8 Bit Development Microcomputer

FEATURES

- PIC microcomputer with ROM removed
- Useful for engineering prototyping of PIC applications
- ROM address & data lines brought out to pins
- HALT pin for single stepping or stopping program execution
- 50/55 pin for selection of PIC16C50 or PIC16C55 emulation
- 32 8-bit RAM registers
- Arithmetic Logic Unit
- Real Time Clock/Counter
- Self-contained oscillator
- Access to RAM registers inherent in instruction
- Wide power supply operating range (2.5V to 6.0V)

DESCRIPTION

The PIC16C63 development microcomputer is a CMOS/LSI device containing RAM, I/O, and a central processing unit on a single chip.

The PIC16C63 CMOS/LSI device is functionally identical to the PIC16C55 microcomputer except that the ROM is removed and the ROM address and data lines are brought out, requiring a 64-pin package. The addition of a HALT pin gives the user the ability to stop as well as single-step the chip. The logic level applied to the $\overline{50}$/55 pin determines whether the PIC16C63 emulates a PIC16C50 or PIC16C55.

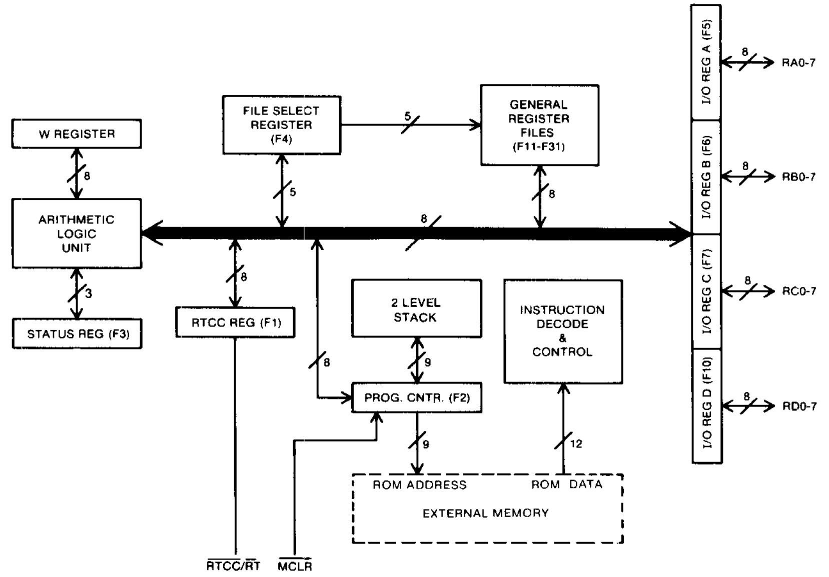
The external ROM can contain a customer-defined program using the PIC's powerful instruction set to specify the overall functional characteristics of the device. The 8-bit input/ouput registers provide latched lines for interfacing to a limitless variety of applications.

The 12-bit instruction word format provides a powerful yet easy to use instruction repertoire emphasizing single bit manipulation as well as logical and arithmetic operations using bytes.

The PIC Series is fabricated with complimentary MOS technology resulting in a high performance product with proven reliability and production history. Only a single wide range power supply is required for operation, and an on-chip oscillator provides the operating clock with only an external RC network (or crystal oscillator for greater accuracy) to establish the frequency.

Extensive hardware and software support is available to aid the user in developing this application program and to verify performance before committing to mask tooling. Application notes and sample programs are used to develop programs which can then be assembled into machine language using PICAL, eliminating the burden of coding with ones and zeros. PICAL is available in a Fortran IV version that can be run on many popular computer systems. Once the application program is developed several options are available to insure proper performance. The PFD Field Demo Systems are available containing a PIC16C63 with sockets for erasable CMOS PROMs. Finally, the PICES II (PIC In-Circuit Emulation System) provides the user with stand-alone emulation and debugging operation or operation as a peripheral to a larger computer system. Easy program debugging and changing is facilitated because the user's program is stored in RAM.

PIC16C63 GENERAL BLOCK DIAGRAM



ARCHITECTURAL DESCRIPTION

The firmware architecture of the PIC16C63 microcomputer is based on a register file concept with simple yet powerful commands designed to emphasize bit, byte, and register transfer operations. The instruction set also supports computing functions as well as these control and interface functions.

Internally, the PIC16C63 is composed of three functional elements connected together by a single bidirectional bus: the Register File composed of 32 addressable 8-bit registers, an Arithmetic Logic Unit, and a user-defined external ROM composed of 512 program words each 12 bits in width. The Register File is divided into two functional groups: operational registers and general register. The operational registers include, among others, the Real Time Clock Counter Register, the Program Counter (PC), the Status Register, and the I/O Registers. The general purpose registers are used for data and control information under command of the instructions.

The Arithmetic Logic Unit contains one temporary working register or accumulator (W Register) and gating to perform Boolean functions between data held in the working register and any file register.

