

Welcome to E-XFL.COM

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

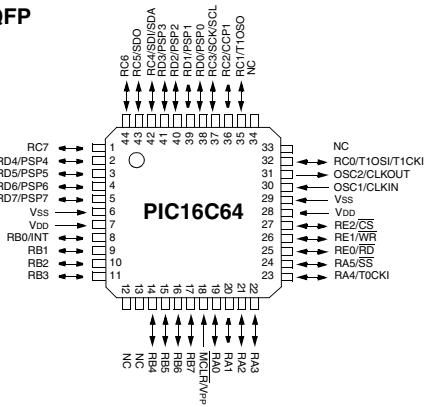
Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	22
Program Memory Size	14KB (8K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c66-04-sp

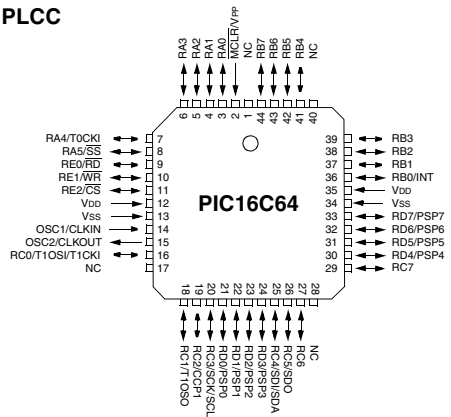
PIC16C6X

Pin Diagrams (Cont'd)

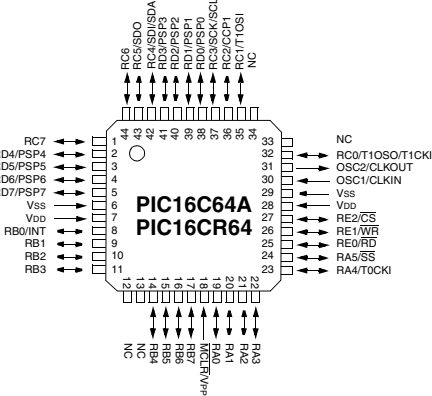
MQFP



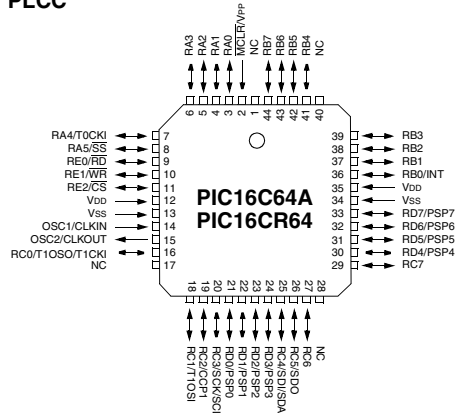
PLCC



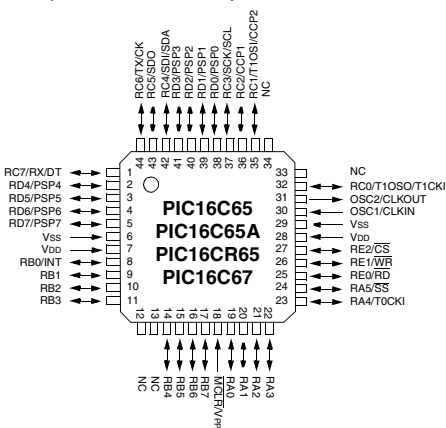
MQFP, TQFP (PIC16C64A only)



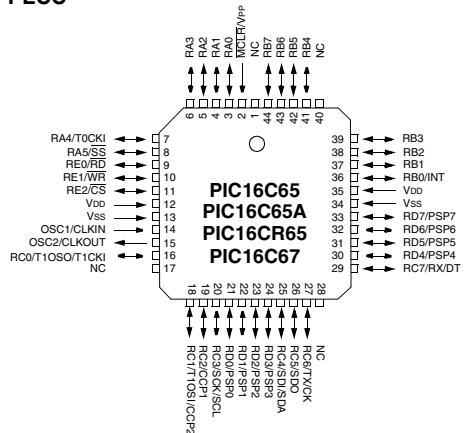
PLCC



MQFP, TQFP (Not on PIC16C65)

