

Welcome to [E-XFL.COM](https://www.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	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 31x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf19176t-i-mv

TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19155/56/75/76/85/86 (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: MCLR
Bank 62 (Continued)											
1F4Eh	ANSEL	ANSC7	ANSC6	—	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0	1111 1111	1111 1111
1F4Fh	WPUC	WPUC7	WPUC6	—	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	0000 0000	0000 0000
1F50h	ODCONC	ODCC7	ODCC6	—	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0	0000 0000	0000 0000
1F51h	SLRCONC	SLRC7	SLRC6	—	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0	1111 1111	1111 1111
1F52h	INLVLC	INLVLC7	INLVLC6	—	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	1111 1111	1111 1111
1F53h	IOCCP	IOCCP7	IOCCP6	—	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	0000 0000	0000 0000
1F54h	IOCCN	IOCCN7	IOCCN6	—	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	0000 0000	0000 0000
1F55h	IOCCF	IOCCF7	IOCCF6	—	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	0000 0000	0000 0000
1F56h	—	Unimplemented								----	----
1F57h	—	Unimplemented								----	----
1F58h	—	Unimplemented								----	----
1F59h	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
1F5Ah	WPUD	WPUD7	WPUD6	WPUD5	WPUD4	WPUD3	WPUD2	WPUD1	WPUD0	0000 0000	0000 0000
1F5Bh	ODCOND	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	0000 0000	0000 0000
1F5Ch	SLRCOND	SLRD7	SLRD6	SLRD5	SLRD4	SLRD3	SLRD2	SLRD1	SLRD0	1111 1111	1111 1111
1F5Dh	INLVLD	INLVLD7	INLVLD6	INLVLD5	INLVLD4	INLVLD3	INLVLD2	INLVLD1	INLVLD0	1111 1111	1111 1111
1F5Eh	—	Unimplemented								----	----
1F5Fh	—	Unimplemented								----	----
1F60h	—	Unimplemented								----	----
1F61h	—	Unimplemented								----	----
1F62h	—	Unimplemented								----	----
1F63h	—	Unimplemented								----	----

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

Note 1: Unimplemented data memory locations, read as '0'.

10.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) of the PIRx[y] registers for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIRx registers)

The PIR1, PIR2, PIR3, PIR4, PIR5, PIR6, PIR7 and PIR8 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while interrupts are enabled:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See “**Section 10.5 “Auto-matic Context Saving”**”)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The `RETFIE` instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupts operation, refer to its peripheral chapter.

- Note 1:** Individual interrupt flag bits are set, regardless of the state of any other enable bits.
- 2:** All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again.

10.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The interrupt is sampled during Q1 of the instruction cycle. The actual interrupt latency then depends on the instruction that is executing at the time the interrupt is detected. See Figure 10-2 and Figure 10-3 for more details.

PIC16(L)F19155/56/75/76/85/86

REGISTER 10-16: PIR5: PERIPHERAL INTERRUPT REQUEST REGISTER 5

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	R/W/HS-0/0
CLC4IF	CLC3IF	CLC2IF	CLC1IF	—	—	—	TMR1GIF
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

HS = Hardware set

- bit 7 **CLC4IF:** CLC4 Interrupt Flag bit
1 = A CLC4OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC4 interrupt event has occurred
- bit 6 **CLC3IF:** CLC3 Interrupt Flag bit
1 = A CLC3OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC3 interrupt event has occurred
- bit 5 **CLC2IF:** CLC2 Interrupt Flag bit
1 = A CLC2OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC2 interrupt event has occurred
- bit 4 **CLC1IF:** CLC1 Interrupt Flag bit
1 = A CLC1OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC1 interrupt event has occurred
- bit 3-1 **Unimplemented:** Read as '0'
- bit 0 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = The Timer1 Gate has gone inactive (the acquisition is complete)
0 = The Timer1 Gate has not gone inactive

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

14.2.5 INPUT THRESHOLD CONTROL

The INLVLA register (Register 14-8) controls the input voltage threshold for each of the available PORTA input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTA register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 39-4 for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

14.2.6 ANALOG CONTROL

The ANSELA register (Register 14-4) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on digital output functions. A pin with its TRIS bit clear and its ANSEL bit set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELA bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

14.2.7 WEAK PULL-UP CONTROL

The WPUA register (Register 14-5) controls the individual weak pull-ups for each PORT pin.

14.2.8 PORTA FUNCTIONS AND OUTPUT PRIORITIES

Each PORTA pin is multiplexed with other functions.

Each pin defaults to the PORT latch data after Reset. Other output functions are selected with the peripheral pin select logic or by enabling an analog output, such as the DAC. See **Section 15.0 "Peripheral Pin Select (PPS) Module"** for more information.

