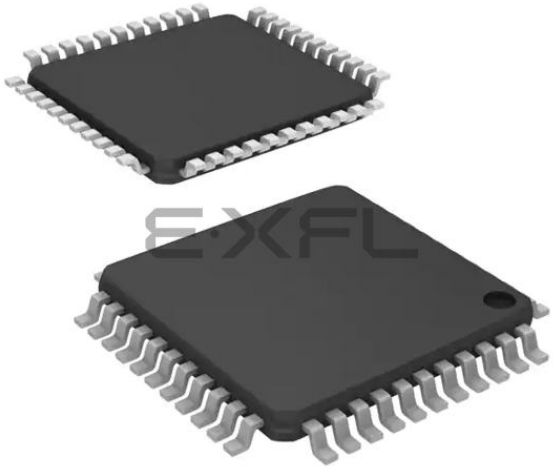


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K 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	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf19175-i-pt

TABLE 1-2: PIC16(L)F19155/56 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC0/T1CKI(1)/SMTWIN1(1)/IOCC0/SOSCO	RC0	TTL/ST	CMOS/OD	General purpose I/O.
	T1CKI(1)	—	—	Timer1 clock input.
	SMTWIN1(1)	—	—	SMT window input.
	IOCC0	TTL/ST	—	Interrupt-on-change input.
	SOSCO	#VALUE!	AN	32.768 kHz secondary oscillator crystal driver output.
RC1/T4IN(1)/SMTSIG1(1)/CCP2(1)/IOCC1/SOSCI	RC1	TTL/ST	CMOS/OD	General purpose I/O.
	T4IN(1)	—	—	Timer4 external input.
	SMTSIG1(1)	—	—	SMT signal input.
	CCP2(1)	—	—	CCP Capture Input.
	IOCC1	TTL/ST	—	Interrupt-on-change input.
	SOSCI	—	—	32.768 kHz secondary oscillator crystal driver input.
RC2/CCP1(1)/IOCC2/ANC2/SEG18/COM2	RC2	TTL/ST	CMOS/OD	General purpose I/O.
	CCP1(1)	—	—	CCP Capture Input.
	IOCC2	TTL/ST	—	Interrupt-on-change input.
	ANC2	AN	—	ADC Channel input.
	SEG18	—	AN	LCD Analog output.
	COM2	—	AN	LCD Driver Common Outputs.
RC3/T2IN/SCL(3,4)/SCK(1)/SEG19	RC3	TTL/ST	CMOS/OD	General purpose I/O.
	T2IN(1)	—	—	Timer2 external input.
	SCL(3,4)	I^2C	OD	MSSP I^2Cclock input/output.
	SCK(1)	TTL/ST	—	MSSP SPI clock input/output
	IOCC3	TTL/ST	—	Interrupt-on-change input.
	ANC3	AN	—	ADC Channel input.
	SEG19	—	AN	LCD Analog output.
RC4/SDA(3,4)/SDI(1)/IOCC4/ANC4/SEG20	RC4	TTL/ST	CMOS/OD	General purpose I/O.
	SDA(3,4)	TTL/ST	—	MSSP I^2C data input/output.
	SDI(1)	I^2C	OD	MSSP SPI serial data in.
	IOCC4	TTL/ST	—	Interrupt-on-change input.
	ANC4	AN	—	ADC Channel input.
	SEG20	—	AN	LCD Analog output.
RC6/CK1(3)/TX1(1)/IOCC6/ANC6/SEG22/COM5/VLCD2	RC6	TTL/ST	CMOS/OD	General purpose I/O.
	CK1(3)	—	—	EUSART synchronous clock out
	TX1(1)	—	—	EUSART asynchronous TX data out
	IOCC6	TTL/ST	—	Interrupt-on-change input.
	ANC6	AN	—	ADC Channel input.
	SEG22	—	AN	LCD Analog output.
	COM5	—	AN	LCD Driver Common Outputs.
	VLCD2	AN	—	LCD analog input

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; OD = Open-Drain; TTL = TTL compatible input'; ST = Schmitt Trigger input with CMOS levels; I^2C = Schmitt Trigger input with I^2C; HV = High Voltage; XTAL = Crystal levels

Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 14-2 for details on which PORT pins may be used for this signal.

2: All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin options as described in Table 14-3.

3: This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.

4: These pins are configured for I^2C logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the I^2C specific or SMBus input buffer thresholds.

4.2 Memory Access Partition (MAP)

User Flash is partitioned into:
- Application Block
- Boot Block, and
- Storage Area Flash (SAF) Block

The user can allocate the memory usage by setting the $\overline{\text{BBEN}}$ bit, selecting the size of the partition defined by BBSIZE<2:0> bits and enabling the Storage Area Flash by the $\overline{\text{SAFEN}}$ bit of the Configuration Word (see Register 5-4). Refer to Table 4-2 for the different user Flash memory partitions.

4.2.1 APPLICATION BLOCK

Default settings of the Configuration bits ($\overline{\text{BBEN}}$ = 1 and $\overline{\text{SAFEN}}$ = 1) assign all memory in the user Flash area to the Application Block.

4.2.2 BOOT BLOCK

If $\overline{\text{BBEN}}$ = 1, the Boot Block is enabled and a specific address range is alloted as the Boot Block based on the value of the BBSIZE bits of Configuration Word (Register 5-4) and the sizes provided in Table 5-1.

4.2.3 STORAGE AREA FLASH

Storage Area Flash (SAF) is enabled by clearing the $\overline{\text{SAFEN}}$ bit of the Configuration Word in Register 5-4. If enabled, the SAF block is placed at the end of memory and spans 128 words. If the Storage Area Flash (SAF) is enabled, the SAF area is not available for program execution.

