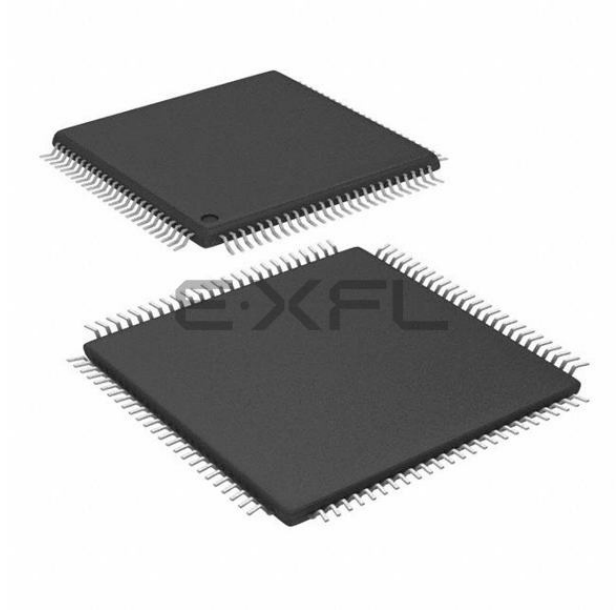
Microchip Technology - PIC18F97J94T-I/PT Datasheet



Welcome to E-XFL.COM

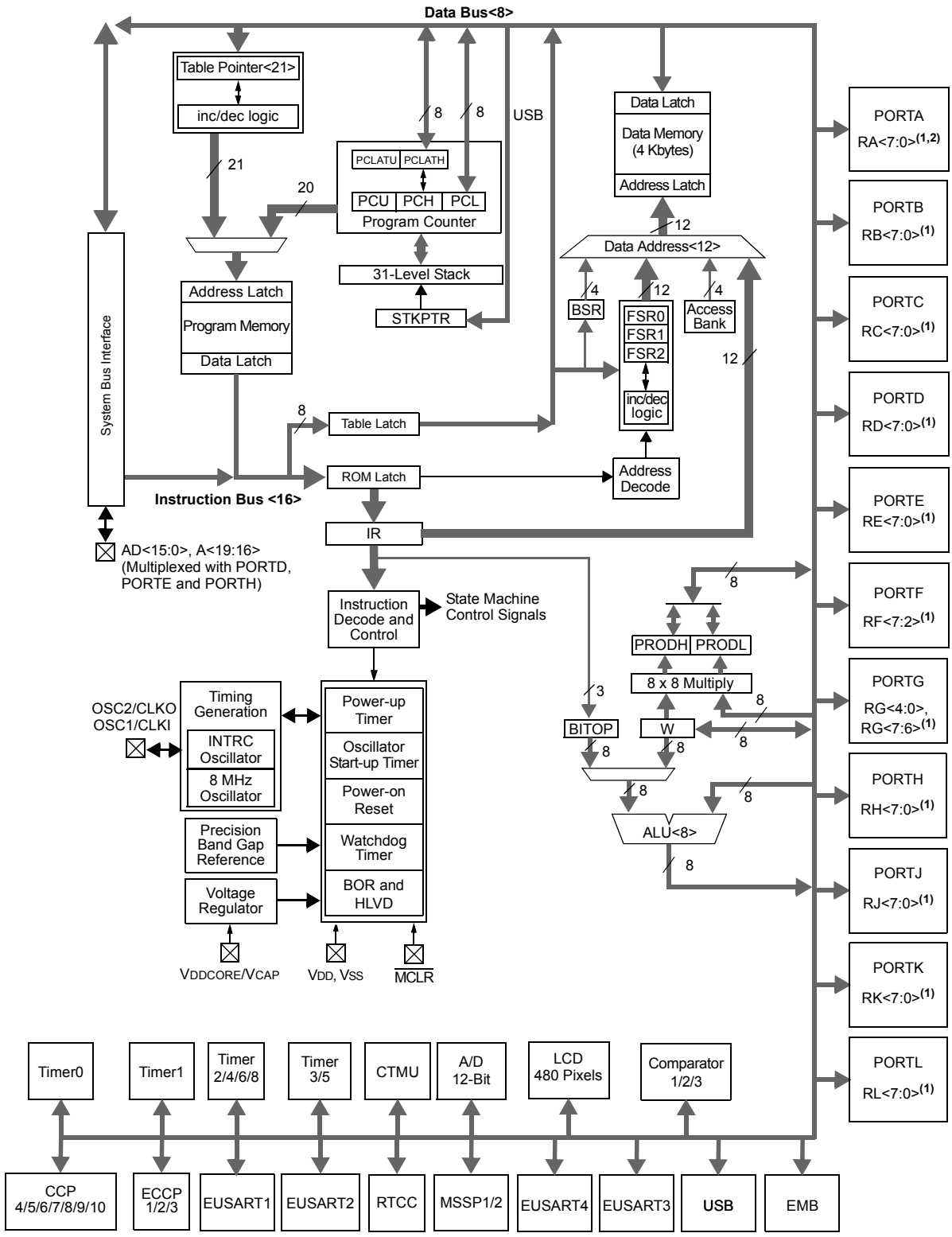
What is "Embedded - Microcontrollers"?

"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	64MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, HLVD, LCD, POR, PWM, WDT
Number of I/O	86
Program Memory Size	128KB (64K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 24x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f97j94t-i-pt

FIGURE 1-3: 100-PIN DEVICE BLOCK DIAGRAM



Note 1: See Table 1-4 for I/O port pin descriptions.

2: RA6 and RA7 are only available as digital I/O in select oscillator modes. See **Section 3.0 "Oscillator Configurations"** for more information.

REGISTER 4-6: PMD1: PERIPHERAL MODULE DISABLE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCP2MD	ECCP1MD	UART4MD	UART3MD	UART2MD	UART1MD	SSP2MD	SSP1MD
bit 7							bit 0

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

bit 7 **ECCP2MD:** ECCP2 Module Disable bit
1 = The ECCP2 module is disabled. All ECCP2 registers are held in Reset and are not writable.
0 = The ECCP2 module is enabled

bit 6 **ECCP1MD:** ECCP1 Module Disable bit
1 = The ECCP1 module is disabled. All ECCP1 registers are held in Reset and are not writable.
0 = The ECCP1 module is enabled

bit 5 **UART4MD:** USART4 Module Disable bit
1 = The USART4 module is disabled. All USART4 registers are held in Reset and are not writable.
0 = The USART4 module is enabled

bit 4 **UART3MD:** USART3 Module Disable bit
1 = The USART3 module is disabled. All USART3 registers are held in Reset and are not writable.
0 = The USART3 module is enabled

bit 3 **UART2MD:** USART2 Module Disable bit
1 = The USART2 module is disabled. All USART2 registers are held in Reset and are not writable.
0 = The USART2 module is enabled

bit 2 **UART1MD:** USART1 Module Disable bit
1 = The USART1 module is disabled. All USART1 registers are held in Reset and are not writable.
0 = The USART1 module is enabled

bit 1 **SSP2MD:** SSP2 Module Disable bit
1 = The SSP2 module is disabled. All SSP2 registers are held in Reset and are not writable.
0 = The SSP2 module is enabled

bit 0 **SSP1MD:** SSP1 Module Disable bit
1 = The SSP1 module is disabled. All SSP1 registers are held in Reset and are not writable.
0 = The SSP1 module is enabled

