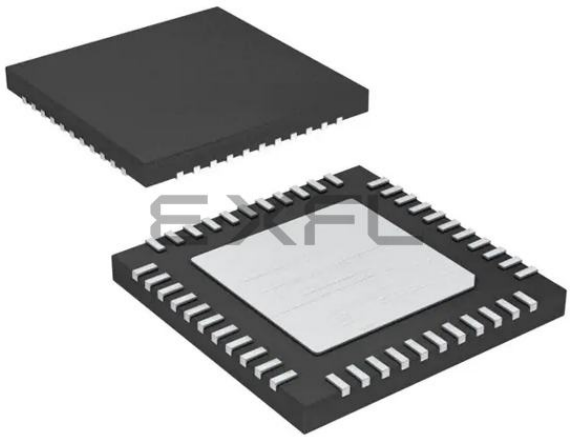


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

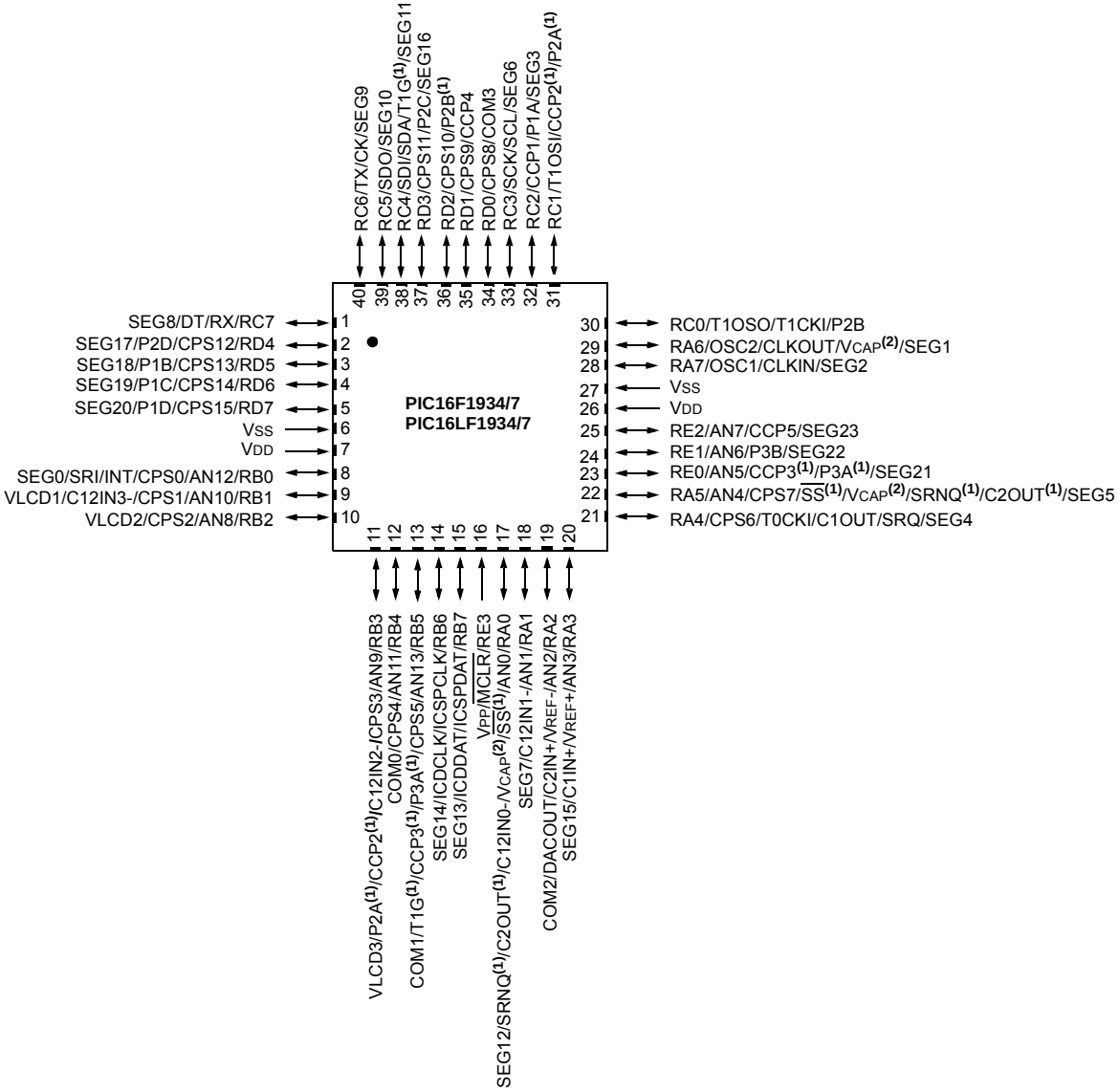
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 14x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1937t-i-ml

Pin Diagram – 40-Pin UQFN 5X5 (PIC16(L)F1934/7)

40-pin UQFN



Note 1: Pin function is selectable via the APFCON register.

2: PIC16F1934/7 devices only.

1.0 DEVICE OVERVIEW

The PIC16(L)F1934/6/7 are described within this data sheet. They are available in 28/40/44-pin packages. Figure 1-1 shows a block diagram of the PIC16(L)F1934/6/7 devices. Table 1-2 shows the pinout descriptions.

Reference Table 1-1 for peripherals available per device.

TABLE 1-1: DEVICE PERIPHERAL SUMMARY

Peripheral		PIC16F193X	PIC16LF193X
ADC		●	●
Capacitive Sensing Module		●	●
Digital-to-Analog Converter (DAC)		●	●
EUSART		●	●
Fixed Voltage Reference (FVR)		●	●
LCD		●	●
SR Latch		●	●
Temperature Indicator		●	●
Capture/Compare/PWM Modules			
	ECCP1	●	●
	ECCP2	●	●
	ECCP3	●	●
	CCP4	●	●
	CCP5	●	●
Comparators			
	C1	●	●
	C2	●	●
Operational Amplifiers			
	OPA1	●	●
	OPA2	●	●
Master Synchronous Serial Ports			
	MSSP1	●	●
Timers			
	Timer0	●	●
	Timer1	●	●
	Timer2	●	●
	Timer4	●	●
	Timer6	●	●

3.2.1.1 STATUS Register

The STATUS register, shown in Register 3-1, contains:

- the arithmetic status of the ALU
- the Reset status

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as '`000u u1uu`' (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (Refer to **Section 29.0 "Instruction Set Summary"**).