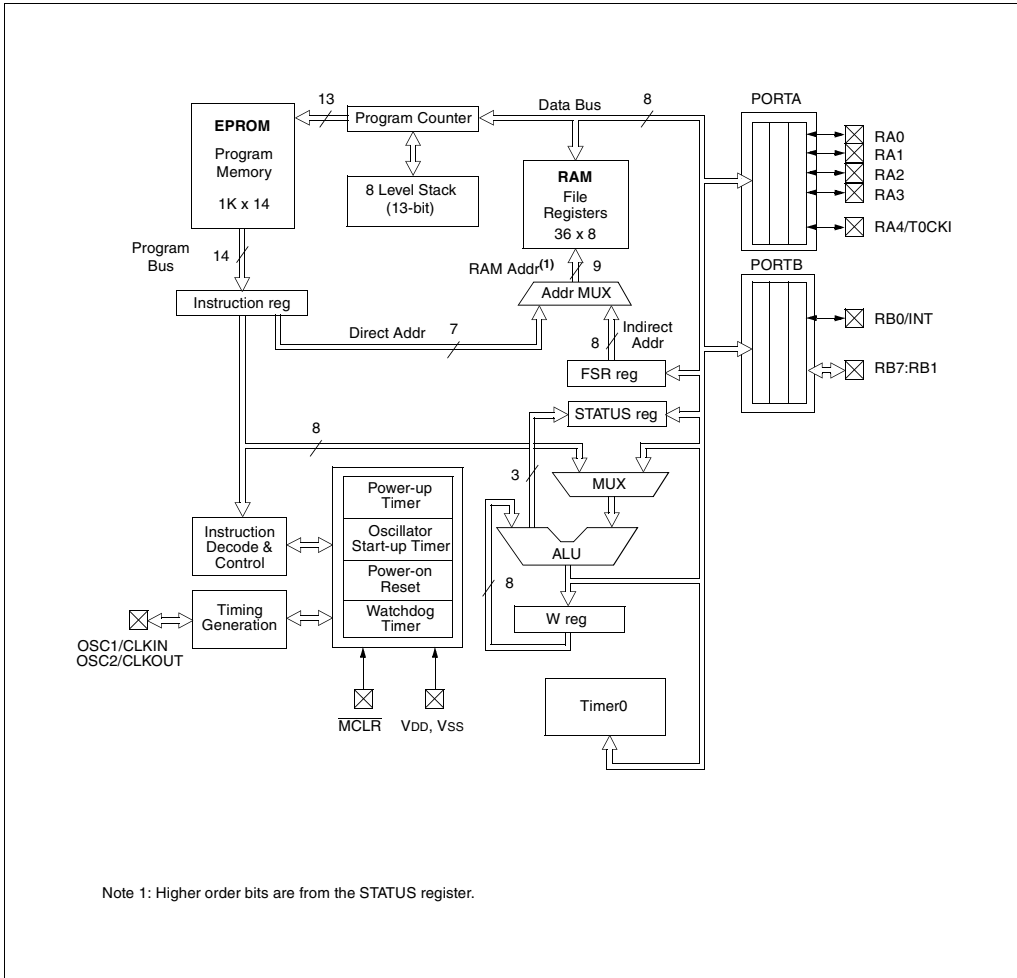


PLCC



PIC16C6X

FIGURE 3-1: PIC16C61 BLOCK DIAGRAM



PIC16C6X

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
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
-n = Value at POR reset

bit 7: **PSPIE**: Parallel Slave Port Read/Write Interrupt Enable bit
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

bit 6: **Reserved**: Always maintain this bit clear.

bit 5: **RCIE**: USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Disables the USART receive interrupt

bit 4: **TXIE**: USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3: **SSPIE**: Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2: **CCP1IE**: CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE**: TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE**: TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

4.2.2.5 PIR1 REGISTER

Applicable Devices

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

This register contains the individual flag bits for the peripheral interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

FIGURE 4-16: PIR1 REGISTER FOR PIC16C62/62A/R62 (ADDRESS 0Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7				bit0			
<p>bit 7-6: Reserved: Always maintain these bits clear.</p> <p>bit 5-4: Unimplemented: Read as '0'</p> <p>bit 3: SSPIF: Synchronous Serial Port Interrupt Flag bit 1 = The transmission/reception is complete (must be cleared in software) 0 = Waiting to transmit/receive</p> <p>bit 2: CCP1IF: CCP1 Interrupt Flag bit <u>Capture Mode</u> 1 = A TMR1 register capture occurred (must be cleared in software) 0 = No TMR1 register capture occurred <u>Compare Mode</u> 1 = A TMR1 register compare match occurred (must be cleared in software) 0 = No TMR1 register compare match occurred <u>PWM Mode</u> Unused in this mode</p> <p>bit 1: TMR2IF: TMR2 to PR2 Match Interrupt Flag bit 1 = TMR2 to PR2 match occurred (must be cleared in software) 0 = No TMR2 to PR2 match occurred</p> <p>bit 0: TMR1IF: TMR1 Overflow Interrupt Flag bit 1 = TMR1 register overflow occurred (must be cleared in software) 0 = No TMR1 register overflow occurred</p>							
<p>R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset</p>							
<p>Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.</p>							

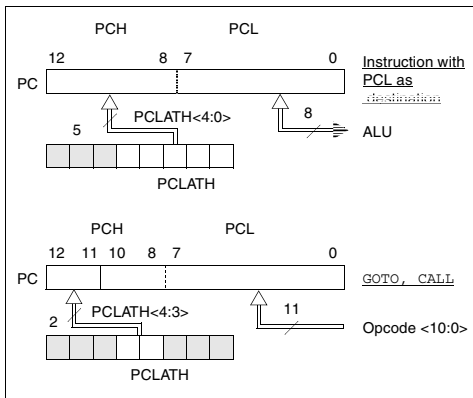
PIC16C6X

4.3 PCL and PCLATH

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

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The upper bits (PC<12:8>) are not readable, but are indirectly writable through the PCLATH register. On any reset, the upper bits of the PC will be cleared. Figure 4-24 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 4-24: LOADING OF PC IN DIFFERENT SITUATIONS



4.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (`ADDWF PCL`). When doing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256 word block). Refer to the application note "Implementing a Table Read" (AN556).

4.3.2 STACK

The PIC16CXX family has an 8 deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or a POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1: There are no status bits to indicate stack overflows or stack underflow conditions.

Note 2: There are no instructions mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW, and RETFIE instructions, or the vectoring to an interrupt address

4.4 Program Memory Paging

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

PIC16C6X devices are capable of addressing a continuous 8K word block of program memory. The CALL and GOTO instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction the upper two bits of the address are provided by PCLATH<4:3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is pushed onto the stack. Therefore, manipulation of the return instructions (which POPs the address from the stack).

Note: PIC16C6X devices with 4K or less of program memory ignore paging bit PCLATH<4>. The use of PCLATH<4> as a general purpose read/write bit is not recommended since this may affect upward compatibility with future products.

PIC16C6X

NOTES:

PIC16C6X

TABLE 5-1: PORTA FUNCTIONS

Name	Bit#	Buffer Type	Function
RA0	bit0	TTL	Input/output
RA1	bit1	TTL	Input/output
RA2	bit2	TTL	Input/output
RA3	bit3	TTL	Input/output
RA4/T0CKI	bit4	ST	Input/output or external clock input for Timer0. Output is open drain type.
RA5/ \overline{SS} (1)	bit5	TTL	Input/output or slave select input for synchronous serial port.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: The PIC16C61 does not have PORTA<5> or TRISA<5>, read as '0'.