REGISTER 4-7: PMD2: PERIPHERAL MODULE DISABLE REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR8MD	TMR6MD	TMR5MD	TMR4MD	TMR3MD	TMR2MD	TMR1MD	TMR0MD
bit 7							bit 0

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

bit 7 **TMR8MD:** Timer8 Module Disable bit
1 = The Timer8 module is disabled. All Timer8 registers are held in Reset and are not writable.
0 = The Timer8 module is enabled

bit 6 **TMR6MD:** Timer6 Module Disable bit
1 = The Timer6 module is disabled. All Timer6 registers are held in Reset and are not writable.
0 = The Timer6 module is enabled

bit 5 **TMR5MD:** Timer5 Module Disable bit
1 = The Timer5 module is disabled. All Timer5 registers are held in Reset and are not writable.
0 = The Timer5 module is enabled

bit 4 **TMR4MD:** Timer4 Module Disable bit
1 = The Timer4 module is disabled. All Timer4 registers are held in Reset and are not writable.
0 = The Timer4 module is enabled

bit 3 **TMR3MD:** Timer3 Module Disable bit
1 = The Timer3 module is disabled. All Timer3 registers are held in Reset and are not writable.
0 = The Timer3 module is enabled

bit 2 **TMR2MD:** Timer2 Module Disable bit
1 = The Timer2 module is disabled. All Timer2 registers are held in Reset and are not writable.
0 = The Timer2 module is enabled

bit 1 **TMR1MD:** Timer1 Module Disable bit
1 = The Timer1 module is disabled. All Timer1 registers are held in Reset and are not writable.
0 = The Timer1 module is enabled

bit 0 **TMR0MD:** Timer0 Module Disable bit
1 = The Timer0 module is disabled. All Timer0 registers are held in Reset and are not writable.
0 = The Timer0 module is enabled

5.6.1 BROWN-OUT RESET (BOR)

Brown-out Reset is the legacy PIC18 “J” feature that monitors the core voltage, VDDCORE. Since the regulator on the PIC18F97J94 family is always enabled, this feature is always active. Its trip point is non-configurable. A Brown-out Reset will occur as the regulator output voltage drops below, approximately 1.6V. After proper operating voltage recovers, the Brown-out Reset condition is exited and execution begins after the Power-up Timer has expired. The $\overline{\text{BOR}}$ (RCON<0>) bit is also cleared. This bit must be set after each Brown-out and Power-on Reset event to detect subsequent Brown-out Reset events.

Note:	Brown-out Reset (BOR) has been provided to support legacy devices that can disable their internal regulator. The PIC18F97J94 family’s regulator is always enabled. Therefore, it’s recommended that new designs use VDDBOR to detect Brown-out conditions.

5.6.2 VDD BOR (VDDBOR)

VDDBOR is enabled by setting the BOREN (CONFIG1H<0>) Configuration bit. The low-power BOR trip level is configurable to either 1.8V or 2.0V, (typ) depending on the BORV (CONFIG1H<1>) Configuration bit setting. When in normal Run mode, Idle or normal Sleep modes, the BOR circuit that monitors VDD is active and will cause the device to be held in BOR if VDD drops below VBOR. Once VDD rises back above VVDDBOR, the device will be held in Reset until the expiration of the Power-up Timer, with period, TPWRT. This event is captured by the VDDBOR flag bit (RCON3<3>).

5.6.3 DETECTING VDD BOR

When the BOR module is enabled, the VDDBOR (RCON3<3>) bit is set on a Brown-out Reset event. This makes it difficult to determine if a Brown-out Reset event has occurred just by reading the state of VDDBOR alone. A more reliable method is to simultaneously check the state of both VDDPOR and VDDBOR. This assumes that the VDDPOR bit is reset to ‘1’ in software immediately after any Power-on Reset event. If VDDBOR is ‘0’ while VDDPOR is ‘1’, it can be reliably assumed that a Brown-out Reset event has occurred. Legacy PIC18 software can use the respective $\overline{\text{POR}}$ (RCON<1>) and $\overline{\text{BOR}}$ (RCON<0>) bits. This technique monitors the regulator output voltage, VDDCORE. To take advantage of the configuration features, it is recommended to use VDDBOR instead of BOR.

5.6.4 VBAT BROWN-OUT RESET (VBATBOR)

The VBAT BOR can be enabled/disabled using the VBTBOR bit in the Configuration register (CONFIG7L<2>). If the VBTBOR enable bit is cleared, the VBATBOR is always disabled and there will be no indication of a VBAT BOR. If the VBTBOR bit is set, the VBAT POR will reset the device when the battery voltage drops below VVBATBOR. After power is restored to the VBAT pin, the device exits Reset and returns to VBAT mode. The device remains in VBAT mode until power returns to the VDD pin. For more information on using the VBAT feature, refer to **Section 4.5 “Vbat Mode”**.

5.6.5 DEEP SLEEP BROWN-OUT RESET (DSBOR)

The PIC18F97J94 has its dedicated BOR for Deep Sleep mode (DSBOR). It is enabled through the DSBOREN (CONFIG7L<3>) Configuration bit. When the device enters Deep Sleep mode and receives a DSBOR event, the device will not wake-up and will remain in Deep Sleep mode. When a valid wake-up event occurs and causes the device to exit Deep Sleep mode, software can determine if a DSBOR event occurred during Deep Sleep mode by reading the DSBOR (DSCONL<1>) Status bit.

5.7 RESET Instruction

Whenever the RESET instruction is executed, the device asserts $\overline{\text{SYSRST}}$. This Reset state does not re-initialize the clock. The clock source that is in effect prior to the RESET instruction remains in effect. Configuration settings are updated and the $\overline{\text{SYSRST}}$ is released at the next instruction cycle. A noise filter in the $\overline{\text{MCLR}}$ Reset path detects and ignores small pulses. The $\overline{\text{RI}}$ bit (RCON<4>) is cleared when a RESET instruction is executed. Software must set this bit to initialize the flag.

5.8 Stack Underflow/Overflow Reset

A Reset can be enabled on stack error conditions by setting the STVREN (CONFIG1L<5>) Configuration bit. See **Section 6.1.4.4 “Stack Full and Underflow Resets”**section for additional information.

7.2.2 TABLE LATCH REGISTER (TABLAT)

The Table Latch (TABLAT) is an 8-bit register mapped into the Special Function Register (SFR) space. The Table Latch register is used to hold 8-bit data during data transfers between program memory and data RAM.

7.2.3 TABLE POINTER REGISTER (TBLPTR)

The Table Pointer (TBLPTR) register addresses a byte within the program memory. The TBLPTR is comprised of three SFR registers: Table Pointer Upper Byte, Table Pointer High Byte and Table Pointer Low Byte (TBLPTRU:TBLPTRH:TBLPTRL). These three registers join to form a 22-bit wide pointer. The low-order 21 bits allow the device to address up to 2 Mbytes of program memory space. The 22nd bit allows access to the Device ID, the User ID and the Configuration bits.