Note 1: The C and DC bits operate as $\overline{Borrow}$ and $\overline{Digit\ Borrow}$ out bits, respectively, in subtraction.

REGISTER 3-1: STATUS: STATUS REGISTER

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC(1)	C(1)
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **$\overline{TO}$:** Time-out bit
1 = After power-up, `CLRWDT` instruction or `SLEEP` instruction
0 = A WDT time-out occurred

bit 3 **$\overline{PD}$:** Power-down bit
1 = After power-up or by the `CLRWDT` instruction
0 = By execution of the `SLEEP` instruction

bit 2 **Z:** Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit Carry/$\overline{Digit\ Borrow}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)(1)
1 = A carry-out from the 4th low-order bit of the result occurred
0 = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/$\overline{Borrow}$ bit(1) (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)(1)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note 1: For $\overline{Borrow}$, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

TABLE 3-12: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

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
Bank 31											
F80h(2)	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
F81h(2)	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
F82h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
F83h(2)	STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu
F84h(2)	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
F85h(2)	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
F86h(2)	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
F87h(2)	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
F88h(2)	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
F89h(2)	WREG	Working Register								0000 0000	uuuu uuuu
F8Ah(1),(2)	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
F8Bh(2)	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
F8Ch — FE3h	—	Unimplemented								—	—
FE4h	STATUS_ SHAD						Z_SHAD	DC_SHAD	C_SHAD	---- -xxx	---- -uuu
FE5h	WREG_ SHAD	Working Register Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FE6h	BSR_ SHAD				Bank Select Register Normal (Non-ICD) Shadow					---x xxxx	---u uuuu
FE7h	PCLATH_ SHAD		Program Counter Latch High Register Normal (Non-ICD) Shadow							-xxx xxxx	uuuu uuuu
FE8h	FSR0L_ SHAD	Indirect Data Memory Address 0 Low Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FE9h	FSR0H_ SHAD	Indirect Data Memory Address 0 High Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FEAh	FSR1L_ SHAD	Indirect Data Memory Address 1 Low Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FEBh	FSR1H_ SHAD	Indirect Data Memory Address 1 High Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FECh	—	Unimplemented								—	—
FEDh	STKPTR	—	—	—	Current Stack pointer					---1 1111	---1 1111
FEEh	TOSL	Top of Stack Low byte								xxxx xxxx	uuuu uuuu
FEFh	TOSH	—	Top of Stack High byte							-xxx xxxx	-uuu uuuu

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<14:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: These registers/bits are not implemented on PIC16(L)F1936 devices, read as '0'.

4: Unimplemented, read as '1'.

© 2008-2011 Microchip Technology Inc.

5.2.2.3 Internal Oscillator Frequency Adjustment

The 500 kHz internal oscillator is factory calibrated. This internal oscillator can be adjusted in software by writing to the OSCTUNE register (Register 5-3). Since the HFINTOSC and MFINTOSC clock sources are derived from the 500 kHz internal oscillator a change in the OSCTUNE register value will apply to both.

The default value of the OSCTUNE register is '0'. The value is a 5-bit two's complement number. A value of 0Fh will provide an adjustment to the maximum frequency. A value of 10h will provide an adjustment to the minimum frequency.

When the OSCTUNE register is modified, the oscillator frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

OSCTUNE does not affect the LFINTOSC frequency. Operation of features that depend on the LFINTOSC clock source frequency, such as the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and peripherals, are *not* affected by the change in frequency.

5.2.2.4 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is an uncalibrated 31 kHz internal clock source.

The output of the LFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). Select 31 kHz, via software, using the IRCF<3:0> bits of the OSCCON register. See **Section 5.2.2.7 "Internal Oscillator Clock Switch Timing"** for more information. The LFINTOSC is also the frequency for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

The LFINTOSC is enabled by selecting 31 kHz (IRCF<3:0> bits of the OSCCON register = 000) as the system clock source (SCS bits of the OSCCON register = 1x), or when any of the following are enabled:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired LF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'

Peripherals that use the LFINTOSC are:

- Power-up Timer (PWRT)
- Watchdog Timer (WDT)
- Fail-Safe Clock Monitor (FSCM)

The Low-Frequency Internal Oscillator Ready bit (LFIOFR) of the OSCSTAT register indicates when the LFINTOSC is running and can be utilized.

5.2.2.5 Internal Oscillator Frequency Selection

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register.

The output of the 16 MHz HFINTOSC and 31 kHz LFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). The Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register select the frequency output of the internal oscillators. One of the following frequencies can be selected via software:

- 32 MHz (requires 4X PLL)
- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz
- 500 kHz (Default after Reset)
- 250 kHz
- 125 kHz
- 62.5 kHz
- 31.25 kHz
- 31 kHz (LFINTOSC)

Note:	Following any Reset, the IRCF<3:0> bits of the OSCCON register are set to '0111' and the frequency selection is set to 500 kHz. The user can modify the IRCF bits to select a different frequency.

The IRCF<3:0> bits of the OSCCON register allow duplicate selections for some frequencies. These duplicate choices can offer system design trade-offs. Lower power consumption can be obtained when changing oscillator sources for a given frequency. Faster transition times can be obtained between frequency changes that use the same oscillator source.

EXAMPLE 11-4: ERASING ONE ROW OF PROGRAM MEMORY -

```
; This row erase routine assumes the following:
; 1. A valid address within the erase block is loaded in ADDRH:ADDRL
; 2. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F (common RAM)


        BCF         INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL     EEADRL
        MOVF        ADDRL,W         ; Load lower 8 bits of erase address boundary
        MOVWF       EEADRL
        MOVF        ADDRH,W         ; Load upper 6 bits of erase address boundary
        MOVWF       EEADRH
        BSF         EECON1,EEPGD    ; Point to program memory
        BCF         EECON1,CFGS     ; Not configuration space
        BSF         EECON1,FREE     ; Specify an erase operation
        BSF         EECON1,WREN     ; Enable writes

Required
Sequence
        MOVLW       55h             ; Start of required sequence to initiate erase
        MOVWF       EECON2          ; Write 55h
        MOVLW       0AAh            ;
        MOVWF       EECON2          ; Write AAh
        BSF         EECON1,WR       ; Set WR bit to begin erase
        NOP                         ; Any instructions here are ignored as processor
                                    ; halts to begin erase sequence
        NOP                         ; Processor will stop here and wait for erase complete.

                                    ; after erase processor continues with 3rd instruction

        BCF         EECON1,WREN     ; Disable writes
        BSF         INTCON,GIE      ; Enable interrupts
```


© 2008-2011 Microchip Technology Inc.

