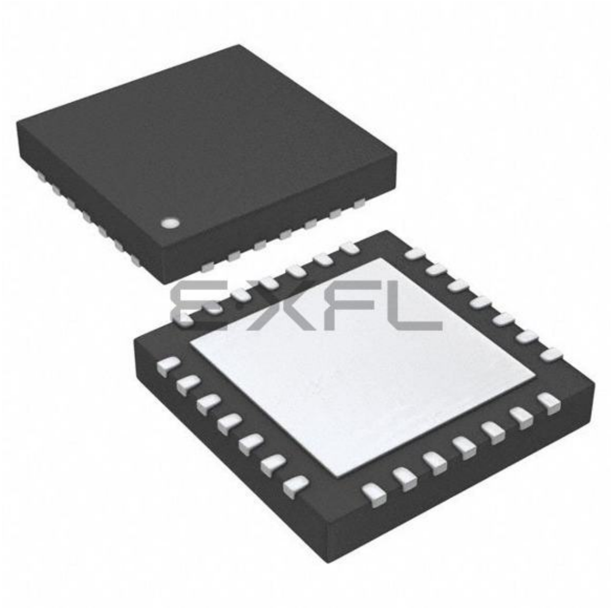


What is "Embedded - Microcontrollers"?

"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, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1827-e-ml

18/20/28-Pin Flash Microcontrollers with nanoWatt XLP Technology

High-Performance RISC CPU:

- C Compiler Optimized Architecture
- 256 bytes Data EEPROM
- Up to 8 Kbytes Linear Program Memory Addressing
- Up to 384 bytes Linear Data Memory Addressing
- Interrupt Capability with Automatic Context Saving
- 16-Level Deep Hardware Stack with Optional Overflow/Underflow Reset
- Direct, Indirect and Relative Addressing modes:
  - Two full 16-bit File Select Registers (FSRs)
  - FSRs can read program and data memory

Flexible Oscillator Structure:

- Precision 32 MHz Internal Oscillator Block:
  - Factory calibrated to ± 1%, typical
  - Software selectable frequencies range of 31 kHz to 32 MHz
- 31 kHz Low-Power Internal Oscillator
- Four Crystal modes up to 32 MHz
- Three External Clock modes up to 32 MHz
- 4X Phase-Lock Loop (PLL)
- Fail-Safe Clock Monitor:
  - Allows for safe shutdown if peripheral clock stops
- Two-Speed Oscillator Start-up
- Reference Clock Module:
  - Programmable clock output frequency and duty-cycle

Special Microcontroller Features:

- 1.8V-5.5V Operation – PIC16F1826/27
- 1.8V-3.6V Operation – PIC16LF1826/27
- Self-Programmable under Software Control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Programmable Brown-out Reset (BOR)
- Extended Watchdog Timer (WDT):
  - Programmable period from 1ms to 268s
- Programmable Code Protection
- In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Enhance Low-Voltage Programming
- Power-Saving Sleep mode

Extreme Low-Power Management PIC16LF1826/27 with nanoWatt XLP:

- Operating Current: 75 μA @ 1 MHz, 1.8V, typical
- Sleep mode: 30 nA
- Watchdog Timer: 500 nA
- Timer1 Oscillator: 600 nA @ 32 kHz

Analog Features:

- Analog-to-Digital Converter (ADC) Module:
  - 10-bit resolution, 12 channels
  - Auto acquisition capability
  - Conversion available during Sleep
- Analog Comparator Module:
  - Two rail-to-rail analog comparators
  - Power mode control
  - Software controllable hysteresis
- Voltage Reference Module:
  - Fixed Voltage Reference (FVR) with 1.024V, 2.048V and 4.096V output levels
  - 5-bit rail-to-rail resistive DAC with positive and negative reference selection

Peripheral Highlights:

- 15 I/O Pins and 1 Input Only Pin:
  - High current sink/source 25 mA/25 mA
  - Programmable weak pull-ups
  - Programmable interrupt-on- change pins
- Timer0: 8-Bit Timer/Counter with 8-Bit Prescaler
- Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Gate Input mode
  - Dedicated, low-power 32 kHz oscillator driver
- Up to three Timer2-types: 8-Bit Timer/Counter with 8-Bit Period Register, Prescaler and Postscaler
- Up to two Capture, Compare, PWM (CCP) Modules
- Up to two Enhanced CCP (ECCP) Modules:
  - Software selectable time bases
  - Auto-shutdown and auto-restart
  - PWM steering
- Up to two Master Synchronous Serial Port (MSSP) with SPI and I^{2}C™ with:
  - 7-bit address masking
  - SMBus/PMBus™ compatibility
- Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) Module
- mTouch™ Sensing Oscillator Module:
  - Up to 12 input channels
- Data Signal Modulator Module:
  - Selectable modulator and carrier sources
- SR Latch:
  - Multiple Set/Reset input options
  - Emulates 555 Timer applications