Sequencing of microinstructions is controlled via the Program Counter (PC) which automatically increments to execute in-line programs. Program control operations can be performed by Bit Test and Skip instructions, Jump instructions, Call instructions, or by loading computed addresses into the PC. In addition, an on-chip two-level stack is employed to provide easy to use subroutine nesting. Activating the MCLR input on power up initializes the external ROM program to address 777_8.

PIN FUNCTIONS

Signal	Function
50/55 (Input)	Used to set the PIC16C63 to emulate the PIC16C55 (logic "one") or the PIC16C50 (logic "zero"). The mode must be selected before $\overline{\text{MCLR}}$ is brought high. This pin has an internal pullup.
OSC1 (Input), OSC2	Oscillator pins. Both OSC 1 and OSC 2 are used as a two pin oscillator clock using either crystal, ceramic resonator or RC network or OSC 1 can be driven by an external clock signal.
$\overline{\text{RTCC}}$ (Input)	Real Time input. This pin increments the Real Time Clock Counter Register 1 on high to low transitions applied to this input.
RA0-7, RB0-7, RC0-7, RD0-7 (Input/output)	User programmable input/output lines. These lines can be inputs and/or outputs and are under direct control of the program. During emulation of the PIC16C55, Register D will become internal general purpose File Register 10; Register B will become an output only file. Any instruction involving a read (all instructions except literal and control operations, NOP, MOVWF, CLRW and CLRF) will not read the pin but will read data from the output latch.
$\overline{\text{MCLR}}$ (Input)	Master Clear. Used to initialize the internal ROM program to address 777_8 and to latch all I/O registers into tri-state mode. This pin should be held low at least 1ms after the power supply is valid. $\overline{\text{MCLR}}$ has no internal pullup resistor.
V_{DD}	Primary Power supply input.
CLK OUT (output)	A signal derived from the internal oscillator. Used by external devices to synchronize themselves to PIC timing. The OSC frequency is divided by 5.
HALT (Input)	Halt. When high this input suspends execution of the next instruction. No data is lost and after HALT is brought low execution proceeds exactly as if no HALT signal has been applied.
HALT ACK (output)	Halt Acknowledge. This output is high when the PIC16C63 is halted either due to an active HALT input or execution of the HALT insruction (0001_8). In the first case HALT ACK is brought back low when the PIC 16C63 begins execution when the HALT input is brought low; and in the second case it is brought low using $\overline{\text{MCLR}}$ or by first raising and then lowering the HALT input.
D0-D11 (Input)	Data input. These 12 lines accept 12 bit PIC instruction codes generated by an external source. D0 is the LSB of the instruction.
A0-A8 (output)	Address Output. These 9 lines represent the address of the next instruction to be executed by the PIC16C63. A0 is the LSB of the address.

MODE PIN OPERATION

The mode pin is used to select either PIC16C50 emulation or PIC16C55 emulation.

With the $\overline{50}$/55 pin set high, the PIC16C63 is set to emulate the PIC16C55. Specifically:

1. I/O port RD is general purpose register.
2. I/O port RB is output only.

When the MODE input is low, the PIC16C63 will emulate the PIC16C50 circuit. Specifically:

1. All ports will be I/O.

To insure proper chip operation, the $\overline{50}$/55 pin should be preset before $\overline{MCLR}$ is brought high at initialization. "Dynamic-type" switching of this pin during processor operation will result in undefined conditions and must be avoided.

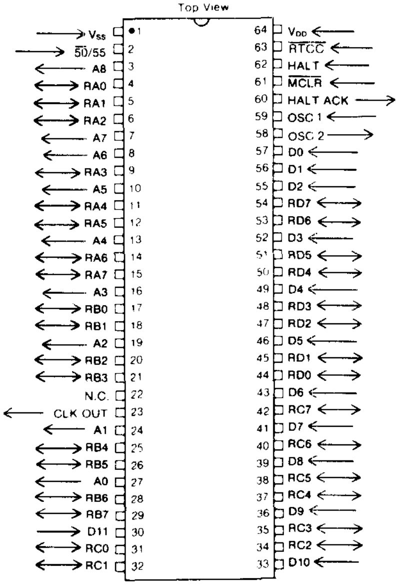
PROGRAMMING CAUTIONS

The PIC16C63 is designed as a development circuit for emulating the operation of the PIC16C50 and PIC16C55. While all circuits in the PIC series have the same basic architecture and instruction set, there are differences which require attention on the part of the user to insure that all conditions are met for proper operation of the PIC16C63 with respect to the target PIC circuit (either PIC16C50 or PIC16C55). The following checklist list should be used to achieve proper emulation.