19.0 SR LATCH

The module consists of a single SR Latch with multiple Set and Reset inputs as well as separate latch outputs. The SR Latch module includes the following features:

- Programmable input selection
- SR Latch output is available externally
- Separate Q and $\overline{Q}$ outputs
- Firmware Set and Reset

The SR Latch can be used in a variety of analog applications, including oscillator circuits, one-shot circuit, hysteretic controllers, and analog timing applications.

19.1 Latch Operation

The latch is a Set-Reset Latch that does not depend on a clock source. Each of the Set and Reset inputs are active-high. The latch can be set or reset by:

- Software control (SRPS and SRPR bits)
- Comparator C1 output (SYNCC1OUT)
- Comparator C2 output (SYNCC2OUT)
- SRI pin
- Programmable clock (SRCLK)

The SRPS and the SRPR bits of the SRCON0 register may be used to set or reset the SR Latch, respectively. The latch is Reset-dominant. Therefore, if both Set and Reset inputs are high, the latch will go to the Reset state. Both the SRPS and SRPR bits are self resetting which means that a single write to either of the bits is all that is necessary to complete a latch Set or Reset operation.

The output from Comparator C1 or C2 can be used as the Set or Reset inputs of the SR Latch. The output of either Comparator can be synchronized to the Timer1 clock source. See **Section 18.0 "Comparator Module"** and **Section 21.0 "Timer1 Module with Gate Control"** for more information.

An external source on the SRI pin can be used as the Set or Reset inputs of the SR Latch.

An internal clock source is available that can periodically set or reset the SR Latch. The SRCLK<2:0> bits in the SRCON0 register are used to select the clock source period. The SRSCKE and SRRCKE bits of the SRCON1 register enable the clock source to set or reset the SR Latch, respectively.

Note:	Enabling both the Set and Reset inputs from any one source at the same time may result in indeterminate operation, as the Reset dominance cannot be assured.

19.2 Latch Output

The SRQEN and SRNQEN bits of the SRCON0 register control the Q and $\overline{Q}$ latch outputs. Both of the SR Latch outputs may be directly output to an I/O pin at the same time. The $\overline{Q}$ latch output pin function can be moved to an alternate pin using the SRNQSEL bit of the APFCON register.

The applicable TRIS bit of the corresponding port must be cleared to enable the port pin output driver.

19.3 Effects of a Reset

Upon any device Reset, the SR Latch output is not initialized to a known state. The user's firmware is responsible for initializing the latch output before enabling the output pins.

FIGURE 24-12: I²C START AND STOP CONDITIONS

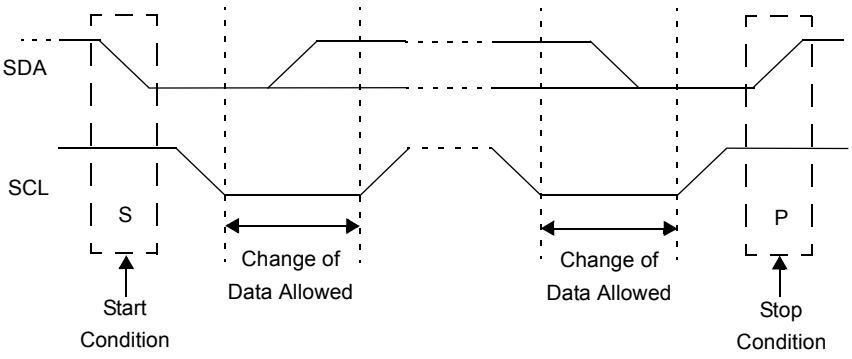
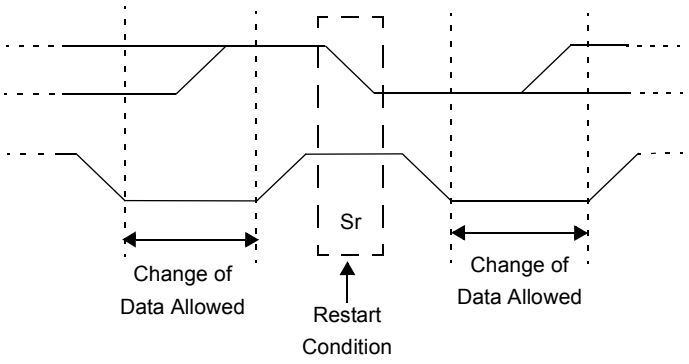


FIGURE 24-13: I²C RESTART CONDITION



24.5.3 SLAVE TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register, and an $\overline{ACK}$ pulse is sent by the slave on the ninth bit.

Following the $\overline{ACK}$, slave hardware clears the CKP bit and the SCL pin is held low (see **Section 24.5.6 "Clock Stretching"** for more detail). By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data.

The transmit data must be loaded into the SSPBUF register which also loads the SSPSR register. Then the SCL pin should be released by setting the CKP bit of the SSPCON1 register. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time.

The $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. This $\overline{ACK}$ value is copied to the ACKSTAT bit of the SSPCON2 register. If ACKSTAT is set (not $\overline{ACK}$), then the data transfer is complete. In this case, when the not $\overline{ACK}$ is latched by the slave, the slave goes Idle and waits for another occurrence of the Start bit. If the SDA line was low ($\overline{ACK}$), the next transmit data must be loaded into the SSPBUF register. Again, the SCL pin must be released by setting bit CKP.

An MSSP interrupt is generated for each data transfer byte. The SSPIF bit must be cleared by software and the SSPSTAT register is used to determine the status of the byte. The SSPIF bit is set on the falling edge of the ninth clock pulse.

24.5.3.1 Slave Mode Bus Collision

A slave receives a Read request and begins shifting data out on the SDA line. If a bus collision is detected and the SBCDE bit of the SSPCON3 register is set, the BCLIF bit of the PIR register is set. Once a bus collision is detected, the slave goes Idle and waits to be addressed again. User software can use the BCLIF bit to handle a slave bus collision.