4.2.4 MEMORY WRITE PROTECTION

All the memory blocks have corresponding write protection fuses WRTAPP and WRTB bits in the Configuration Word (Register 5-4). If write-protected locations are written from NVMCON registers, memory is not changed and the WRERR bit defined in Register 12-5 is set as explained in **Section 13.4.9 "WRERR Bit"**.

4.2.5 MEMORY VIOLATION

A Memory Execution Violation Reset occurs while executing an instruction that has been fetched from outside a valid execution area, clearing the $\overline{\text{MEMV}}$ bit. Refer to **Section 8.13 "Memory Execution Violation"** for the available valid program execution areas and the PCON1 register definition (Register 8-3) for $\overline{\text{MEMV}}$ bit conditions.

TABLE 4-2: MEMORY ACCESS PARTITION

<table>
<tr><th rowspan="2">REG</th><th rowspan="2">Address</th><th colspan="4">Partition</th></tr>
<tr><th>$\overline{\text{BBEN}}$ = 1
SAFEN = 1</th><th>$\overline{\text{BBEN}}$ = 1
SAFEN = 0</th><th>$\overline{\text{BBEN}}$ = 0
SAFEN = 1</th><th>$\overline{\text{BBEN}}$ = 0
SAFEN = 0</th></tr>
<tr><td rowspan="3">PFM</td><td>00 0000h
• • •
Last Boot Block Memory Address</td><td rowspan="3">APPLICATION BLOCK
(Note 4)</td><td rowspan="2">APPLICATION BLOCK
(Note 4)</td><td>BOOT BLOCK
(Note 4)</td><td>BOOT BLOCK
(Note 4)</td></tr>
<tr><td>Last Boot Block Memory Address + 1(1)
• • •
Last Program Memory Address - 80h</td><td rowspan="2">APPLICATION BLOCK
(Note 4)</td><td>APPLICATION BLOCK
(Note 4)</td></tr>
<tr><td>Last Program Memory Address - 7Fh(2)
• • •
Last Program Memory Address</td><td>SAF
(Note 4)</td><td>SAF
(Note 4)</td></tr>
<tr><td>CONF IG</td><td>Config Memory Address(3)</td><td colspan="4">CONFIG
(Note 4)</td></tr>
</table>

Note 1: Last Boot Block Memory Address is based on BBSIZE[2:0] given in Table 5-1.
2: Last Program Memory Address is the Flash size given in Table 4-1.
3: Config Memory Address are the address locations of the Configuration Words given in Table 13-2.
4: Each memory block has a corresponding write protection fuse defined by the $\overline{\text{WRTAPP}}$, $\overline{\text{WRTB}}$ and $\overline{\text{WRTSAF}}$ bits in the Configuration Word (Register 5-4).

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 61 (Continued)											
1ECFh	—	Unimplemented								---- ----	---- ----
1ED0h	—	Unimplemented								---- ----	---- ----
1ED1h	—	Unimplemented								---- ----	---- ----
1ED2h	—	Unimplemented								---- ----	---- ----
1ED3h	—	Unimplemented								---- ----	---- ----
1ED4h	—	Unimplemented								---- ----	---- ----
1ED5h	—	Unimplemented								---- ----	---- ----
1ED6h	—	Unimplemented								---- ----	---- ----
1ED7h	—	Unimplemented								---- ----	---- ----
1ED8h	—	Unimplemented								---- ----	---- ----
1ED9h	—	Unimplemented								---- ----	---- ----
1EDAh	—	Unimplemented								---- ----	---- ----
1EDBh	—	Unimplemented								---- ----	---- ----
1EDCh	—	Unimplemented								---- ----	---- ----
1EDDh	—	Unimplemented								---- ----	---- ----
1EDEh	—	Unimplemented								---- ----	---- ----
1EDFh	—	Unimplemented								---- ----	---- ----
1EE0h	—	Unimplemented								---- ----	---- ----
1EE1h	—	Unimplemented								---- ----	---- ----
1EE2h	—	Unimplemented								---- ----	---- ----
1EE3h	—	Unimplemented								---- ----	---- ----
1EE4h	—	Unimplemented								---- ----	---- ----

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'.

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 63 (Continued)											
1FD0h	—	Unimplemented								---- ----	---- ----
1FD1h	—	Unimplemented								---- ----	---- ----
1FD2h	—	Unimplemented								---- ----	---- ----
1FD3h	—	Unimplemented								---- ----	---- ----
1FD4h	—	Unimplemented								---- ----	---- ----
1FD5h	—	Unimplemented								---- ----	---- ----
1FD6h	—	Unimplemented								---- ----	---- ----
1FD7h	—	Unimplemented								---- ----	---- ----
1FD8h	—	Unimplemented								---- ----	---- ----
1FD9h	—	Unimplemented								---- ----	---- ----
1FDAh	—	Unimplemented								---- ----	---- ----
1FDBh	—	Unimplemented								---- ----	---- ----
1FDCh	—	Unimplemented								---- ----	---- ----
1FDDh	—	Unimplemented								---- ----	---- ----
1FDEh	—	Unimplemented								---- ----	---- ----
1FDFh	—	Unimplemented								---- ----	---- ----
1FE0h	—	Unimplemented								---- ----	---- ----
1FE1h	—	Unimplemented								---- ----	---- ----
1FE2h	—	Unimplemented								---- ----	---- ----
1FE3h	—	Unimplemented								---- ----	---- ----

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'.

5.3 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection and data memory are controlled independently. Internal access to the program memory is unaffected by any code protection setting.