Analog input functions, such as ADC and comparator inputs are not shown in the peripheral pin select lists. Digital output functions may continue to control the pin when it is in Analog mode.

PIC16(L)F19155/56/75/76/85/86

14.7 Register Definitions: PORTC

REGISTER 14-18: PORTC: PORTC REGISTER

R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
RC7	RC6	—	RC4	RC3	RC2	RC1	RC0
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 7-6 **RC<7:6>**: PORTC General Purpose I/O Pin bits⁽¹⁾

1 = Port pin is $\geq V_{IH}$

0 = Port pin is $\leq V_{IL}$

bit 5 **Unimplemented**: Read as '0'.

bit 4-0 **RC<4:0>**: PORTC General Purpose I/O Pin bits⁽¹⁾

1 = Port pin is $\geq V_{IH}$

0 = Port pin is $\leq V_{IL}$

Note 1: Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

REGISTER 14-19: TRISC: PORTC TRI-STATE REGISTER

R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISC7	TRISC6	—	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
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 7-6 **TRISC<7:6>**: PORTC Tri-State Control bits

1 = PORTC pin configured as an input (tri-stated)

0 = PORTC pin configured as an output

bit 5 **Unimplemented**: Read as '0'.

bit 4-0 **TRISC<4:0>**: PORTC Tri-State Control bits

1 = PORTC pin configured as an input (tri-stated)

0 = PORTC pin configured as an output

14.8 PORTD Registers

Note: PORTD functionality is not available on the PIC16(L)F19155/56 family of devices.

14.8.1 DATA REGISTER

PORTD is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISD (Register 14-2). Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., disable the output driver). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 14.2.8 shows how to initialize PORTD.

Reading the PORTD register (Register 14-1) reads the status of the pins, whereas writing to it will write to the PORT latch.

The PORT data latch LATD (Register 14-3) holds the output port data, and contains the latest value of a LATD or PORTD write.

EXAMPLE 14-3: INITIALIZING PORTD

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

BANKSEL PORTD      ;
CLRF PORTD          ;Init PORTD
BANKSEL LATD        ;Data Latch
CLRF LATD           ;
BANKSEL ANSEL        ;
CLRF ANSEL          ;digital I/O
BANKSEL TRISD       ;
MOVLW B'00111000'  ;Set RD<5:3> as inputs
MOVWF TRISD         ;and set RD<2:0> as
                   ;outputs
```

14.8.2 DIRECTION CONTROL

The TRISD register (Register 14-2) controls the PORTD pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISD register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

14.8.3 OPEN-DRAIN CONTROL

The ODCOND register (Register 14-6) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCOND bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCOND bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

Note: It is not necessary to set open-drain control when using the pin for I²C; the I²C module controls the pin and makes the pin open-drain.

14.8.4 SLEW RATE CONTROL

The SLRCOND register (Register 14-7) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCOND bit is set, the corresponding port pin drive is slew rate limited. When an SLRCOND bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

14.8.5 INPUT THRESHOLD CONTROL

The INLVLD register (Register 14-8) controls the input voltage threshold for each of the available PORTD input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTD register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 39-4 for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

14.10 PORTE Registers

14.10.1 DATA REGISTER

PORTE is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISE (Register 14-2). Setting a TRISE bit (= 1) will make the corresponding PORTE pin an input (i.e., disable the output driver). Clearing a TRISE bit (= 0) will make the corresponding PORTE pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 14.2.8 shows how to initialize PORTE.

Reading the PORTE register (Register 14-1) reads the status of the pins, whereas writing to it will write to the PORT latch.

The PORT data latch LATE (Register 14-3) holds the output port data, and contains the latest value of a LATE or PORTE write.

EXAMPLE 14-4: INITIALIZING PORTE

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

BANKSEL PORTE      ;
CLRFB  PORTE       ;Init PORTE
BANKSEL LATE        ;Data Latch
CLRFB  LATE         ;
BANKSEL ANSELE      ;
CLRFB  ANSELE       ;digital I/O
BANKSEL TRISE       ;
MOVLW  B'00111000' ;Set RE<5:3> as inputs
MOVWF  TRISE        ;and set RE<2:0> as
                   ;outputs
```

14.10.2 DIRECTION CONTROL

The TRISE register (Register 14-2) controls the PORTE pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISE register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

14.10.3 OPEN-DRAIN CONTROL

The ODCONE register (Register 14-6) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONE bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONE bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

Note: It is not necessary to set open-drain control when using the pin for I²C; the I²C module controls the pin and makes the pin open-drain.

