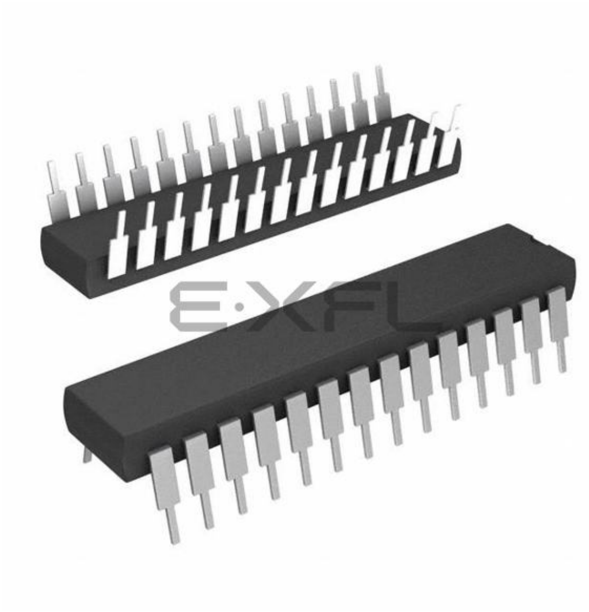
Microchip Technology - PIC18F258-E/SP Datasheet



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	40MHz
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	22
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 5x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f258-e-sp

REGISTER 8-3: INTCON3: INTERRUPT CONTROL REGISTER 3

R/W-1	R/W-1	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
bit 7							bit 0

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 **Unimplemented:** Read as '0'

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 **Unimplemented:** Read as '0'

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

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

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 software polling.

9.4 PORTD, TRISD and LATD Registers

Note:	This port is only available on the PIC18F448 and PIC18F458.

PORTD is an 8-bit wide, bidirectional port. The corresponding Data Direction register for the port is TRISD. Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., put the corresponding output driver in a high-impedance mode). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., put the contents of the output latch on the selected pin).

Read-modify-write operations on the LATD register read and write the latched output value for PORTD.

PORTD uses Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

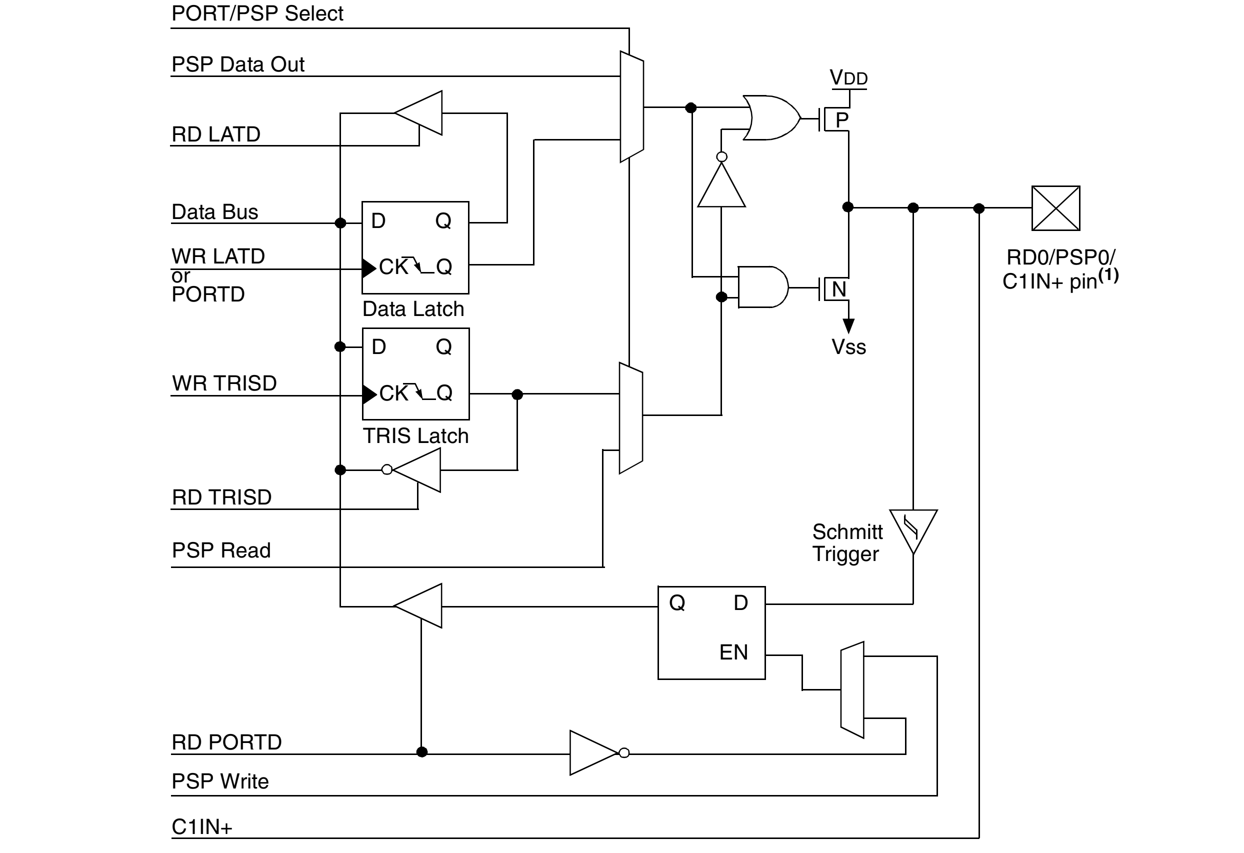
PORTD can be configured as an 8-bit wide, microprocessor port (Parallel Slave Port or PSP) by setting the control bit PSPMODE (TRISE<4>). In this mode, the input buffers are TTL. See **Section 10.0 "Parallel Slave Port"** for additional information.

PORTD is also multiplexed with the analog comparator module and the ECCP module.

EXAMPLE 9-4: INITIALIZING PORTD

```
CLRF    PORTD        ; Initialize PORTD by
                     ; clearing output
                     ; data latches
CLRF    LATD         ; Alternate method
                     ; to clear output
                     ; data latches
MOVLW   07h          ; comparator off
MOVWF   CMCON
MOVLW   0CFh         ; Value used to
                     ; initialize data
                     ; direction
MOVWF   TRISD        ; Set RD3:RD0 as inputs
                     ; RD5:RD4 as outputs
                     ; RD7:RD6 as inputs
```

FIGURE 9-9: PORTD BLOCK DIAGRAM IN I/O PORT MODE



Note 1: I/O pins have diode protection to VDD and VSS.