5.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the $\overline{CP}$ bit in Configuration Words. When $\overline{CP}$ = 0, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Self-writing the program memory is dependent upon the write protection setting. See **Section 5.4 "Write Protection"** for more information.

5.3.2 DATA MEMORY PROTECTION

The entire data EEPROM is protected from external reads and writes by the $\overline{WRTD}$ bit in the Configuration Words. When $\overline{WRTD}$ = 0, external reads and writes of EEPROM memory are inhibited and a read will return all '0's. The CPU can continue to read and write EEPROM memory, regardless of the protection bit settings.

5.4 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The $\overline{WRTAPP}$, $\overline{WRTSAF}$, $\overline{WRTB}$, $\overline{WRTC}$ bits in Configuration Words (Register 5-4) define whether the corresponding region of the program memory block is protected or not.

5.5 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See **Section 13.4.7 "NVMREG Access to Device Information Area, Device Configuration Area, User ID, Device ID and Configuration Words"** for more information on accessing these memory locations. For more information on checksum calculation, see the *"PIC16(L)F191XX Memory Programming Specification"* (DS40001880).

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 PIEx[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 PIEx 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 "Automatic 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.

TABLE 12-3: SUMMARY OF REGISTERS ASSOCIATED WITH WINDOWED WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCCON1	—	NOSC<2:0>			NDIV<3:0>				152
OSCCON2	—	COSC<2:0>			CDIV<3:0>				152
OSCCON3	CSWHOLD	SOSCPWR	—	ORDY	NOSCR	—	—	—	154
PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	140
STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	50
WDTCON0	—	—	WDTPS<4:0>					SWDTEN	196
WDTCON1	—	WDTCS<2:0>			—	WINDOW<2:0>			197
WDTPSL	PSCNT<7:0>								198
WDTPSH	PSCNT<15:8>								198
WDTTMR	—	WDTTMR<3:0>				STATE	PSCNT<17:16>		198

Legend: – = unimplemented locations read as '0'. Shaded cells are not used by Watchdog Timer.

TABLE 12-4: SUMMARY OF CONFIGURATION WORD WITH WATCHDOG TIMER

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG1	13:8	—	—	FCMEN	—	CSWEN	LCDPEN	$\overline{\text{VBATEN}}$	$\overline{\text{CLKOUTEN}}$	120
	7:0	—	RSTOSC<2:0>			—	FEXTOSC<2:0>			

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by Watchdog Timer.

19.0 ANALOG-TO-DIGITAL CONVERTER WITH COMPUTATION (ADC^2) MODULE

The Analog-to-Digital Converter with Computation (ADC^2) allows conversion of an analog input signal to a 12-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 12-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair).

Additionally, the following features are provided within the ADC module:

- 13-bit Acquisition Timer
- Hardware Capacitive Voltage Divider (CVD) Support:
  - 13-bit Precharge Timer
  - Adjustable sample and hold capacitor array
  - Guard ring digital output drive
- Automatic Repeat and Sequencing:
  - Automated double sample conversion for CVD
  - Two sets of result registers (Result and Previous result)
  - Auto-conversion trigger
  - Internal retrigger
- Computation Features:
  - Averaging and Low-Pass Filter functions
  - Reference Comparison
  - 2-level Threshold Comparison
  - Selectable Interrupts

Figure 19-1 shows the block diagram of the ADC.

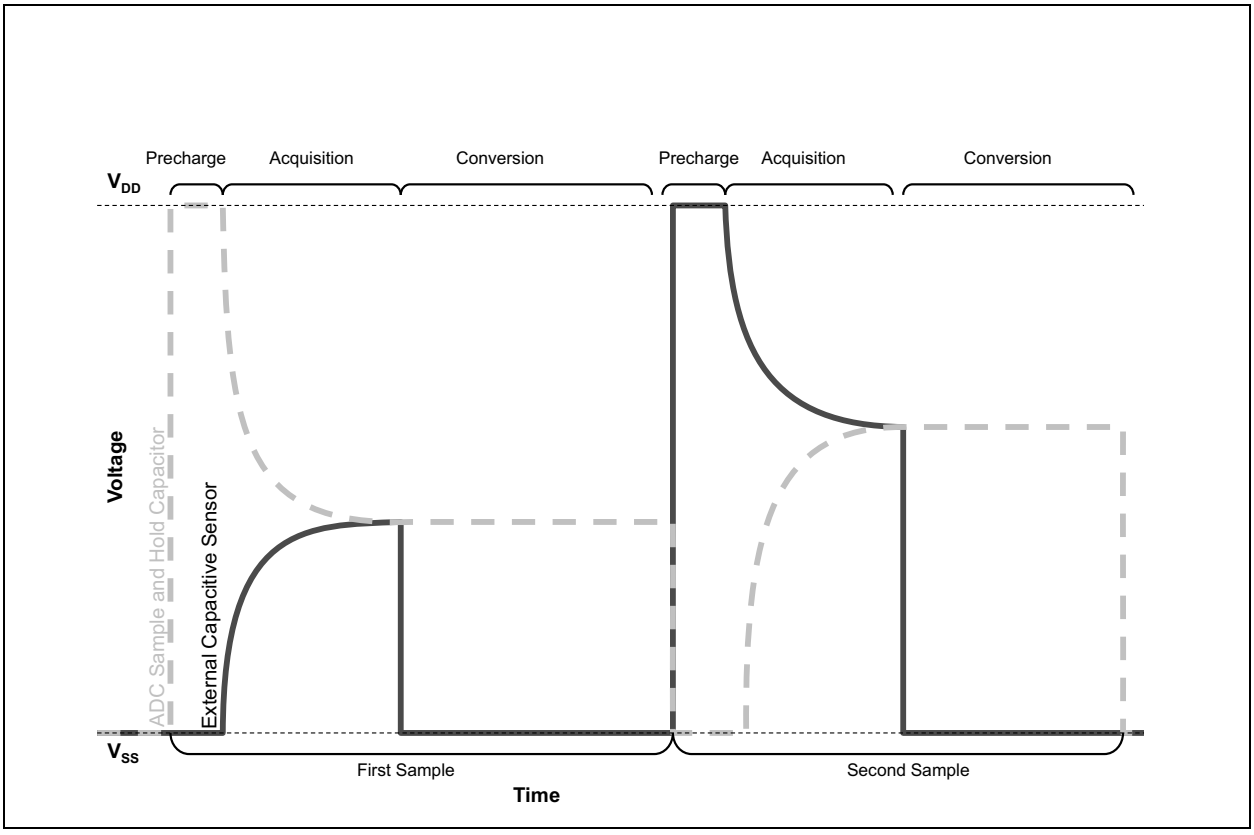
The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion and upon threshold comparison. These interrupts can be used to wake-up the device from Sleep.