1. The $\overline{50}$/55 pin must be properly set (high for PIC16C55 or low for PIC16C50).
2. For PIC16C55 emulation bits 4-7 of F5 (the input only file) should be tied to V_{SS} (ground) as these bits are always read as low inputs.
3. For PIC16C55 emulation the pins corresponding to $F10_8$ (I/O port RD on the PIC16C50) should be left unconnected. In this way $F10_8$ will operate as an internal register as is appropriate for the PIC16C55.
4. The HALT instruction (0001_8) is not recognized by any PIC circuit other than the PIC16C63.

PIN CONFIGURATION

64 LEAD DUAL IN LINE



REGISTER FILE ARRANGEMENT

File (Octal)	Function
F0	Not a physically implemented register. F0 calls for the contents of the File Select Register (low order 5 bits) to be used to select a file register. F0 is thus useful as an indirect address pointer. For example, W+F0→W will add the contents of the file register pointed to by the FSR (F4) to W and place the result in W.
F1	Real Time Clock Counter Register. This register can be loaded and read by the microprogram. The RTCC register keeps counting up after zero is reached. The counter increments on the falling edge of a pulse on the $\overline{\text{RTCC}}$ pin. However, if data are being stored in the RTCC register simultaneously with a negative transition on the $\overline{\text{RTCC}}$ pin, the RTCC register will contain the new stored value and the external transition will be ignored by the microcomputer.
F2	Program Counter (PC). The PC is automatically incremented and each instruction cycle can be written into under program control e.g., MOVWF F2. The PC is nine bits wide, but only the low order 8 bits can be read under program control.
F3	Status Word Register. When F3 is the destination register the status flags are overwritten.

(7)	(6)	(5)	(4)	(3)	(2)	(1)	(0)
1	1	1	1	1	Z	DC	C

BIT 0: Carry
(C) bit — For ADD and SUB instructions, this bit is set if there is a carry out from the most significant bit of the resultant.

For ROTATE instructions, this bit is loaded with either the high or low order bit of the source.

BIT 1: Digit Carry
(DC) bit — For ADD and SUB instructions, this bit is set if there is a carry out from the 4th low order bit of the resultant.

BIT 2: Zero
(Z) bit — Set if the result of an ALU operation is zero.

BITS: 3-7 — These bits are defined as logic ones.

File (Octal)	Function
F4	File Select Register (FSR). Low order 5 bits only are used. The FSR is used in generating effective file register addresses under program control. When accessed as a directly addressed file, the upper 3 bits are read as ones.
F5	I/O Register A (A0-A7)
F6	I/O Register B (B0-B7)
F7	I/O Register C (C0-C7)
F10	I/O Register D (D0-D7)
F11-F37	General Purpose Registers

Basic Instruction Set Summary

Each PIC instruction is 12-bit word divided into an OP code which specifies the instruction type and one or more operands which further specify the operation of the instruction. The following PIC instruction summary lists byte-oriented, bit-oriented, and literal and control operations.

For byte-oriented instructions, "f" represents a file register designator and "d" represents a destination designator. The file register designator specifies which one of the 32 PIC file registers is to be utilized by the instruction. The destination designator specifies where the result of the operation performed by the instruction is to be placed. If "d" is zero, the result is placed in the PIC W register. If "d" is one, the result is returned to the file register specified in the instruction.

For bit-oriented instructions, "b" represents a bit field designator which selects the number of the bit affected by the operation, while "f" represents the number of the file in which the bit is located.

For literal and control operations, "k" represents an eight or nine bit constant or literal value.

For an oscillator frequency of 1 MHz the instruction execution time is 5 µsec, unless a conditional test is true or the program counter is changed as a result of an instruction. In these two cases, the instruction execution time is 10 µsec.

BYTE-ORIENTED FILE REGISTER OPERATIONS

(11-6)	(5)	(4-0)
OP CODE	d	f (FILE #)

For d = 0, f→W (PICAL accepts d = 0 or d = W in the mnemonic)
d = 1, f→f (If d is omitted, assembler assigns d = 1.)