24.5.3.2 7-bit Transmission

A master device can transmit a read request to a slave, and then clock data out of the slave. The list below outlines what software for a slave will need to do to accomplish a standard transmission. Figure 24-17 can be used as a reference to this list.

1. Master sends a Start condition on SDA and SCL.
2. S bit of SSPSTAT is set; SSPIF is set if interrupt on Start detect is enabled.
3. Matching address with R/$\overline{W}$ bit set is received by the Slave setting SSPIF bit.
4. Slave hardware generates an $\overline{ACK}$ and sets SSPIF.
5. SSPIF bit is cleared by user.
6. Software reads the received address from SSPBUF, clearing BF.
7. R/$\overline{W}$ is set so CKP was automatically cleared after the $\overline{ACK}$.
8. The slave software loads the transmit data into SSPBUF.
9. CKP bit is set releasing SCL, allowing the master to clock the data out of the slave.
10. SSPIF is set after the $\overline{ACK}$ response from the master is loaded into the ACKSTAT register.
11. SSPIF bit is cleared.
12. The slave software checks the ACKSTAT bit to see if the master wants to clock out more data.

Note 1: If the master $\overline{ACK}$s the clock will be stretched.

2: ACKSTAT is the only bit updated on the rising edge of SCL (9th) rather than the falling.

13. Steps 9-13 are repeated for each transmitted byte.
14. If the master sends a not $\overline{ACK}$; the clock is not held, but SSPIF is still set.
15. The master sends a Restart condition or a Stop.
16. The slave is no longer addressed.

© 2008-2011 Microchip Technology Inc.

REGISTER 25-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER(1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **SPEN:** Serial Port Enable bit
1 = Serial port enabled (configures RX/DT and TX/CK pins as serial port pins)
0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit
Asynchronous mode:
Don't care
Synchronous mode – Master:
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode – Slave
Don't care

bit 4 **CREN:** Continuous Receive Enable bit
Asynchronous mode:
1 = Enables receiver
0 = Disables receiver
Synchronous mode:
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit
Asynchronous mode 9-bit (RX9 = 1):
1 = Enables address detection, enable interrupt and load the receive buffer when RSR<8> is set
0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit
Asynchronous mode 8-bit (RX9 = 0):
Don't care

bit 2 **FERR:** Framing Error bit
1 = Framing error (can be updated by reading RCREG register and receive next valid byte)
0 = No framing error

bit 1 **OERR:** Overrun Error bit
1 = Overrun error (can be cleared by clearing bit CREN)
0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data
This can be address/data bit or a parity bit and must be calculated by user firmware.

FIGURE 25-7: AUTO-WAKE-UP BIT (WUE) TIMING DURING NORMAL OPERATION

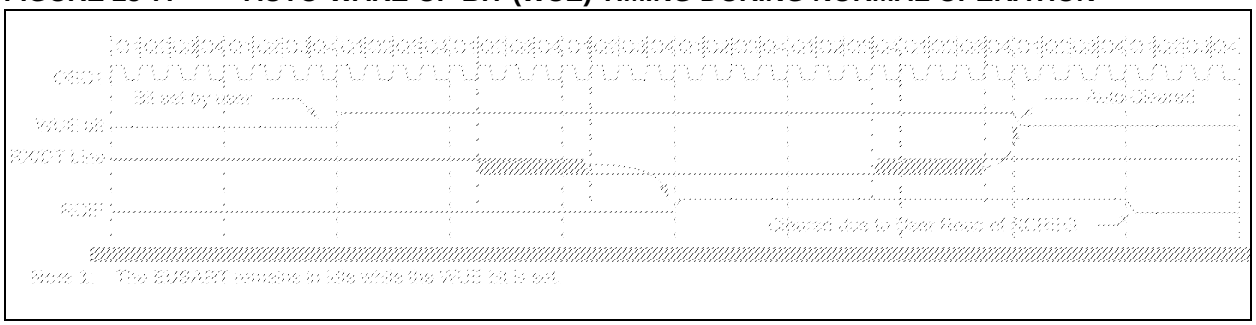
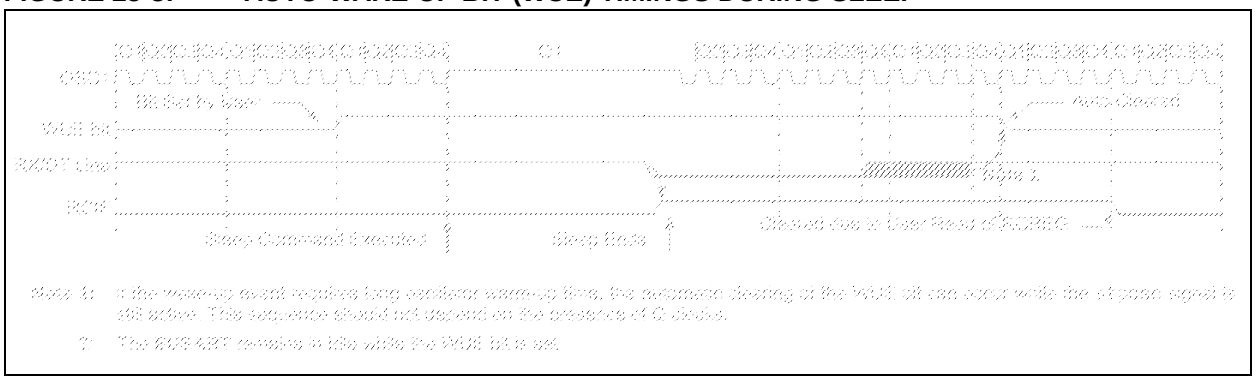


FIGURE 25-8: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING SLEEP



© 2008-2011 Microchip Technology Inc.

27.9 LCD Waveform Generation

LCD waveforms are generated so that the net AC voltage across the dark pixel should be maximized and the net AC voltage across the clear pixel should be minimized. The net DC voltage across any pixel should be zero.

The COM signal represents the time slice for each common, while the SEG contains the pixel data.

The pixel signal (COM-SEG) will have no DC component and it can take only one of the two RMS values. The higher RMS value will create a dark pixel and a lower RMS value will create a clear pixel.