19.4.1 CVD OPERATION

A CVD operation begins with the ADC's internal sample and hold capacitor (C_{HOLD}) being disconnected from the path which connects it to the external capacitive sensor node. While disconnected, C_{HOLD} is precharged to V_{DD} or V_{SS}, while the path to the sensor node is precharged to the level opposite that of C_{HOLD}. When the precharge phase is complete, the V_{DD}/V_{SS} precharge paths for the two nodes are shut off and C_{HOLD} and the path to the external sensor node are re-connected, at which time the acquisition phase of the CVD operation begins. During acquisition, a capacitive voltage divider is formed between the precharged C_{HOLD} and sensor nodes, which results in a final voltage level setting on C_{HOLD}, which is determined by the capacitances and precharge levels of the two nodes. After acquisition, the ADC converts the voltage level on C_{HOLD}. This process is then repeated with inverted precharge levels for both the C_{HOLD} and external sensor nodes. Figure 19-7 shows the waveform for two inverted CVD measurements, which is known as differential CVD measurement.

FIGURE 19-7: DIFFERENTIAL CVD MEASUREMENT WAVEFORM



19.6 Register Definitions: ADC Control

REGISTER 19-1: ADCON0: ADC CONTROL REGISTER 0

R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0	R/W-0/0	U-0	R/W/HC-0
ON	CONT	—	CS	—	FM	—	GO
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 **ON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled

bit 6 **CONT:** ADC Continuous Operation Enable bit(2)
1 = GO is retriggered upon completion of each conversion trigger until ADTIF is set (if ADSOI is set) or until GO is cleared (regardless of the value of ADSOI)
0 = ADC is cleared upon completion of each conversion trigger

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

bit 4 **CS:** ADC Clock Selection bit
1 = Clock supplied from FRC dedicated oscillator
0 = Clock supplied by FOSC, divided according to ADCLK register

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

bit 2 **FM:** ADC results Format/alignment Selection
1 = ADRES and PREV data are right-justified
0 = ADRES and PREV data are left-justified, zero-filled

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

bit 0 **GO:** ADC Conversion Status bit(1)
1 = ADC conversion cycle in progress. Setting this bit starts an ADC conversion cycle. The bit is cleared by hardware as determined by the CONT bit
0 = ADC conversion completed/not in progress

Note 1: This bit requires ON bit to be set.

2: If cleared by software while a conversion is in progress, the results of the conversion up to this point will be transfered to ADRES and the state machine will be reset, but the ADIF interrupt flag bit will not be set; filter and threshold operations will not be performed.

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 retriggered
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.

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’

REGISTER 19-36: ADCP: ADC CHARGE PUMP CONTROL REGISTER

R/W-0/0	U-0	U-0	U-0	U-0	U-0	U-0	R-0/0
CPON	—	—	—	—	—	—	CPRDY
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 **CPON**: Charge Pump On Control bit
1 = Charge Pump On when requested by the ADC
0 = Charge Pump Off

bit 6-1 **Unimplemented:** Read as ‘0’

bit 0 CPRDY: Charge Pump Ready Status bit
1 = Charge Pump is ready
0 = Charge Pump is not ready (or never started)

REGISTER 24-2: RTCCAL: RTC CALIBRATION REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0
bit 7							bit 0

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

bit 7-0 **CAL<7:0>:** RTC Drift Calibration bits

01111111 = Maximum positive adjustment; adds 508 RTC clock pulses every one minute
.
.
.
00000001 = Minimum positive adjustment; adds four RTC clock pulses every one minute
00000000 = No adjustment
11111111 = Minimum negative adjustment; subtracts four RTC clock pulses every one minute
.
.
.
10000000 = Maximum negative adjustment; subtracts 512 RTC clock pulses every one minute

REGISTER 24-3: YEAR(1): YEAR VALUE REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YEARH<3:0>				YEARL<3:0>			
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 7-4 **YEARH<3:0>:** Binary Coded Decimal value of years '10' digit; contains a value from 0 to 9

bit 3-0 **YEARL<3:0>:** Binary Coded Decimal value of years '1' digit; contains a value from 0 to 9

Note 1: Writes to the YEAR register is only allowed when RTCWREN = 1.

REGISTER 24-4: MONTH(1): MONTH VALUE REGISTER

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MONTHH	MONTHL<3:0>			
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 7-5 **Unimplemented:** Read as '0'

bit 4 **MONTHH<0>:** Binary Coded Decimal value of months '10' digit; valid values from 0 to 1

bit 3-0 **MONTHL<3:0>:** Binary Coded Decimal value of months '1' digit; valid values from 0 to 9

Note 1: Writes to the MONTH registers are only allowed when RTCWREN = 1.