The Table Pointer register, TBLPTR, is used by the TBLRD and TBLWT instructions. These instructions can update the TBLPTR in one of four ways, based on the table operation. These operations are shown in Table 7-1 and only affect the low-order 21 bits.

7.2.4 TABLE POINTER BOUNDARIES

TBLPTR is used in reads, writes and erases of the Flash program memory.

When a TBLRD is executed, all 22 bits of the TBLPTR determine which byte is read from program memory into TABLAT.

When a TBLWT is executed, the seven Least Significant bits (LSbs) of the Table Pointer register (TBLPTR<6:0>) determine which of the 64 program memory holding registers is written to. When the timed write to program memory begins (via the WR bit), the 12 Most Significant bits (MSbs) of the TBLPTR (TBLPTR<21:10>) determine which program memory block of 1024 bytes is written to. For more detail, see **Section 7.5 "Writing to Flash Program Memory"**.

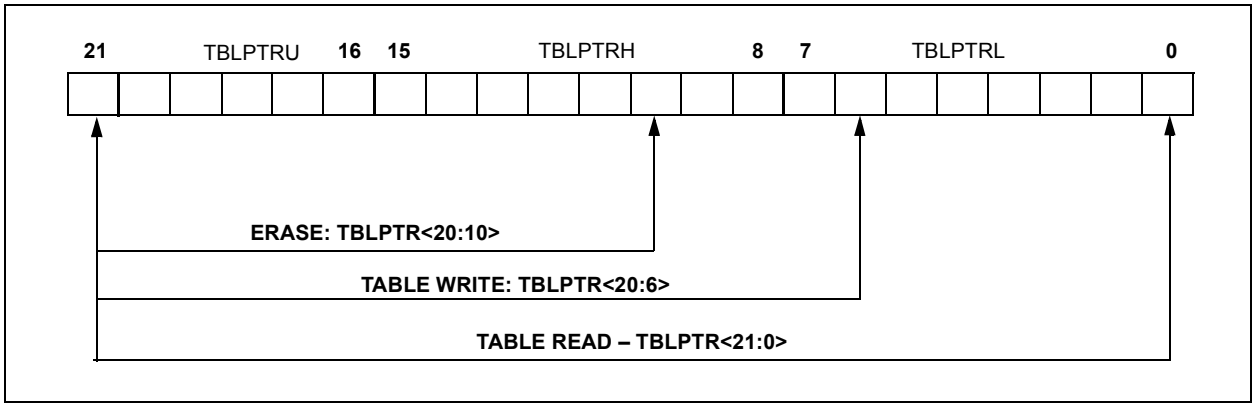
When an erase of program memory is executed, the 12 MSbs of the Table Pointer register point to the 1024-byte block that will be erased. The LSbs are ignored.

Figure 7-3 describes the relevant boundaries of the TBLPTR based on Flash program memory operations.

TABLE 7-1: TABLE POINTER OPERATIONS WITH TBLRD AND TBLWT INSTRUCTIONS

Example	Operation on Table Pointer
TBLRD* TBLWT*	TBLPTR is not modified
TBLRD*+ TBLWT*+	TBLPTR is incremented after the read/write
TBLRD*- TBLWT*-	TBLPTR is decremented after the read/write
TBLRD+* TBLWT+*	TBLPTR is incremented before the read/write

FIGURE 7-3: TABLE POINTER BOUNDARIES BASED ON OPERATION



10.4 INTCON Registers

The INTCON registers are readable and writable registers that contain various enable, priority and flag bits.

Note: Interrupt flag bits are set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

REGISTER 10-1: INTCON: INTERRUPT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0
GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	IOCIE	TMR0IF	INT0IF	IOCIF
bit 7							bit 0

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

bit 7 **GIE/GIEH:** Global Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked interrupts
0 = Disables all interrupts including peripherals
When IPEN = 1:
1 = Enables all high-priority interrupts
0 = Disables all interrupts including low priority

bit 6 **PEIE/GIEL:** Peripheral Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
When IPEN = 1:
1 = Enables all low-priority peripheral interrupts
0 = Disables all low-priority peripheral interrupts

bit 5 **TMR0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 overflow interrupt
0 = Disables the TMR0 overflow interrupt

bit 4 **INT0IE:** INT0 External Interrupt Enable bit
1 = Enables the INT0 external interrupt
0 = Disables the INT0 external interrupt

bit 3 **IOCIE:** I/O Change Interrupt Enable bit
1 = Enables the I/O port change interrupt
0 = Disables the I/O port change interrupt

bit 2 **TMR0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register has not overflowed

bit 1 **INT0IF:** INT0 External Interrupt Flag bit
1 = The INT0 external interrupt occurred (must be cleared in software)
0 = The INT0 external interrupt did not occur

bit 0 **IOCIF:** I/O Port Change Interrupt Flag bit
1 = At least one of the IOC<7:0> pins changed state (must be cleared by clearing all the IOCF bits in the IOC module)
0 = None of the IOC<7:0> pins have changed state

REGISTER 10-3: INTCON3: INTERRUPT CONTROL REGISTER 3

R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF
bit 7							bit 0

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

bit 7 **INT2IP:** INT2 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **INT1IP:** INT1 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **INT3IE:** INT3 External Interrupt Enable bit
1 = Enables the INT3 external interrupt
0 = Disables the INT3 external interrupt

bit 4 **INT2IE:** INT2 External Interrupt Enable bit
1 = Enables the INT2 external interrupt
0 = Disables the INT2 external interrupt

bit 3 **INT1IE:** INT1 External Interrupt Enable bit
1 = Enables the INT1 external interrupt
0 = Disables the INT1 external interrupt

bit 2 **INT3IF:** INT3 External Interrupt Flag bit
1 = The INT3 external interrupt occurred (must be cleared in software)
0 = The INT3 external interrupt did not occur

bit 1 **INT2IF:** INT2 External Interrupt Flag bit
1 = The INT2 external interrupt occurred (must be cleared in software)
0 = The INT2 external interrupt did not occur

bit 0 **INT1IF:** INT1 External Interrupt Flag bit
1 = The INT1 external interrupt occurred (must be cleared in software)
0 = The INT1 external interrupt did not occur

Note: Interrupt flag bits are set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

REGISTER 10-19: IPR4: PERIPHERAL INTERRUPT PRIORITY REGISTER 4

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CCP10IP	CCP9IP	CCP8IP	CCP7IP	CCP6IP	CCP5IP	CCP4IP	ECCP3IP
bit 7							bit 0

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

bit 7 **CCP10IP:** CCP10 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **CCP9IP:** CCP9 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **CCP8IP:** CCP8 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **CCP7IP:** CCP7 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 3 **CCP6IP:** CCP6 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 2 **CCP5IP:** CCP5 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **CCP4IP:** CCP4 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **ECCP3IP:** ECCP3 Interrupt Priority bits
1 = High priority
0 = Low priority

11.0 I/O PORTS

Depending on the device selected and features enabled, there are up to eleven ports available. Some pins of the I/O ports are multiplexed with an alternate function from the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Each port has three memory mapped registers for its operation:

- TRIS register (Data Direction register)
- PORT register (reads the levels on the pins of the device)
- LAT register (Output Latch register)