TABLE 15-3: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, TIMER1 AND TIMER3

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
INTCON	GIE/ GIEH	PEIE/ GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	`0000 000x`	`0000 000u`
PIR1	PSPIF[1]	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	`0000 0000`	`0000 0000`
PIE1	PSPIE[1]	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	`0000 0000`	`0000 0000`
IPR1	PSPIP[1]	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	`1111 1111`	`1111 1111`
TRISD	PORTD Data Direction Register								`1111 1111`	`1111 1111`
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								`xxxx xxxx`	`uuuu uuuu`
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								`xxxx xxxx`	`uuuu uuuu`
T1CON	RD16	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	`0-00 0000`	`u-uu uuuu`
CCPR1L	Capture/Compare/PWM Register 1 (LSB)								`xxxx xxxx`	`uuuu uuuu`
CCPR1H	Capture/Compare/PWM Register 1 (MSB)								`xxxx xxxx`	`uuuu uuuu`
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	`--00 0000`	`--00 0000`
PIR2	—	CMIF	—	EEIF	BCLIF	LVDIF	TMR3IF	ECCP1IF	`-0-0 0000`	`-0-0 0000`
PIE2	—	CMIE	—	EEIE	BCLIE	LVDIE	TMR3IE	ECCP1IE	`-0-0 0000`	`-0-0 0000`
IPR2	—	CMIP	—	EEIP	BCLIP	LVDIP	TMR3IP	ECCP1IP	`-1-1 1111`	`-1-1 1111`
TMR3L	Holding Register for the Least Significant Byte of the 16-bit TMR3 Register								`xxxx xxxx`	`uuuu uuuu`
TMR3H	Holding Register for the Most Significant Byte of the 16-bit TMR3 Register								`xxxx xxxx`	`uuuu uuuu`
T3CON	RD16	T3ECCP1	T3CKPS1	T3CKPS0	T3CCP1	$\overline{\text{T3SYNC}}$	TMR3CS	TMR3ON	`0000 0000`	`uuuu uuuu`

Legend: `x` = unknown, `u` = unchanged, `-` = unimplemented, read as '`0`'. Shaded cells are not used by Capture and Timer1.

Note 1: These registers or register bits are not implemented on the PIC18F248 and PIC18F258 and read as '`0`'s.

FIGURE 17-5: SPI™ MODE WAVEFORM (SLAVE MODE WITH CKE = 0)

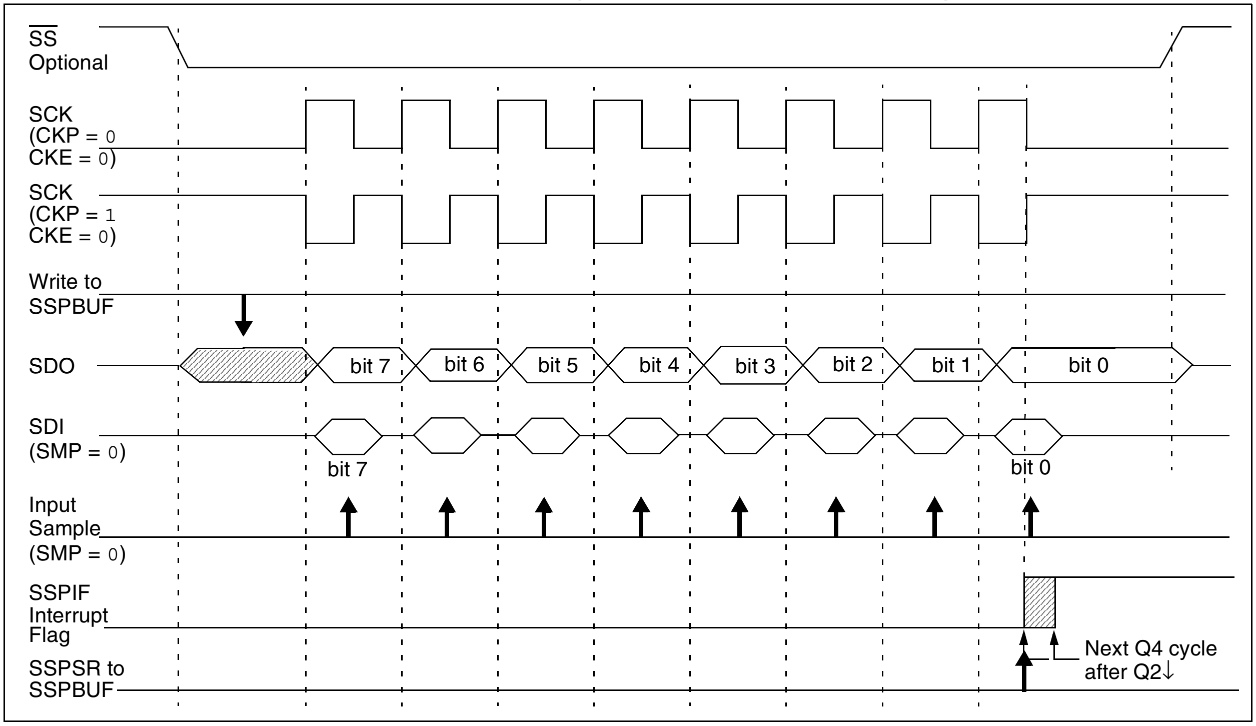


FIGURE 17-6: SPI™ MODE WAVEFORM (SLAVE MODE WITH CKE = 1)