27.0 TIMER2/4 MODULE WITH HARDWARE LIMIT TIMER (HLT)

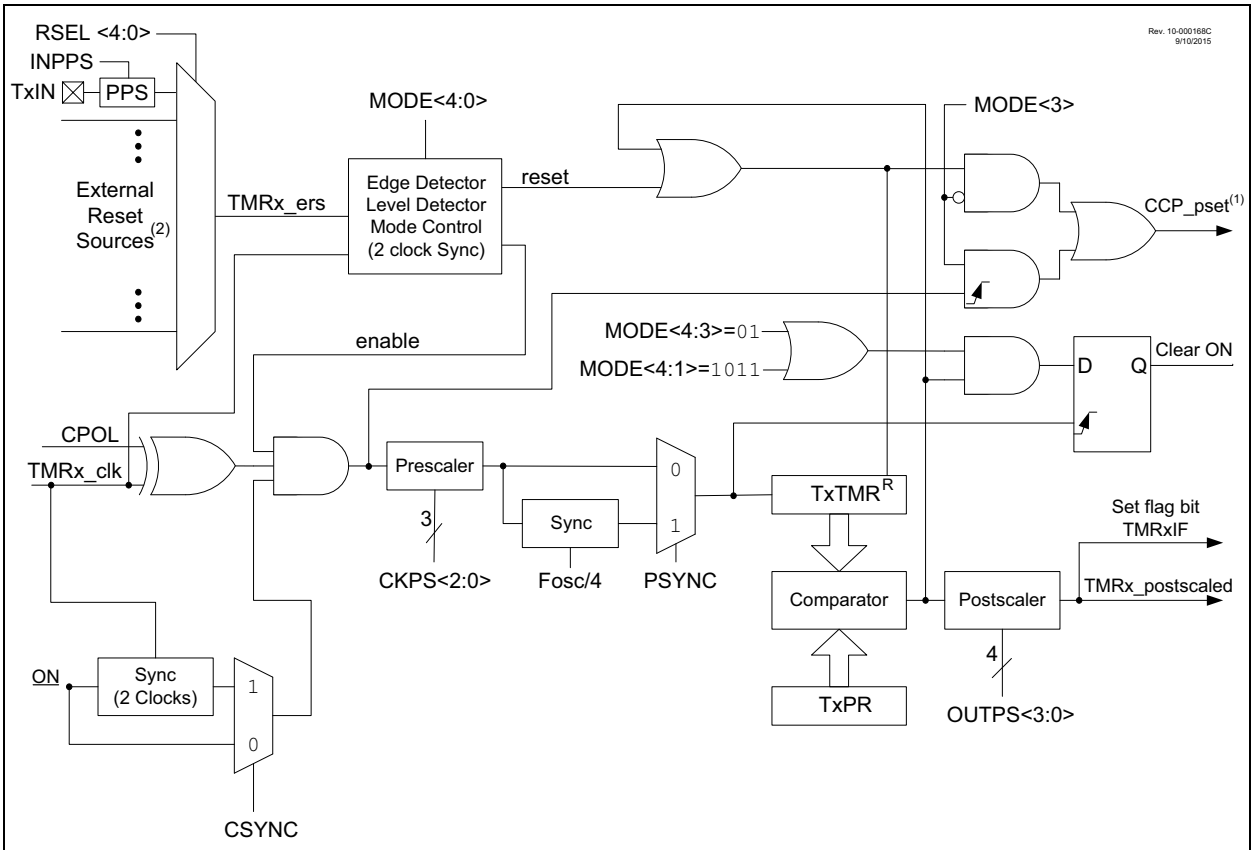
The Timer2/4 modules are 8-bit timers that can operate as free-running period counters or in conjunction with external signals that control start, run, freeze, and reset operation in One-Shot and Monostable modes of operation. Sophisticated waveform control like pulse density modulation are possible by combining the operation of these timers with other internal peripherals such as the comparators and CCP modules. Features of the timer include:

- 8-bit timer register
- 8-bit period register
- Selectable external hardware timer Resets
- Programmable prescaler (1:1 to 1:128)
- Programmable postscaler (1:1 to 1:16)
- Selectable synchronous/asynchronous operation
- Alternate clock sources
- Interrupt-on-period
- Three modes of operation:
  - Free Running Period
  - One-shot
  - Monostable

See Figure 27-1 for a block diagram of Timer2/4. See Figure 27-2 for the clock source block diagram.

Note: Two identical Timer2 modules are implemented on this device. The timers are named Timer2 and Timer4. All references to Timer2 apply as well to Timer4. All references to T2PR apply as well to T4PR.

FIGURE 27-1: TIMER2/4 BLOCK DIAGRAM



Note 1: Signal to the CCP to trigger the PWM pulse.
2: See Register 27-4 for external Reset sources.

28.6.3 GO STATUS

Timer run status is determined by the TS bit of the SMTxSTAT register, and will be delayed in time by synchronizer delays in non-Counter modes.

28.7 Modes of Operation

The modes of operation are summarized in Table 28-2. The following sections provide detailed descriptions, examples of how the modes can be used. Note that all waveforms assume WPOL/SPOL/CPOL = 0. When WPOL/SPOL/CPOL = 1, all SMTSIGx, SMTWINx and SMT clock signals will have a polarity opposite to that indicated. For all modes, the REPEAT bit controls whether the acquisition is repeated or single. When REPEAT = 0 (Single Acquisition mode), the timer will stop incrementing and the SMTxGO bit will be reset upon the completion of an acquisition. Otherwise, the timer will continue and allow for continued acquisitions to overwrite the previous ones until the timer is stopped in software.