TABLE 5-2: REGISTERS/BITS ASSOCIATED WITH PORTA

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
05h	PORTA	—	—	RA5 ⁽¹⁾	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
85h	TRISA	—	—	PORTA Data Direction Register ⁽¹⁾						--11 1111	--11 1111

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

Note 1: PORTA<5> and TRISA<5> are not implemented on the PIC16C61, read as '0'.

PIC16C6X

FIGURE 13-2: CONFIGURATION WORD FOR PIC16C62/64/65

—	—	—	—	—	—	—	—	—	CP1	CP0	PWRTE	WDTE	FOSC1	FOSC0	bit0	Register: CONFIG
bit13															Address: 2007h	
bit 13-6: Unimplemented: Read as '1'																
bit 5-4: CP1:CP0: Code Protection bits																
11 = Code protection off																
10 = Upper half of program memory code protected																
01 = Upper 3/4th of program memory code protected																
00 = All memory is code protected																
bit 3: PWRTE: Power-up Timer Enable bit																
1 = Power-up Timer enabled																
0 = Power-up Timer disabled																
bit 2: WDTE: Watchdog Timer Enable bit																
1 = WDT enabled																
0 = WDT disabled																
bit 1-0: FOSC1:FOSC0: Oscillator Selection bits																
11 = RC oscillator																
10 = HS oscillator																
01 = XT oscillator																
00 = LP oscillator																

FIGURE 13-3: CONFIGURATION WORD FOR PIC16C62A/R62/63/R63/64A/R64/65A/R65/66/67

CP1	CP0	CP1	CP0	CP1	CP0	—	BODEN	CP1	CP0	PWRTE	WDTE	FOSC1	FOSC0	bit0	Register: CONFIG
bit13															Address: 2007h
bit 13-8: CP1:CP0: Code Protection bits ⁽²⁾															
bit 5-4: 11 = Code protection off															
10 = Upper half of program memory code protected															
01 = Upper 3/4th of program memory code protected															
00 = All memory is code protected															
bit 7: Unimplemented: Read as '1'															
bit 6: BODEN: Brown-out Reset Enable bit ⁽¹⁾															
1 = Brown-out Reset enabled															
0 = Brown-out Reset disabled															
bit 3: PWRTE: Power-up Timer Enable bit ⁽¹⁾															
1 = Power-up Timer disabled															
0 = Power-up Timer enabled															
bit 2: WDTE: Watchdog Timer Enable bit															
1 = WDT enabled															
0 = WDT disabled															
bit 1-0: FOSC1:FOSC0: Oscillator Selection bits															
11 = RC oscillator															
10 = HS oscillator															
01 = XT oscillator															
00 = LP oscillator															
Note 1: Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit PWRTE . Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.															
Note 2: All of the CP1:CP0 pairs have to be given the same value to implement the code protection scheme listed.															

TABLE 13-9: STATUS BITS AND THEIR SIGNIFICANCE FOR PIC16C62A/R62/63/R63/64A/R64/65A/R65/66/67

POR	BOR	\overline{TO}	PD	
0	x	1	1	Power-on Reset
0	x	0	x	Illegal, \overline{TO} is set on a Power-on Reset
0	x	x	0	Illegal, PD is set on a Power-on Reset
1	0	x	x	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake-up
1	1	u	u	MCLR reset during normal operation
1	1	1	0	MCLR reset during SLEEP or interrupt wake-up from SLEEP

Legend: x = unknown, u = unchanged

TABLE 13-10: RESET CONDITION FOR SPECIAL REGISTERS ON PIC16C61/62/64/65

	Program Counter	STATUS	PCON ⁽²⁾
Power-on Reset	000h	0001 1xxx	---- --0-
MCLR reset during normal operation	000h	000u uuuu	---- --u-
MCLR reset during SLEEP	000h	0001 0uuu	---- --u-
WDT Reset	000h	0000 1uuu	---- --u-
WDT Wake-up	PC + 1	uuu0 0uuu	---- --u-
Interrupt wake-up from SLEEP	PC + 1 ⁽¹⁾	uuu1 0uuu	---- --u-

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

Note 1: When the wake-up is due to an interrupt and the global enable bit, GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC+1.

2: The PCON register is not implemented on the PIC16C61.

TABLE 13-11: RESET CONDITION FOR SPECIAL REGISTERS ON PIC16C62A/R62/63/R63/64A/R64/65A/R65/66/67

	Program Counter	STATUS	PCON
Power-on Reset	000h	0001 1xxx	---- --0x
MCLR reset during normal operation	000h	000u uuuu	---- --uu
MCLR reset during SLEEP	000h	0001 0uuu	---- --uu
WDT Reset	000h	0000 1uuu	---- --uu
Brown-out Reset	000h	0001 1uuu	---- --u0
WDT Wake-up	PC + 1	uuu0 0uuu	---- --uu
Interrupt wake-up from SLEEP	PC + 1 ⁽¹⁾	uuu1 0uuu	---- --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

Note 1: When the wake-up is due to an interrupt and global enable bit, GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC+1.

13.6 Context Saving During Interrupts

Applicable Devices

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

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt i.e., W register and STATUS register. This will have to be implemented in software.

Example 13-1 stores and restores the STATUS and W registers. Example 13-2 stores and restores the STATUS, W, and PCLATH registers (Devices with paged program memory). For all PIC16C6X devices with greater than 1K of program memory (all devices except PIC16C61), the register, W_TEMP, must be

defined in banks and must be defined at the same offset from the bank base address (i.e., if W_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1, 0x120 in bank 2, and 0x1A0 in bank 3).