Instruction-Binary (Octal)	Name	Mnemonic, Operands		Operation	Status Affected
000 000 000 000 (0000)	No Operation	NOP	—	—	None
000 000 0ff fff (0000)	Tri-state port f	TRIS	f	W→Tristate status f	None
000 000 1ff fff (0040)	Move W to f (Note 1)	MOVWF	f	W→f	None
000 001 000 000 (0100)	Clear W	CLRW	—	0→W	Z
000 001 1ff fff (0140)	Clear f	CLRF	f	0→f	Z
000 010 dff fff (0200)	Subtract W from f	SUBWF	f, d	f - W→d [$f+\overline{W}+1$→d]	C,DC,Z
000 011 dff fff (0300)	Decrement f	DECF	f, d	f - 1→d	Z
000 100 dff fff (0400)	Inclusive OR W and f	IORWF	f, d	WVf→d	Z
000 101 dff fff (0500)	AND W and f	ANDWF	f, d	W•f→d	Z
000 110 dff fff (0600)	Exclusive OR W and f	XORWF	f, d	W⊕f→d	Z
000 111 dff fff (0700)	Add W and f	ADDWF	f, d	W+f→d	C,DC,Z
001 000 dff fff (1000)	Move f	MOVF	f, d	f→d	Z
001 001 dff fff (1100)	Complement f	COMF	f, d	$\overline{f}$→d	Z
001 010 dff fff (1200)	Increment f	INCF	f, d	f+1→d	Z
001 011 dff fff (1300)	Decrement f, Skip if Zero	DECFSZ	f, d	f - 1→d, skip if Zero	None
001 100 dff fff (1400)	Rotate Right f	RRF	f, d	f(n)→d(n-1), f(0)→C, C→d(7)	C
001 101 dff fff (1500)	Rotate Left f	RLF	f, d	f(n)→d(n+1), f(7)→C, C→d(0)	C
001 110 dff fff (1600)	Swap halves f	SWAPF	f, d	f(0-3)⇆f(4-7)→d	None
001 111 dff fff (1700)	Increment f, Skip if Zero	INCFSZ	f, d	f+1→d, skip if zero	None

BIT-ORIENTED FILE REGISTER OPERATIONS

(11-8)	(7-5)	(4-0)
OP CODE	b (BIT #)	f (FILE #)

Instruction-Binary (Octal)	Name	Mnemonic, Operands		Operation	Status Affected
010 0bb bff fff (2000)	Bit Clear f	BCF	f, b	0→f(b)	None
010 1bb bff fff (2400)	Bit Set f	BSF	f, b	1→f(b)	None
011 0bb bff fff (3000)	Bit Test f, Skip if Clear	BTFSC	f, b	Bit Test f(b): skip if clear	None
011 1bb bff fff (3400)	Bit Test f, Skip if Set	BTFSS	f, b	Bit Test f(b): skip is set	None

LITERAL AND CONTROL OPERATIONS

(11-8)	(7-0)
OP CODE	k (LITERAL)

Instruction-Binary (Octal)	Name	Mnemonic, Operands		Operation	Status Affected
000 000 000 010 (0002)	Return	RETURN	—	Stack→PC	None
100 0kk kkk kkk (4000)	Return and place Literal in W	RETLW	k	k→W, Stack→PC	None
100 1kk kkk kkk (4400)	Call subroutine (Note 1)	CALL	k	PC+1 → Stack, k → PC	None
101 kkk kkk kkk (5000)	Go To address (k is 9 bits)	GOTO	k	k→PC	None
110 0kk kkk kkk (6000)	Move Literal to W	MOVLW	k	k→W	None
110 1kk kkk kkk (6400)	Inclusive OR Literal and W	IORLW	k	kVW→W	Z
111 0kk kkk kkk (7000)	AND Literal and W	ANDLW	k	k•W→W	Z
111 1kk kkk kkk (7400)	Exclusive OR Literal and W	XORLW	k	k⊕W→W	Z

NOTES:

1. The 9th bit of the program counter in the PIC is zero for a CALL and a MOVWF F2. Therefore, subroutines must be located in program memory locations 0-377$_8$. However, subroutines can be called from anywhere in the program memory since the Stack is 9 bits wide.
2. When an I/O register is modified as a function of itself, the value used will be that value present on the output pins. For example, an output pin which has been latched high but is driven low by an external device, will be relatched in the low state.
3. TRIS f (where f=6 or 7 for PIC16C55 or 5, 6, 7, 10 for PIC16C50) causes the contents of W to be written to the tri-state latches of the specified file. A one forces the pin to tri-state the output buffer to a high impedance state.