14.10.4 SLEW RATE CONTROL

The SLRCONE register (Register 14-7) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONE bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONE bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

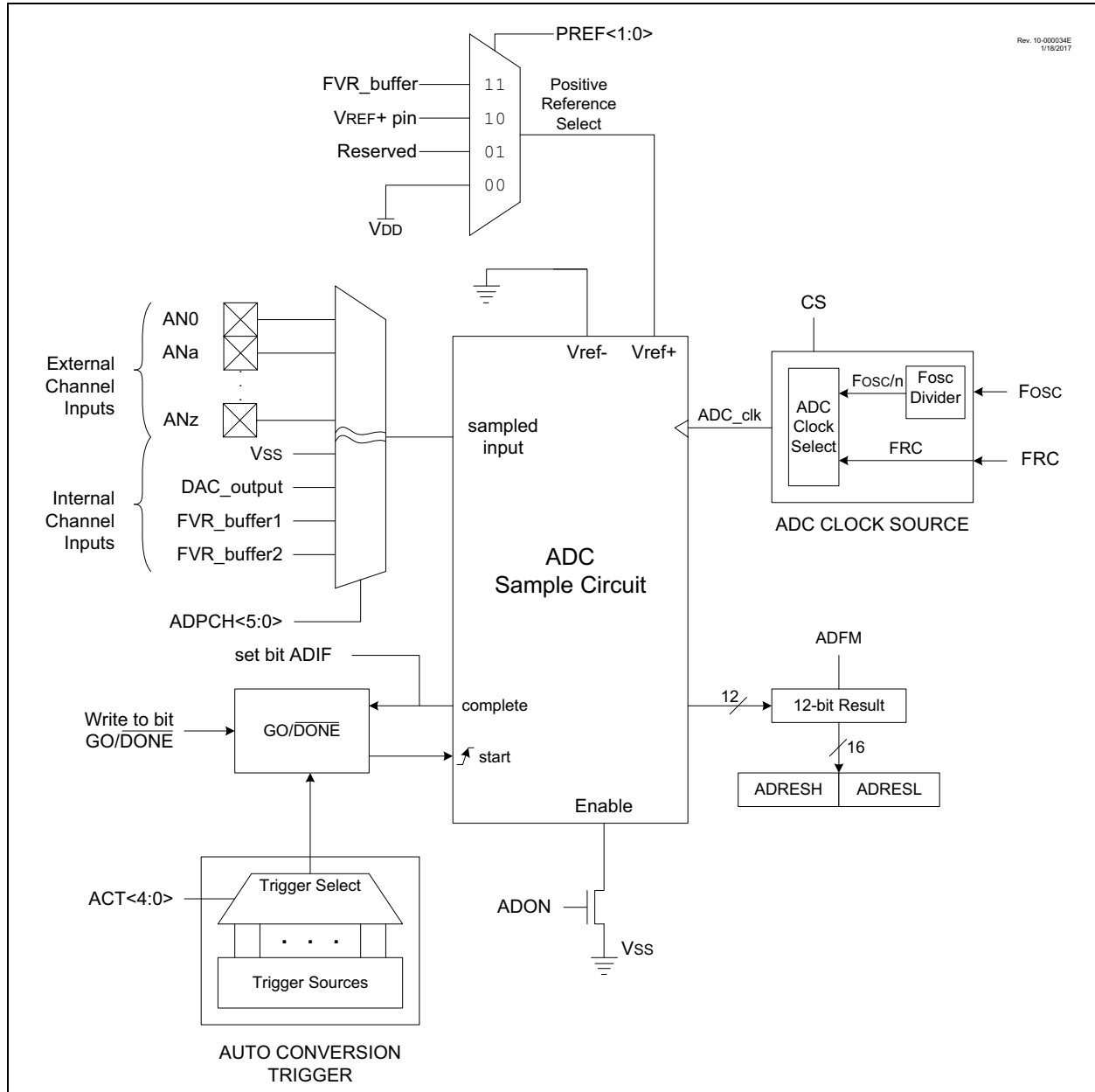
14.10.5 INPUT THRESHOLD CONTROL

The INLVLE register (Register 14-8) controls the input voltage threshold for each of the available PORTE input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTE register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 39-4 for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

PIC16(L)F19155/56/75/76/85/86

FIGURE 19-1: ADC² BLOCK DIAGRAM



19.3 ADC Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 19-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), refer to Figure 19-4. **The maximum recommended impedance for analog sources is 1 kΩ.** If the source

impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an ADC acquisition must be completed before the conversion can be started. To calculate the minimum acquisition time, Equation 19-1 may be used. This equation assumes that 1/2 LSB error is used (4,096 steps for the ADC). The 1/2 LSB error is the maximum error allowed for the ADC to meet its specified accuracy.