28.7.1 TIMER MODE

Timer mode is the simplest mode of operation where the SMTxTMR is used as a 16/24-bit timer. No data acquisition takes place in this mode. The timer increments as long as the SMTxGO bit has been set by software. No SMT window or SMT signal events affect the SMTxGO bit. Everything is synchronized to the SMT clock source. When the timer experiences a period match (SMTxTMR = SMTxPR), SMTxTMR is reset and the period match interrupt trips. See Figure 28-3.

TABLE 28-2: MODES OF OPERATION

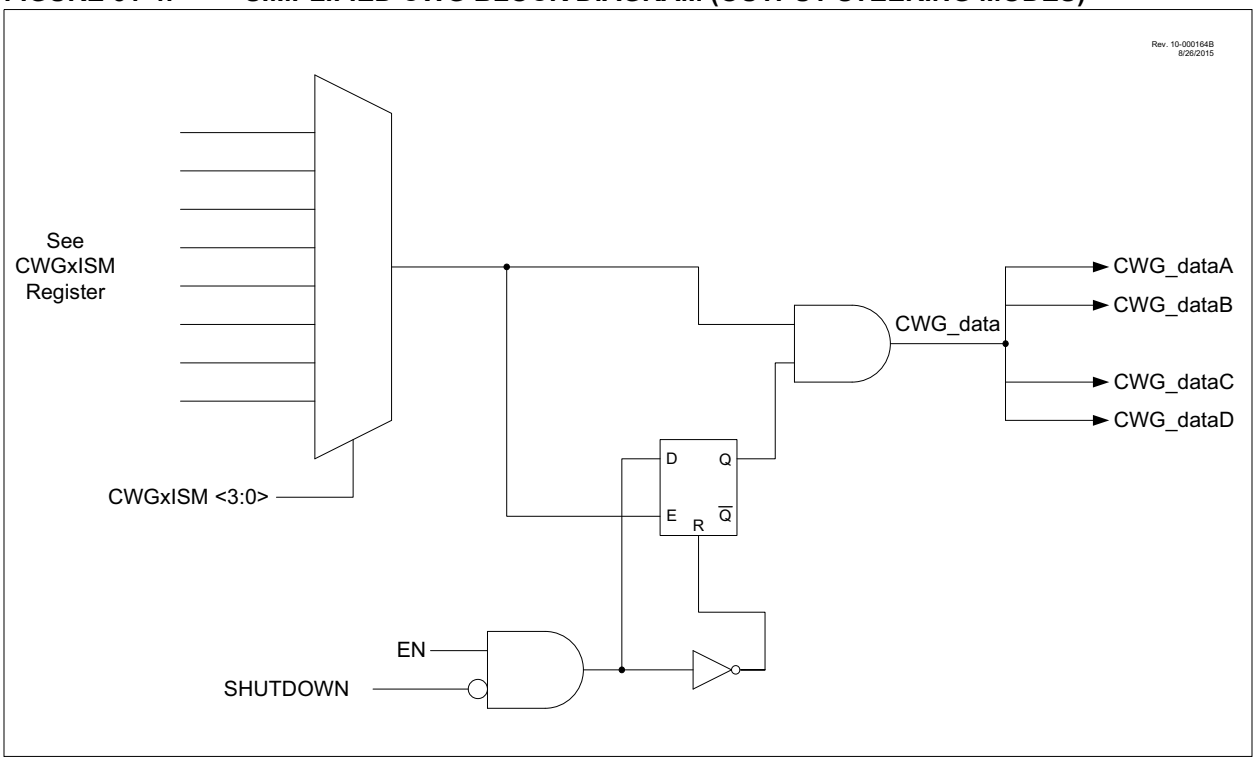
MODE	Mode of Operation	Synchronous Operation	Reference
0000	Timer	Yes	**Section 28.7.1 “Timer Mode”**
0001	Gated Timer	Yes	**Section 28.7.2 “Gated Timer Mode”**
0010	Period and Duty Cycle Acquisition	Yes	**Section 28.7.3 “Period and Duty-Cycle Mode”**
0011	High and Low Time Measurement	Yes	**Section 28.7.4 “High and Low-Measure Mode”**
0100	Windowed Measurement	Yes	**Section 28.7.5 “Windowed Measure Mode”**
0101	Gated Windowed Measurement	Yes	**Section 28.7.6 “Gated Window Measure Mode”**
0110	Time of Flight	Yes	**Section 28.7.7 “Time of Flight Measure Mode”**
0111	Capture	Yes	**Section 28.7.8 “Capture Mode”**
1000	Counter	No	**Section 28.7.9 “Counter Mode”**
1001	Gated Counter	No	**Section 28.7.10 “Gated Counter Mode”**
1010	Windowed Counter	No	**Section 28.7.11 “Windowed Counter Mode”**
1011-1111	Reserved	—	—

31.1.4 STEERING MODES

In Steering modes, the data input can be steered to any or all of the four CWG output pins. In Synchronous Steering mode, changes to steering selection registers take effect on the next rising input.

In Non-Synchronous mode, steering takes effect on the next instruction cycle. Additional details are provided in **Section 31.9 "CWG Steering Mode"**.

FIGURE 31-4: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)



31.2 Clock Source

The CWG module allows the following clock sources to be selected:

- Fosc (system clock)
- HFINTOSC (16 MHz only)

The clock sources are selected using the CS bit of the CWG1CLKCON register.