SUPPLEMENTAL INSTRUCTION SET SUMMARY

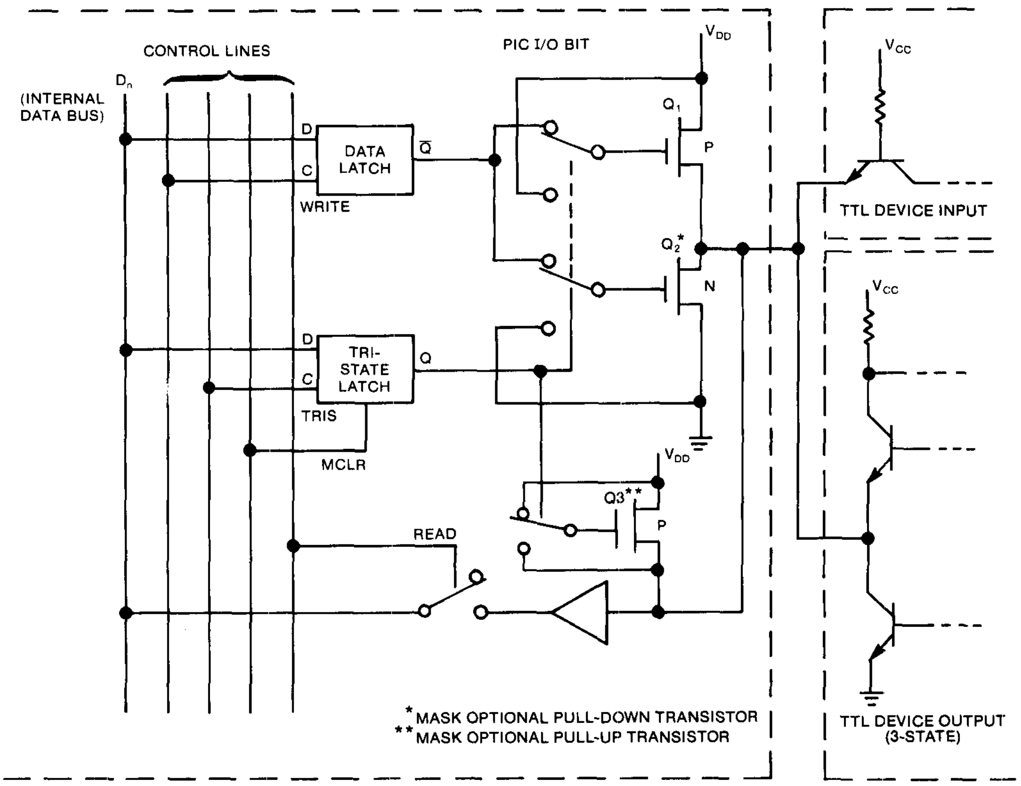
The following supplemental instructions summarized below represent specific applications of the basic PIC instructions. For example, the "CLEAR CARRY" supplemental instruction is equivalent to the basic instruction BCF 3,0 ("Bit Clear, File 3, Bit 0"). These instruction mnemonics are recognized by the PIC Cross Assembler (PICAL).

Instruction-Binary (Octal)	Name	Mnemonic, Operands	Equivalent Operation(s)	Status Affected
010 000 000 011 (2003)	Clear Carry	CLRC	BCF 3, 0	—
010 100 000 011 (2403)	Set Carry	SETC	BSF 3, 0	—
010 000 100 011 (2043)	Clear Digit Carry	CLRDC	BCF 3, 1	—
010 100 100 011 (2443)	Set Digit Carry	SETDC	BSF 3, 1	—
010 001 000 011 (2103)	Clear Zero	CLRZ	BCF 3, 2	—
010 101 000 011 (2503)	Set Zero	SETZ	BSF 3, 2	—
011 100 000 011 (3403)	Skip on Carry	SKPC	BTFSS 3, 0	—
011 000 000 011 (3003)	Skip on No Carry	SKPNC	BTFSC 3, 0	—
011 100 100 011 (3443)	Skip on Digit Carry	SKPDC	BTFSS 3, 1	—
011 000 100 011 (3043)	Skip on No Digit Carry	SKPNDC	BTFSC 3, 1	—
011 101 000 011 (3503)	Skip on Zero	SKPZ	BTFSS 3, 2	—
011 001 000 011 (3103)	Skip on No Zero	SKPNZ	BTFSC 3, 2	—
001 000 1ff fff (1040)	Test File	TSTF f	MOVF f, 1	Z
001 000 0ff fff (1000)	Move File to W	MOVFW f	MOVF f, 0	Z
001 001 1ff fff (1140) 001 010 dff fff (1200)	Negate File	NEGF f,d	COMF f, 1 INCF f, d	Z
011 000 000 011 (3003) 001 010 dff fff (1200)	Add Carry to File	ADDCF f, d	BTFSC 3,0 INCF f, d	Z
011 000 000 011 (3003) 000 011 dff fff (0300)	Subtract Carry from File	SUBCF f,d	BTFSC 3,0 DECF f, d	Z
011 000 100 011 (3043) 001 010 dff fff (1200)	Add Digit Carry to File	ADDDCF f,d	BTFSG 3,1 INCF f,d	Z
011 000 100 011 (3043) 000 011 dff fff (0300)	Subtract Digit Carry from File	SUBDCF f,d	BTFSC 3,1 DECF f,d	Z
101 kkk kkk kkk (5000)	Branch	B k	GOTO k	—
011 000 000 011 (3003) 101 kkk kkk kkk (5000)	Branch on Carry	BC k	BTFSC 3,0 GOTO k	—
011 100 000 011 (3403) 101 kkk kkk kkk (5000)	Branch on No Carry	BNC k	BTFSS 3,0 GOTO k	—
011 100 100 011 (3043) 101 kkk kkk kkk (5000)	Branch on Digit Carry	BDC k	BTFSC 3,1 GOTO k	—
011 001 000 011 (3443) 101 kkk kkk kkk (5000)	Branch on No Digit Carry	BNDC k	BTFSS 3,1 GOTO k	—
011 101 000 011 (3103) 101 kkk kkk kkk (5000)	Branch on Zero	BZ k	BTFSC 3,2 GOTO k	—
011 101 000 011 (3503) 101 kkk kkk kkk (5000)	Branch on No Zero	BNZ k	BTFSS 3,2 GOTO k	—