EQUATION 19-1: ACQUISITION TIME EXAMPLE

Assumptions: Temperature = 50°C and external impedance of 1kΩ 5.0V VDD

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(Temperature - 25^\circ C)(0.05\mu s/^\circ C)] \end{aligned}$$

The value for TC can be approximated with the following equations:

$$V_{APPLIED} \left(1 - \frac{1}{(2^{n+1}) - 1} \right) = V_{CHOLD} \quad ;[1] \text{ } V_{CHOLD} \text{ charged to within } 1/2 \text{ lsb}$$

$$V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) = V_{CHOLD} \quad ;[2] \text{ } V_{CHOLD} \text{ charge response to } V_{APPLIED}$$

$$V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) = V_{APPLIED} \left(1 - \frac{1}{(2^{n+1}) - 1} \right) \quad ;\text{combining [1] and [2]}$$

Note: Where n = number of bits of the ADC.

Solving for TC:

$$\begin{aligned} T_C &= -CHOLD(RIC + RSS + RS) \ln(1/8191) \\ &= -28 \text{ pF}(1k\Omega + 7k\Omega + 1k\Omega) \ln(0.0001221) \\ &= 2.27\mu s \end{aligned}$$

Therefore:

$$\begin{aligned} T_{ACQ} &= 2\mu s + 4.54\mu s + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \\ &= 5.52\mu s \end{aligned}$$

Note 1: The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (CHOLD) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 1kΩ. This is required to meet the pin leakage specification.

PIC16(L)F19155/56/75/76/85/86

REGISTER 19-4: ADCON3: ADC CONTROL REGISTER 3

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W/HC-0	R/W-0/0	R/W-0/0	R/W-0/0
—	CALC<2:0>			SOI	TMD<2:0>		
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	HC = Bit is cleared by hardware

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

bit 6-4 **CALC<2:0>:** ADC Error Calculation Mode Select bits

CALC	DSEN = 0 Single-Sample Mode	DSEN = 1 CVD Double-Sample Mode ⁽¹⁾	Application
111	Reserved	Reserved	Reserved
110	Reserved	Reserved	Reserved
101	FLTR-STPT	FLTR-STPT	Average/filtered value vs. setpoint
100	PREV-FLTR	PREV-FLTR	First derivative of filtered value ⁽³⁾ (negative)
011	Reserved	Reserved	Reserved
010	RES-FLTR	(RES-PREV)-FLTR	Actual result vs. averaged/filtered value
001	RES-STPT	(RES-PREV)-STPT	Actual result vs. setpoint
000	RES-PREV	RES-PREV	First derivative of single measurement ⁽²⁾
			Actual CVD result in CVD mode ⁽²⁾

bit 3 **SOI:** ADC Stop-on-Interrupt bit

If **CONT = 1**:

1 = GO is cleared when the threshold conditions are met, otherwise the conversion is retrigged
0 = GO is not cleared by hardware, must be cleared by software to stop retriggers

bit 2-0 **TMD<2:0>:** Threshold Interrupt Mode Select bits

111 = Interrupt regardless of threshold test results
110 = Interrupt if ERR>UTH
101 = Interrupt if ERR≤UTH
100 = Interrupt if ERR<LTH or ERR>UTH
011 = Interrupt if ERR>LTH and ERR<UTH
010 = Interrupt if ERR≥LTH
001 = Interrupt if ERR<LTH
000 = Never interrupt

Note 1: When PSIS = 0, the value of (RES-PREV) is the value of (S2-S1) from Table 19-2.

2: When ADPSIS = 0.

3: When ADPSIS = 1.

PIC16(L)F19155/56/75/76/85/86

REGISTER 19-18: ADRESH: ADC RESULT REGISTER HIGH, FM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<11:4>							
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 7-0

ADRES<11:4>: ADC Result Register bits
Upper eight bits of 12-bit conversion result.

REGISTER 19-19: ADRESL: ADC RESULT REGISTER LOW, FM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
ADRES<3:0>				—	—	—	—
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 7-4

ADRES<3:0>: ADC Result Register bits. Lower four bits of 12-bit conversion result.

bit 3-0

Unimplemented: Read as '0'

PIC16(L)F19155/56/75/76/85/86

REGISTER 24-10: ALRMCON: ALARM 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	U-0	U-0
ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	—	—
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 **ALRMEN:** Alarm Enable bit⁽¹⁾

1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 0000 0000 and CHIME = 0)

0 = Alarm is disabled

bit 6 **CHIME:** Chime Enable bit

1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 00h to FFh

0 = Chime is disabled; ARPT<7:0> bits stop once they reach 00h

bit 5-2 **AMASK<3:0>:** Alarm Mask Configuration bits

0000 = Every half second

0001 = Every second

0010 = Every 10 seconds

0011 = Every minute

0100 = Every 10 minutes

0101 = Every hour

0110 = Once a day

0111 = Once a week

1000 = Once a month

1001 = Once a year (except when configured for February 29th, once every four years)

101x = Reserved – do not use

11xx = Reserved – do not use

bit 1-0 **Unimplemented:** Read as '0'

Note 1: ALRMEN is cleared automatically any time an alarm event occurs when ARPT<7:0> = 00 and CHIME = 0

REGISTER 24-11: ALRMRPT: ALARM REPEAT REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits⁽¹⁾

00000000 = Alarm will repeat 0 more times

.