FIGURE 33-28: I^2C MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)

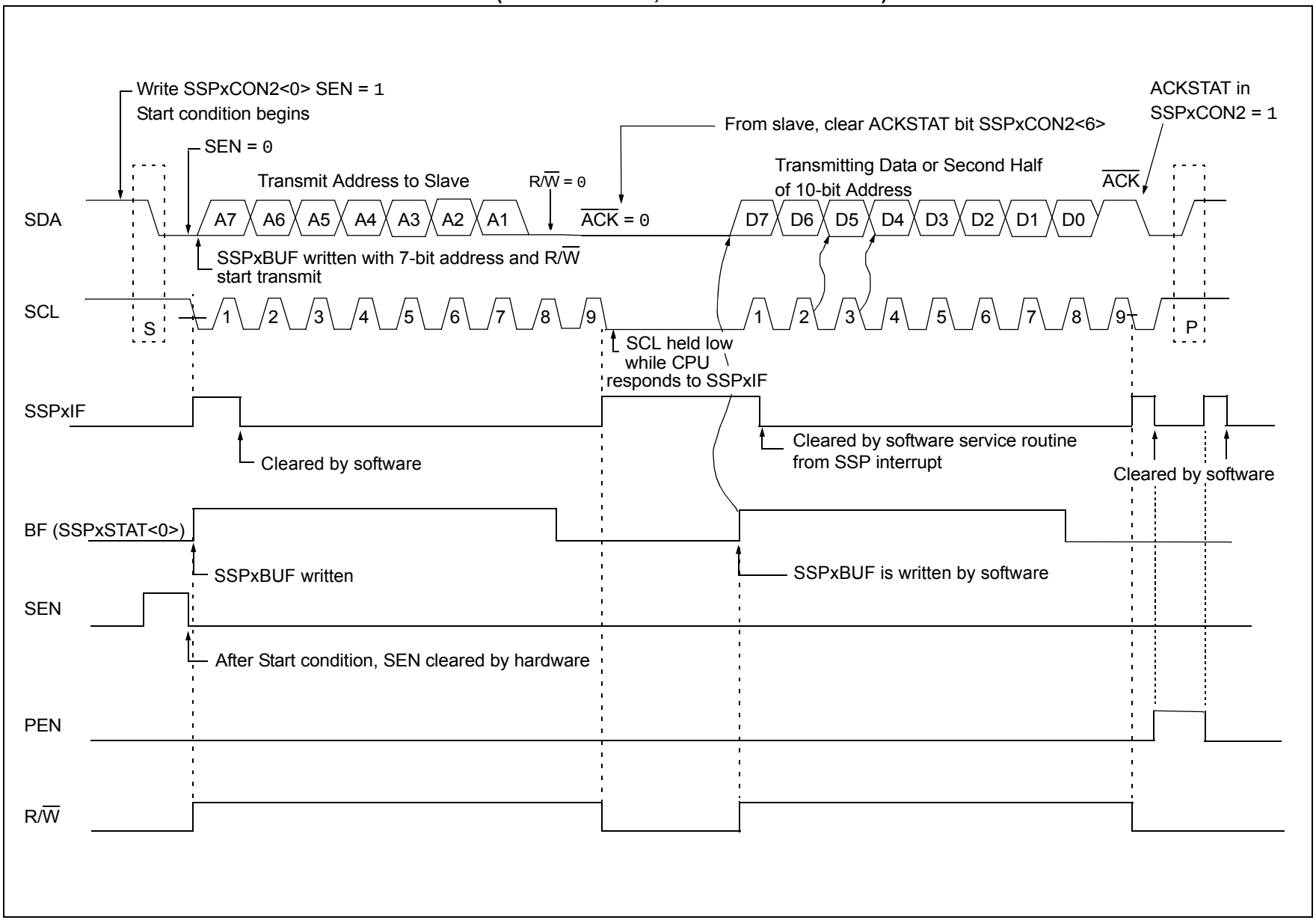


FIGURE 39-11: ADC CONVERSION TIMING (ADC CLOCK FROM ADCRC)