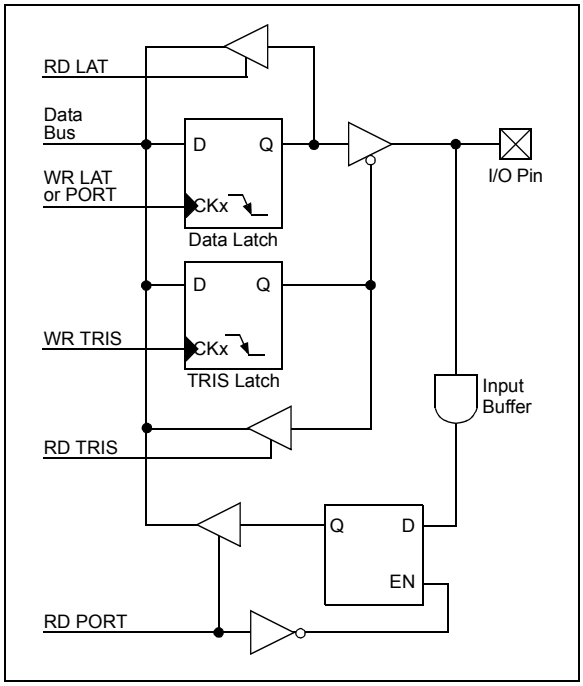
Reading the PORT register reads the current status of the pins, whereas writing to the PORT register, writes to the Output Latch (LAT) register.

Setting a TRIS bit (= 1) makes the corresponding PORT pin an input (putting the corresponding output driver in a High-Impedance mode). Clearing a TRIS bit (= 0) makes the corresponding port pin an output (i.e., driving the contents of the corresponding LAT bit on the selected pin).

The Output Latch (LAT register) is useful for read-modify-write operations on the value that the I/O pins are driving. Read-modify-write operations on the LAT register read and write the latched output value for the PORT register.

A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 11-1.

FIGURE 11-1: GENERIC I/O PORT OPERATION



11.1 I/O Port Pin Capabilities

When developing an application, the capabilities of the port pins must be considered.

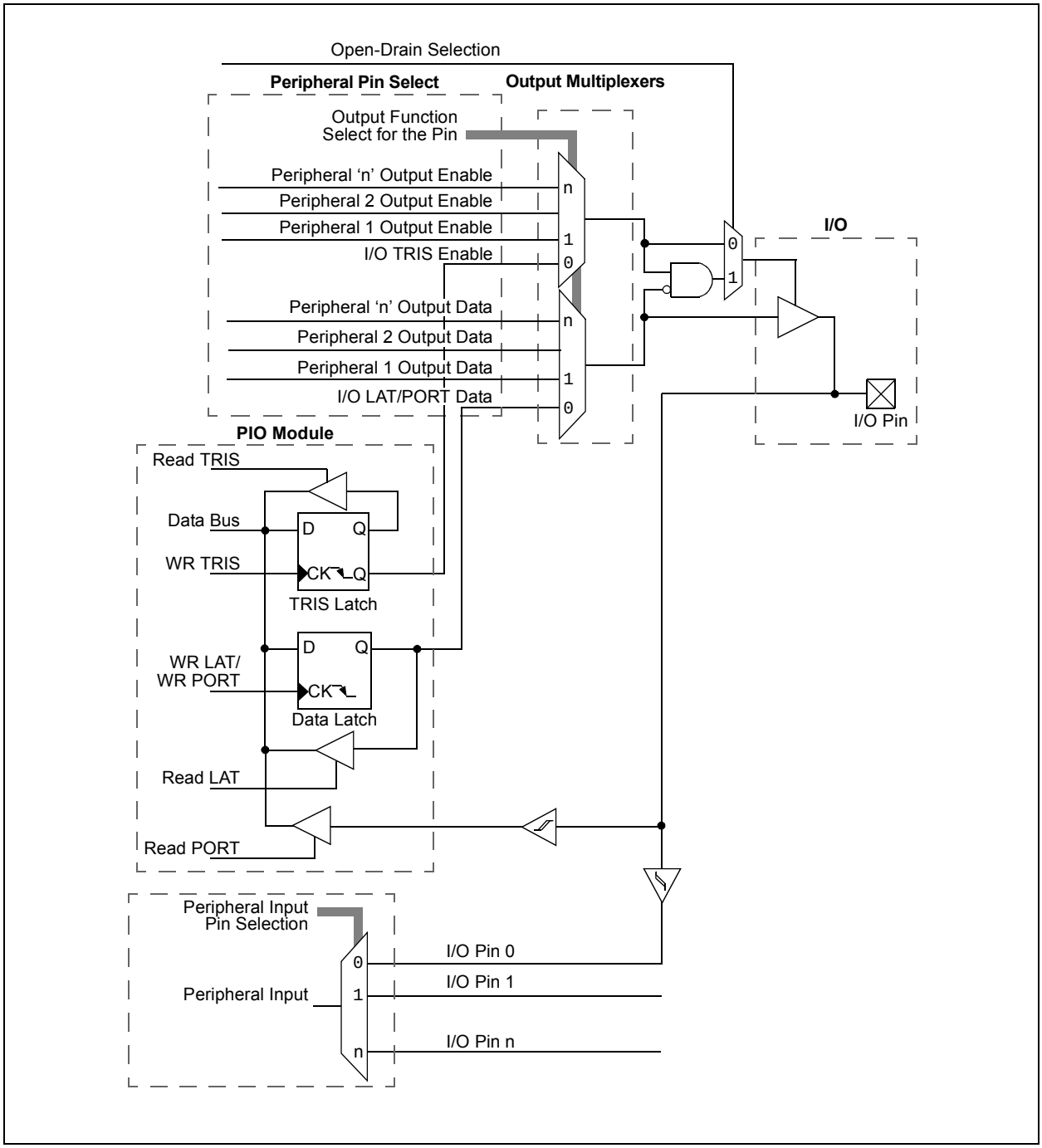
The Absolute Maximum Ratings of the I/O pins are as follows:

- RA2, RA3 = -300mV to (V_{DD} + 300 mV)
- RA6, RA7, RC0, RC1 = -300 mV to (V_{DD} +300 mV)**(1)**
- RF3/RF4 (the USB D+/D- pins) = supports “USB specific levels” (e.g.: -1.0V to +4.6V, but only when the external source impedance is >/= 28 ohms, and the VUSB3V3 pin voltage is >/= 3.0V, otherwise: -500 mV to (VUSB3V3 +500 mV)
- All other general purpose I/O pins (including $\overline{MCLR}$), when V_{DD} is < 2.0V: -300 mV to +4.0V.
- All other general purpose I/O pins (including $\overline{MCLR}$), when V_{DD} is >= 2.0V: -300 mV to +6.0V**(2)**.

Note 1: When the pins are used to drive a crystal or ceramic resonator, natural oscillation waveforms slightly exceeding the -300 mV to (V_{DD} +300 mV) range may sometimes occur, and if present, such waveforms are allowed. If these pins are instead used as general purpose inputs, the external driving source should adhere to the -300 mV to (V_{DD} +300 mV) specification.

2: In addition to the above absolute maximums, any I/O pin voltage that is actively selected at runtime by the ADC channel select MUX must also meet the V_{AIN} requirements (parameter A25 in Table 30-40).