TABLE 3-3: PIC16(L)F1826/27 MEMORY MAP

	BANK 0		BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7
000h–00Bh	Core Registers (Table 3-2)	080h–08Bh	Core Registers (Table 3-2)	100h–10Bh	Core Registers (Table 3-2)	180h–18Bh	Core Registers (Table 3-2)	200h–20Bh	Core Registers (Table 3-2)	280h–28Bh	Core Registers (Table 3-2)	300h–30Bh	Core Registers (Table 3-2)	380h–38Bh	Core Registers (Table 3-2)
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	—	30Ch	—	38Ch	—
00Dh	PORTB	08Dh	TRISB	10Dh	LATB	18Dh	ANSELB	20Dh	WPUB	28Dh	—	30Dh	—	38Dh	—
00Eh	—	08Eh	—	10Eh	—	18Eh	—	20Eh	—	28Eh	—	30Eh	—	38Eh	—
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh	—	28Fh	—	30Fh	—	38Fh	—
010h	—	090h	—	110h	—	190h	—	210h	—	290h	—	310h	—	390h	—
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	EEADRL	211h	SSP1BUF	291h	CCPR1L	311h	CCPR3L(1)	391h	—
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	EEADRH	212h	SSP1ADD	292h	CCPR1H	312h	CCPR3H(1)	392h	—
013h	PIR3(1)	093h	PIE3(1)	113h	CM2CON0	193h	EEDATL	213h	SSP1MASK	293h	CCP1CON	313h	CCP3CON(1)	393h	—
014h	PIR4(1)	094h	PIE4(1)	114h	CM2CON1	194h	EEDATH	214h	SSP1STAT	294h	PWM1CON	314h	—	394h	IOCBP
015h	TMR0	095h	OPTION	115h	CMOUT	195h	EECON1	215h	SSP1CON	295h	CCP1AS	315h	—	395h	IOCBN
016h	TMR1L	096h	PCON	116h	BORCON	196h	EECON2	216h	SSP1CON2	296h	PSTR1CON	316h	—	396h	IOCBF
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	—	217h	SSP1CON3	297h	—	317h	—	397h	—
018h	T1CON	098h	OSCTUNE	118h	DACCON0	198h	—	218h	—	298h	CCPR2L(1)	318h	CCPR4L(1)	398h	—
019h	T1GCON	099h	OSCCON	119h	DACCON1	199h	RCREG	219h	SSP2BUF(1)	299h	CCPR2H(1)	319h	CCPR4H(1)	399h	—
01Ah	TMR2	09Ah	OSCSTAT	11Ah	SRCON0	19Ah	TXREG	21Ah	SSP2ADD(1)	29Ah	CCP2CON(1)	31Ah	CCP4CON(1)	39Ah	CLKRCON
01Bh	PR2	09Bh	ADRESL	11Bh	SRCON1	19Bh	SPBRGL	21Bh	SSP2MASK(1)	29Bh	PWM2CON(1)	31Bh	—	39Bh	—
01Ch	T2CON	09Ch	ADRESH	11Ch	—	19Ch	SPBRGH	21Ch	SSP2STAT(1)	29Ch	CCP2AS(1)	31Ch	—	39Ch	MDCON
01Dh	—	09Dh	ADCON0	11Dh	APFCON0	19Dh	RCSTA	21Dh	SSP2CON(1)	29Dh	PSTR2CON(1)	31Dh	—	39Dh	MDSRC
01Eh	CPSCON0	09Eh	ADCON1	11Eh	APFCON1	19Eh	TXSTA	21Eh	SSP2CON2(1)	29Eh	CCPTMRS(1)	31Eh	—	39Eh	MDCARL
01Fh	CPSCON1	09Fh	—	11Fh	—	19Fh	BAUDCON	21Fh	SSP2CON3(1)	29Fh	—	31Fh	—	39Fh	MDCARH
020h–06Fh	General Purpose Register 96 Bytes	0A0h–0EFh	General Purpose Register 80 Bytes	120h–16Fh	General Purpose Register 80 Bytes	1A0h–1EFh	General Purpose Register 80 Bytes(1)	220h–26Fh	General Purpose Register 48 Bytes(1); Unimplemented Read as '0'	2A0h–2EFh	Unimplemented Read as '0'	320h–36Fh	Unimplemented Read as '0'	3A0h–3EFh	Unimplemented Read as '0'
070h–07Fh	(General Purpose Register 96 Bytes, cont.)	0F0h–0FFh	Accesses 70h – 7Fh	170h–17Fh	Accesses 70h – 7Fh	1F0h–1FFh	Accesses 70h – 7Fh	270h–27Fh	Accesses 70h – 7Fh	2F0h–2FFh	Accesses 70h – 7Fh	370h–37Fh	Accesses 70h – 7Fh	3F0h–3FFh	Accesses 70h – 7Fh

Legend: = Unimplemented data memory locations, read as '0'

Note 1: Available only on PIC16(L)F1827.

TABLE 3-3: PIC16(L)F1826/27 MEMORY MAP (CONTINUED)