I/O Interfacing

The equivalent circuit for an I/O port bit is shown below as it would interface with either the input of a TTL device (PIC is outputting) or the output of a tri-state TTL device (PIC is inputting). Each I/O port bit can be individually time multiplexed between input and output functions under software control. When outputting thru a PIC I/O Port, the data is latched at the port and the pin can be connected directly to a TTL gate input. When inputting data thru an I/O Port, the port must first be set to the high impedance state under program control. This turns off Q_1 and Q_2 and turns on Q_3 (if present), allowing the TTL tri-state device to drive the pin.

TYPICAL INTERFACE-BIDIRECTIONAL I/O LINE
(shown in active I/O state)



*MASK OPTIONAL PULL-DOWN TRANSISTOR
**MASK OPTIONAL PULL-UP TRANSISTOR

Programming Cautions

The use of the bidirectional I/O ports are subject to certain rules of operation. These rules must be carefully followed in the instruction sequences written for I/O operation. (Note that for an output only port the latch, not the pin is read.)

Bidirectional I/O Ports

The bidirectional ports may be used for both input and output operations. For input operations these ports are non-latching. Any input must be present until read by an input instruction. The outputs are latched and remain unchanged until the output latch is rewritten. For use as an input port the output must be set to the high impedance state via the tri-state latch. Thus the external device inputs to the PIC circuit by forcing the input line high or low. If the input lines are not tri-stated then refer to PIC1650A programming cautions. This principle is the same whether operating on individual bits or the entire port.

Some instructions operate internally as input followed by output operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation, and re-output the result.

Successive Operations on Bidirectional I/O Ports

Care must be exercised if successive instructions operate on the same I/O port. The sequence of instructions should be such to allow the pin voltage to stabilize (load dependent) before the next instruction which causes that file to be read into the CPU (MOVF, BIT SET, BIT CLEAR, and BIT TEST) is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. This will happen if t_{pd} (See I/O Timing Diagram) is greater than $^1/st_{cy}$ (min). When in doubt, it is better to separate these instructions with a NOP or other instruction.

ELECTRICAL CHARACTERISTICS

Maximum Ratings*

Ambient Temperature Under Bias .... 125° C
Storage Temperature .... −55° C to +150° C
Voltage on any Pin with Respect to V_{SS} (Note 1) ..... −0.3V to V_{DD} +0.3V
Power Dissipation (Note 5) 300mW
Voltage on V_{DD} with Respect to V_{SS} .... −0.3V to +6.5V

*Exceeding these ratings could cause permanent damage to the device. This is a stress rating only and functional operation of this device at these conditions is not implied—operating ranges are specified in Standard Conditions. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Data labeled "typical" is presented for design guidance only and is not guaranteed.

Standard Conditions (unless otherwise stated):

DC CHARACTERISTICS

Operating Temperature $T_A = 0° C$ to $+70° C$

Characteristics	Sym	Min	Typ†	Max	Units	Conditions
Supply Voltage	V_{DD}	2.5	—	6.0	V	
Supply Current	I_{DD}	— —	— —	2 5	mA mA	$V_{DD} = 4V$ } All I/O pins tri-state, $V_{DD} = 6V$ } $t_{CY} = 4\mu sec$
Input Low Voltage	V_{IL}	V_{SS}	—	$0.2V_{DD}$	V	
Input High Voltage	V_{IH}	$0.8V_{DD}$	—		V	
Output High Voltage (RB0-7, RC0-7)	V_{OH}	$V_{DD}-0.4$ $V_{DD}-0.4$	— —	— —	V V	$I_{SOURCE} = 0.2mA$, $V_{DD} = 4.75V$ $I_{SOURCE} = 80\mu A$, $V_{DD} = 2.5V$
Output Low Voltage (RB0-7, RC0-7) (Note 1)	V_{OL}	—	—	0.4	V	$I_{SINK} = 0.2mA$, $V_{DD} = 2.5V$ $I_{SINK} = 1.6mA$, $V_{DD} = 4.75V$
Input Low Current (RA0-3, RC0-7) (Note 2)	I_{IL}	—	—	250	μA	$V_{DD} = 6V$, $V_{IN} = 0.4V$
Input High Current (RA0-3, RC0-7) (Note 2)	I_{IH}	2	—	—	μA	$V_{IN} = V_{DD} - 0.4V$
Leakage Current (Note 3)	I_{LC}	−1	—	1	μA	$V_{SS} \leqslant V_{PIN} \leqslant V_{DD}$
Input Low Current (HALT)	I_{IL}	15	—	—	μA	$V_{IL} = 0.4V$, $V_{DD} = 5V \pm 5\%$
Input High Current (HALT)	I_{IH}	— —	— —	250 40	μA μA	$V_{IH} = V_{DD} - 0.4V$, $V_{DD} = 5V \pm 5\%$ $V_{IH} = 2.1V$, $V_{DD} = 2.5V$

†Typical data is at $T_A = 25° C$, $V_{DD} = 5.0V$.