The examples:

- Stores the W register
- Stores the STATUS register in bank 0
- Stores PCLATH
- Executes ISR code
- Restores PCLATH
- Restores STATUS register (and bank select bit)
- Restores W register

EXAMPLE 13-1: SAVING STATUS AND W REGISTERS IN RAM (PIC16C61)

```

MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF   STATUS,W          ;Swap status to be saved into W
MOVWF   STATUS_TEMP       ;Save status to bank zero STATUS_TEMP register
:
:(ISR)
:
SWAPF   STATUS_TEMP,W     ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)

MOVWF   STATUS            ;Move W into STATUS register
SWAPF   W_TEMP,F         ;Swap W_TEMP
SWAPF   W_TEMP,W         ;Swap W_TEMP into W
    
```

EXAMPLE 13-2: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM (ALL OTHER PIC16C6X DEVICES)

```

MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF   STATUS,W          ;Swap status to be saved into W
CLRF    STATUS            ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF   STATUS_TEMP       ;Save status to bank zero STATUS_TEMP register
MOVF    PCLATH, W        ;Only required if using pages 1, 2 and/or 3
MOVWF   PCLATH_TEMP      ;Save PCLATH into W
CLRF    PCLATH           ;Page zero, regardless of current page
BCF     STATUS, IRP      ;Return to Bank 0
MOVF    FSR, W           ;Copy FSR to W
MOVWF   FSR_TEMP         ;Copy FSR from W to FSR_TEMP
:(ISR)
:
MOVF    PCLATH_TEMP, W   ;Restore PCLATH
MOVWF   PCLATH           ;Move W into PCLATH
SWAPF   STATUS_TEMP,W     ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)

MOVWF   STATUS            ;Move W into STATUS register
SWAPF   W_TEMP,F         ;Swap W_TEMP
SWAPF   W_TEMP,W         ;Swap W_TEMP into W
    
```

PIC16C6X

IORWF **Inclusive OR W with f**

Syntax: [*label*] IORWF *f*,*d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: (W) .OR. (f) → (destination)

Status Affected: Z

Encoding:

00	0100	dfff	ffff
----	------	------	------

Description: Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to destination

Example `IORWF RESULT, 0`

Before Instruction
 RESULT = 0x13
 W = 0x91

After Instruction
 RESULT = 0x13
 W = 0x93
 Z = 1

MOVF **Move f**

Syntax: [*label*] MOVF *f*,*d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: (f) → (destination)

Status Affected: Z

Encoding:

00	1000	dfff	ffff
----	------	------	------

Description: The contents of register *f* is moved to a destination dependant upon the status of *d*. If *d* = 0, destination is W register. If *d* = 1, the destination is file register *f* itself. *d* = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write to destination

Example `MOVF FSR, 0`

After Instruction
 W = value in FSR register
 Z = 1

MOVLW **Move Literal to W**

Syntax: [*label*] MOVLW *k*

Operands: $0 \leq k \leq 255$

Operation: $k \rightarrow (W)$

Status Affected: None

Encoding:

11	00xx	kkkk	kkkk
----	------	------	------

Description: The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process data	Write to W

Example `MOVLW 0x5A`

After Instruction
 W = 0x5A

MOVWF **Move W to f**

Syntax: [*label*] MOVWF *f*

Operands: $0 \leq f \leq 127$

Operation: (W) → (f)

Status Affected: None

Encoding:

00	0000	1fff	ffff
----	------	------	------

Description: Move data from W register to register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write register 'f'

Example `MOVWF OPTION_REG`

Before Instruction
 OPTION = 0xFF
 W = 0x4F

After Instruction
 OPTION = 0x4F
 W = 0x4F

15.0 ELECTRICAL CHARACTERISTICS FOR PIC16C61

Absolute Maximum Ratings †

Ambient temperature under bias.....	-55°C to +125°C
Storage temperature.....	-65°C to +150°C
Voltage on any pin with respect to V _{SS} (except V _{DD} , $\overline{\text{MCLR}}$, and RA4).....	-0.3V to (V _{DD} + 0.3V)
Voltage on V _{DD} with respect to V _{SS}	-0.3V to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to V _{SS} (Note 2).....	0V to +14V
Voltage on RA4 pin with respect to V _{SS}	0V to +14V
Total power dissipation (Note 1).....	800 mW
Maximum current out of V _{SS} pin.....	150 mA
Maximum current into V _{DD} pin.....	100 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > V _{DD}).....	± 20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > V _{DD}).....	± 20 mA
Maximum output current sunk by any I/O pin.....	25 mA
Maximum output current sourced by any I/O pin.....	20 mA
Maximum current sunk by PORTA.....	80 mA
Maximum current sourced by PORTA.....	50 mA
Maximum current sunk by PORTB.....	150 mA
Maximum current sourced by PORTB.....	100 mA

Note 1: Power dissipation is calculated as follows: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

Note 2: Voltage spikes below V_{SS} at the $\overline{\text{MCLR}}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the $\overline{\text{MCLR}}$ pin rather than pulling this pin directly to V_{SS}.

† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

TABLE 15-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

OSC	PIC16C61-04	PIC16C61-20	PIC16LC61-04	JW Devices
RC	V _{DD} : 4.0V to 6.0V I _{DD} : 3.3 mA max. at 5.5V I _{PD} : 14 μA max. at 4V Freq: 4 MHz max.	V _{DD} : 4.5V to 5.5V I _{DD} : 1.8 mA typ. at 5.5V I _{PD} : 1.0 μA typ. at 4V Freq: 4 MHz max.	V _{DD} : 3.0V to 6.0V I _{DD} : 1.4 mA typ. at 3.0V I _{PD} : 0.6 μA typ. at 3V Freq: 4 MHz max.	V _{DD} : 4.0V to 6.0V I _{DD} : 3.3 mA max. at 5.5V I _{PD} : 14 μA max. at 4V Freq: 4 MHz max.
XT	V _{DD} : 4.0V to 6.0V I _{DD} : 3.3 mA max. at 5.5V I _{PD} : 14 μA max. at 4V Freq: 4 MHz max.	V _{DD} : 4.5V to 5.5V I _{DD} : 1.8 mA typ. at 5.5V I _{PD} : 1.0 μA typ. at 4V Freq: 4 MHz max.	V _{DD} : 3.0V to 6.0V I _{DD} : 1.4 mA typ. at 3.0V I _{PD} : 0.6 μA typ. at 3V Freq: 4 MHz max.	V _{DD} : 4.0V to 6.0V I _{DD} : 3.3 mA max. at 5.5V I _{PD} : 14 μA max. at 4V Freq: 4 MHz max.
HS	V _{DD} : 4.5V to 5.5V I _{DD} : 13.5 mA typ. at 5.5V I _{PD} : 1.0 μA typ. at 4.5V Freq: 4 MHz max.	V _{DD} : 4.5V to 5.5V I _{DD} : 30 mA max. at 5.5V I _{PD} : 1.0 μA typ. at 4.5V Freq: 20 MHz max.	Not recommended for use in HS mode	V _{DD} : 4.5V to 5.5V I _{DD} : 30 mA max. at 5.5V I _{PD} : 1.0 μA typ. at 4.5V Freq: 20 MHz max.
LP	V _{DD} : 4.0V to 6.0V I _{DD} : 15 μA typ. at 32 kHz, 4.0V I _{PD} : 0.6 μA typ. at 4.0V Freq: 200 kHz max.	Not recommended for use in LP mode	V _{DD} : 3.0V to 6.0V I _{DD} : 32 μA max. at 32 kHz, 3.0V I _{PD} : 9 μA max. at 3.0V Freq: 200 kHz max.	V _{DD} : 3.0V to 6.0V I _{DD} : 32 μA max. at 32 kHz, 3.0V I _{PD} : 9 μA max. at 3.0V Freq: 200 kHz max.

