40
	Section				48
	Cummon.	04 06	00 00		04
PCI	Summary	. 24, 26,	28, 30	, 32,	34
PCL	Summary ATH Section	24, 26,	28, 30	, 32,	34 48
PCL	Summary ATH Section Summary	24, 26, 24, 26,	28, 30 28, 30	, 32, , 32,	34 48 34
PCL	Summary ATH Section Summary DN	24, 26, 24, 26,	28, 30 28, 30	, 32, , 32,	34 48 34
PCL PCC	Summary ATH Section Summary N Diagram	24, 26, 24, 26,	28, 30 28, 30	, 32, , 32,	34 48 34 47
PCL PCC	Summary ATH Section Summary N Diagram Section	24, 26, 24, 26,	28, 30 28, 30	, 32, , 32,	 34 48 34 47 47 47 20
PCL	Summary ATH Section Summary N Diagram Section Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29	, 32, , 32, , 32,	34 48 34 47 47 33
PCL PCC PIE ⁻	Summary ATH Section Summary N Diagram Summary Diagram	24, 26, 24, 26, 	28, 30 28, 30 27, 29	, 32, , 32, , 31,	 34 48 34 47 47 33 40
PCL PCC PIE ⁻	Summary ATH Section Summary N Diagram Summary Diagram Section	. 24, 26, 24, 26, 	28, 30 28, 30 27, 29	, 32, , 32, , 31,	 34 48 34 47 47 33 40 38
PCL PCC PIE ⁻	Summary ATH Section Summary N Diagram Section Diagram Section Section Summary	. 24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29	, 32, , 32, , 31, , 31,	 34 48 34 47 47 33 40 38 33
PCL PCC PIE ⁻ PIE2	Summary ATH Section Summary N Diagram Section Summary Section Section Section Section Summary	. 24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29	, 32, , 32, , 31, , 31,	34 48 34 47 47 33 40 38 33
PCL PCC PIE ⁻ PIE2	SummaryATH SectionSummary DiagramSectionSummary DiagramSectionSummary SectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionS	. 24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29	, 32, , 32, , 31, , 31,	34 48 34 47 47 33 40 38 33 45
PCL PCC PIE	Summary ATH Section Summary Diagram Section Summary Diagram Summary Diagram 2 Diagram Section Section	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29	, 32, , 32, , 31, , 31,	 34 48 34 47 47 33 40 38 33 45 45 45 22
PCL PCC PIE	Summary ATH Section Summary Diagram Summary Diagram Summary Diagram Diagram Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29	, 32, , 32, , 31, , 31, , 31,	34 48 34 47 47 33 40 38 33 45 45 33
PCL PCC PIE PIE	Summary ATH Section Summary Diagram Summary Diagram Diagram Diagram Summary Diagram Diagram Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 27, 29	, 32, , 32, , 31, , 31, , 31,	 34 48 34 47 47 33 40 38 33 45 45 33 45 45 33 44
PCL PCC PIE PIE	Summary ATH Section	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29	, 32, , 32, , 31, , 31, , 31,	 34 48 34 47 47 33 40 38 33 45 45 33 45 33 44 41
PCL PCC PIE PIE	SummaryATH SectionSummary Summary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 27, 29 26, 28	, 32, , 32, , 31, , 31, , 31, , 31,	 34 48 34 47 47 33 40 38 33 45 45 33 45 45 33 44 41 32
PCL PCC PIE PIE	Summary ATH Section	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 27, 29 26, 28	, 32, , 32, , 31, , 31, , 31, , 31,	34 48 34 47 33 40 38 33 45 45 33 45 33 44 41 32
PCC PIE PIE PIR	SummaryATH SectionSummary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary SectionSummary SectionSummary Summary DiagramSectionSummary Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28	, 32, , 32, , 31, , 31, , 31, , 31,	 34 48 34 47 33 40 38 33 45 45 33 45 45 33 44 41 32 46
PCC PIE PIE PIR	SummaryATH SectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSummarySectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSectionSec	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28	, 32, , 32, , 31, , 31, , 31, , 31, , 33,	34 48 34 47 47 33 40 38 33 45 45 33 44 41 32 46 46 20