FIGURE 11-6: STRUCTURE OF PORT SHARED WITH PPS PERIPHERALS



13.6 BIAS CONFIGURATIONS

PIC18F97J94 family devices have four distinct circuit configurations for LCD bias generation:

- M0: Regulator with Boost
- M1: Regulator without Boost
- M2: Resistor Ladder with Software Contrast
- M3: Resistor Ladder with Hardware Contrast

13.6.1 M0 (REGULATOR WITH BOOST)

In M0 operation, the LCD charge pump feature is enabled. This allows the regulator to generate voltages up to +3.6V to the LCD (as measured at LCDBIAS3).

M0 uses a flyback capacitor connected between VLCAP1 and VLCAP2, as well as filter capacitors on LCDBIAS0 through LCDBIAS3, to obtain the required voltage boost (Figure 13-6). The output voltage (VBIAS) is the difference of the potential between LCDBIAS3 and LCDBIAS0. It is set by the BIAS<2:0> bits which adjust the offset between LCDBIAS0 and VSS. The flyback capacitor (CFLY) acts as a charge storage element for large LCD loads. This mode is useful in those cases where the voltage requirements of the LCD are higher than the microcontroller's VDD. It also permits software control of the display's contrast, by adjustment of bias voltage, by changing the value of the BIAS bits.

M0 supports static and 1/3 bias types. Generation of the voltage levels for 1/3 bias is handled automatically, but must be configured in software.

M0 is enabled by selecting a valid regulator clock source (CLKSEL<1:0> set to any value except '00') and setting the CPEN bit. If static bias type is required, the MODE13 bit must be cleared.

13.6.2 M1 (REGULATOR WITHOUT BOOST)

M1 operation is similar to M0, but does not use the LCD charge pump. It can provide VBIAS up to the voltage level supplied directly to LCDBIAS3. It can be used in cases where VDD for the application is expected to never drop below a level that can provide adequate contrast for the LCD. The connection of external components is very similar to M0, except that LCDBIAS3 must be tied directly to VDD (Figure 13-6).

Note:	When the device is put to Sleep while operating in mode M0 or M1, make sure that the bias capacitors are fully discharged to get the lowest Sleep current.

The BIAS<2:0> bits can still be used to adjust contrast in software by changing the VBIAS. As with M0, changing these bits changes the offset between LCDBIAS0 and VSS. In M1, this is reflected in the change between the LCDBIAS0 and the voltage tied to LCDBIAS3. Thus, if VDD should change, VBIAS will also change; where in M0, the level of VBIAS is constant.

Like M0, M1 supports static and 1/3 bias types. Generation of the voltage levels for 1/3 bias is handled automatically but must be configured in software. M1 is enabled by selecting a valid regulator clock source (CLKSEL<1:0> set to any value except '00') and clearing the CPEN bit. If 1/3 bias type is required, the MODE13 bit should also be set.