.

.

11111111 = Alarm will repeat 255 more times

Note 1: The counter decrements on any alarm event. The counter is prevented from rolling over from '255' to '0' unless CHIME = 1.

PIC16(L)F19155/56/75/76/85/86

REGISTER 28-10: SMTxCPR_L: SMT CAPTURED PERIOD REGISTER – LOW BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<7:0>							
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 7-0 **SMTxCPR<7:0>**: Significant bits of the SMT Period Latch – Low Byte

REGISTER 28-11: SMTxCPR_H: SMT CAPTURED PERIOD REGISTER – HIGH BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<15:8>							
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 7-0 **SMTxCPR<15:8>**: Significant bits of the SMT Period Latch – High Byte

REGISTER 28-12: SMTxCPR_U: SMT CAPTURED PERIOD REGISTER – UPPER BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<23:16>							
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 7-0 **SMTxCPR<23:16>**: Significant bits of the SMT Period Latch – Upper Byte

PIC16(L)F19155/56/75/76/85/86

30.2 Register Definitions: PWM Control

REGISTER 30-1: PWMxCON: PWM CONTROL REGISTER

R/W-0/0	U-0	R-0	R/W-0/0	U-0	U-0	U-0	U-0
PWMxEN	—	PWMxOUT	PWMxPOL	—	—	—	—
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 **PWMxEN:** PWM Module Enable bit

1 = PWM module is enabled

0 = PWM module is disabled

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

bit 5 **PWMxOUT:** PWM Module Output Level when Bit is Read

bit 4 **PWMxPOL:** PWMx Output Polarity Select bit

1 = PWM output is active-low

0 = PWM output is active-high

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

32.2 CLCx Interrupts

An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

The CLCxIF bit of the associated PIR5 register will be set when either edge detector is triggered and its associated enable bit is set. The LCxINTP enables rising edge interrupts and the LCxINTN bit enables falling edge interrupts. Both are located in the CLCxCON register.

To fully enable the interrupt, set the following bits:

- CLCxIE bit of the PIE5 register
- LCxINTP bit of the CLCxCON register (for a rising edge detection)
- LCxINTN bit of the CLCxCON register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The CLCxIF bit of the PIR5 register, must be cleared in software as part of the interrupt service. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

32.3 Output Mirror Copies

Mirror copies of all LCxCON output bits are contained in the CLCxDATA register. Reading this register reads the outputs of all CLCs simultaneously. This prevents any reading skew introduced by testing or reading the LCxOUT bits in the individual CLCxCON registers.

32.4 Effects of a Reset

The CLCxCON register is cleared to zero as the result of a Reset. All other selection and gating values remain unchanged.

32.5 Operation During Sleep

The CLC module operates independently from the system clock and will continue to run during Sleep, provided that the input sources selected remain active.

The HFINTOSC remains active during Sleep when the CLC module is enabled and the HFINTOSC is selected as an input source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and as a CLC input source, when the CLC is enabled, the CPU will go idle during Sleep, but the CLC will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

32.6 CLCx Setup Steps

The following steps should be followed when setting up the CLCx:

- Disable CLCx by clearing the LCxEN bit.
- Select desired inputs using CLCxSEL0 through CLCxSEL3 registers (See Table 32-2).
- Clear any associated ANSEL bits.
- Enable the chosen inputs through the four gates using CLCxGLS0, CLCxGLS1, CLCxGLS2, and CLCxGLS3 registers.
- Select the gate output polarities with the LCxGyPOL bits of the CLCxPOL register.
- Select the desired logic function with the LCxMODE<2:0> bits of the CLCxCON register.
- Select the desired polarity of the logic output with the LCxPOL bit of the CLCxPOL register. (This step may be combined with the previous gate output polarity step).
- If driving a device pin, set the desired pin PPS control register and also clear the TRIS bit corresponding to that output.
- If interrupts are desired, configure the following bits:
 - Set the LCxINTP bit in the CLCxCON register for rising event.
 - Set the LCxINTN bit in the CLCxCON register for falling event.
 - Set the CLCxIE bit of the PIE5 register.
 - Set the GIE and PEIE bits of the INTCON register.
- Enable the CLCx by setting the LCxEN bit of the CLCxCON register.