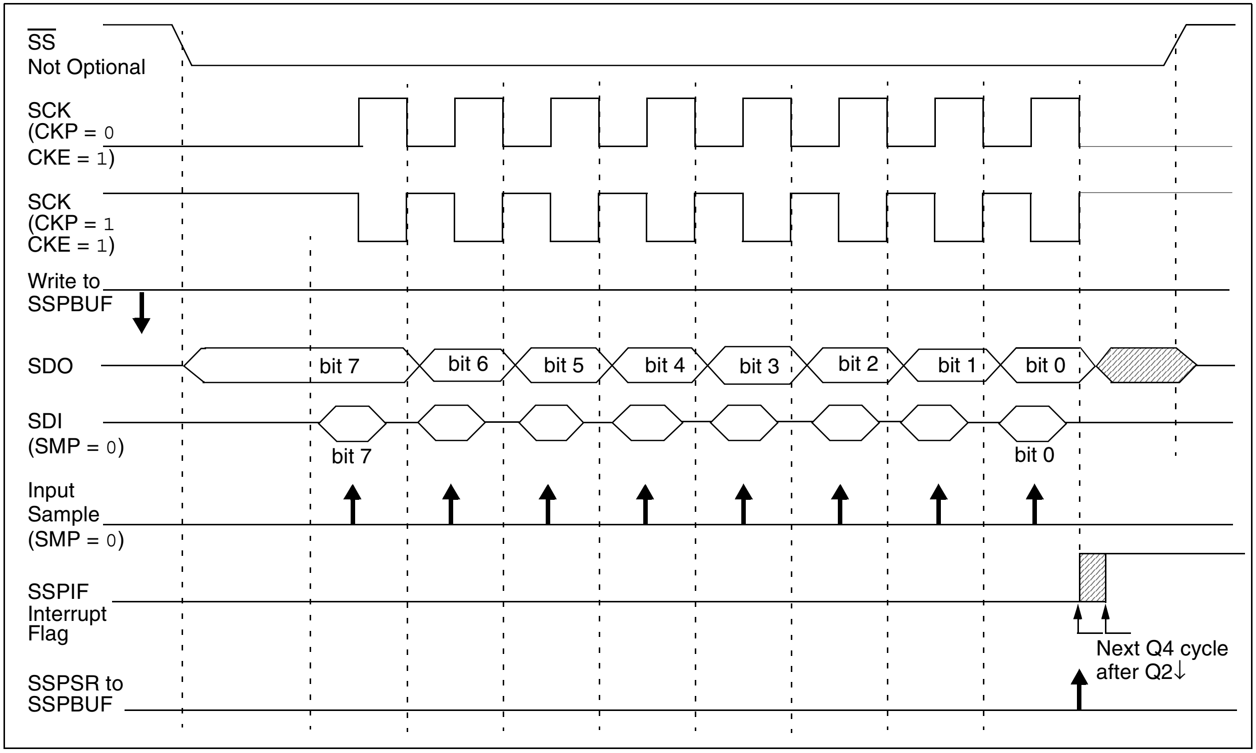
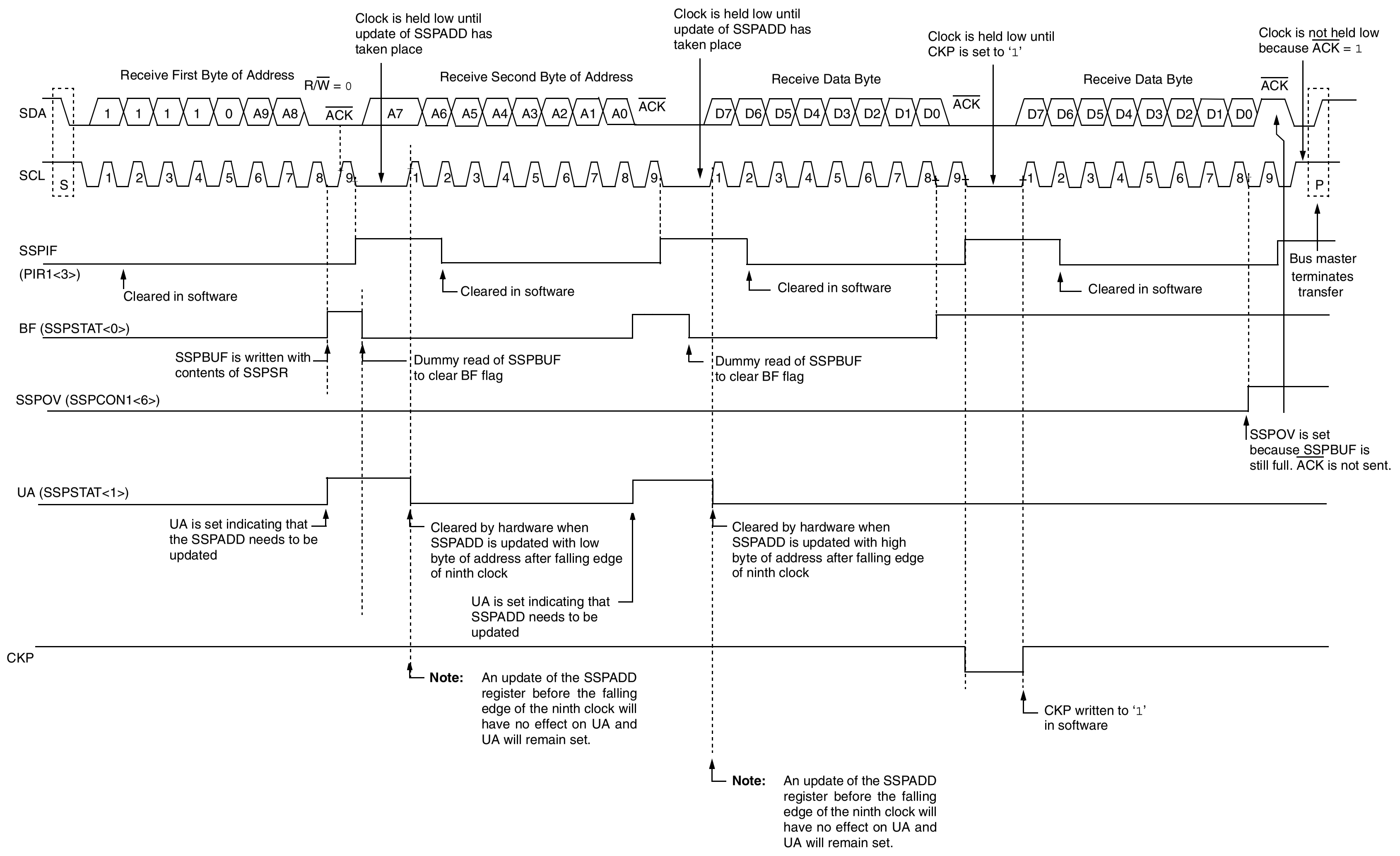


FIGURE 17-14: I²C™ SLAVE MODE TIMING WITH SEN = 1 (RECEPTION, 10-BIT ADDRESS)


Clock is held low until update of SSPADD has taken place
Clock is held low until update of SSPADD has taken place
Clock is held low until CKP is set to '1'
Clock is not held low because ACK = 1
Receive First Byte of Address
R/W = 0
Receive Second Byte of Address
Receive Data Byte
Receive Data Byte
ACK
SDA
1 1 1 1 0 A9 A8
ACK
A7 A6 A5 A4 A3 A2 A1 A0
ACK
D7 D6 D5 D4 D3 D2 D1 D0
ACK
D7 D6 D5 D4 D3 D2 D1 D0
SCL
S
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
P
SSPIF
(PIR1<3>)
Cleared in software
Cleared in software
Cleared in software
Cleared in software
Bus master terminates transfer
BF (SSPSTAT<0>)
SSPBUF is written with contents of SSPSR
Dummy read of SSPBUF to clear BF flag
Dummy read of SSPBUF to clear BF flag
SSPOV (SSPCON1<6>)
SSPOV is set because SSPBUF is still full. ACK is not sent.
UA (SSPSTAT<1>)
UA is set indicating that the SSPADD needs to be updated
Cleared by hardware when SSPADD is updated with low byte of address after falling edge of ninth clock
UA is set indicating that SSPADD needs to be updated
Cleared by hardware when SSPADD is updated with high byte of address after falling edge of ninth clock
CKP
Note: An update of the SSPADD register before the falling edge of the ninth clock will have no effect on UA and UA will remain set.
CKP written to '1' in software
Note: An update of the SSPADD register before the falling edge of the ninth clock will have no effect on UA and UA will remain set.

19.2.3.1 Message Acceptance Filters and Masks

This subsection describes the message acceptance filters and masks for the CAN receive buffers.

REGISTER 19-21: RXFnSIDH: RECEIVE ACCEPTANCE FILTER n STANDARD IDENTIFIER FILTER, HIGH BYTE REGISTERS

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							bit 0

bit 7-0 **SID10:SID3:** Standard Identifier Filter bits if EXIDEN = `0` or
Extended Identifier Filter bits EID28:EID21 if EXIDEN = `1`

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

REGISTER 19-22: RXFnSIDL: RECEIVE ACCEPTANCE FILTER n STANDARD IDENTIFIER FILTER, LOW BYTE REGISTERS

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16
bit 7							bit 0

bit 7-5 **SID2:SID0:** Standard Identifier Filter bits if EXIDEN = `0` or
Extended Identifier Filter bits EID20:EID18 if EXIDEN = `1`

bit 4 **Unimplemented:** Read as ‘`0`’

bit 3 **EXIDEN:** Extended Identifier Filter Enable bit
`1` = Filter will only accept extended ID messages
`0` = Filter will only accept standard ID messages

bit 2 **Unimplemented:** Read as ‘`0`’

bit 1-0 **EID17:EID16:** Extended Identifier Filter bits

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

19.2.5 CAN MODULE I/O CONTROL REGISTER

This register controls the operation of the CAN module's I/O pins in relation to the rest of the microcontroller.