PCC PIE PIE PIR	SummaryATH SectionSummary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary SectionSummary DiagramSectionSummary SummarySummary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 31, , 30,	34 48 34 47 47 33 40 38 33 45 45 33 44 41 32 46 46 32
PCC PIE PIE PIR PIR	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 31, , 30,	34 48 34 47 47 33 40 38 33 45 45 33 44 41 32 46 46 32 51
PCC PIE PIE PIR PIR	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 31, , 30, , 30,	 34 48 34 47 47 33 40 38 33 45 45 33 45 45 33 45 46 46 32 51 32
PCL PCC PIE PIE PIR PIR POF	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 30, , 30,	34 48 34 47 33 40 38 33 45 45 33 45 45 33 44 41 32 46 46 32 51 32
PCL PCC PIE PIE PIR PIR	SummaryATH SectionSummary DiagramSectionSummary Summary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary DiagramSectionSummary SectionSummary SectionSummary SectionSummary TA SectionSummary Summary Summary Summary Summary TB Section	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 30, , 30,	 34 48 34 47 47 33 40 33 45 33 45 33 45 33 45 33 45 33 45 33 46 46 32 51 32 53
PCL PCC PIE PIE PIR PIR	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28 26, 28 28, 30	, 32, , 32, , 31, , 31, , 31, , 31, , 30, , 30, , 30,	34 48 34 47 47 33 40 38 33 45 45 33 45 45 33 44 41 32 46 46 32 51 32 53 34
PCL PCC PIE PIE PIR PIR POF	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28 26, 28 28, 30	, 32, , 32, , 31, , 31, , 31, , 31, , 30, , 30, , 30,	34 48 34 47 33 40 33 40 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 44 32 53 34 53 54 53 54 55
PCL PCC PIE PIE PIR PIR POF POF	Summary	24, 26, 24, 26, 	28, 30 28, 30 27, 29 27, 29 27, 29 26, 28 26, 28 26, 28 26, 28 26, 28 26, 28 26, 28 26, 28	, 32, , 32, , 31, , 31, , 31, , 31, , 30, , 30, , 30, , 30,	34 48 34 47 33 40 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 45 33 46 32 53 34 55 32 53 53 53 53 53 53 53 53 53 53 53 53 53 54

DODTD
PORID
Section57
Summary
PORTE
Section
Summary
PB2
Summany 25 27 20 31 33
Summary 25, 27, 29, 31, 30
RUREG
Summary 26, 30, 32
RCSTA
Diagram 106
Summary
SPBBG
Summary
SSPBUF
Section 86
Summary 24, 26, 28, 30, 32
SSPCON
Diagram 85
Summany 24 26 28 30 30
Summary 24, 20, 20, 30, 32
SSPSR
Section 86
SSPSTAT 89
Diagram
Section 84
Summary 05 07 00 21 23
OTATIO
STATUS
Diagram 35
Section 35
Summary
T1CON
Diagram 71
Diagram
Section
Summary
T2CON
Diagram 75
Section
Summary 24 26 28 30 32
TMP0
Summary 24, 26, 28, 30, 32, 34
TMR1H
Summary 24, 26, 28, 30, 32
TMR1L
Summary
TMB2 7F
Summony 04 06 09 20 20
Summary 24, 20, 20, 30, 32
IRISA
Section51
Summary 25, 27, 29, 31, 33
TRISB
Section 53
Summany 05 07 00 31 33 3/
TDICO
TRISC
Section
Summary 25, 27, 29, 31, 33
TRISD
Section
Summary 20 31 33
TRISE
Diagram
Diagram
Section
Summary 00.01.00
Summary 29, 31, 33
TXREG

LIST OF EQUATION AND EXAMPLES

Example 3-1:	Instruction Pipeline Flow
Example 4-1:	Call of a Subroutine in Page 1
	from Page 0
Example 4-2:	Indirect Addressing
Example 5-1:	Initializing PORTA
Example 5-2:	Initializing PORTB53
Example 5-3:	Initializing PORTC55
Example 5-4:	Read-Modify-Write Instructions on an
	I/O Port60
Example 7-1:	Changing Prescaler (Timer0→WDT)69
Example 7-2:	Changing Prescaler (WDT→Timer0)69
Example 8-1:	Reading a 16-bit
	Free-running Timer73
Example 10-1:	Changing Between
	Capture Prescalers79
Example 10-2:	PWM Period and Duty
	Cycle Calculation81
Example 11-1:	Loading the SSPBUF
	(SSPSR) Register86
Example 11-2:	Loading the SSPBUF
	(SSPSR) Register (PIC16C66/67)91
Example 12-1:	Calculating Baud Rate Error107
Example 13-1:	Saving Status and W
	Registers in RAM139
Example 13-2:	Saving Status, W, and
	PCLATH Registers in RAM
	(All other PIC16C6X devices) 139

LIST OF FIGURES