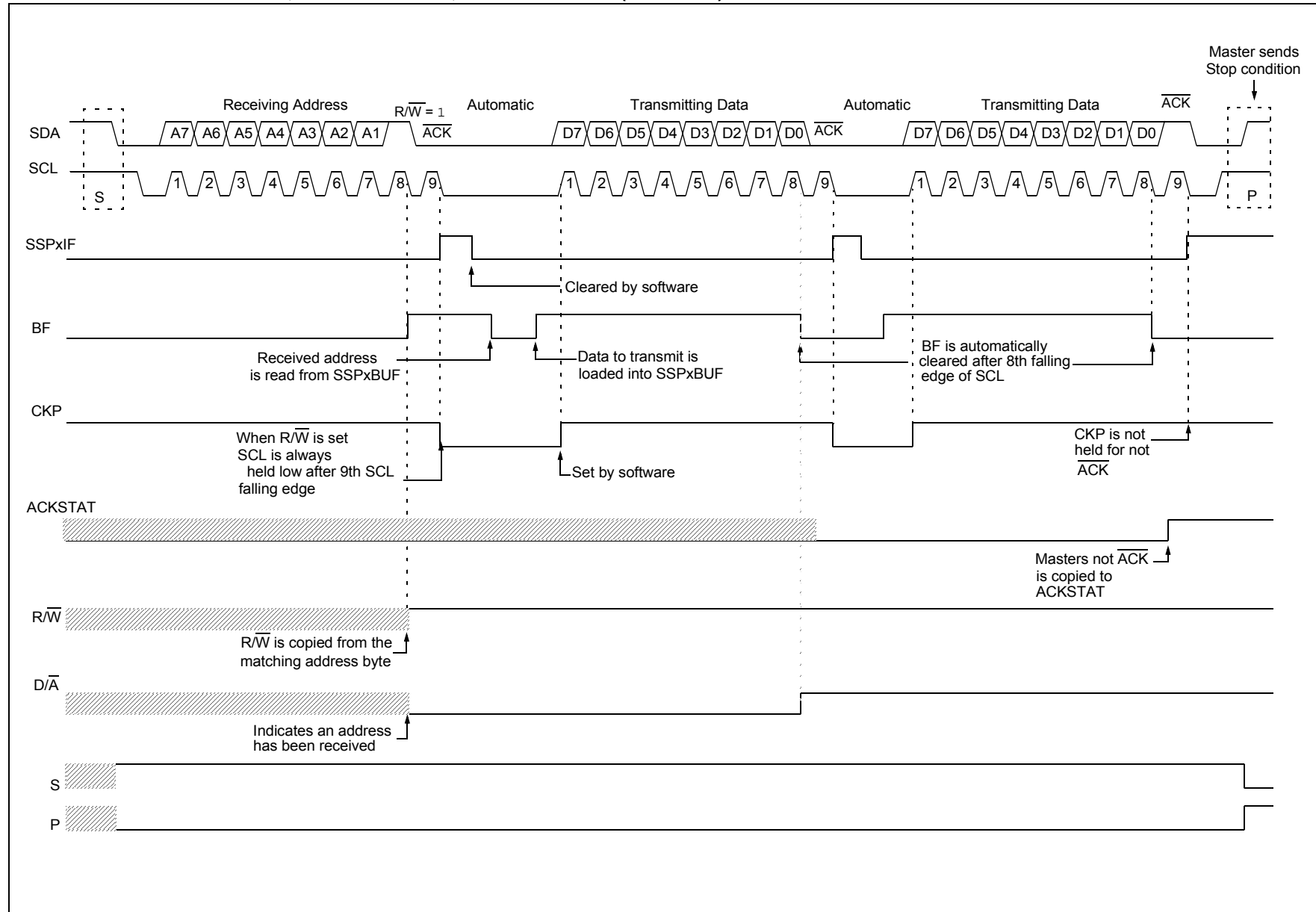
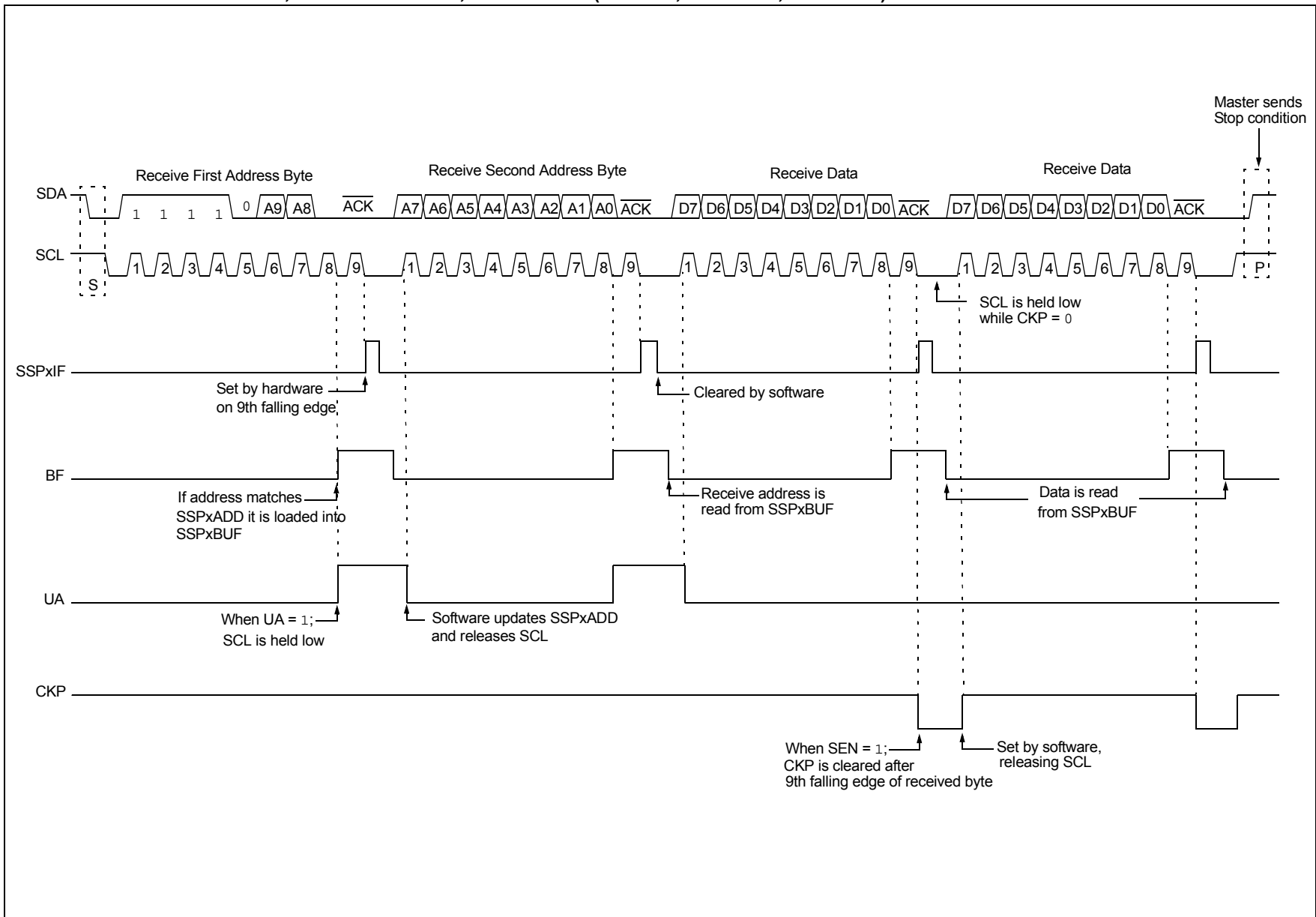
FIGURE 33-18: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 0)