REGISTER 19-32: CIOCON: CAN I/O CONTROL REGISTER

U-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	—	ENDRHI	CANCAP	—	—	—	—
bit 7							bit 0

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

bit 5 **ENDRHI:** Enable Drive High bit

1 = CANTX pin will drive V_{DD} when recessive
0 = CANTX pin will tri-state when recessive

bit 4 **CANCAP:** CAN Message Receive Capture Enable bit

1 = Enable CAN capture, CAN message receive signal replaces input on RC2/CCP1
0 = Disable CAN capture, RC2/CCP1 input to CCP1 module

bit 3-0 **Unimplemented:** Read as '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

19.3 CAN Modes of Operation

The PIC18FXX8 has six main modes of operation:

- Configuration mode
- Disable mode
- Normal Operation mode
- Listen Only mode
- Loopback mode
- Error Recognition mode

All modes, except Error Recognition, are requested by setting the REQOP bits (CANCON<7:5>); Error Recognition is requested through the RXM bits of the Receive Buffer register(s). Entry into a mode is Acknowledged by monitoring the OPMODE bits.

When changing modes, the mode will not actually change until all pending message transmissions are complete. Because of this, the user must verify that the device has actually changed into the requested mode before fUrther Operations Are Executed.

19.3.1 CONFIGURATION MODE

The CAN module has to be initialized before the activation. This is only possible if the module is in the Configuration mode. The Configuration mode is requested by setting the REQOP2 bit. Only when the OPMODE2 status bit has a high level can the initialization be performed. Afterwards, the Configuration registers, the Acceptance Mask registers and the Acceptance Filter registers can be written. The module is activated by setting the REQOP control bits to zero.

The module will protect the user from accidentally violating the CAN protocol through programming errors. All registers which control the configuration of the module can not be modified while the module is on-line. The CAN module will not be allowed to enter the Configuration mode while a transmission is taking place. The CONFIG bit serves as a lock to protect the following registers.

- Configuration registers
- Bus Timing registers
- Identifier Acceptance Filter registers
- Identifier Acceptance Mask registers

In the Configuration mode, the module will not transmit or receive. The error counters are cleared and the interrupt flags remain unchanged. The programmer will have access to Configuration registers that are access restricted in other modes.

19.3.2 DISABLE MODE

In Disable mode, the module will not transmit or receive. The module has the ability to set the WAKIF bit due to bus activity, however, any pending interrupts will remain and the error counters will retain their value.

If REQOP<2:0> is set to '001', the module will enter the Module Disable mode. This mode is similar to disabling other peripheral modules by turning off the module enables. This causes the module internal clock to stop unless the module is active (i.e., receiving or transmitting a message). If the module is active, the module will wait for 11 recessive bits on the CAN bus, detect that condition as an IDLE bus, then accept the module disable command. OPMODE<2:0> = 001 indicates whether the module successfully went into Module Disable mode.

The WAKIF interrupt is the only module interrupt that is still active in the Module Disable mode. If the WAKIE is set, the processor will receive an interrupt whenever the CAN bus detects a dominant state, as occurs with a SOF. If the processor receives an interrupt while it is sleeping, more than one message may get lost. User firmware must anticipate this condition and request retransmission. If the processor is running while it receives an interrupt, only the first message may get lost.

The I/O pins will revert to normal I/O function when the module is in the Module Disable mode.

19.3.3 NORMAL MODE

This is the standard operating mode of the PIC18FXX8. In this mode, the device actively monitors all bus messages and generates Acknowledge bits, error frames, etc. This is also the only mode in which the PIC18FXX8 will transmit messages over the CAN bus.

19.3.4 LISTEN ONLY MODE

Listen Only mode provides a means for the PIC18FXX8 to receive all messages, including messages with errors. This mode can be used for bus monitor applications or for detecting the baud rate in 'hot plugging' situations. For auto-baud detection, it is necessary that there are at least two other nodes which are communicating with each other. The baud rate can be detected empirically by testing different values until valid messages are received. The Listen Only mode is a silent mode, meaning no messages will be transmitted while in this state, including error flags or Acknowledge signals. The filters and masks can be used to allow only particular messages to be loaded into the receive registers, or the filter masks can be set to all zeros to allow a message with any identifier to pass. The error counters are reset and deactivated in this state. The Listen Only mode is activated by setting the mode request bits in the CANCON register.

The analog reference voltage is software selectable to either the device's positive and negative supply voltage (VDD and VSS) or the voltage level on the RA3/AN3/VREF+ pin and RA2/AN2/VREF- pin.

The A/D converter has a unique feature of being able to operate while the device is in Sleep mode. To operate in Sleep, the A/D conversion clock must be derived from the A/D's internal RC oscillator.

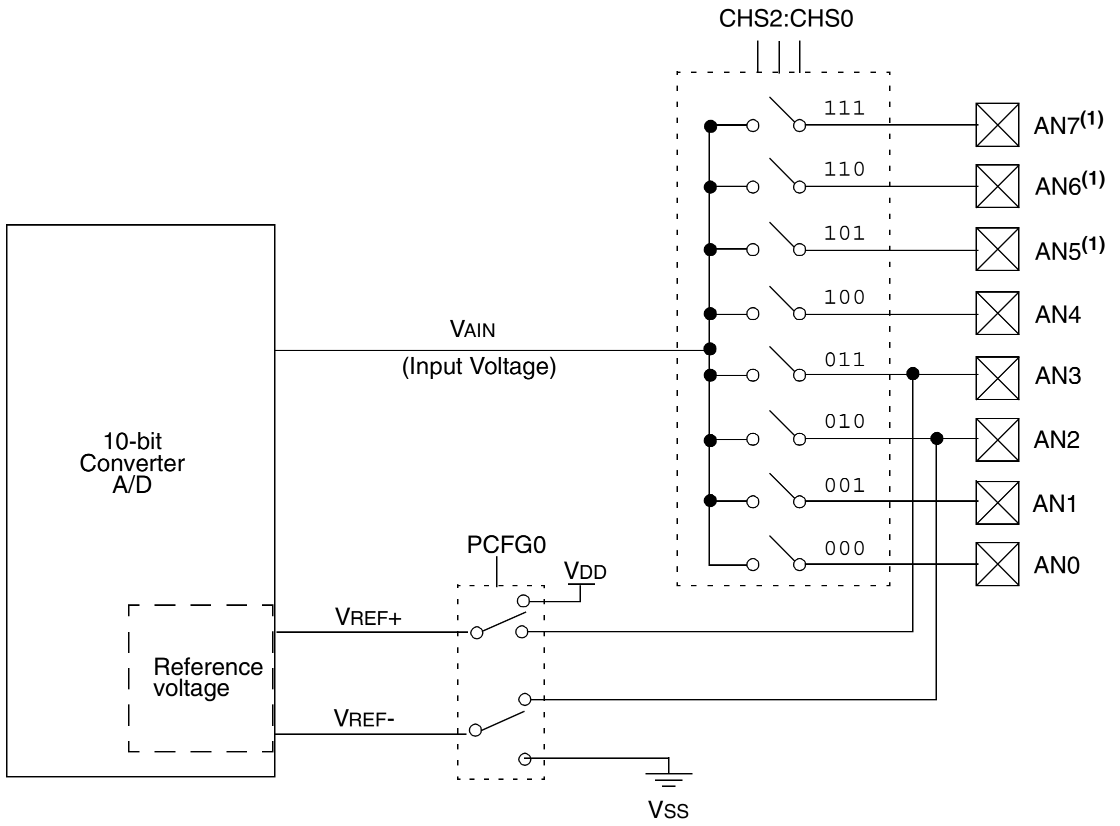
The output of the sample and hold is the input into the converter which generates the result via successive approximation.