The shaded sections indicate oscillator selections which are tested for functionality, but not for MIN/MAX specifications. It is recommended that the user select the device type that ensures the specifications required.

PIC16C6X

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

FIGURE 19-5: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

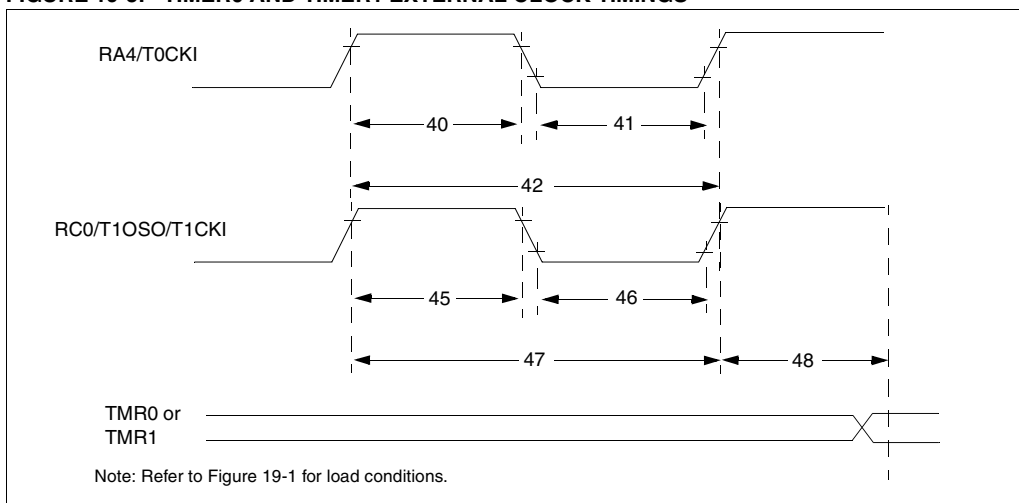


TABLE 19-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions		
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 42		
			With Prescaler	10	—	—	ns			
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 42		
			With Prescaler	10	—	—	ns			
42*	Tt0P	T0CKI Period	No Prescaler	$T_{CY} + 40$	—	—	ns			
			With Prescaler	Greater of: 20 or $T_{CY} + 40$ N	—	—	ns	N = prescale value (2, 4, ..., 256)		
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 47		
			Synchronous, Prescaler = 2,4,8	PIC16C6X PIC16LC6X	15 25	—	—		ns	
			Asynchronous	PIC16C6X PIC16LC6X	30 50	—	—		ns	
46*	Tt1L	T1CKI Low Time	Synchronous, Prescaler = 1	$0.5T_{CY} + 20$	—	—	ns	Must also meet parameter 47		
			Synchronous, Prescaler = 2,4,8	PIC16C6X PIC16LC6X	15 25	—	—		ns	
			Asynchronous	PIC16C6X PIC16LC6X	30 50	—	—		ns	
47*	Tt1P	T1CKI input period	Synchronous	PIC16C6X	Greater of: 30 OR $T_{CY} + 40$ N	—	—	ns	N = prescale value (1, 2, 4, 8)	
				PIC16LC6X	Greater of: 50 OR $T_{CY} + 40$ N				ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	PIC16C6X PIC16LC6X	60 100	—	—	ns		
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)		DC	—	200	kHz			
48	TCKEZtmr1	Delay from external clock edge to timer increment		$2T_{osc}$	—	$7T_{osc}$	—			

* 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.

PIC16C6X

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

FIGURE 21-8: PARALLEL SLAVE PORT TIMING (PIC16CR65)