FIGURE 13-9: TYPE-A/TYPE-B WAVEFORMS IN STATIC DRIVE

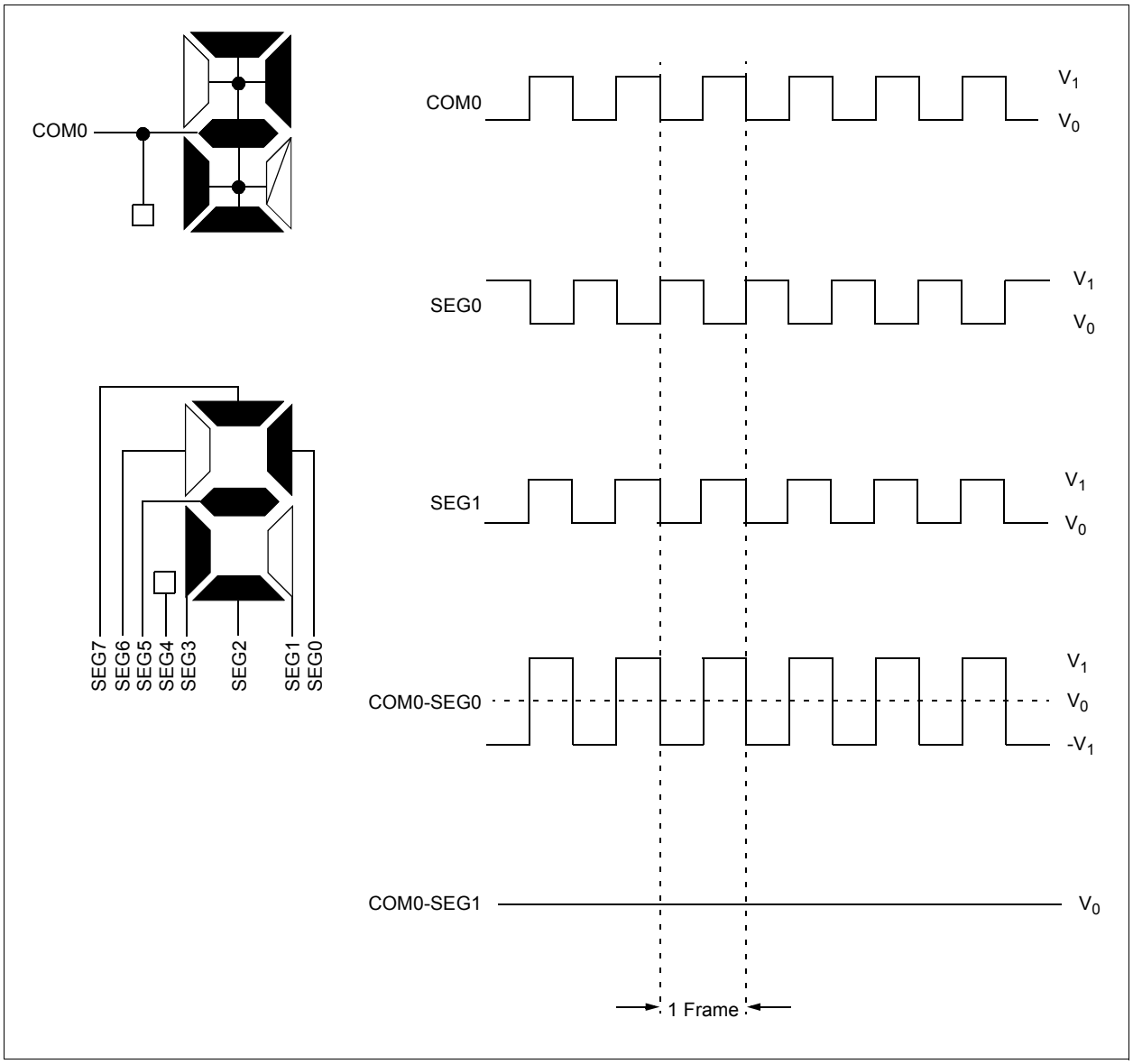


FIGURE 13-11: TYPE-B WAVEFORMS IN 1/2 MUX, 1/2 BIAS DRIVE

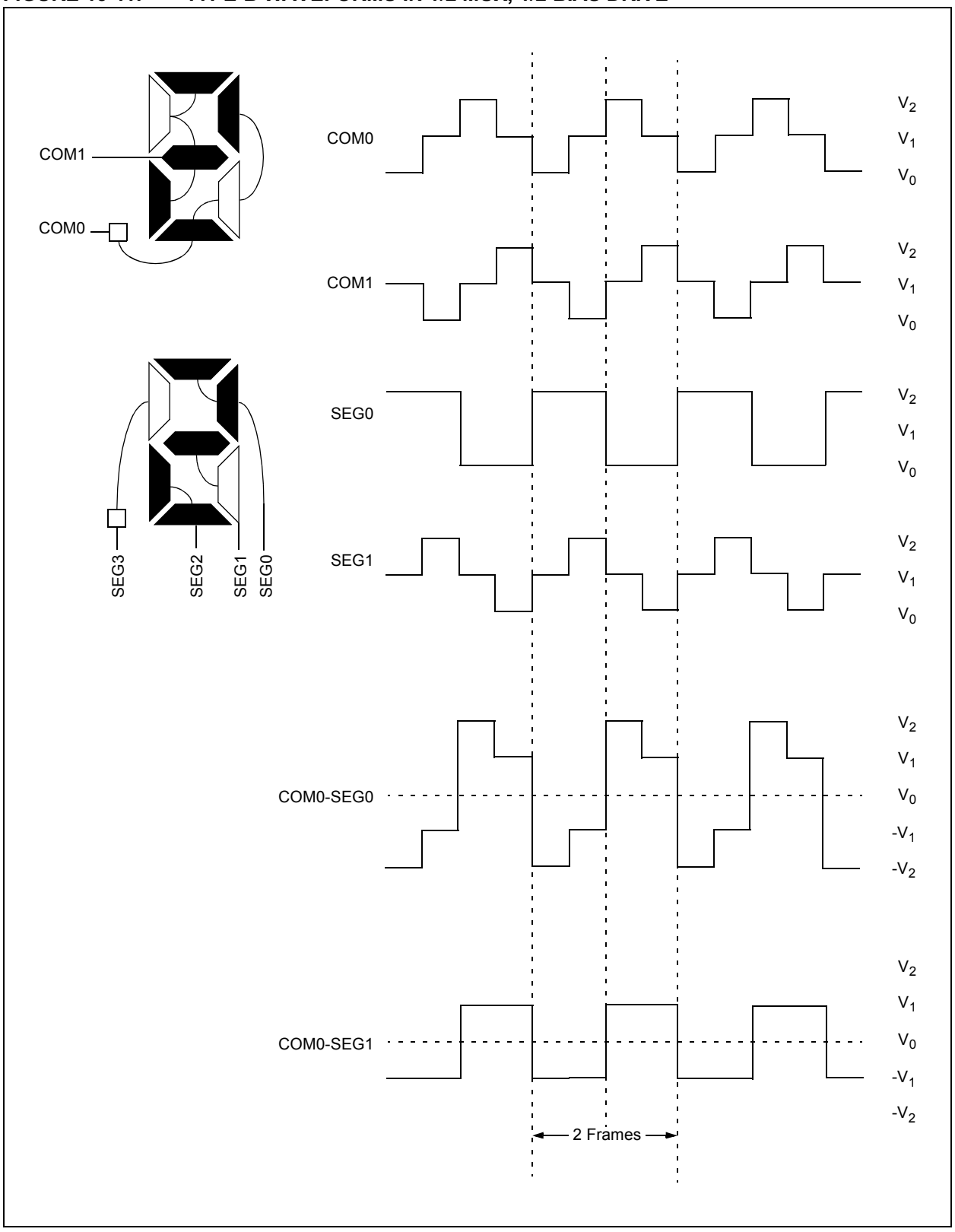
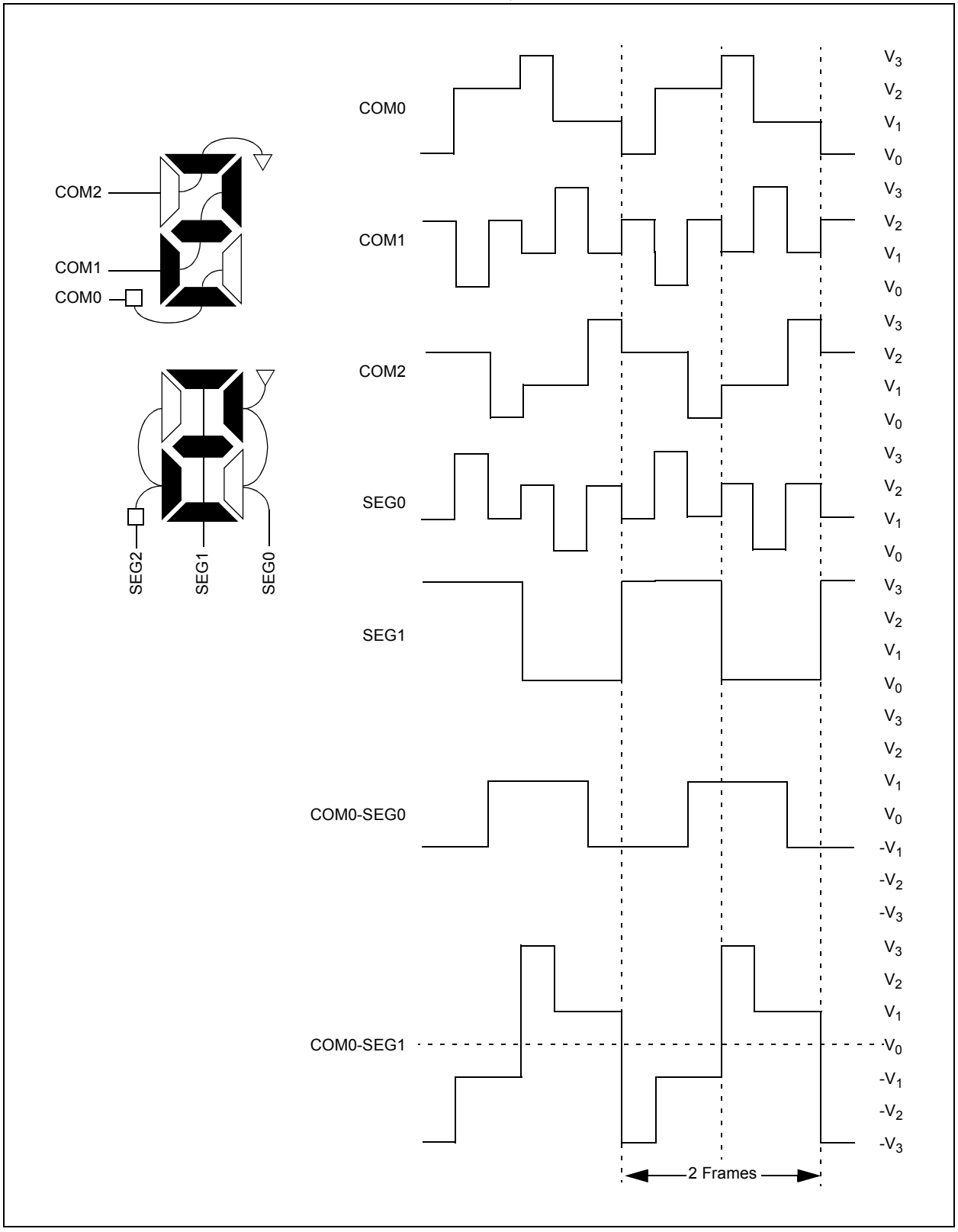