FIGURE 33-20: I²C SLAVE, 10-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)

33.6.7 I²C MASTER MODE RECEPTION

Master mode reception (Figure 33-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSPxCON2 register.

Note: The MSSP module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPxSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPxSR are loaded into the SSPxBUF, the BF flag bit is set, the SSPxIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSP is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPxCON2 register.

33.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPxBUF from SSPxSR. It is cleared when the SSPxBUF register is read.

33.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when eight bits are received into the SSPxSR and the BF flag bit is already set from a previous reception.

33.6.7.3 WCOL Status Flag

If the user writes the SSPxBUF when a receive is already in progress (i.e., SSPxSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

33.6.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
2. SSPxIF is set by hardware on completion of the Start.
3. SSPxIF is cleared by software.
4. User writes SSPxBUF with the slave address to transmit and the R/W bit set.
5. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
6. The MSSP module shifts in the $\overline{\text{ACK}}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
7. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
8. User sets the RCEN bit of the SSPxCON2 register and the master clocks in a byte from the slave.
9. After the eighth falling edge of SCL, SSPxIF and BF are set.
10. Master clears SSPxIF and reads the received byte from SSPxBUF, clears BF.
11. Master sets $\overline{\text{ACK}}$ value sent to slave in ACKDT bit of the SSPxCON2 register and initiates the $\overline{\text{ACK}}$ by setting the ACKEN bit.
12. Master's $\overline{\text{ACK}}$ is clocked out to the slave and SSPxIF is set.
13. User clears SSPxIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{\text{ACK}}$ or Stop to end communication.