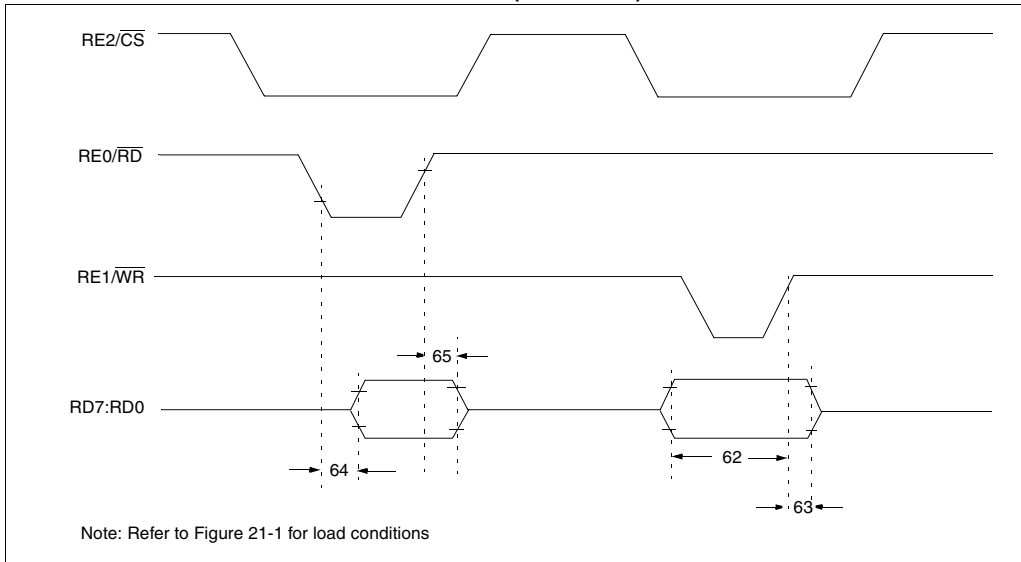


TABLE 21-7: PARALLEL SLAVE PORT REQUIREMENTS (PIC16CR65)

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions	
62*	TdtV2wrH	Data in valid before $\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ (setup time)	20	—	—	ns		
63*	TwrH2dtI	$\overline{WR}\uparrow$ or $\overline{CS}\uparrow$ to data-in invalid (hold time)	PIC16CR65	20	—	—	ns	
			PIC16LCR65	35	—	—	ns	
64	TrdL2dtV	$\overline{RD}\downarrow$ and $\overline{CS}\downarrow$ to data-out valid	—	—	80	ns		
65*	TrdH2dtI	$\overline{RD}\uparrow$ or $\overline{CS}\uparrow$ 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 tested.

PIC16C6X

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

FIGURE 23-3: TYPICAL IPD vs. VDD @ 25°C (WDT ENABLED, RC MODE)

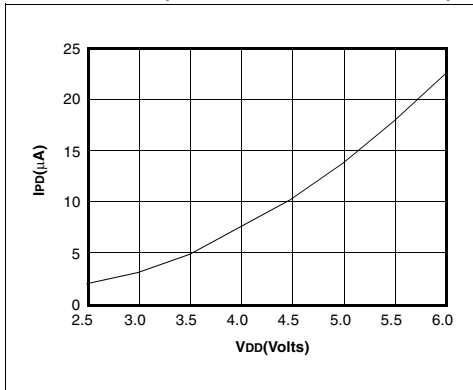


FIGURE 23-4: MAXIMUM IPD vs. VDD (WDT ENABLED, RC MODE)

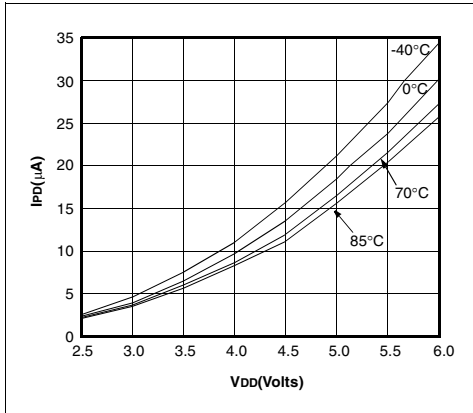


FIGURE 23-5: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

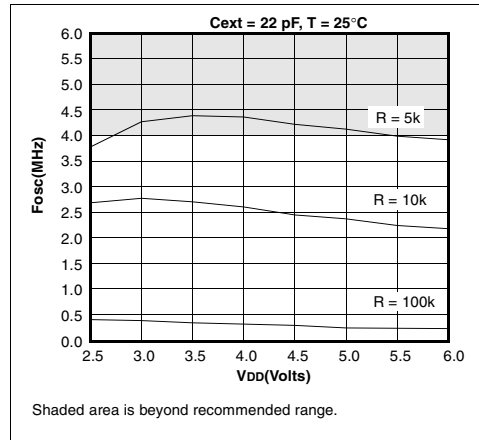


FIGURE 23-6: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

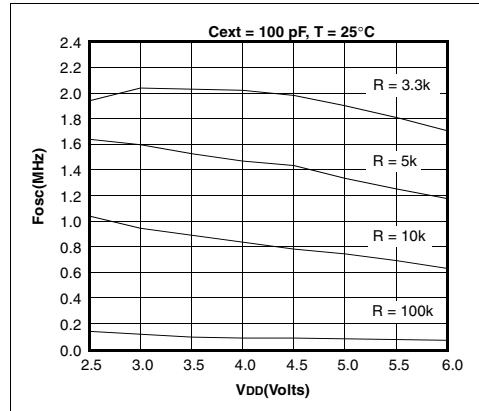
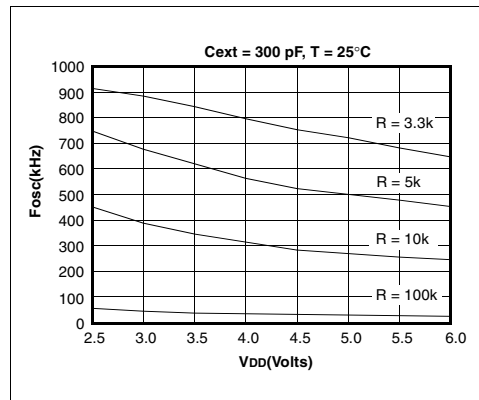