A device Reset forces all registers to their Reset state. This forces the A/D module to be turned off and any conversion is aborted.

Each port pin associated with the A/D converter can be configured as an analog input (RA3 can also be a voltage reference) or as a digital I/O.

The ADRESH and ADRESL registers contain the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRESH/ADRESL registers, the GO/$\overline{\text{DONE}}$ bit (ADCON0<2>) is cleared and A/D Interrupt Flag bit, ADIF, is set. The block diagram of the A/D module is shown in Figure 20-1.

FIGURE 20-1: A/D BLOCK DIAGRAM



Note 1: Channels AN5 through AN7 are not available on PIC18F2X8 devices.

2: All I/O pins have diode protection to VDD and VSS.

To calculate the minimum acquisition time, Equation 20-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

Example 20-1 shows the calculation of the minimum required acquisition time T_{ACQ}. This calculation is based on the following application system assumptions:

- C_{HOLD} = 120 pF
- Rs = 2.5 kΩ
- Conversion Error ≤ 1/2 LSb
- V_{DD} = 5V → Rss = 7 kΩ
- Temperature = 50°C (system max.)
- V_{HOLD} = 0V @ time = 0

EQUATION 20-1: ACQUISITION TIME

$$
\begin{aligned}
T_{ACQ} &= \text{Amplifier Settling Time + Holding Capacitor Charging Time + Temperature Coefficient} \\
&= T_{AMP} + T_C + T_{COFF}
\end{aligned}
$$

EQUATION 20-2: A/D MINIMUM CHARGING TIME

$$V_{HOLD} = (V_{REF} - (V_{REF}/2048)) \bullet (1 - e^{(-T_C/C_{HOLD}(R_{IC} + R_{SS} + R_S))})$$

or

$$T_C = -(120\text{ pF})(1\text{ k}\Omega + R_{SS} + R_S)\ \ln(1/2047)$$

EXAMPLE 20-1: CALCULATING THE MINIMUM REQUIRED ACQUISITION TIME

$$T_{ACQ} = T_{AMP} + T_C + T_{COFF}$$

Temperature coefficient is only required for temperatures > 25°C.

$$
\begin{aligned}
T_{ACQ} &= 2\ \mu s + T_C + [(\text{Temp} - 25°C)(0.05\ \mu s/°C)] \\
T_C &= -C_{HOLD}\ (R_{IC} + R_{SS} + R_S)\ \ln(1/2047) \\
&\ -120\text{ pF } (1\text{ k}\Omega + 7\text{ k}\Omega + 2.5\text{ k}\Omega)\ \ln(0.0004885) \\
&\ -120\text{ pF } (10.5\text{ k}\Omega)\ \ln(0.0004885) \\
&\ -1.26\ \mu s\ (-7.6241) \\
&\ 9.61\ \mu s \\
T_{ACQ} &= 2\ \mu s + 9.61\ \mu s + [(50°C - 25°C)(0.05\ \mu s/°C)] \\
&\ 11.61\ \mu s + 1.25\ \mu s \\
&\ 12.86\ \mu s
\end{aligned}
$$

Note: When using external voltage references with the A/D converter, the source impedance of the external voltage references must be less than 20Ω to obtain the specified A/D resolution. Higher reference source impedances will increase both offset and gain errors. Resistive voltage dividers will not provide a sufficiently low source impedance.

To maintain the best possible performance in A/D conversions, external V_{REF} inputs should be buffered with an operational amplifier or other low output impedance circuit.

20.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires 12 TAD per 10-bit conversion. The source of the A/D conversion clock is software selectable. The seven possible options for TAD are:

- 2 TOSC
- 4 TOSC
- 8 TOSC
- 16 TOSC
- 32 TOSC
- 64 TOSC
- Internal RC oscillator.

For correct A/D conversions, the A/D conversion clock (TAD) must be selected to ensure a minimum TAD time of 1.6 µs.

Table 20-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

20.3 Configuring Analog Port Pins

The ADCON1, TRISA and TRISE registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

Note 1: When reading the port register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.

2: Analog levels on any pin that is defined as a digital input (including the AN4:AN0 pins) may cause the input buffer to consume current that is out of the device's specification.

TABLE 20-1: TAD vs. DEVICE OPERATING FREQUENCIES

AD Clock Source (TAD)		Device Frequency			
Operation	**ADCS2:ADCS0**	**20 MHz**	**5 MHz**	**1.25 MHz**	**333.33 kHz**
2 TOSC	`000`	100 ns[2]	400 ns[2]	1.6 µs	6 µs
4 TOSC	`100`	200 ns[2]	800 ns[2]	3.2 µs	12 µs
8 TOSC	`001`	400 ns[2]	1.6 µs	6.4 µs	24 µs[3]
16 TOSC	`101`	800 ns[2]	3.2 µs	12.8 µs	48 µs[3]
32 TOSC	`010`	1.6 µs	6.4 µs	25.6 µs[3]	96 µs[3]
64 TOSC	`110`	3.2 µs	12.8 µs	51.2 µs[3]	192 µs[3]
RC	`011`	2-6 µs[1]	2-6 µs[1]	2-6 µs[1]	2-6 µs[1]

Legend: Shaded cells are outside of recommended range.

Note 1: The RC source has a typical TAD time of 4 µs.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

TABLE 20-2: TAD vs. DEVICE OPERATING FREQUENCIES (FOR EXTENDED, LF DEVICES)

AD Clock Source (TAD)		Device Frequency			
Operation	**ADCS2:ADCS0**	**4 MHz**	**2 MHz**	**1.25 MHz**	**333.33 kHz**
2 TOSC	`000`	500 ns[2]	1.0 µs[2]	1.6 µs[2]	6 µs
4 TOSC	`100`	1.0 µs[2]	2.0 µs[2]	3.2 µs[2]	12 µs
8 TOSC	`001`	2.0 µs[2]	4.0 µs	6.4 µs	24 µs[3]
16 TOSC	`101`	4.0 µs[2]	8.0 µs	12.8 µs	48 µs[3]
32 TOSC	`010`	8.0 µs	16.0 µs	25.6 µs[3]	96 µs[3]
64 TOSC	`110`	16.0 µs	32.0 µs	51.2 µs[3]	192 µs[3]
RC	`011`	3-9 µs[1]	3-9 µs[1]	3-9 µs[1]	3-9 µs[1]