FIGURE 13-17: TYPE-B WAVEFORMS IN 1/3 MUX, 1/3 BIAS DRIVE



17.0 REAL-TIME CLOCK AND CALENDAR (RTCC)

The key features of the Real-Time Clock and Calendar (RTCC) module are:

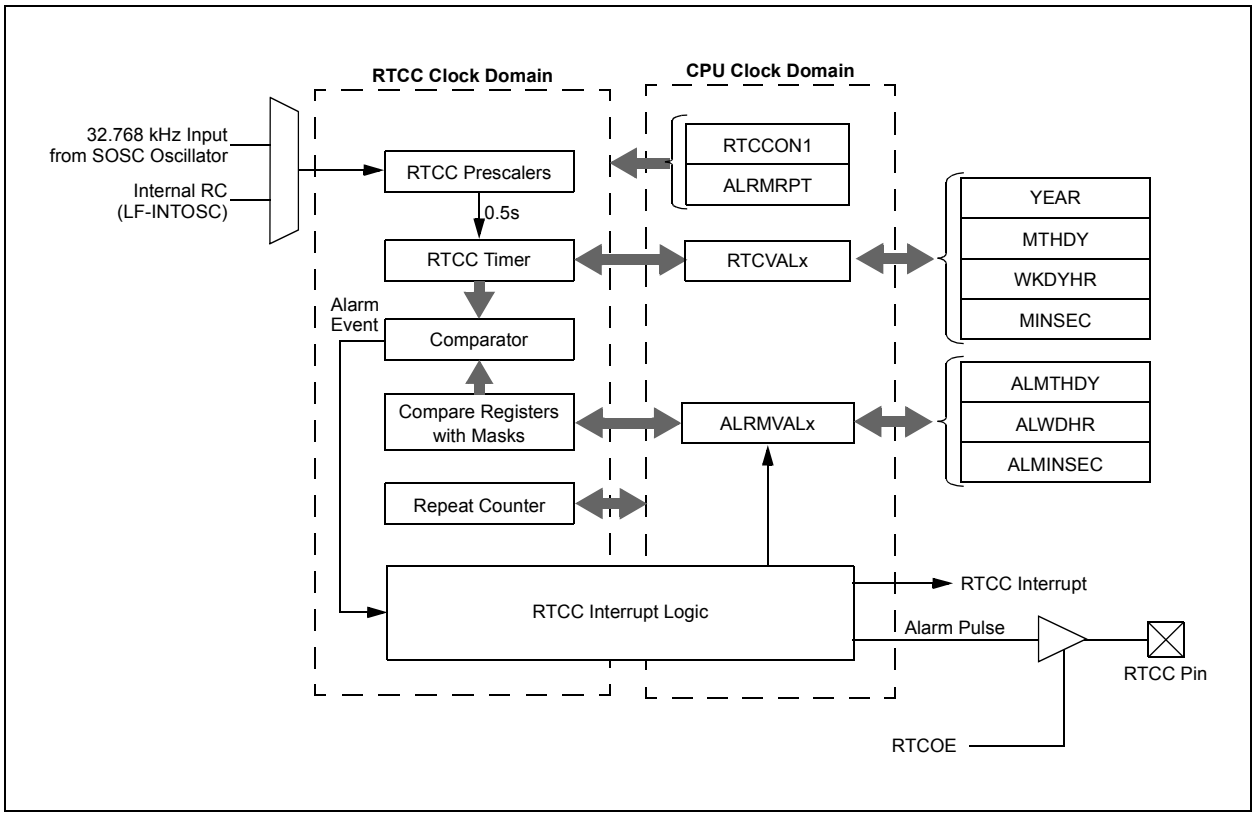
- Hardware Real-Time Clock and Calendar (RTCC)
- Provides hours, minutes and seconds using 24- hour format
- Visibility of one-half second period
- Provides calendar – weekday, date, month and year
- Alarm configurable for half a second, one second, 10 seconds, one minute, 10 minutes, one hour, one day, one week or one month
- Alarm repeat with decrementing counter
- Alarm with indefinite repeat – chime
- Year 2000 to 2099 leap year correction
- BCD format for smaller software overhead
- Optimized for long term battery operation
- Fractional second synchronization
- Multiple clock sources
  - SOSC
  - LPRC
  - 50 Hz
  - 60 Hz
- User calibration of the 32.768 kHz clock crystal frequency with periodic auto-adjust
- Calibration to within ±2.64 seconds error per month
- Calibrates up to 260 ppm of crystal error

The RTCC module is intended for applications where accurate time must be maintained for an extended period with minimum to no intervention from the CPU. The module is optimized for low-power usage in order to provide extended battery life, while keeping track of time.

The module is a 100-year clock and calendar with automatic leap year detection. The range of the clock is from 00:00:00 (midnight) on January 1, 2000 to 23:59:59 on December 31, 2099.

Hours are measured in 24-hour (military time) format. The clock provides a granularity of one second with half-second visibility to the user.

FIGURE 17-1: RTCC BLOCK DIAGRAM



REGISTER 20-11: SSPxCON3: MSSP CONTROL REGISTER 3 (I^2C SLAVE MODE)

R/HS/HC-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

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

bit 7 **ACKTIM:** Acknowledge Time Status bit
1 = Indicates the I^2C bus is in an Acknowledge sequence, set on 8th falling edge of SCL clock
0 = Not an Acknowledge sequence, cleared on 9th rising edge of SCL clock

bit 6 **PCIE:** Stop Condition Interrupt Enable bit(1)
1 = Enable interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled

bit 5 **SCIE:** Start Condition Interrupt Enable bit(1)
1 = Enable interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled

bit 4 **BOEN:** Buffer Overwrite Enable bit
1 = SSPBUF is updated every time a new data byte is available, ignoring the SSPOV effect on updating the buffer
0 = SSPBUF is only updated when SSPOV is clear

bit 3 **SDAHT:** SDA Hold Time Selection bit
1 = Minimum of 300ns hold time on SDA after the falling edge of SCL
0 = Minimum of 100ns hold time on SDA after the falling edge of SCL

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit
If, on the rising edge of SCL, SDA is sampled low when the module is outputting a high state, the BCLIF bit is set, and bus goes Idle.
1 = Enable slave bus collision interrupts
0 = Slave bus collision interrupts are disabled

bit 1 **AHEN:** Address Hold Enable bit
1 = Following the 8th falling edge of SCL for a matching received address byte; CKP bit of SSPxCON1 will be cleared and the SCL will be held low.
0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit
1 = Following the 8th falling edge of SCL for a received data byte; slave hardware clears the CKP bit of SSPCON register and SCL is held low.
0 = Data holding is disabled

Note 1: This bit has no effect in Slave modes that Start and Stop condition detection is explicitly listed as enabled.