Address	BANK 8	Address	BANK 9	Address	BANK 10	Address	BANK 11	Address	BANK 12	Address	BANK 13	Address	BANK 14	Address	BANK 15
400h–40Bh	Core Registers (Table 3-2)	480h–48Bh	Core Registers (Table 3-2)	500h–50Bh	Core Registers (Table 3-2)	580h–58Bh	Core Registers (Table 3-2)	600h–60Bh	Core Registers (Table 3-2)	680h–68Bh	Core Registers (Table 3-2)	700h–70Bh	Core Registers (Table 3-2)	780h–78Bh	Core Registers (Table 3-2)
40Ch	—	48Ch	—	50Ch	—	58Ch	—	60Ch	—	68Ch	—	70Ch	—	78Ch	—
40Dh	—	48Dh	—	50Dh	—	58Dh	—	60Dh	—	68Dh	—	70Dh	—	78Dh	—
40Eh	—	48Eh	—	50Eh	—	58Eh	—	60Eh	—	68Eh	—	70Eh	—	78Eh	—
40Fh	—	48Fh	—	50Fh	—	58Fh	—	60Fh	—	68Fh	—	70Fh	—	78Fh	—
410h	—	490h	—	510h	—	590h	—	610h	—	690h	—	710h	—	790h	—
411h	—	491h	—	511h	—	591h	—	611h	—	691h	—	711h	—	791h	—
412h	—	492h	—	512h	—	592h	—	612h	—	692h	—	712h	—	792h	—
413h	—	493h	—	513h	—	593h	—	613h	—	693h	—	713h	—	793h	—
414h	—	494h	—	514h	—	594h	—	614h	—	694h	—	714h	—	794h	—
415h	TMR4(1)	495h	—	515h	—	595h	—	615h	—	695h	—	715h	—	795h	—
416h	PR4(1)	496h	—	516h	—	596h	—	616h	—	696h	—	716h	—	796h	—
417h	T4CON(1)	497h	—	517h	—	597h	—	617h	—	697h	—	717h	—	797h	—
418h	—	498h	—	518h	—	598h	—	618h	—	698h	—	718h	—	798h	—
419h	—	499h	—	519h	—	599h	—	619h	—	699h	—	719h	—	799h	—
41Ah	—	49Ah	—	51Ah	—	59Ah	—	61Ah	—	69Ah	—	71Ah	—	79Ah	—
41Bh	—	49Bh	—	51Bh	—	59Bh	—	61Bh	—	69Bh	—	71Bh	—	79Bh	—
41Ch	TMR6(1)	49Ch	—	51Ch	—	59Ch	—	61Ch	—	69Ch	—	71Ch	—	79Ch	—
41Dh	PR6(1)	49Dh	—	51Dh	—	59Dh	—	61Dh	—	69Dh	—	71Dh	—	79Dh	—
41Eh	T6CON(1)	49Eh	—	51Eh	—	59Eh	—	61Eh	—	69Eh	—	71Eh	—	79Eh	—
41Fh	—	49Fh	—	51Fh	—	59Fh	—	61Fh	—	69Fh	—	71Fh	—	79Fh	—
420h–46Fh	Unimplemented Read as ‘0’	4A0h–4EFh	Unimplemented Read as ‘0’	520h–56Fh	Unimplemented Read as ‘0’	5A0h–5EFh	Unimplemented Read as ‘0’	620h–66Fh	Unimplemented Read as ‘0’	6A0h–6EFh	Unimplemented Read as ‘0’	720h–76Fh	Unimplemented Read as ‘0’	7A0h–7EFh	Unimplemented Read as ‘0’
470h–47Fh	Accesses 70h – 7Fh	4F0h–4FFh	Accesses 70h – 7Fh	570h–57Fh	Accesses 70h – 7Fh	5F0h–5FFh	Accesses 70h – 7Fh	670h–67Fh	Accesses 70h – 7Fh	6F0h–6FFh	Accesses 70h – 7Fh	770h–77Fh	Accesses 70h – 7Fh	7F0h–7FFh	Accesses 70h – 7Fh

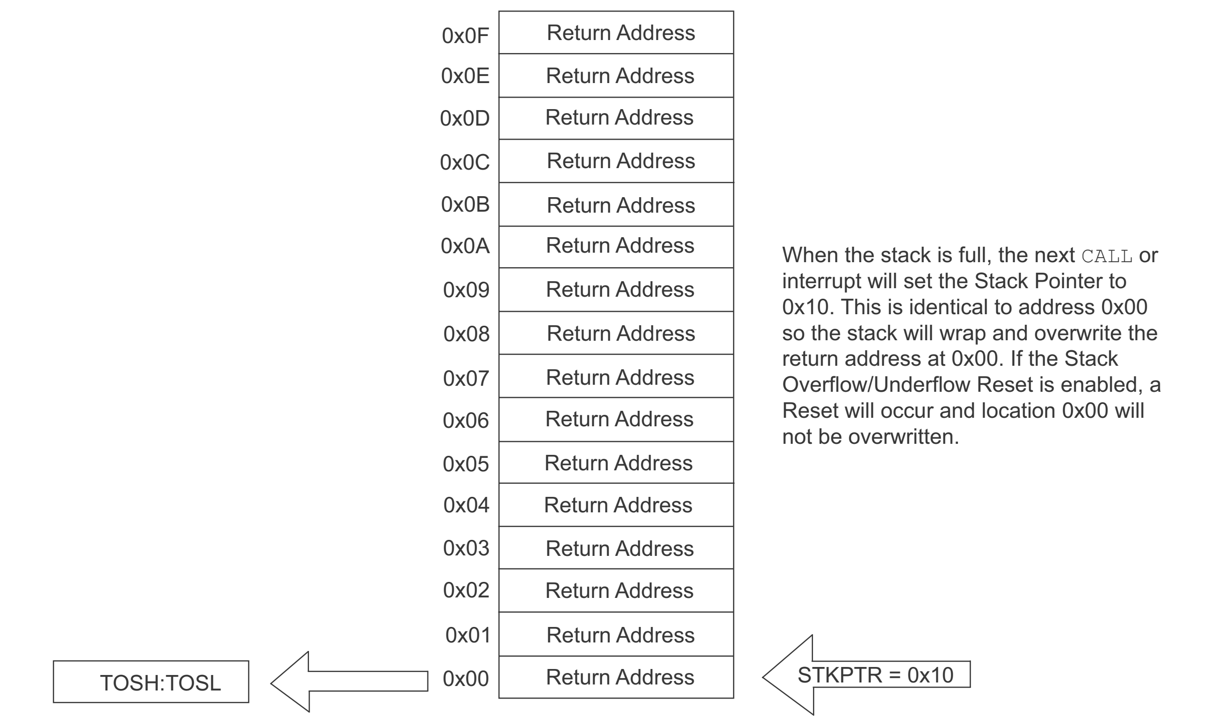
Legend: = Unimplemented data memory locations, read as ‘0’

TABLE 3-4: PIC16(L)F1826/27 MEMORY MAP (CONTINUED)

Address	Bank 31
F80h–F8Bh	Core Registers (Table 3-2)
F8Ch–FE3h	Unimplemented Read as '0'
FE4h	STATUS_SHAD
FE5h	WREG_SHAD
FE6h	BSR_SHAD
FE7h	PCLATH_SHAD
FE8h	FSR0L_SHAD
FE9h	FSR0H_SHAD
FEAh	FSR1L_SHAD
FEBh	FSR1H_SHAD
FECh	—
FEDh	STKPTR
FEEh	TOSL
FEFh	TOSH
FF0h–FFFh	Common RAM (Accesses 70h – 7Fh)

= Unimplemented data memory locations, read as '0',

FIGURE 3-8: ACCESSING THE STACK EXAMPLE 4



3.4.2 OVERFLOW/UNDERFLOW RESET

If the STVREN bit in Configuration Word 2 is programmed to '1', the device will be reset if the stack is PUSHed beyond the sixteenth level or POPed beyond the first level, setting the appropriate bits (STKOVF or STKUNF, respectively) in the PCON register.