PIC16(L)F19155/56/75/76/85/86

REGISTER 33-5: SSPxMSK: SSPx MASK REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
SSPxMSK<7:0>							
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 7-1 **SSPxMSK<7:1>**: Mask bits
 1 = The received address bit n is compared to SSPxADD<n> to detect I²C address match
 0 = The received address bit n is not used to detect I²C address match
- bit 0 **SSPxMSK<0>**: Mask bit for I²C Slave mode, 10-bit Address
I²C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
 1 = The received address bit 0 is compared to SSPxADD<0> to detect I²C address match
 0 = The received address bit 0 is not used to detect I²C address match
I²C Slave mode, 7-bit address:
 MSK0 bit is ignored.

REGISTER 33-6: SSPxADD: MSSPx ADDRESS AND BAUD RATE REGISTER (I²C MODE)

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
SSPxADD<7:0>							
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	

Master mode:

- bit 7-0 **SSPxADD<7:0>**: Baud Rate Clock Divider bits

$$\text{SCL pin clock period} = ((\text{ADD}<7:0> + 1) * 4) / \text{Fosc}$$

10-Bit Slave mode – Most Significant Address Byte:

- bit 7-3 **Not used:** Unused for Most Significant Address Byte. Bit state of this register is a “don’t care”. Bit pattern sent by master is fixed by I²C specification and must be equal to ‘11110’. However, those bits are compared by hardware and are not affected by the value in this register.
- bit 2-1 **SSPxADD<2:1>**: Two Most Significant bits of 10-bit address
- bit 0 **Not used:** Unused in this mode. Bit state is a “don’t care”.

10-Bit Slave mode – Least Significant Address Byte:

- bit 7-0 **SSPxADD<7:0>**: Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

- bit 7-1 **SSPxADD<7:1>**: 7-bit address
- bit 0 **Not used:** Unused in this mode. Bit state is a “don’t care”.

PIC16(L)F19155/56/75/76/85/86

TABLE 38-1: REGISTER FILE SUMMARY FOR PIC16(L)F19155/56/75/76/85/86 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
1F33h	—	Unimplemented								
1F34h	—	Unimplemented								
1F35h	—	Unimplemented								
1F36h	—	Unimplemented								
1F37h	—	Unimplemented								
1F38h	ANSELA	ANSA7	ANSA6	—	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	223
1F39h	WPUA	WPUA7	WPUA6	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	224
1F3Ah	ODCONA	ODCA7	ODCA6	—	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	224
1F3Bh	SLRCONA	SLRA7	SLRA6	—	SLRA4	SLRA3	SLRA2	SLRA1	SLRA0	225
1F3Ch	INLVLA	INLVLA7	INLVLA6	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	225
1F3Dh	IOCAP	IOCAP7	IOCAP6	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	
1F3Eh	IOCAN	IOCAN7	IOCAN6	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	
1F3Fh	IOCAF	IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	
1F40h	—	Unimplemented								
1F41h	—	Unimplemented								
1F42h	—	Unimplemented								
1F43h	ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	230
1F44h	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	231
1F45h	ODCONB	ODCB7	ODCB6	ODCB5	ODCB4	ODCB3	ODCB2	ODCB1	ODCB0	231
1F46h	SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	SLRB3	SLRB2	SLRB1	SLRB0	232
1F47h	INVLB	INVLB7	INVLB6	INVLB5	INVLB4	INVLB3	INVLB2	INVLB1	INVLB0	232
1F48h	IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0	277
1F49h	IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0	277
1F4Ah	IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0	277
1F4Bh	—	Unimplemented								
1F4Ch	—	Unimplemented								
1F4Dh	—	Unimplemented								
1F4Eh	—	Unimplemented								
1F4Fh	WPUC	WPUC7	WPUC6	—	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	236
1F50h	ODCONC	ODCC7	ODCC6	—	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0	237
1F51h	SLRCONC	SLRC7	SLRC6	—	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0	237
1F52h	INLVLC	INLVLC7	INLVLC6	—	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	238
1F53h	IOCCP	IOCCP7	IOCCP6	—	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	278
1F54h	IOCCN	IOCCN7	IOCCN6	—	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	278
1F55h	IOCCF	IOCCF7	IOCCF6	—	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	278
1F56h	—	Unimplemented								
1F57h	—	Unimplemented								
1F58h	—	Unimplemented								
1F59h	ANSELD	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	242
1F5Ah	WPUD	WPUD7	WPUD6	WPUD5	WPUD4	WPUD3	WPUD2	WPUD1	WPUD0	243
1F5Bh	ODCOND	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	243
1F5Ch	SLRCOND	SLRD7	SLRD6	SLRD5	SLRD4	SLRD3	SLRD2	SLRD1	SLRD0	244
1F5Dh	INVLVD	INVLVD7	INVLVD6	INVLVD5	INVLVD4	INVLVD3	INVLVD2	INVLVD1	INVLVD0	244

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

Note 1: Unimplemented data memory locations, read as '0'.