REGISTER 22-4: ADCON1H: A/D CONTROL REGISTER 1 HIGH

R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	—	—	—	MODE12	FORM1	FORM0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7 **ADON:** A/D Operating Mode bit
1 = A/D Converter module is operating
0 = A/D Converter is off

bit 6-3 **Unimplemented:** Read as ‘0’

bit 2 **MODE12:** 12-Bit Operation Mode bit
1 = 12-bit A/D operation
0 = 10-bit A/D operation

bit 1-0 **FORM<1:0>:** Data Output Format bits (see following formats)
11 = Fractional result, signed, left-justified
10 = Absolute fractional result, unsigned, left-justified
01 = Decimal result, signed, right-justified
00 = Absolute decimal result, unsigned, right-justified

27.5.3 USB ERROR INTERRUPT STATUS REGISTER (UEIR)

The USB Error Interrupt STATUS register (Register 27-9) contains the flag bits for each of the error sources within the USB peripheral. Each of these sources is controlled by a corresponding interrupt enable bit in the UEIE register. All of the USB error flags are ORed together to generate the USB Error Interrupt Flag (UERRIF) at the top level of the interrupt logic.

Each error bit is set as soon as the error condition is detected. Thus, the interrupt will typically not correspond with the end of a token being processed.

Once an interrupt bit has been set by the SIE, it must be cleared in software by writing a '0'.

Register 27-9: UEIR: USB ERROR INTERRUPT STATUS REGISTER (ACCESS F63h)

R/C-0	U-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
BTSEF	—	—	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF
bit 7							bit 0

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

bit 7 **BTSEF:** Bit Stuff Error Flag bit
1 = A bit stuff error has been detected
0 = No bit stuff error has been detected

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **BTOEF:** Bus Turnaround Time-out Error Flag bit
1 = Bus turnaround time-out has occurred (more than 16 bit times of Idle from previous EOP elapsed)
0 = No bus turnaround time-out has occurred

bit 3 **DFN8EF:** Data Field Size Error Flag bit
1 = The data field was not an integral number of bytes
0 = The data field was an integral number of bytes

bit 2 **CRC16EF:** CRC16 Failure Flag bit
1 = The CRC16 failed
0 = The CRC16 passed

bit 1 **CRC5EF:** CRC5 Host Error Flag bit
1 = The token packet was rejected due to a CRC5 error
0 = The token packet was accepted

bit 0 **PIDEF:** PID Check Failure Flag bit
1 = PID check failed
0 = PID check passed

29.0 INSTRUCTION SET SUMMARY

The PIC18FXXJ94 of devices incorporates the standard set of 75 PIC18 core instructions, as well as an extended set of eight new instructions for the optimization of code that is recursive or that utilizes a software stack. The extended set is discussed later in this section.

29.1 Standard Instruction Set

The standard PIC18 MCU instruction set adds many enhancements to the previous PIC® MCU instruction sets, while maintaining an easy migration from these PIC MCU instruction sets. Most instructions are a single program memory word (16 bits), but there are four instructions that require two program memory locations.

Each single-word instruction is a 16-bit word divided into an opcode, which specifies the instruction type and one or more operands, which further specify the operation of the instruction.

The instruction set is highly orthogonal and is grouped into four basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal** operations
- **Control** operations

The PIC18 instruction set summary in Table 29-2 lists **byte-oriented**, **bit-oriented**, **literal** and **control** operations. Table 29-1 shows the opcode field descriptions.

Most **byte-oriented** instructions have three operands:

1. The file register (specified by ‘f’)
2. The destination of the result (specified by ‘d’)
3. The accessed memory (specified by ‘a’)

The file register designator, ‘f’, specifies which file register is to be used by the instruction. The destination designator, ‘d’, specifies where the result of the operation is to be placed. If ‘d’ is zero, the result is placed in the WREG register. If ‘d’ is one, the result is placed in the file register specified in the instruction.

All **bit-oriented** instructions have three operands:

1. The file register (specified by ‘f’)
2. The bit in the file register (specified by ‘b’)
3. The accessed memory (specified by ‘a’)

The bit field designator, ‘b’, selects the number of the bit affected by the operation, while the file register designator, ‘f’, represents the number of the file in which the bit is located.

The **literal** instructions may use some of the following operands:

- A literal value to be loaded into a file register (specified by ‘k’)
- The desired FSR register to load the literal value into (specified by ‘f’)
- No operand required (specified by ‘—’)

The **control** instructions may use some of the following operands:

- A program memory address (specified by ‘n’)
- The mode of the CALL or RETURN instructions (specified by ‘s’)
- The mode of the table read and table write instructions (specified by ‘m’)
- No operand required (specified by ‘—’)

All instructions are a single word, except for four double-word instructions. These instructions were made double-word to contain the required information in 32 bits. In the second word, the 4 MSbs are ‘1’s. If this second word is executed as an instruction (by itself), it will execute as a NOP.

All single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the Program Counter is changed as a result of the instruction. In these cases, the execution takes two instruction cycles with the additional instruction cycle(s) executed as a NOP.

The double-word instructions execute in two instruction cycles.

One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true, or the Program Counter is changed as a result of an instruction, the instruction execution time is 2 μs. Two-word branch instructions (if true) would take 3 μs.

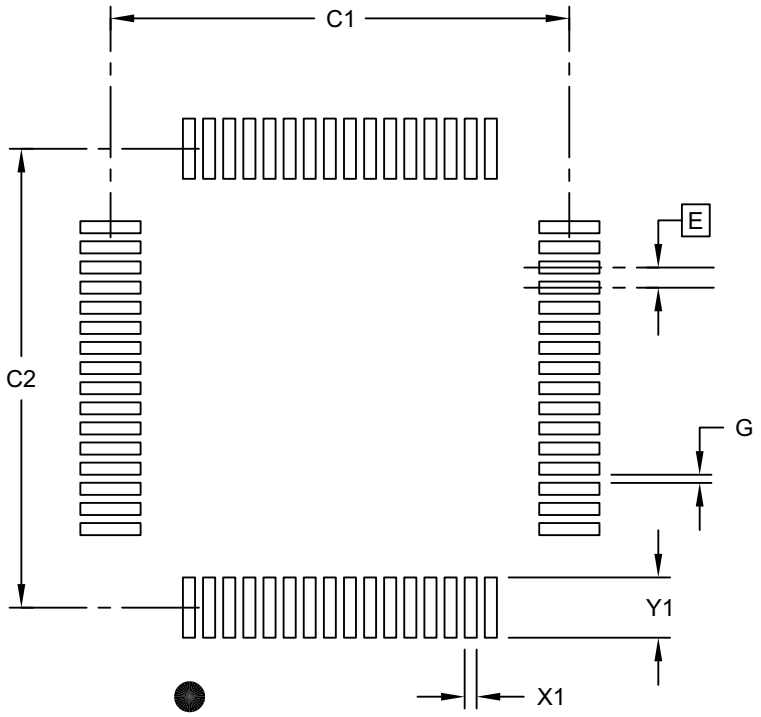
Figure 29-1 shows the general formats that the instructions can have. All examples use the convention ‘nnh’ to represent a hexadecimal number.

The Instruction Set Summary, shown in Table 29-2, lists the standard instructions recognized by the Microchip MPASM™ Assembler.

Section 29.1.1 “Standard Instruction Set” provides a description of each instruction.

64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X28)	X1			0.30
Contact Pad Length (X28)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2085B Sheet 1 of 1