3.5 Indirect Addressing

The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the File Select Registers (FSR). If the FSRn address specifies one of the two INDFn registers, the read will return '0' and the write will not occur (though Status bits may be affected). The FSRn register value is created by the pair FSRnH and FSRnL.

The FSR registers form a 16-bit address that allows an addressing space with 65536 locations. These locations are divided into three memory regions:

- Traditional Data Memory
- Linear Data Memory
- Program Flash Memory

4.5 Device ID and Revision ID

The memory location 8006h is where the Device ID and Revision ID are stored. The upper nine bits hold the Device ID. The lower five bits hold the Revision ID. See **Section 11.5 "User ID, Device ID and Configuration Word Access"** for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

10.6 Watchdog Control Register

REGISTER 10-1: WDTCON: WATCHDOG TIMER CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
—	—	WDTPS<4:0>					SWDTEN
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	-m/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **Unimplemented:** Read as '0'

bit 5-1 **WDTPS<4:0>:** Watchdog Timer Period Select bits(1)

Bit Value = Prescale Rate

00000 = 1:32 (Interval 1 ms nominal)
00001 = 1:64 (Interval 2 ms nominal)
00010 = 1:128 (Interval 4 ms nominal)
00011 = 1:256 (Interval 8 ms nominal)
00100 = 1:512 (Interval 16 ms nominal)
00101 = 1:1024 (Interval 32 ms nominal)
00110 = 1:2048 (Interval 64 ms nominal)
00111 = 1:4096 (Interval 128 ms nominal)
01000 = 1:8192 (Interval 256 ms nominal)
01001 = 1:16384 (Interval 512 ms nominal)
01010 = 1:32768 (Interval 1s nominal)
01011 = 1:65536 (Interval 2s nominal) (Reset value)
01100 = 1:131072 (2^{17}) (Interval 4s nominal)
01101 = 1:262144 (2^{18}) (Interval 8s nominal)
01110 = 1:524288 (2^{19}) (Interval 16s nominal)
01111 = 1:1048576 (2^{20}) (Interval 32s nominal)
10000 = 1:2097152 (2^{21}) (Interval 64s nominal)
10001 = 1:4194304 (2^{22}) (Interval 128s nominal)
10010 = 1:8388608 (2^{23}) (Interval 256s nominal)

10011 = Reserved. Results in minimum interval (1:32)
•
•
•
11111 = Reserved. Results in minimum interval (1:32)

bit 0 **SWDTEN:** Software Enable/Disable for Watchdog Timer bit

If WDTE<1:0> = 00:
This bit is ignored.
If WDTE<1:0> = 01:
1 = WDT is turned on
0 = WDT is turned off
If WDTE<1:0> = 1x:
This bit is ignored.

Note 1: Times are approximate. WDT time is based on 31 kHz LFINTOSC.

12.0 I/O PORTS

Depending on the device selected and peripherals enabled, there are two ports available. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Each port has three registers for its operation. These registers are:

- TRISx registers (data direction register)
- PORTx registers (reads the levels on the pins of the device)
- LATx registers (output latch)

Some ports may have one or more of the following additional registers. These registers are:

- ANSELx (analog select)
- WPUx (weak pull-up)

TABLE 12-1: PORT AVAILABILITY PER DEVICE

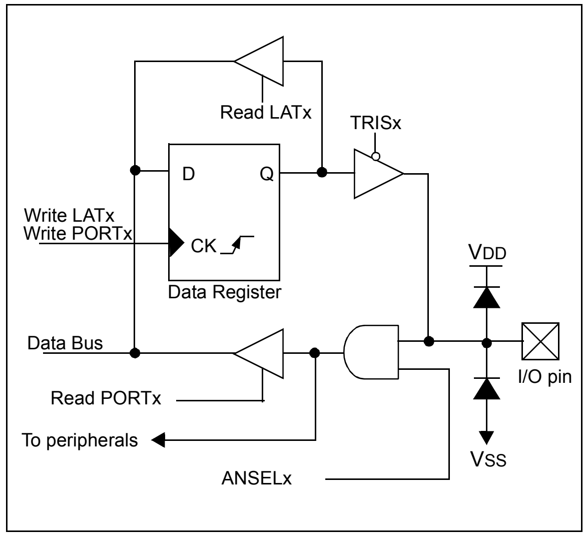
Device	PORTA	PORTB	PORTC
PIC16(L)F1826	●	●	
PIC16(L)F1827	●	●	●

The Data Latch (LATx registers) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATx register has the same effect as a write to the corresponding PORTx register. A read of the LATx register reads of the values held in the I/O PORT latches, while a read of the PORTx register reads the actual I/O pin value.

Ports with analog functions also have an ANSELx register which can disable the digital input and save power. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 12-1.