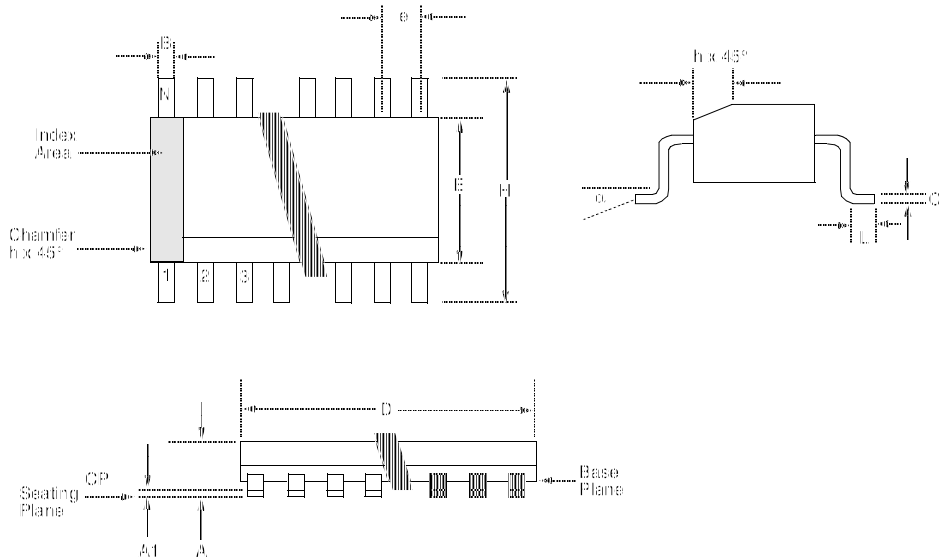
FIGURE 23-7: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD



Data based on matrix samples. See first page of this section for details.

24.5 28-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body) (SO)

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



Package Group: Plastic SOIC (SO)						
Symbol	Millimeters			Inches		
	Min	Max	Notes	Min	Max	Notes
α	0°	8°		0°	8°	
A	2.362	2.642		0.093	0.104	
A1	0.101	0.300		0.004	0.012	
B	0.355	0.483		0.014	0.019	
C	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	
e	1.270	1.270	Typical	0.050	0.050	Typical
H	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	

PIC16C6X

F.3 PIC16C15X Family of Devices

		PIC16C154	PIC16CR154	PIC16C156	PIC16CR156	PIC16C158	PIC16CR158
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
Memory	EPROM Program Memory (x12 words)	512	—	1K	—	2K	—
	ROM Program Memory (x12 words)	—	512	—	1K	—	2K
	RAM Data Memory (bytes)	25	25	25	25	73	73
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0	TMR0
Features	I/O Pins	12	12	12	12	12	12
	Voltage Range (Volts)	3.0-5.5	2.5-5.5	3.0-5.5	2.5-5.5	3.0-5.5	2.5-5.5
	Number of Instructions	33	33	33	33	33	33
	Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.

F.4 PIC16C5X Family of Devices

		PIC16C52	PIC16C54	PIC16C54A	PIC16CR54A	PIC16C55	PIC16C56
Clock	Maximum Frequency of Operation (MHz)	4	20	20	20	20	20
Memory	EPROM Program Memory (x12 words)	384	512	512	—	512	1K
	ROM Program Memory (x12 words)	—	—	—	512	—	—
	RAM Data Memory (bytes)	25	25	25	25	24	25
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0	TMR0
Features	I/O Pins	12	12	12	12	20	12
	Voltage Range (Volts)	2.5-6.25	2.5-6.25	2.0-6.25	2.0-6.25	2.5-6.25	2.5-6.25
	Number of Instructions	33	33	33	33	33	33
	Packages	18-pin DIP, SOIC	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	28-pin DIP, SOIC, SSOP	18-pin DIP, SOIC; 20-pin SSOP

		PIC16C57	PIC16CR57B	PIC16C58A	PIC16CR58A
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20
Memory	EPROM Program Memory (x12 words)	2K	—	2K	—
	ROM Program Memory (x12 words)	—	2K	—	2K
	RAM Data Memory (bytes)	72	72	73	73
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0
Features	I/O Pins	20	20	12	12
	Voltage Range (Volts)	2.5-6.25	2.5-6.25	2.0-6.25	2.5-6.25
	Number of Instructions	33	33	33	33
	Packages	28-pin DIP, SOIC, SSOP	28-pin DIP, SOIC, SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer (except PIC16C52), selectable code protect and high I/O current capability.

INDEX

Numerics

9-bit Receive Enable bit, RX9	106
9-bit Transmit Enable bit, TX9	105
9th bit of received data, RX9D	106
9th bit of transmit data, TX9D	105

A

Absolute Maximum	
Ratings	163, 183, 199, 215, 231, 247, 263
ACK	96, 100, 101
ALU	9
Application Notes	
AN552 (Implementing Wake-up on Key Stroke)	53
AN556 (Implementing a Table Read)	48
AN594 (Using the CCP Modules)	77
Architectural Overview	9

B

Baud Rate Formula	107
Baud Rate Generator	107
Baud Rates	
Asynchronous Mode	108
Error, Calculating	107
RX Pin Sampling, Timing Diagrams	110, 111
Sampling	110
Synchronous Mode	108
BF	84, 89, 100

Block Diagrams

Capture Mode Operation	78
Compare Mode	79
Crystal Oscillator, Ceramic Resonator	125
External Brown-out Protection	135
External Parallel Resonant Crystal Circuit	127
External Power-on Reset	135
External Series Resonant Crystal Circuit	127
I ² C Mode	99
In-circuit Programming Connections	142
Interrupt Logic	137
On-chip Reset Circuit	128
Parallel Slave Port, PORTD-PORTE	61
PIC16C61	10
PIC16C62	11
PIC16C62A	11
PIC16C63	12
PIC16C64	11
PIC16C64A	11
PIC16C65	12
PIC16C65A	12
PIC16C66	13
PIC16C67	13
PIC16CR62	11
PIC16CR63	12
PIC16CR64	11
PIC16CR65	12
PORTC	55
PORTD (I/O Mode)	57
PORTE (I/O Mode)	58
PWM	80
RA3:RA0 pins	51
RA4/T0CK1 pin	51
RA5 pin	51
RB3:RB0 pins	54
RB7:RB4 pins	53, 54
RC Oscillator Mode	127

SPI Master/Slave Connection	87
SSP in I ² C Mode	99
SSP in SPI Mode	86, 91
Timer0	65
Timer0/WDT Prescaler	68
Timer1	72
Timer2	75
USART Receive	114
USART Transmit	112
Watchdog Timer	140
BOR	129
$\overline{\text{BOR}}$	47, 131
BRGH	105
Brown-out Reset (BOR)	129
Brown-out Reset Status bit, $\overline{\text{BOR}}$	47
Buffer Full Status bit, BF	84, 89