NOTES:

1. The output pull-down transistor can be removed via a mask option to facilitate interfacing with external circuitry which has signal swings below V_{SS}. If this is the case, the maximum voltage permitted to be applied to the pin is −12V with respect to V_{DD}.
2. Current is being sourced by the internal pull-up resistors which are available as a mask option on ports RA0-3 and RC0-7. (RC0-7 have their pull-ups turned off when selected as outputs.)
3. This applies to ports RA0-3 and RC0-7 without the mask optional internal pull-up resistors, port RB0-7 and RC0-7 in the high impedance state, $\overline{\text{RTCC}}$, $\overline{\text{MCLR}}$ and OSC 1.
4. Total output sink current for all output pins (including CLK OUT) must not exceed 50mA. Total output source current must not exceed 20mA. Maximum output sink or source current for each individual output must not exceed 10mA.
5. Total power dissipation should not exceed 300mW for the package. Power dissipation is calculated as follows:
   $P_{DIS} = V_{DD}\,[(I_{DD}) - \Sigma\,(I_{IN} + I_{OH})] + \Sigma\,(V_{DD} - V_{IN})\,(I_{IN}) + \Sigma\,(V_{DD} - V_{OH})\,(I_{OH}) + \Sigma\,(V_{OL})\,(I_{OL})$

Standard Conditions (unless otherwise stated):

AC CHARACTERISTICS

Operating Temperature $T_A = 0°C$ to $+70°C$, $V_{DD} = 2.5 - 6.0V$ except as noted.

Characteristics	Sym	Min	Typ	Max	Units	Conditions
Oscillator Frequency	f_{OSC}	25	—	1000	kHz	$V_{DD} = 2.5V$
	f_{OSC}	25	—	1250	kHz	$V_{DD} = 3.2V$
	f_{OSC}	25	—	1650	kHz	$V_{DD} = 5.0V$
	f_{OSC}	25	—	1800	kHz	$V_{DD} = 6.0V$
CLOCK OUT						
Period (Instruction Cycle Time)	t_{CY}	—	$5/f_{OSC}$	—	μs	(Note 1)
Pulse Width	t_{CLKH}	—	$1/f_{OSC}$	—	μs	
Rise/Fall Time	t_r/t_f	—	—	200	ns	1 TTL Load + 60pF $V_{DD} = 5V$
RTCC Input						
Period	t_{RT}	$t_{CY}+0.2\mu s$	—	—	—	(Notes 2 and 3)
Pulse Width (High or Low Level)	t_{pw}	500	—	—	ns	
I/O Ports						
Data Input Setup Time	t_S	—	—	$1/5t_{CY}-300$	ns	
Data Input Hold Time	t_h	0	—	—	ns	
Data Output Propagation Delay	t_{pd}	—	—	1.6	μs	60pF + 2.2K to $0.8V_{DD}$
HALT ACK Output Propogation Delay	t_{HA}	—	—	600	ns	$V_{DD} = 5V \pm 5\%$, 1 TTL + 60pF load
A_0-A_8 Output Propogation Delay	t_{AD}	—	—	1	μs	$V_{DD} = 5V \pm 5\%$, $10\mu A$ + 60pF load
D_0-D_{11} Input Set-up time	t_{DS}	30	—	—	ns	$V_{DD} = 5V \pm 5\%$
D_0-D_{11} Input Hold Time	t_{DH}	300	—	—	ns	$V_{DD} = 5V \pm 5\%$

†Typical data is at $T_A = 25°C$, $V_{DD} = 5.0V$

NOTES:

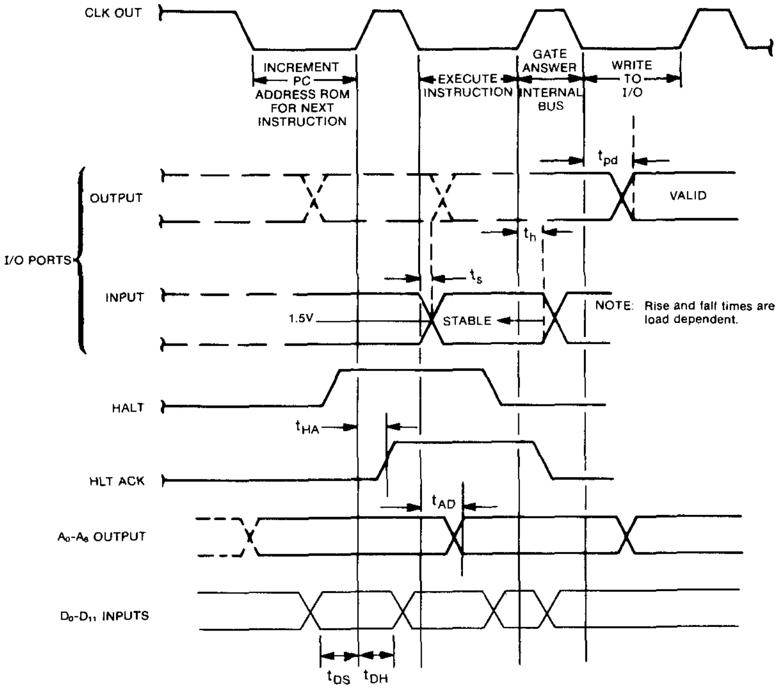
1. Instruction cycle period (t_{CY}) equals five times the input oscillator time base period.
2. Due to the synchronous timing nature between CLK OUT and the sampling circuit used on the $\overline{RTCC}$ input, CLK OUT may be directly tied to the $\overline{RTCC}$ input.
3. The maximum frequency which may be input to the $\overline{RTCC}$ pin is calculated as follows:

$$f_{(max)} = \frac{1}{t_{RT\ (min)}} = \frac{1}{t_{CY\ (min)} + 0.2\mu s}$$

For example:

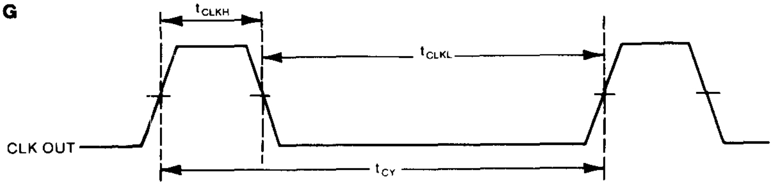
$$\text{if } t_{CY} = 4\mu s,\ f_{(max)} = \frac{1}{4.2\mu s} = 238KHz.$$

I/O TIMING

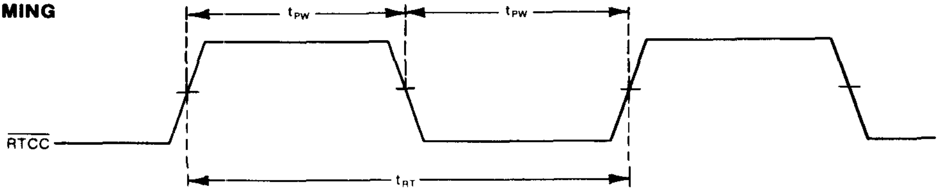


NOTE: Rise and fall times are load dependent.

CLK OUT TIMING

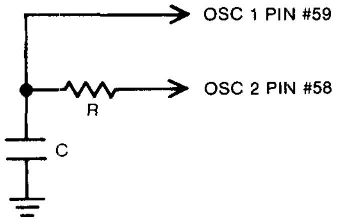


$\overline{RTCC}$ TIMING



PIC16C63 EMULATION OSCILLATOR OPTIONS (TYPICAL CIRCUITS)

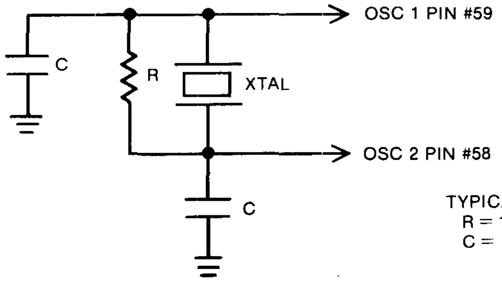
RC OPTION OPERATION



10K ⩽ R ⩽ 1M.

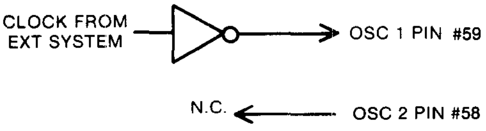
TYPICAL VALUES
R ⩾ 10K
C ⩾ 100pF

CRYSTAL INPUT OPERATION

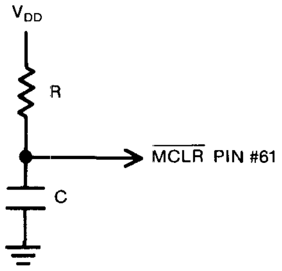


TYPICAL VALUES
R = 1M
C = −20pF

EXTERNAL CLOCK INPUT OPERATION



MASTER CLEAR



TYPICAL VALUES
R = 1M
C = 0.1μF

Master Clear may require up to a 75ms delay before activation after power is applied to the V_{DD} pin for a 1MHz crystal to start up. To achieve this an external RC configuration as shown can be used (assuming V_{DD} is applied as a step function). The RC oscillator option, shown above, should start up in less time.