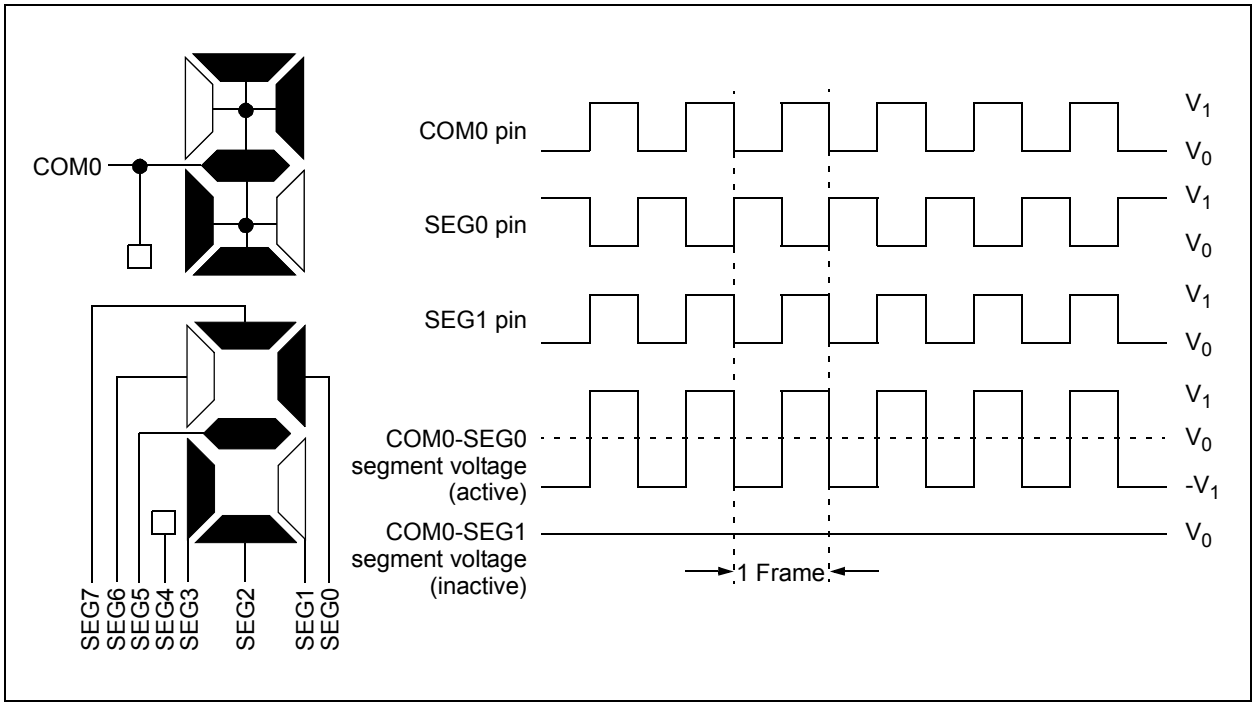
As the number of commons increases, the delta between the two RMS values decreases. The delta represents the maximum contrast that the display can have.

The LCDs can be driven by two types of waveform: Type-A and Type-B. In Type-A waveform, the phase changes within each common type, whereas in Type-B waveform, the phase changes on each frame boundary. Thus, Type-A waveform maintains 0 VDC over a single frame, whereas Type-B waveform takes two frames.

Note 1:	If Sleep has to be executed with LCD Sleep disabled (LCDCON<SLPEN> is '1'), then care must be taken to execute Sleep only when VDC on all the pixels is '0'.
2:	When the LCD clock source is FOSC/256, if Sleep is executed, irrespective of the LCDCON<SLPEN> setting, the LCD immediately goes into Sleep. Thus, take care to see that VDC on all pixels is '0' when Sleep is executed.

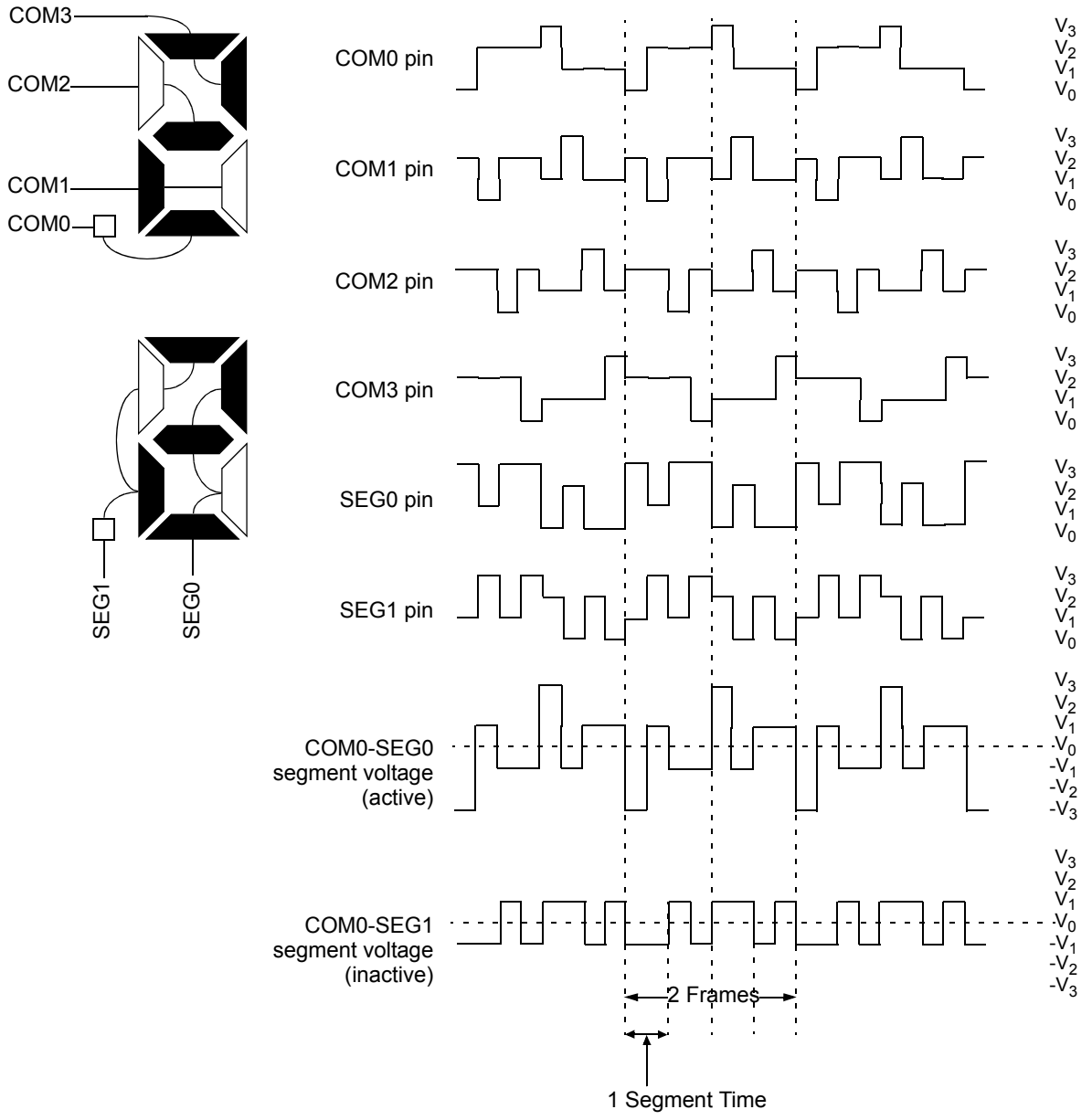
Figure 27-8 through Figure 27-18 provide waveforms for static, half-multiplex, 1/3-multiplex and 1/4-multiplex drives for Type-A and Type-B waveforms.