Figure 3-1:	PIC16C61 Block Diagram	10
Figure 3-2:	PIC16C62/62A/R62/64/64A/R64	
	Block Diagram	11
Figure 3-3:	PIC16C63/R63/65/65A/R65	
	Block Diagram	12
Figure 3-4:	PIC16C66/67 Block Diagram	13
Figure 3-5:	Clock/Instruction Cycle	18
Figure 4-1:	PIC16C61 Program Memory Map	
	and Stack	19
Figure 4-2:	PIC16C62/62A/R62/64/64A/	
	R64 Program Memory Map and Stack	19
Figure 4-3:	PIC16C63/R63/65/65A/R65 Program	
	Memory Map and Stack	19
Figure 4-4:	PIC16C66/67 Program Memory	
	Map and Stack	20
Figure 4-5:	PIC16C61 Register File Map	20
Figure 4-6:	PIC16C62/62A/R62/64/64A/	
	R64 Register File Map	21
Figure 4-7:	PIC16C63/R63/65/65A/R65	
	Register File Map	21
Figure 4-8:	PIC16C66/67 Data Memory Map	22
Figure 4-9:	STATUS Register	
	(Address 03h, 83h, 103h, 183h)	35
Figure 4-10:	OPTION Register	
	(Address 81h, 181h)	36
Figure 4-11:	INTCON Register	
	(Address 0Bh, 8Bh, 10Bh 18Bh)	37
Figure 4-12:	PIE1 Register for PIC16C62/62A/R62	
	(Address 8Ch)	38
Figure 4-13:	PIE1 Register for PIC16C63/R63/66	
	(Address 8Ch)	39
Figure 4-14:	PIE1 Register for PIC16C64/64A/R64	
	(Address 8Ch)	39

1 igule 4-15.	PIE1 Register for PIC16C65/65A/R65/67
Figure 4-16:	(Address 8Ch) 40 PIR1 Register for PIC16C62/62A/R62
	(Address 0Ch)
Figure 4-17:	PIR1 Register for PIC16C63/R63/66 Address 0Cb) 42
Figure 4-18:	PIR1 Register for PIC16C64/64A/R64
Figure 4-19:	PIR1 Register for PIC16C65/65A/R65/67
0	(Address 0Ch) 44
Figure 4-20:	PIE2 Register (Address 8Dh) 45
Figure 4-21:	PIR2 Register (Address 0Dh) 46
Figure 4-22:	PCON Register for PIC16C62/64/65
Figure 4-23	(Address 8En)
Figure 4-23.	R63/644/R64/654/R65/66/67
	(Address 8Eh) 47
Figure 4-24:	Loading of PC in Different Situations
Figure 4-25	Direct/Indirect Addressing 49
Figure 5-1	Block Diagram of the
i iguio o ii	BA3:BA0 Pins and the BA5 Pin 51
Figure 5-2	Block Diagram of the BA4/T0CKI Pin 51
Figure 5-3	Block Diagram of the
. iguio o oi	BB7 BB4 Pins for PIC16C61/62/64/65 53
Figure 5-4	Block Diagram of the
i iguio o il	BB7 BB4 Pins for PIC16C62A/63/B63/
	64A/65A/B65/66/67 54
Figure 5-5:	Block Diagram of the
	BB3:BB0 Pins
Figure 5-6:	PORTC Block Diagram
Figure 5-7:	PORTD Block Diagram
J	(In I/O Port Mode)
Figure 5-8:	PORTE Block Diagram
	(In I/O Port Mode) 58
Figure 5-9:	TRISE Register (Address 89h) 58
Figure 5-10:	Successive I/O Operation 60
Figure 5-11:	PORTD and PORTE as a Parallel
	Slave Port 61
Figure 5-12:	Parallel Slave Port Write Waveforms 62
Figure 5-13:	Parallel Slave Port Read Waveforms 62
Figure 7-1:	
	Timer0 Block Diagram 65
Figure 7-2:	Timer0 Block Diagram
Figure 7-2:	Timer0 Block Diagram
Figure 7-2: Figure 7-3:	Timer0 Block Diagram
Figure 7-2: Figure 7-3:	Timer0 Block Diagram
Figure 7-2: Figure 7-3: Figure 7-4:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 65 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing: Clock/Prescale 1:2 66 Timer0 Timing: Clock/Prescale 1:2 66
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Plack Descent of the Timer0 AUDT 67
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 60 65 Clock/Prescale 1:2 66 66 TMR0 Interrupt Timing. 66 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 68 Descention 67
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 9 1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal Clock/Prescale 1:2 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Bosister 68
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10b) 71
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer0 Timigram 71
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Eigure 8-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Register 71 (Address 10h) 71 Timer1 Block Diagram 72