FIGURE 12-1: GENERIC I/O PORT OPERATION

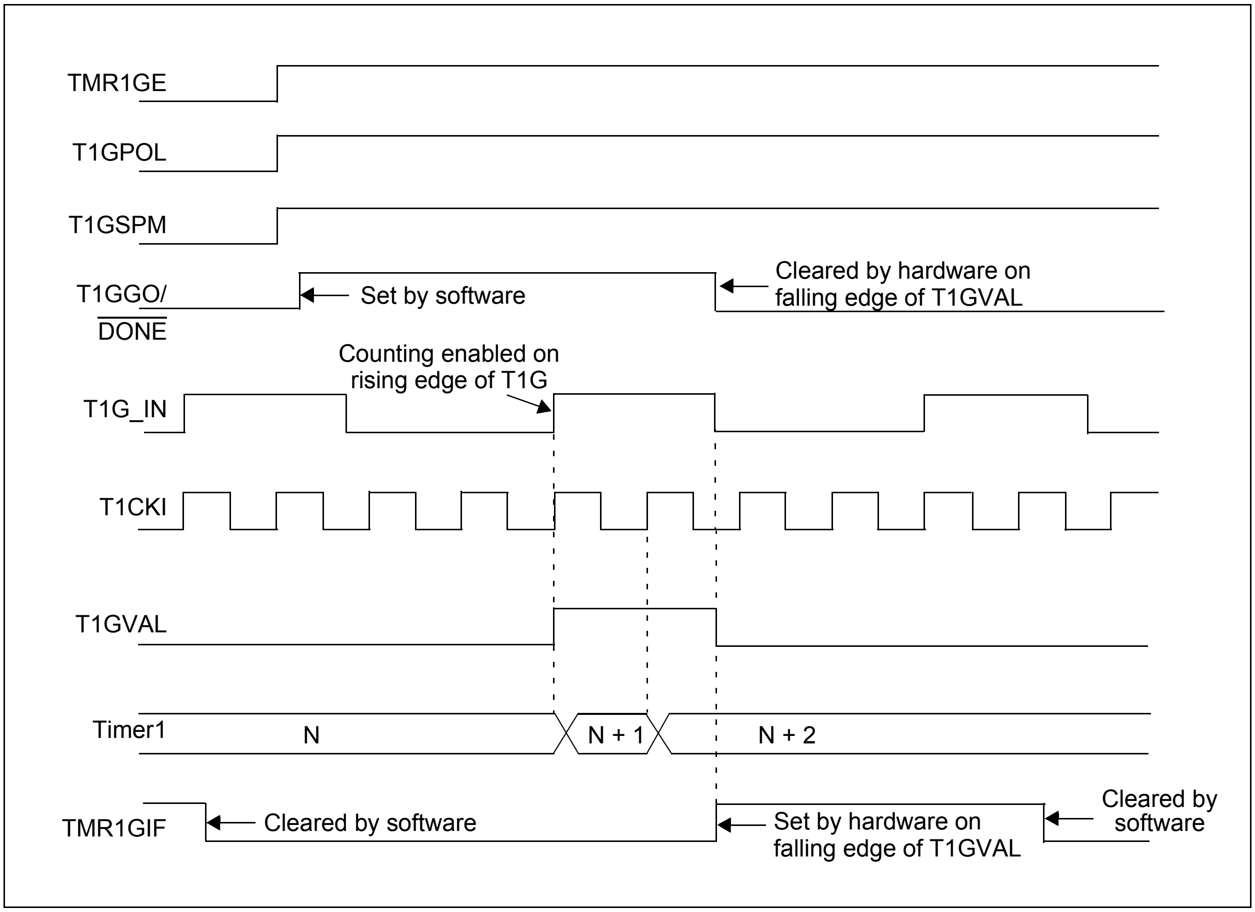


EXAMPLE 12-1: INITIALIZING PORTA

```
; This code example illustrates
; initializing the PORTA register. The
; other ports are initialized in the same
; manner.

BANKSEL  PORTA        ;
CLRF     PORTA        ;Init PORTA
BANKSEL  LATA         ;Data Latch
CLRF     LATA         ;
BANKSEL  ANSELA       ;
CLRF     ANSELA       ;digital I/O
BANKSEL  TRISA        ;
MOVLW    B'00111000'  ;Set RA<5:3> as inputs
MOVWF    TRISA        ;and set RA<2:0> as
                      ;outputs
```

FIGURE 21-5: TIMER1 GATE SINGLE-PULSE MODE



REGISTER 24-4: PWMxCON: ENHANCED PWM CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PxRSEN	PxDC<6:0>						
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 **PxRSEN:** PWM Restart Enable bit
- `1` = Upon auto-shutdown, the CCPxASE bit clears automatically once the shutdown event goes away; the PWM restarts automatically
- `0` = Upon auto-shutdown, CCPxASE must be cleared in software to restart the PWM

bit 6-0 **PxDC<6:0>:** PWM Delay Count bits

PxDCx = Number of FOSC/4 (4 * TOSC) cycles between the scheduled time when a PWM signal **should** transition active and the **actual** time it transitions active

Note 1: Bit resets to ‘0’ with Two-Speed Start-up and LP, XT or HS selected as the Oscillator mode or Fail-Safe mode is enabled.

25.5.3.3 7-bit Transmission with Address Hold Enabled

Setting the AHEN bit of the SSPxCON3 register enables additional clock stretching and interrupt generation after the 8th falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPxIF interrupt is set.

Figure 25-18 displays a standard waveform of a 7-bit Address Slave Transmission with AHEN enabled.

1. Bus starts Idle.
2. Master sends Start condition; the S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Master sends matching address with R/$\overline{W}$ bit set. After the 8th falling edge of the SCLx line the CKP bit is cleared and SSPxIF interrupt is generated.
4. Slave software clears SSPxIF.
5. Slave software reads ACKTIM bit of SSPxCON3 register, and R/$\overline{W}$ and D/$\overline{A}$ of the SSPxSTAT register to determine the source of the interrupt.
6. Slave reads the address value from the SSPxBUF register clearing the BF bit.
7. Slave software decides from this information if it wishes to $\overline{ACK}$ or not $\overline{ACK}$ and sets ACKDT bit of the SSPxCON2 register accordingly.
8. Slave sets the CKP bit releasing SCLx.
9. Master clocks in the $\overline{ACK}$ value from the slave.
10. Slave hardware automatically clears the CKP bit and sets SSPxIF after the $\overline{ACK}$ if the R/$\overline{W}$ bit is set.
11. Slave software clears SSPxIF.
12. Slave loads value to transmit to the master into SSPxBUF setting the BF bit.