FIGURE 27-8: TYPE-A/TYPE-B WAVEFORMS IN STATIC DRIVE



© 2008-2011 Microchip Technology Inc.

FIGURE 27-18: TYPE-B WAVEFORMS IN 1/4 MUX, 1/3 BIAS DRIVE



Note: 1 Frame = 2 single segment times.

© 2008-2011 Microchip Technology Inc.

29.0 INSTRUCTION SET SUMMARY

Each PIC16 instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 29-3 lists the instructions recognized by the MPASM™ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of 4 oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

29.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

TABLE 29-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
n	FSR or INDF number. (0-1)
mm	Pre-post increment-decrement mode selection

TABLE 29-2: ABBREVIATION DESCRIPTIONS

Field	Description
PC	Program Counter
$\overline{TO}$	Time-out bit
C	Carry bit
DC	Digit carry bit
Z	Zero bit
$\overline{PD}$	Power-down bit

30.2 DC Characteristics: PIC16(L)F1934/6/7-I/E (Industrial, Extended) (Continued)

PIC16LF1934/36/37			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
PIC16F1934/36/37			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Device Characteristics**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**	
						VDD	**Note**
	Supply Current (IDD)(1, 2)						
D013		—	50	100	µA	1.8	FOSC = 500 kHz EC Oscillator Low-Power mode
		—	85	150	µA	3.0	
D013		—	70	120	µA	1.8	FOSC = 500 kHz EC Oscillator Low-Power mode **(Note 5)**
		—	115	170	µA	3.0	
		—	120	200	µA	5.0	
D014		—	400	550	µA	1.8	FOSC = 4 MHz EC Oscillator mode Medium Power mode
		—	700	1100	µA	3.0	
D014		—	430	650	µA	1.8	FOSC = 4 MHz EC Oscillator mode **(Note 5)** Medium Power mode
		—	720	1000	µA	3.0	
		—	850	1200	µA	5.0	
D015		—	5.3	6.2	mA	3.0	FOSC = 32 MHz EC Oscillator High-Power mode
		—	6.3	7.5	mA	3.6	
D015		—	5.3	6.5	mA	3.0	FOSC = 32 MHz EC Oscillator High-Power mode **(Note 5)**
		—	5.4	7.5	mA	5.0	
D016		—	5	12	µA	1.8	FOSC = 32 kHz, LFINTOSC mode **(Note 4)** -40°C ≤ TA ≤ +85°C
		—	8	16	µA	3.0	
D016		—	27	70	µA	1.8	FOSC = 32 kHz, LFINTOSC mode **(Note 4, Note 5)** -40°C ≤ TA ≤ +85°C
		—	34	80	µA	3.0	
		—	36	90	µA	5.0	

Note 1: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; $\overline{MCLR}$ = VDD; WDT disabled.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

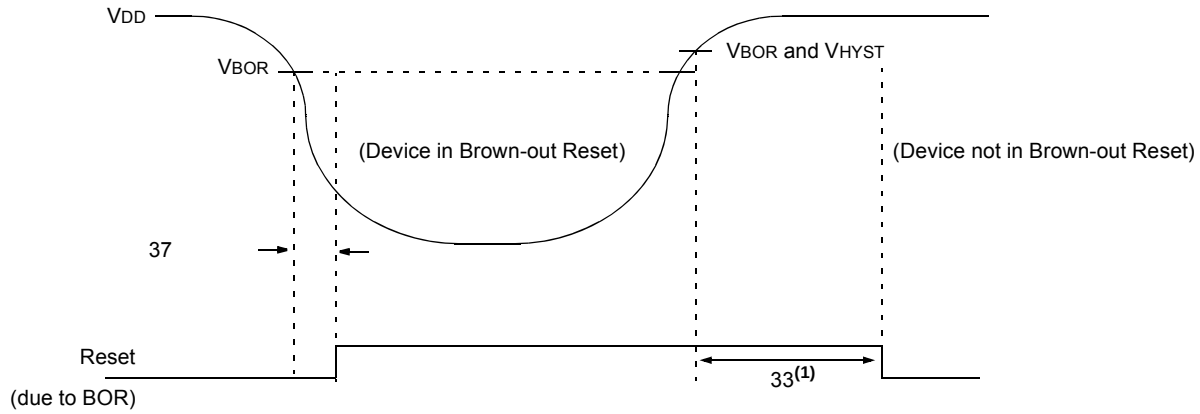
3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in kΩ.

4: FVR and BOR are disabled.

5: 0.1 µF capacitor on VCAP (RA0).

6: 8 MHz crystal oscillator with 4x PLL enabled.

FIGURE 30-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS



Note 1: 64 ms delay only if $\overline{\text{PWRTE}}$ bit in the Configuration Word register is programmed to '0'.
2 ms delay if $\overline{\text{PWRTE}}$ = 0 and VREGEN = 1.

© 2008-2011 Microchip Technology Inc.