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register 75
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register 64 (Address 12h) 75
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 68 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register (Address 12h) 75 CCP1CON Begister (Address 12h)
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 68 Prescaler 68 T1CON: Timer1 Control Register 71 (Address 10h) 71 Timer2 Block Diagram 75 T2CON: Timer2 Control Register (Address 12h) 75 CCP1CON Register (Address 17h) / CCP2CON Begister (Address 17h) / 78
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer2 Block Diagram 75 T2CON: Timer2 Control Register (Address 12h) (Address 12h) 75 CCP1CON Register (Address 17h) / 78 Capture Mode Operation 78
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 8-2: Figure 9-2: Figure 10-1: Figure 10-2:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 72 Timer2 Block Diagram 75 CCP1CON Register (Address 17h) / 78 Capture Mode Operation 80 Block Diagram 78
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register 74 (Address 12h) 75 CCP1CON Register (Address 17h) / 78 Capture Mode Operation 8 Block Diagram 78 Compare Mode Operation 78
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 65 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register 64 (Address 12h) 75 CCP1CON Register (Address 17h) / 75 CCP2CON Register (Address 10h) 78 Capture Mode Operation 8 Block Diagram 78 Compare Mode Operation 79 Block Diagram 79
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3: Figure 10-4:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 665 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register 64 (Address 12h) 75 CCP1CON Register (Address 17h) / 75 Capture Mode Operation 80 Block Diagram 78 Compare Mode Operation 80 Block Diagram 79 Simplified PWM Block Diagram 80
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 8-1: Figure 8-2: Figure 9-2: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3: Figure 10-4: Figure 10-5:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 68 Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer2 Block Diagram 72 Timer2 Block Diagram 75 T2CON: Timer2 Control Register (Address 12h) (Address 12h) 75 CCP1CON Register (Address 17h) / 76 Capture Mode Operation 78 Block Diagram 78 Compare Mode Operation 79 Block Diagram 79 Simplified PWM Block Diagram 80 PWM Output 80
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 8-1: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3: Figure 10-4: Figure 10-5: Figure 11-1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT Prescaler Prescaler 68 T1CON: Timer1 Control Register 64 (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 CCP1CON Register (Address 17h) / 75 CCP2CON Register (Address 1Dh) 78 Compare Mode Operation 80 Block Diagram 79 Simplified PWM Block Diagram 80 PWM Output 80 SSPSTAT: Sync Serial Port Status 80
Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6: Figure 8-1: Figure 8-1: Figure 8-2: Figure 9-1: Figure 9-2: Figure 10-1: Figure 10-2: Figure 10-3: Figure 10-4: Figure 10-5: Figure 11-1:	Timer0 Block Diagram 65 Timer0 Timing: Internal Clock/No Prescaler 65 Timer0 Timing: Internal 66 Clock/Prescale 1:2 66 TMR0 Interrupt Timing. 66 Timer0 Timing With External Clock 67 Block Diagram of the Timer0/WDT 7 Prescaler 68 T1CON: Timer1 Control Register (Address 10h) (Address 10h) 71 Timer1 Block Diagram 72 Timer2 Block Diagram 75 CCP1CON Register (Address 17h) / 75 CCP2CON Register (Address 10h) 78 Capture Mode Operation 80 Block Diagram 79 Simplified PWM Block Diagram 80 PWM Output 80 SSPSTAT: Sync Serial Port Status 84