> **Note:** SSPxBUF cannot be loaded until after the $\overline{ACK}$.

13. Slave sets CKP bit releasing the clock.
14. Master clocks out the data from the slave and sends an $\overline{ACK}$ value on the 9th SCLx pulse.
15. Slave hardware copies the $\overline{ACK}$ value into the ACKSTAT bit of the SSPxCON2 register.
16. Steps 10-15 are repeated for each byte transmitted to the master from the slave.
17. If the master sends a not $\overline{ACK}$ the slave releases the bus allowing the master to send a Stop and end the communication.

> **Note:** Master must send a not $\overline{ACK}$ on the last byte to ensure that the slave releases the SCLx line to receive a Stop.

26.1.2.8 Asynchronous Reception Set-up:

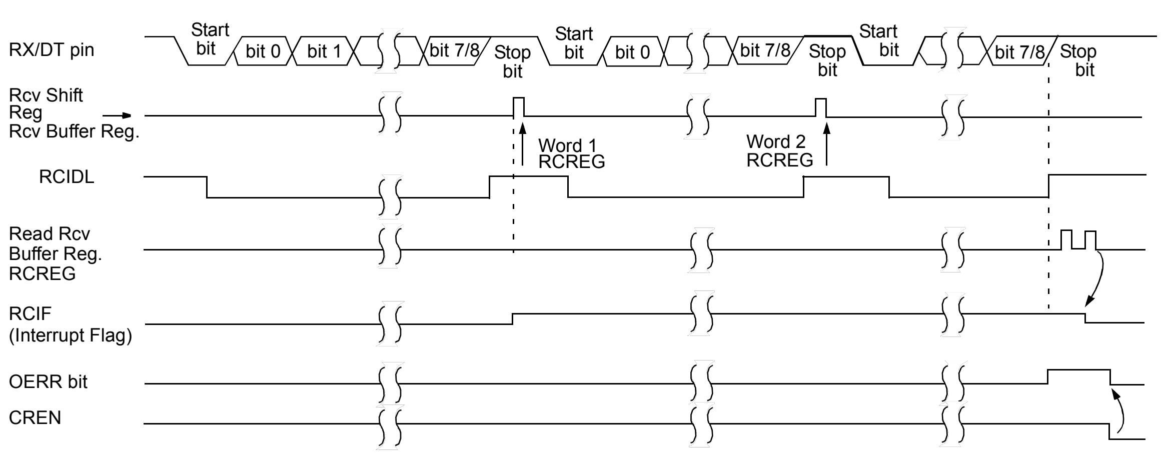
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 26.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

26.1.2.9 9-bit Address Detection Mode Set-up

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 26.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
10. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 26-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

26.3.2 AUTO-BAUD OVERFLOW

During the course of automatic baud detection, the ABDOVF bit of the BAUDCON register will be set if the baud rate counter overflows before the fifth rising edge is detected on the RX pin. The ABDOVF bit indicates that the counter has exceeded the maximum count that can fit in the 16 bits of the SPBRGH:SPBRGL register pair. After the ABDOVF has been set, the counter continues to count until the fifth rising edge is detected on the RX pin. Upon detecting the fifth RX edge, the hardware will set the RCIF interrupt flag and clear the ABDEN bit of the BAUDCON register. The RCIF flag can be subsequently cleared by reading the RCREG register. The ABDOVF flag of the BAUDCON register can be cleared by software directly.

To terminate the auto-baud process before the RCIF flag is set, clear the ABDEN bit then clear the ABDOVF bit of the BAUDCON register. The ABDOVF bit will remain set if the ABDEN bit is not cleared first.

26.3.3 AUTO-WAKE-UP ON BREAK

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the Baud Rate Generator is inactive and a proper character reception cannot be performed. The Auto-Wake-up feature allows the controller to wake-up due to activity on the RX/DT line. This feature is available only in Asynchronous mode.

The Auto-Wake-up feature is enabled by setting the WUE bit of the BAUDCON register. Once set, the normal receive sequence on RX/DT is disabled, and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RX/DT line. (This coincides with the start of a Sync Break or a wake-up signal character for the LIN protocol.)

The EUSART module generates an RCIF interrupt coincident with the wake-up event. The interrupt is generated synchronously to the Q clocks in normal CPU operating modes (Figure 26-7), and asynchronously if the device is in Sleep mode (Figure 26-8). The interrupt condition is cleared by reading the RCREG register.

The WUE bit is automatically cleared by the low-to-high transition on the RX line at the end of the Break. This signals to the user that the Break event is over. At this point, the EUSART module is in Idle mode waiting to receive the next character.

26.3.3.1 Special Considerations

Break Character

To avoid character errors or character fragments during a wake-up event, the wake-up character must be all zeros.

When the wake-up is enabled the function works independent of the low time on the data stream. If the WUE bit is set and a valid non-zero character is received, the low time from the Start bit to the first rising edge will be interpreted as the wake-up event. The remaining bits in the character will be received as a fragmented character and subsequent characters can result in framing or overrun errors.

Therefore, the initial character in the transmission must be all '0's. This must be 10 or more bit times, 13-bit times recommended for LIN bus, or any number of bit times for standard RS-232 devices.