FIGURE 31-56: PIC16LF1937 COMPARATOR 2, HIGH POWER

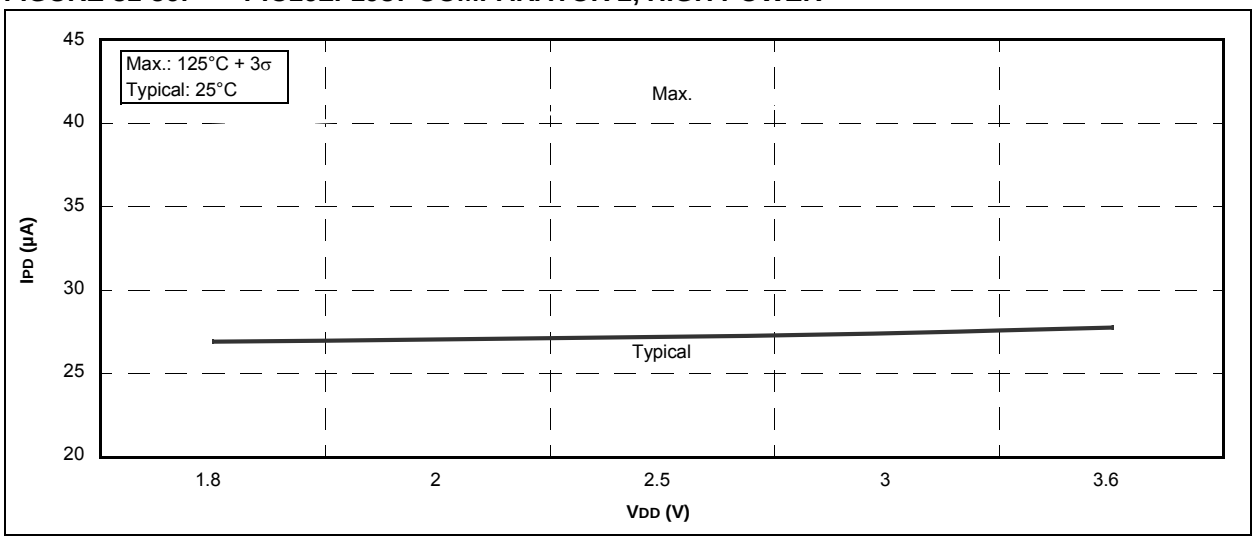


FIGURE 31-57: PIC16F1937 COMPARATOR 2, HIGH POWER

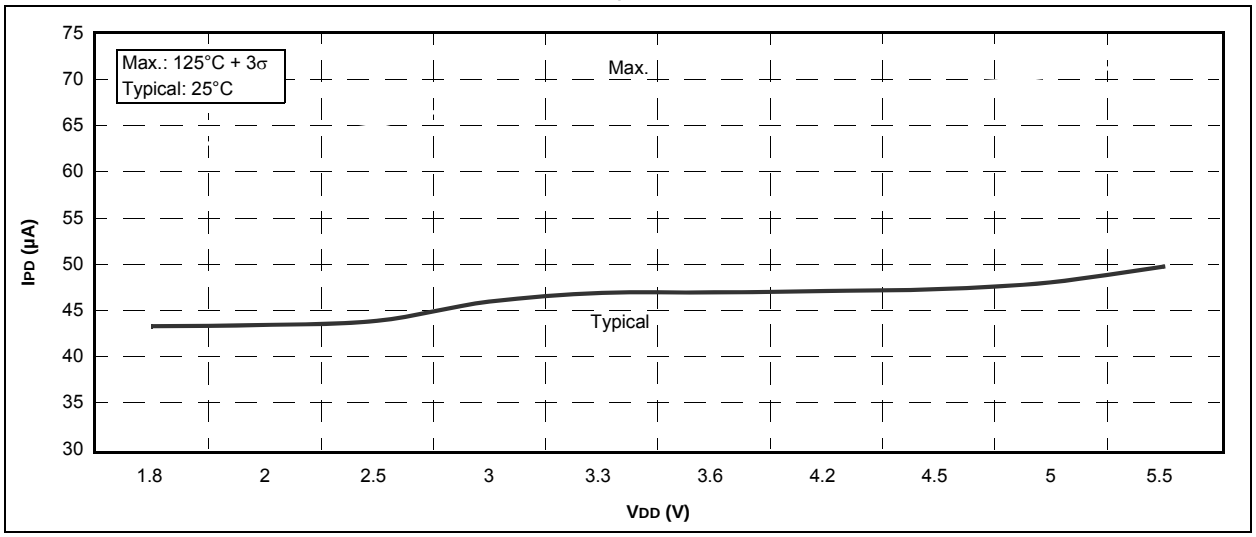
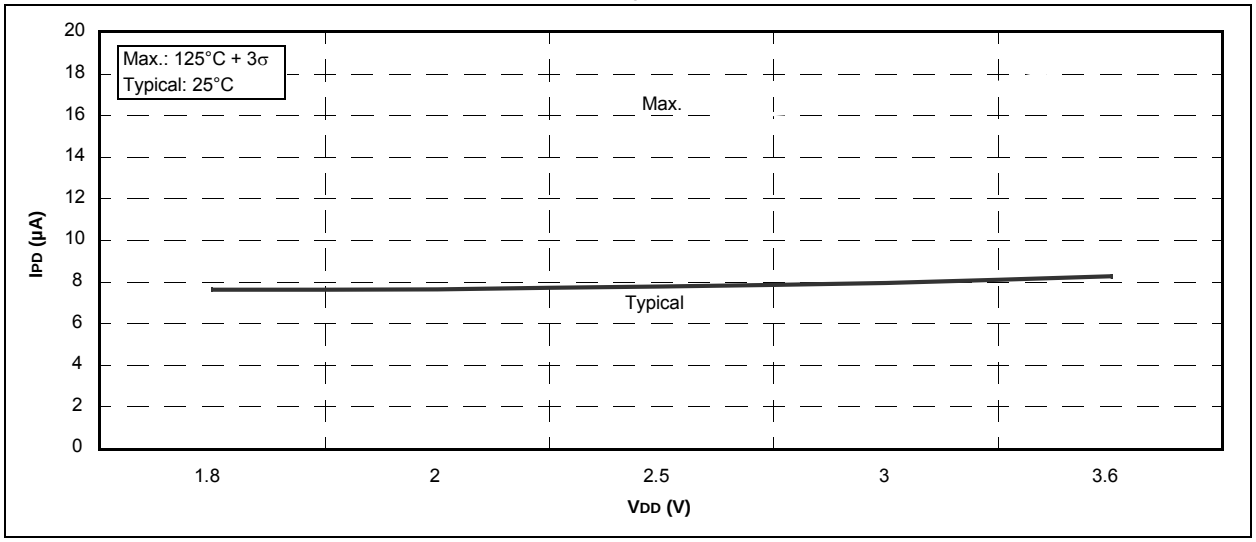