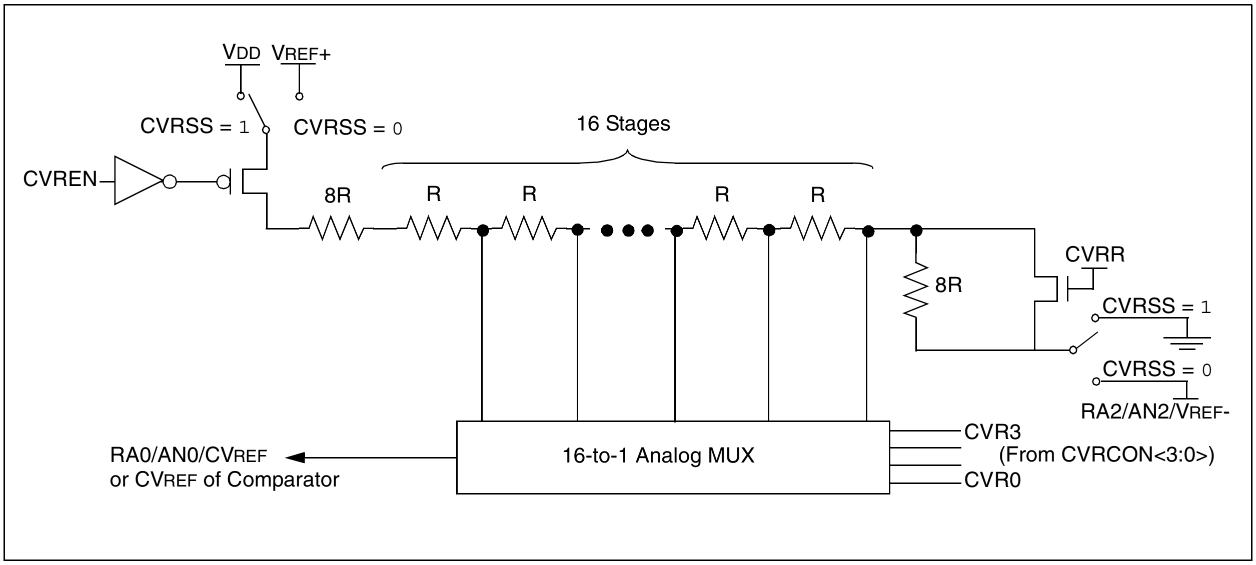
Legend: Shaded cells are outside of recommended range.

Note 1: The RC source has a typical TAD time of 6 µs.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

FIGURE 22-1: VOLTAGE REFERENCE BLOCK DIAGRAM



22.2 Voltage Reference Accuracy/Error

The full range of voltage reference cannot be realized due to the construction of the module. The transistors on the top and bottom of the resistor ladder network (Figure 22-1) keep VREF from approaching the reference source rails. The voltage reference is derived from the reference source; therefore, the VREF output changes with fluctuations in that source. The absolute accuracy of the voltage reference can be found in **Section 27.0 "Electrical Characteristics"**.

22.3 Operation During Sleep

When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the CVRCON register are not affected. To minimize current consumption in Sleep mode, the voltage reference should be disabled.

22.4 Effects of a Reset

A device Reset disables the voltage reference by clearing bit CVREN (CVRCON register). This Reset also disconnects the reference from the RA2 pin by clearing bit CVROE (CVRCON register) and selects the high-voltage range by clearing bit CVRR (CVRCON register). The CVRSS value select bits, CVRCON<3:0>, are also cleared.

22.5 Connection Considerations

The voltage reference module operates independently of the comparator module. The output of the reference generator may be connected to the RA0/AN0 pin if the TRISA<0> bit is set and the CVROE bit (CVRCON<6>) is set. Enabling the voltage reference output onto the RA0/AN0 pin, with an input signal present, will increase current consumption. Connecting RA0/AN0 as a digital output, with CVRSS enabled, will also increase current consumption.

The RA0/AN0 pin can be used as a simple D/A output with limited drive capability. Due to the limited current drive capability, a buffer must be used on the voltage reference output for external connections to VREF. Figure 22-2 shows an example buffering technique.

REGISTER 24-1: CONFIG1H: CONFIGURATION REGISTER 1 HIGH (BYTE ADDRESS 300001h)

U-0	U-0	R/P-1	U-0	U-0	R/P-1	R/P-1	R/P-1
—	—	$\overline{\text{OSCSEN}}$	—	—	FOSC2	FOSC1	FOSC0
bit 7							bit 0

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

bit 5 **$\overline{\text{OSCSEN}}$**: Oscillator System Clock Switch Enable bit

1 = Oscillator system clock switch option is disabled (main oscillator is source)
0 = Oscillator system clock switch option is enabled (oscillator switching is enabled)

bit 4-3 **Unimplemented:** Read as '0'

bit 2-0 **FOSC2:FOSC0**: Oscillator Selection bits

111 = RC oscillator w/OSC2 configured as RA6
110 = HS oscillator with PLL enabled/clock frequency = (4 x F_{OSC})
101 = EC oscillator w/OSC2 configured as RA6
100 = EC oscillator w/OSC2 configured as divide-by-4 clock output
011 = RC oscillator
010 = HS oscillator
001 = XT oscillator
000 = LP oscillator

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

REGISTER 24-2: CONFIG2L: CONFIGURATION REGISTER 2 LOW (BYTE ADDRESS 300002h)

U-0	U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	BORV1	BORV0	BOREN	$\overline{\text{PWRTEN}}$
bit 7							bit 0

bit 7-4 **Unimplemented:** Read as '0'

bit 3-2 **BORV1:BORV0:** Brown-out Reset Voltage bits

11 = V_{BOR} set to 2.0V
10 = V_{BOR} set to 2.7V
01 = V_{BOR} set to 4.2V
00 = V_{BOR} set to 4.5V

bit 1 **BOREN:** Brown-out Reset Enable bit

1 = Brown-out Reset enabled
0 = Brown-out Reset disabled

bit 0 **$\overline{\text{PWRTEN}}$:** Power-up Timer Enable bit

1 = PWRT disabled
0 = PWRT enabled

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

REGISTER 24-11: DEVID1: DEVICE ID REGISTER 1 FOR PIC18FXX8 DEVICES (BYTE ADDRESS 3FFFFEh)

R	R	R	R	R	R	R	R
DEV2	DEV1	DEV0	REV4	REV3	REV2	REV1	REV0
bit 7							bit 0

bit 7-5 **DEV2:DEV0:** Device ID bits

These bits are used with the DEV<10:3> bits in the Device ID Register 2 to identify the part number.

`000` = PIC18F248
`001` = PIC18F448
`010` = PIC18F258
`011` = PIC18F458

bit 4-0 **REV4:REV0:** Revision ID bits

These bits are used to indicate the device revision.