Oscillator Start-up Time

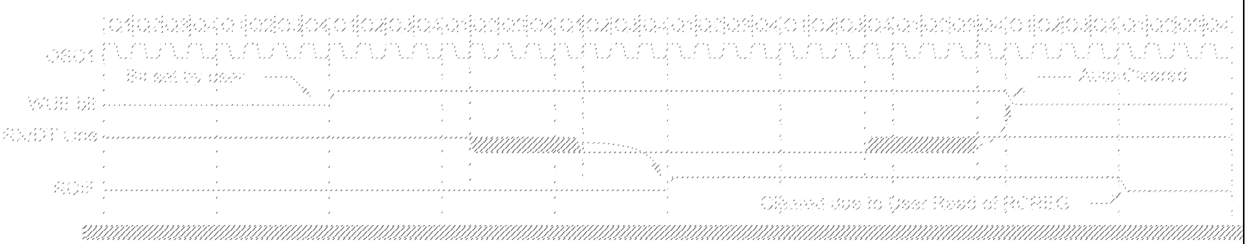
Oscillator start-up time must be considered, especially in applications using oscillators with longer start-up intervals (i.e., LP, XT or HS/PLL mode). The Sync Break (or wake-up signal) character must be of sufficient length, and be followed by a sufficient interval, to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART.

WUE Bit

The wake-up event causes a receive interrupt by setting the RCIF bit. The WUE bit is cleared in hardware by a rising edge on RX/DT. The interrupt condition is then cleared in software by reading the RCREG register and discarding its contents.

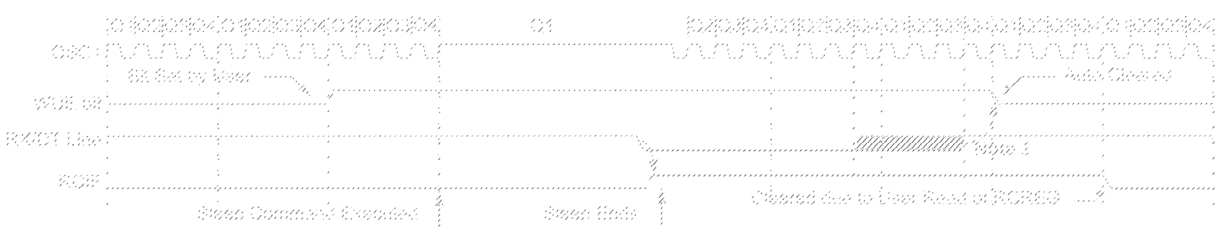
To ensure that no actual data is lost, check the RCIDL bit to verify that a receive operation is not in process before setting the WUE bit. If a receive operation is not occurring, the WUE bit may then be set just prior to entering the Sleep mode.

FIGURE 26-7: AUTO-WAKE-UP BIT (WUE) TIMING DURING NORMAL OPERATION



Note 1: The EUSART remains in Idle while the WUE bit is set.

FIGURE 26-8: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING SLEEP



Note 1: If the wake-up event requires long oscillator warm-up time, the automatic clearing of the WUE bit can occur while the stposc signal is still active. This sequence should not depend on the presence of Q clocks.

2: The EUSART remains in Idle while the WUE bit is set.

NOTES:

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	$0 \leq k \leq 2047$
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → PC<14:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CALLW	Subroutine Call With W
Syntax:	[*label*] CALLW
Operands:	None
Operation:	(PC) +1 → TOS, (W) → PC<7:0>, (PCLATH<6:0>) → PC<14:8>
Status Affected:	None
Description:	Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into PC<14:8>. CALLW is a two-cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	$0 \leq f \leq 127$
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF	Complement f
Syntax:	[*label*] COMF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	$(\bar{f})$ → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

DECF	Decrement f
Syntax:	[*label*] DECF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

RETFIE — Return from Interrupt

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

Description: Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.

Words: 1

Cycles: 2

<u>Example:</u>

```
RETFIE
```

After Interrupt
PC = TOS
GIE = 1

RETLW — Return with literal in W

Syntax: [*label*] RETLW k

Operands: $0 \leq k \leq 255$

Operation: k → (W);
TOS → PC

Status Affected: None

Description: The W register is loaded with the eight bit literal ‘k’. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.

Words: 1

Cycles: 2

<u>Example:</u>

```
      CALL TABLE;W contains table
          ;offset value
      •   ;W now has table value
TABLE •
      •
      ADDWF PC  ;W = offset
      RETLW k1  ;Begin table
      RETLW k2  ;
      •
      •
      •
      RETLW kn  ; End of table
```

Before Instruction
W = 0x07
After Instruction
W = value of k8

RETURN — Return from Subroutine

Syntax: [*label*] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

RLF — Rotate Left f through Carry

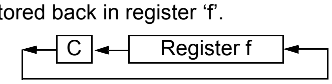
Syntax: [*label*] RLF f,d

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

Operation: See description below

Status Affected: C

Description: The contents of register ‘f’ are rotated one bit to the left through the Carry flag. If ‘d’ is ‘0’, the result is placed in the W register. If ‘d’ is ‘1’, the result is stored back in register ‘f’.



Words: 1

Cycles: 1

<u>Example:</u>

```
RLF     REG1,0
```

Before Instruction
REG1 = 1110 0110
C = 0
After Instruction
REG1 = 1110 0110
W = 1100 1100
C = 1

30.2 DC Characteristics: PIC16(L)F1826/27-I/E (Industrial, Extended) (Continued)

PIC16LF1826/27			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
PIC16F1826/27			**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)						
D019		—	4.0	7.3	mA	3.0	FOSC = 32 MHz HFINTOSC mode **(Note 3)**
		—	4.4	7.5	mA	3.6	
D019		—	4.2	7.3	mA	3.0	FOSC = 32 MHz HFINTOSC mode **(Note 3)**
		—	4.6	7.5	mA	5.0	
D020		—	4.0	6.0	mA	3.0	FOSC = 32 MHz HS Oscillator mode **(Note 4)**
		—	4.7	7.0	mA	3.6	
D020		—	4.2	6.8	mA	3.0	FOSC = 32 MHz HS Oscillator mode **(Note 4)**
		—	4.9	7.6	mA	5.0	
D021		—	410	0.65	mA	1.8	FOSC = 4 MHz EXTRC mode **(Note 5)**
		—	710	1.25	mA	3.0	
D021		—	430	0.695	mA	1.8	FOSC = 4 MHz EXTRC mode **(Note 5)**
		—	730	1.3	mA	3.0	
		—	860	1.35	mA	5.0	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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 as inputs, 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: 8 MHz internal RC oscillator with 4x PLL enabled.