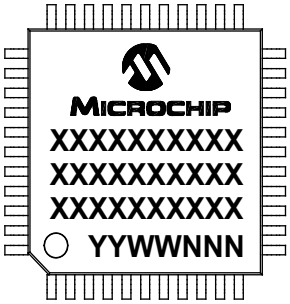
FIGURE 31-58: PIC16LF1937 COMPARATOR 2, LOW POWER



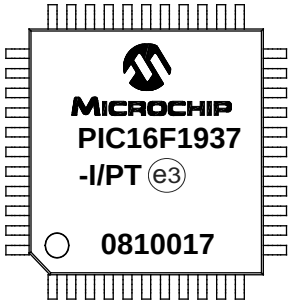
© 2008-2011 Microchip Technology Inc.

Package Marking Information (Continued)

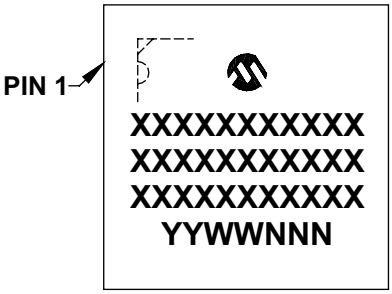
44-Lead TQFP (10x10x1 mm)



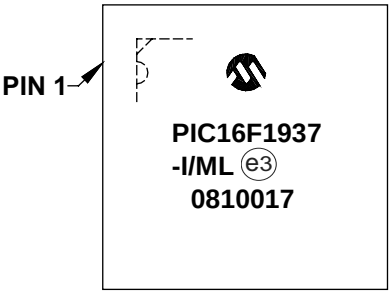
Example



44-Lead QFN (8x8x0.9 mm)

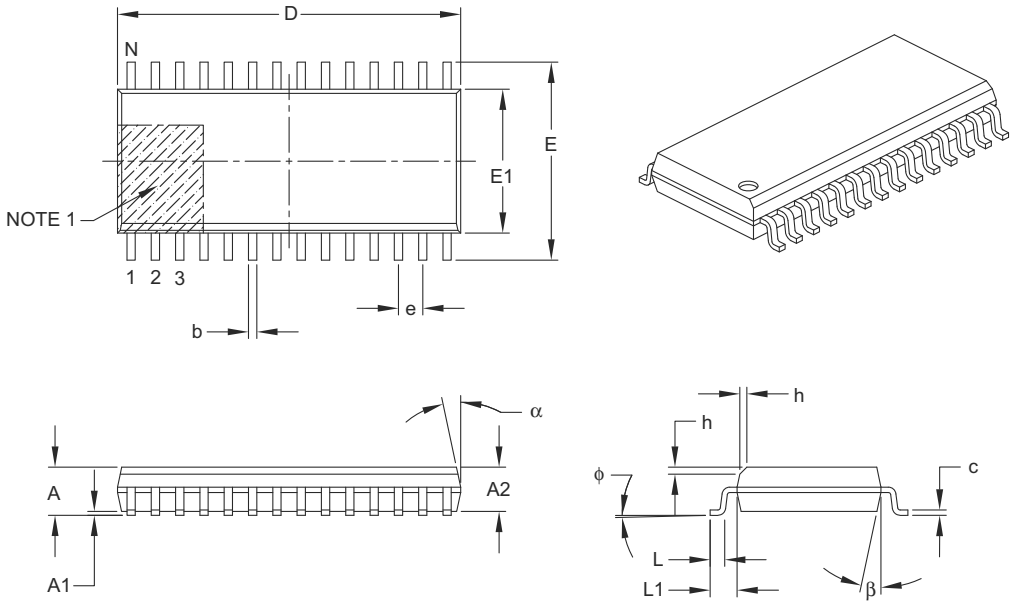


Example



28-Lead Plastic Small Outline (SO) – Wide, 7.50 mm Body [SOIC]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	2.65
Molded Package Thickness	A2	2.05	–	–
Standoff §	A1	0.10	–	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		17.90 BSC	
Chamfer (optional)	h	0.25	–	0.75
Foot Length	L	0.40	–	1.27
Footprint	L1		1.40 REF	
Foot Angle Top	ϕ	0°	–	8°
Lead Thickness	c	0.18	–	0.33
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

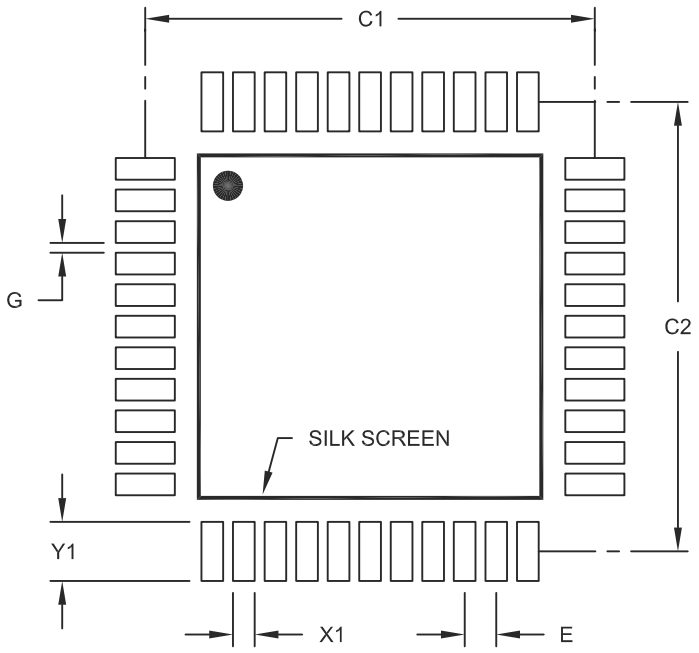
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-052B

© 2008-2011 Microchip Technology Inc.

44-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [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		0.80 BSC	
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2076A

© 2008-2011 Microchip Technology Inc.