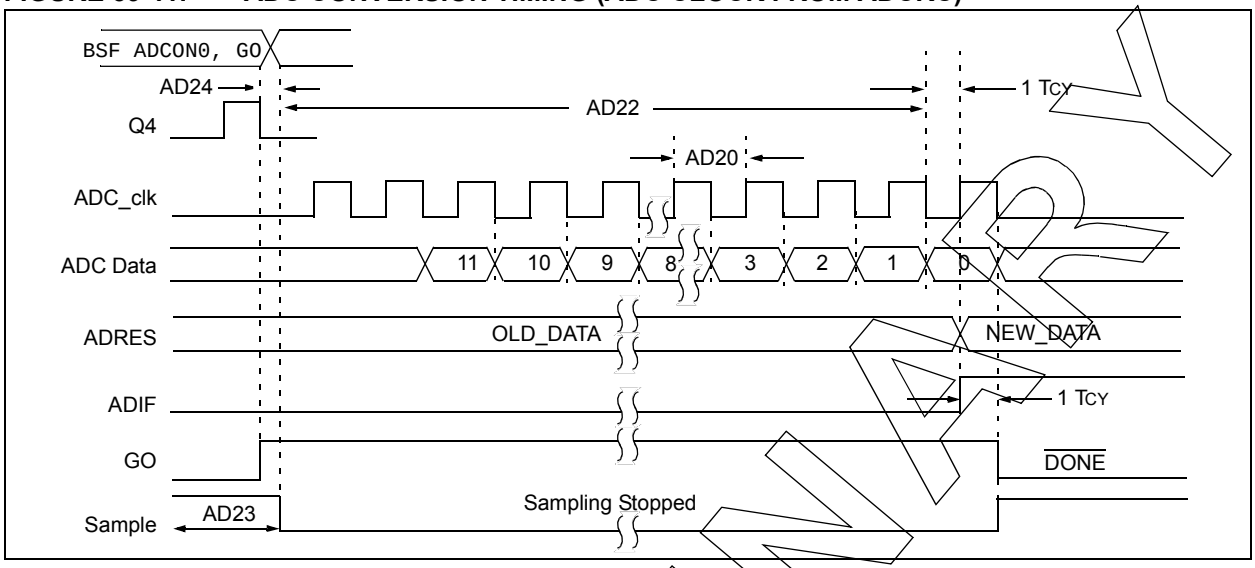


TABLE 39-14: COMPARATOR SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) VDD = 3.0V, TA = 25°C							
Param. No.	**Sym.**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
CM01	VIOFF	Input Offset Voltage	—	±30	—	mV	VICM = VDD/2
CM02	VICM	Input Common Mode Range	GND	—	VDD	V	
CM03	CMRR	Common Mode Input Rejection Ratio	—	50	—	dB	
CM04	VHYST	Comparator Hysteresis	—	25	—	mV	
CM05	TRESP(1)	Response Time, Rising Edge	—	300	600	ns	
		Response Time, Falling Edge	—	220	500	ns	
CMOS6	TMCV2VO(2)	Mode Change to Valid Output	—	—	10	µs	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at VDD/2, while the other input transitions from VSS to VDD.

2: A mode change includes changing any of the control register values, including module enable.

TABLE 39-15: LOW-POWERED CLOCKED COMPARATOR SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) VDD = 3.0V, TA = 25°C							
Param. No.	**Sym.**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
CM07	VIOFF	Input Offset Voltage	—	±30	—	mV	VICM = VDD/2
CM08	VICM	Input Common Mode Range	GND	—	VDD	V	
CM09	CMRR	Common Mode Input Rejection Ratio	—	50	—	dB	
CM010	VHYST	Comparator Hysteresis	—	25	—	mV	
CM011	TRESP(1)(3)	Response Time, Rising Edge	—	300	—	ns	
		Response Time, Falling Edge	—	220	—	ns	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at VDD/2, while the other input transitions from VSS to VDD.

2: A mode change includes changing any of the control register values, including module enable.

3: Comparator output state change occurs on the rising edge of LFINTOSC.

FIGURE 39-19: SPI SLAVE MODE TIMING (CKE = 0)

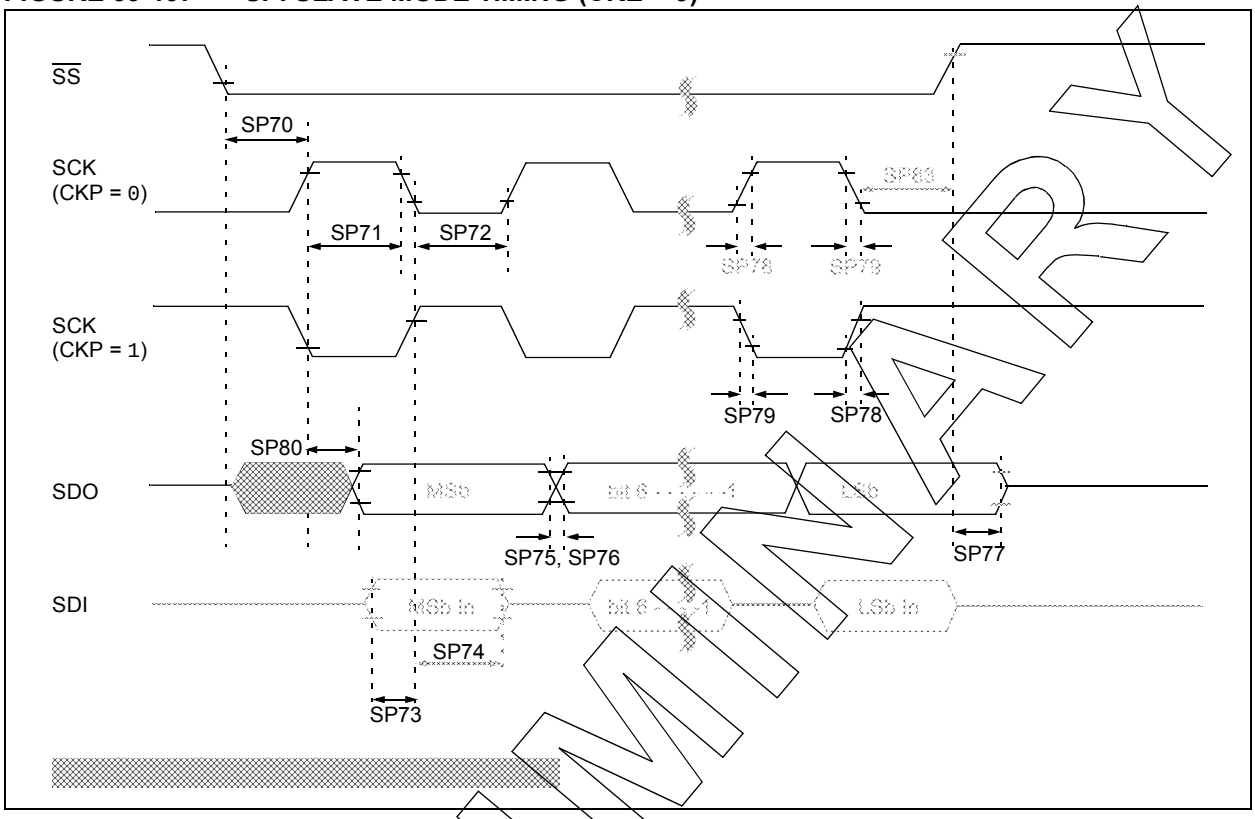


FIGURE 39-20: SPI SLAVE MODE TIMING (CKE = 1)