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

5: 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.

TABLE 30-8: PIC16(L)F1826/27 A/D CONVERTER (ADC) CHARACTERISTICS:

Standard Operating Conditions (unless otherwise stated)
Operating temperature Tested at +25°C

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
AD01	NR	Resolution	—	—	10	bit	
AD02	EIL	Integral Error	—	—	±1.7	LSb	VREF = 3.0V
AD03	EDL	Differential Error	—	—	±1	LSb	No missing codes VREF = 3.0V
AD04	EOFF	Offset Error	—	—	±2.5	LSb	VREF = 3.0V
AD05	EGN	Gain Error	—	—	±2.0	LSb	VREF = 3.0V
AD06	VREF	Reference Voltage(3)	—	—	VDD	V	VREF = (VREF+ minus VREF-) (**NOTE 5**)
AD07	VAIN	Full-Scale Range	—	—	VREF	V	
AD08	ZAIN	Recommended Impedance of Analog Voltage Source	—	—	10	kΩ	Can go higher if external 0.01μF capacitor is present on input pin.

* These parameters are characterized but not tested.
† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Total Absolute Error includes integral, differential, offset and gain errors.
2: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.
3: ADC VREF is from external VREF, VDD pin or FVR, whichever is selected as reference input.
4: When ADC is off, it will not consume any current other than leakage current. The power-down current specification includes any such leakage from the ADC module.
5: FVR voltage selected must be 2.048V or 4.096V.

TABLE 30-9: PIC16(L)F1826/27 A/D CONVERSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)
Operating temperature -40°C ≤ TA ≤ +125°C

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
AD130*	TAD	A/D Clock Period	1.0	—	9.0	μs	TOSC-based
		A/D Internal RC Oscillator Period	1.0	2.5	6.0	μs	ADCS<1:0> = 11 (ADRC mode)
AD131	TCNV	Conversion Time (not including Acquisition Time)(1)	—	11	—	TAD	Set GO/$\overline{\text{DONE}}$ bit to conversion complete
AD132*	TACQ	Acquisition Time	—	5.0	—	μs	

* These parameters are characterized but not tested.
† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The ADRES register may be read on the following TCY cycle.

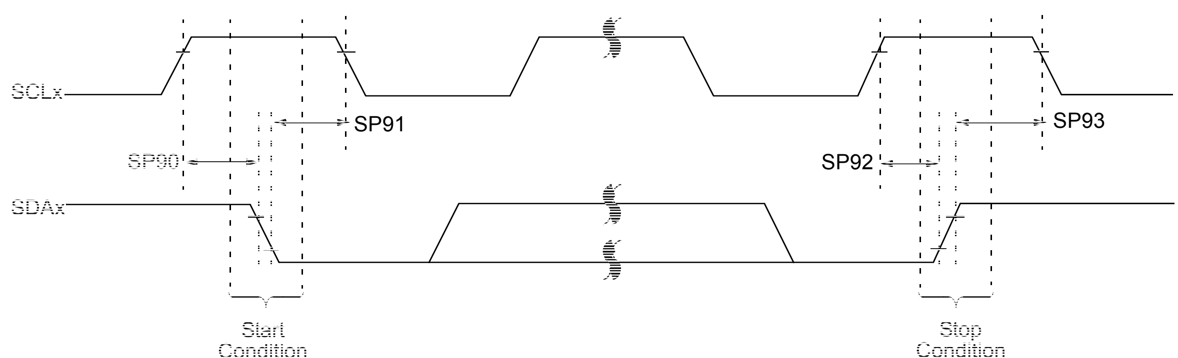
TABLE 30-14: SPI MODE REQUIREMENTS

Param No.	Symbol	Characteristic		Min.	Typ†	Max.	Units	Conditions
SP70*	TssL2scH, TssL2scL	$\overline{\text{SSx}}$↓ to SCKx↓ or SCKx↑ input		Tcy	—	—	ns	
SP71*	TscH	SCKx input high time (Slave mode)		Tcy + 20	—	—	ns	
SP72*	TscL	SCKx input low time (Slave mode)		Tcy + 20	—	—	ns	
SP73*	TdiV2scH, TdiV2scL	Setup time of SDIx data input to SCKx edge		100	—	—	ns	
SP74*	TscH2diL, TscL2diL	Hold time of SDIx data input to SCKx edge		100	—	—	ns	
SP75*	TdoR	SDO data output rise time	3.0-5.5V	—	10	25	ns	
			1.8-5.5V	—	25	50	ns	
SP76*	TdoF	SDOx data output fall time		—	10	25	ns	
SP77*	TssH2doZ	$\overline{\text{SSx}}$↑ to SDOx output high-impedance		10	—	50	ns	
SP78*	TscR	SCKx output rise time (Master mode)	3.0-5.5V	—	10	25	ns	
			1.8-5.5V	—	25	50	ns	
SP79*	TscF	SCKx output fall time (Master mode)		—	10	25	ns	
SP80*	TscH2doV, TscL2doV	SDOx data output valid after SCKx edge	3.0-5.5V	—	—	50	ns	
			1.8-5.5V	—	—	145	ns	
SP81*	TdoV2scH, TdoV2scL	SDOx data output setup to SCKx edge		Tcy	—	—	ns	
SP82*	TssL2doV	SDOx data output valid after $\overline{\text{SS}}$↓ edge		—	—	50	ns	
SP83*	TscH2ssH, TscL2ssH	$\overline{\text{SSx}}$ ↑ after SCKx edge		1.5Tcy + 40	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 30-20: I²C™ BUS START/STOP BITS TIMING



Note: Refer to Figure 30-5 for load conditions.