C

C	35
C Compiler	161
Capture	
Block Diagram	78
Mode	78
Pin Configuration	78
Prescaler	79
Software Interrupt	78
Capture Interrupt	78
Capture/Compare/PWM (CCP)	
Capture Mode	78
Capture Mode Block Diagram	78
CCP1	77
CCP2	77
Compare Mode	79
Compare Mode Block Diagram	79
Overview	63
Prescaler	79
PWM Block Diagram	80
PWM Mode	80
PWM, Example Frequencies/Resolutions	81
Section	77
Carry	9
Carry bit	35
CCP Module Interaction	77
CCP pin Configuration	78
CCP to Timer Resource Use	77
CCP1 Interrupt Enable bit, CCP1IE	38
CCP1 Interrupt Flag bit, CCP1IF	41
CCP1 Mode Select bits	78
CCP1CON	24, 26, 28, 30, 32, 34
CCP1IE	38
CCP1IF	41
CCP1M3:CCM1M0	78
CCP1X:CCP1Y	78
CCP2 Interrupt Enable bit, CCP2IE	45
CCP2 Interrupt Flag bit, CCP2IF	46
CCP2 Mode Select bits	78
CCP2CON	24, 26, 28, 30, 32, 34
CCP2IE	45
CCP2IF	46
CCP2M3:CCP2M0	78
CCP2X:CCP2Y	78
CCPR1H	24, 26, 28, 30, 32, 34
CCPR1L	24, 26, 28, 30, 32, 34
CCPR2H	24, 26, 28, 30, 32, 34
CCPR2L	24, 26, 28, 30, 32, 34
CKE	89
CKP	85, 90

PIC16C6X

LIST OF EQUATION AND EXAMPLES

Example 3-1: Instruction Pipeline Flow	18
Example 4-1: Call of a Subroutine in Page 1 from Page 0	49
Example 4-2: Indirect Addressing	49
Example 5-1: Initializing PORTA	51
Example 5-2: Initializing PORTB	53
Example 5-3: Initializing PORTC	55
Example 5-4: Read-Modify-Write Instructions on an I/O Port	60
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 Timer	73
Example 10-1: Changing Between Capture Prescalers	79
Example 10-2: PWM Period and Duty Cycle Calculation	81
Example 11-1: Loading the SSPBUF (SSPSR) Register	86
Example 11-2: Loading the SSPBUF (SSPSR) Register (PIC16C66/67)	91
Example 12-1: Calculating Baud Rate Error	107
Example 13-1: Saving Status and W Registers in RAM	139
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

Figure 4-15: PIE1 Register for PIC16C65/65A/R65/67 (Address 8Ch)	40
Figure 4-16: PIR1 Register for PIC16C62/62A/R62 (Address 0Ch)	41
Figure 4-17: PIR1 Register for PIC16C63/R63/66 Address 0Ch	42
Figure 4-18: PIR1 Register for PIC16C64/64A/R64 (Address 0Ch)	43
Figure 4-19: PIR1 Register for PIC16C65/65A/R65/67 (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 (Address 8Eh)	47
Figure 4-23: PCON Register for PIC16C62A/R62/63/ R63/64A/R64/65A/R65/66/67 (Address 8Eh)	47
Figure 4-24: Loading of PC in Different Situations	48
Figure 4-25: Direct/Indirect Addressing	49
Figure 5-1: Block Diagram of the RA3:RA0 Pins and the RA5 Pin	51
Figure 5-2: Block Diagram of the RA4/T0CKI Pin	51
Figure 5-3: Block Diagram of the RB7:RB4 Pins for PIC16C61/62/64/65	53
Figure 5-4: Block Diagram of the RB7:RB4 Pins for PIC16C62A/63/R63/ 64A/65A/R65/66/67	54
Figure 5-5: Block Diagram of the RB3:RB0 Pins	54
Figure 5-6: PORTC Block Diagram	55
Figure 5-7: PORTD Block Diagram (In I/O Port Mode)	57
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 Timing: Internal Clock/No Prescaler	65
Figure 7-3: Timer0 Timing: Internal Clock/Prescale 1:2	66
Figure 7-4: TMR0 Interrupt Timing	66
Figure 7-5: Timer0 Timing With External Clock	67
Figure 7-6: Block Diagram of the Timer0/WDT Prescaler	68
Figure 8-1: T1CON: Timer1 Control Register (Address 10h)	71
Figure 8-2: Timer1 Block Diagram	72
Figure 9-1: Timer2 Block Diagram	75
Figure 9-2: T2CON: Timer2 Control Register (Address 12h)	75
Figure 10-1: CCP1CON Register (Address 17h) / CCP2CON Register (Address 1Dh)	78
Figure 10-2: Capture Mode Operation Block Diagram	78
Figure 10-3: Compare Mode Operation Block Diagram	79
Figure 10-4: Simplified PWM Block Diagram	80
Figure 10-5: PWM Output	80
Figure 11-1: SSPSTAT: Sync Serial Port Status Register (Address 94h)	84

PIC16C6X

Table 23-5:	Timer0 and Timer1 External Clock Requirements	272
Table 23-6:	Capture/Compare/PWM Requirements (CCP1 and CCP2)	273
Table 23-7:	Parallel Slave Port Requirements (PIC16C67) 274	
Table 23-8:	SPI Mode Requirements.....	277
Table 23-9:	I ² C Bus Start/Stop Bits Requirements	278
Table 23-10:	I ² C Bus Data Requirements	279
Table 23-11:	USART Synchronous Transmission Requirements	280
Table 23-12:	USART Synchronous Receive Requirements	280
Table 24-1:	RC Oscillator Frequencies	287
Table 24-2:	Capacitor Selection for Crystal Oscillators	288
Table E-1:	Pin Compatible Devices.....	315