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’
-n = Value when device is unprogrammed		u = Unchanged from programmed state

REGISTER 24-12: DEVID2: DEVICE ID REGISTER 2 FOR PIC18FXX8 DEVICES (BYTE ADDRESS 3FFFFFh)

R	R	R	R	R	R	R	R
DEV10	DEV9	DEV8	DEV7	DEV6	DEV5	DEV4	DEV3
bit 7							bit 0

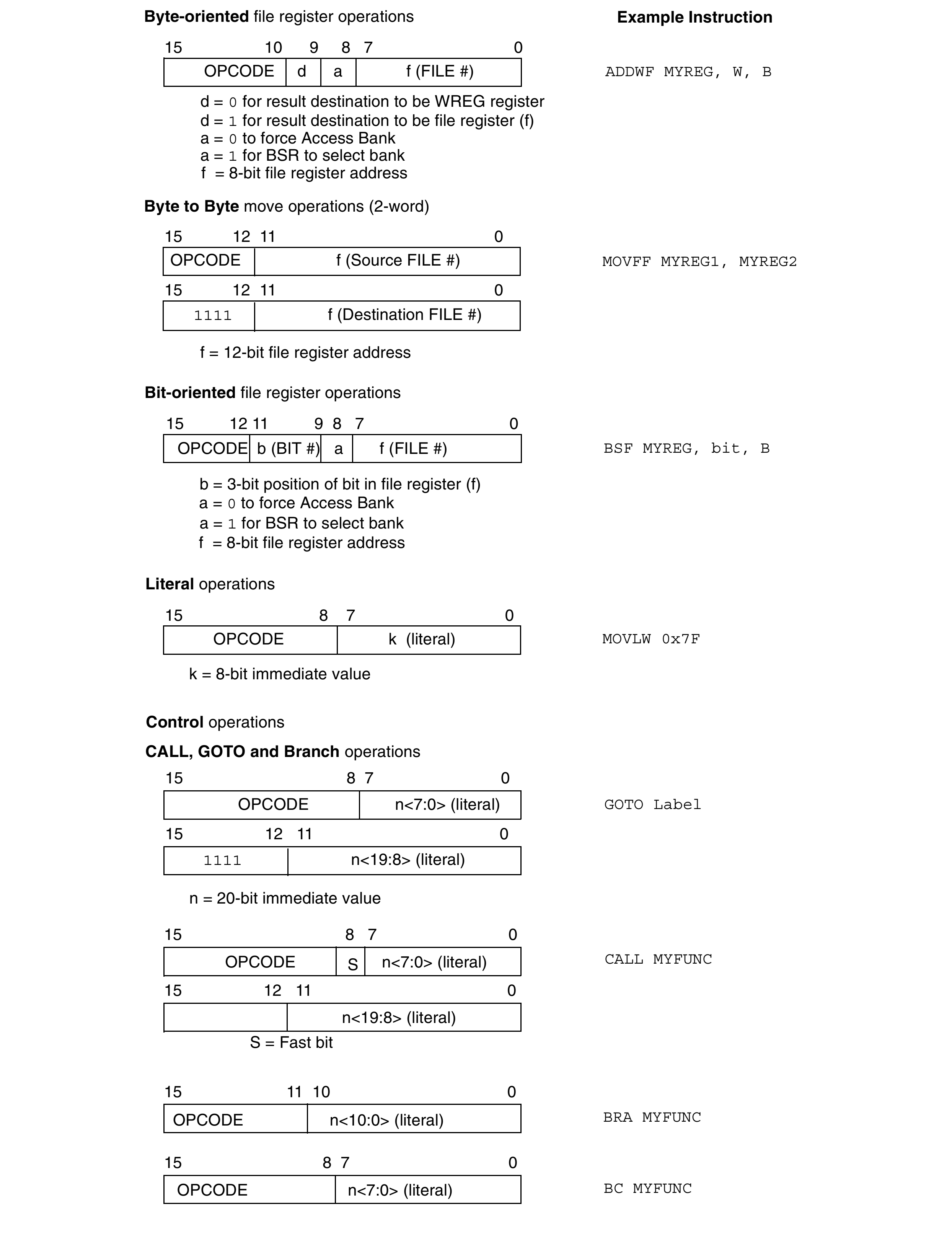
bit 7-0 **DEV10:DEV3:** Device ID bits

These bits are used with the DEV<2:0> bits in the Device ID Register 1 to identify the part number.

`00001000` = PIC18FXX8

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’
-n = Value when device is unprogrammed		u = Unchanged from programmed state

FIGURE 25-1: GENERAL FORMAT FOR INSTRUCTIONS


Byte-oriented file register operations
Example Instruction
15
10
9
8
7
0
OPCODE
d
a
f (FILE #)
ADDWF MYREG, W, B
d = 0 for result destination to be WREG register
d = 1 for result destination to be file register (f)
a = 0 to force Access Bank
a = 1 for BSR to select bank
f = 8-bit file register address
Byte to Byte move operations (2-word)
15
12
11
0
OPCODE
f (Source FILE #)
MOVFF MYREG1, MYREG2
15
12
11
0
1111
f (Destination FILE #)
f = 12-bit file register address
Bit-oriented file register operations
15
12
11
9
8
7
0
OPCODE
b (BIT #)
a
f (FILE #)
BSF MYREG, bit, B
b = 3-bit position of bit in file register (f)
a = 0 to force Access Bank
a = 1 for BSR to select bank
f = 8-bit file register address
Literal operations
15
8
7
0
OPCODE
k (literal)
MOVLW 0x7F
k = 8-bit immediate value
Control operations
CALL, GOTO and Branch operations
15
8
7
0
OPCODE
n<7:0> (literal)
GOTO Label
15
12
11
0
1111
n<19:8> (literal)
n = 20-bit immediate value
15
8
7
0
OPCODE
S
n<7:0> (literal)
CALL MYFUNC
15
12
11
0
n<19:8> (literal)
S = Fast bit
15
11
10
0
OPCODE
n<10:0> (literal)
BRA MYFUNC
15
8
7
0
OPCODE
n<7:0> (literal)
BC MYFUNC

GOTO — Unconditional Branch

Syntax: [*label*] GOTO k

Operands: 0 ≤ k ≤ 1048575

Operation: k → PC<20:1>

Status Affected: None

Encoding:

1st word (k<7:0>)	1110	1111	k$_7$kkk	kkkk$_0$
2nd word(k<19:8>)	1111	k$_{19}$kkk	kkkk	kkkk$_8$

Description: GOTO allows an unconditional branch anywhere within entire 2-Mbyte memory range. The 20-bit value 'k' is loaded into PC<20:1>. GOTO is always a two-cycle instruction.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'<7:0>	No operation	Read literal 'k'<19:8>, Write to PC
No operation	No operation	No operation	No operation

Example: GOTO THERE

After Instruction

PC	=	Address (THERE)

INCF — Increment f

Syntax: [*label*] INCF f [,d [,a]]

Operands: 0 ≤ f ≤ 255
d ∈ [0,1]
a ∈ [0,1]

Operation: (f) + 1 → dest

Status Affected: C, DC, N, OV, Z

Encoding:

0010	10da	ffff	ffff

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default). If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' = 1, then the bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: INCF CNT,

Before Instruction

CNT	=	0xFF
Z	=	0
C	=	?
DC	=	?

After Instruction

CNT	=	0x00
Z	=	1
C	=	1
DC	=	1

IORLW — Inclusive OR Literal with W

Syntax:	[*label*] IORLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .OR. k → W
Status Affected:	N, Z

Encoding:

`0000`	`1001`	`kkkk`	`kkkk`

Description:	The contents of W are ORed with the eight-bit literal 'k'. The result is placed in W.
Words:	1
Cycles:	1

Q Cycle Activity:

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

Example: `IORLW 0x35`

Before Instruction
W = 0x9A
After Instruction
W = 0xBF

IORWF — Inclusive OR W with f

Syntax:	[*label*] IORWF f [,d [,a]]
Operands:	0 ≤ f ≤ 255 d ∈ [0,1] a ∈ [0,1]
Operation:	(W) .OR. (f) → dest
Status Affected:	N, Z

Encoding:

`0001`	`00da`	`ffff`	`ffff`

Description:	Inclusive OR W with register 'f'. If 'd' is '`0`', the result is placed in W. If 'd' is '`1`', the result is placed back in register 'f' (default). If 'a' is '`0`', the Access Bank will be selected, overriding the BSR value. If 'a' = `1`, then the bank will be selected as per the BSR value (default).
Words:	1
Cycles:	1

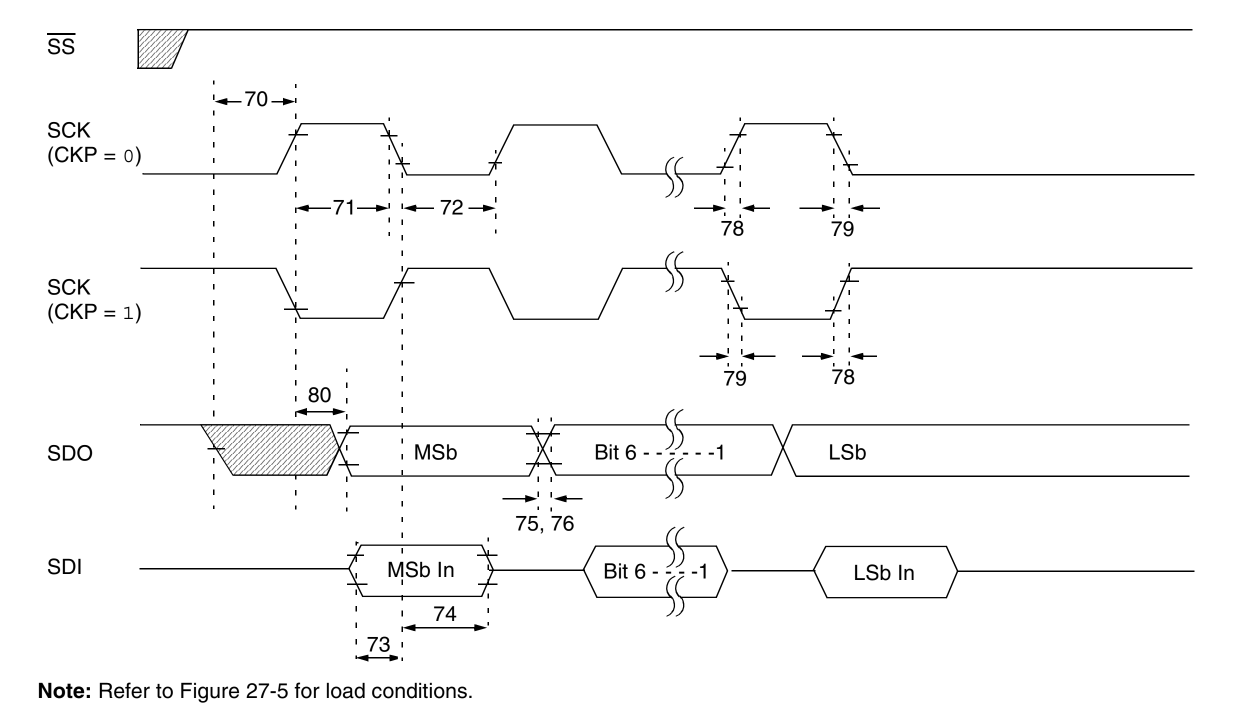
Q Cycle Activity:

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

Example: `IORWF RESULT, W`

Before Instruction
RESULT = 0x13
W = 0x91
After Instruction
RESULT = 0x13
W = 0x93

FIGURE 27-13: EXAMPLE SPI™ MASTER MODE TIMING (CKE = 0)



Note: Refer to Figure 27-5 for load conditions.

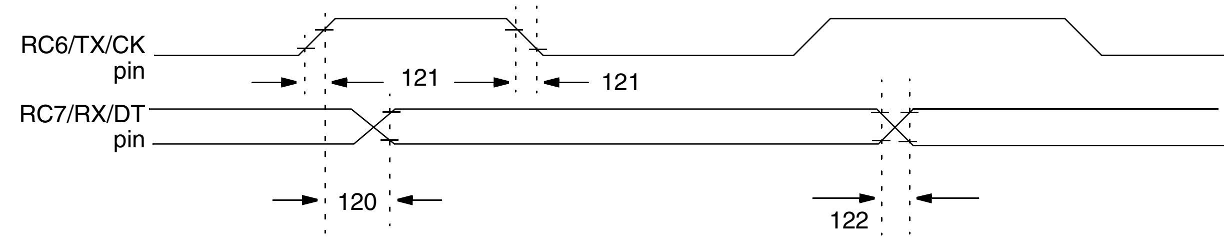
TABLE 27-13: EXAMPLE SPI™ MODE REQUIREMENTS (MASTER MODE, CKE = 0)

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK ↓ or SCK ↑ Input		TCY	—	ns	
71	TscH	SCK Input High Time (Slave mode)	Continuous	1.25 TCY + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCK Input Low Time (Slave mode)	Continuous	1.25 TCY + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
73	TdiV2scH, TdiV2scL	Setup Time of SDI Data Input to SCK Edge		100	—	ns	
73A	TB2B	Last Clock Edge of Byte 1 to the 1st Clock Edge of Byte 2		1.5 TCY + 40	—	ns	**(Note 2)**
74	TscH2diL, TscL2diL	Hold Time of SDI Data Input to SCK Edge		100	—	ns	
75	TdoR	SDO Data Output Rise Time	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	45	ns	
76	TdoF	SDO Data Output Fall Time		—	25	ns	
78	TscR	SCK Output Rise Time (Master mode)	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	45	ns	
79	TscF	SCK Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO Data Output Valid after SCK Edge	PIC18**F**XX8	—	50	ns	
			PIC18**LF**XX8	—	100	ns	

Note 1: Requires the use of parameter #73A.

2: Only if parameter #71A and #72A are used.

FIGURE 27-21: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

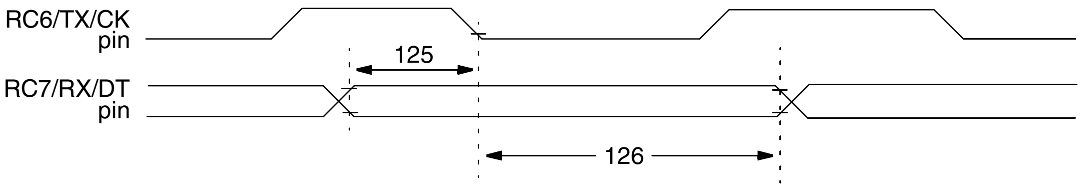


Note: Refer to Figure 27-5 for load conditions.

TABLE 27-21: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (Master & Slave) Clock High to Data-Out Valid	PIC18**F**XX8	—	50	ns	
			PIC18**LF**XX8	—	150	ns	
121	Tckrf	Clock Out Rise Time and Fall Time (Master mode)	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	60	ns	
122	Tdtrf	Data-Out Rise Time and Fall Time	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	60	ns	

FIGURE 27-22: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



Note: Refer to Figure 27-5 for load conditions.

TABLE 27-22: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
125	TdtV2ckl	SYNC RCV (Master & Slave) Data-Hold before CK ↓ (DT hold time)	10	—	ns	
126	TckL2dtl	Data-Hold after CK ↓ (DT hold time)	15	—	ns	