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, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 12x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1619-e-ss

TABLE 4: 20-PIN ALLOCATION TABLE (PIC16(L)F1619)

I/O	20-Pin PDIP, SOIC, SSOP	20-Pin UQFN	A/D	Reference	Comparator	Timers	CCP	CWG	ZCD	CLC	EUSART	SMT	Angular Timer	MSSP	PWM	High Current I/O	Interrupt	Pull-up	Basic
RA0	19	16	AN0	DAC1OUT	C1IN+	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	ICSPDAT
RA1	18	15	AN1	VREF+	C1IN0- C2IN0-	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	ICSPCLK
RA2	17	14	AN2	—	—	T0CKI(1)	—	CWG1IN(1)	ZCD1IN	—	—	—	—	—	—	—	INT IOC	Y	—
RA3	4	1	—	—	—	T6IN(1)	—	—	—	—	—	SMTWIN2(1)	—	—	—	—	IOC	Y	$\overline{\text{MCLR}}$ VPP
RA4	3	20	AN3	—	—	T1G(1)	—	—	—	—	—	SMTSIG1(1)	—	—	—	—	IOC	Y	CLKOUT
RA5	2	19	—	—	—	T1CKI(1) T2IN(1)	—	—	—	CLCIN3(1)	—	SMTWIN1(1)	—	—	—	—	IOC	Y	CLKIN
RB4	13	10	AN10	—	—	—	—	—	—	—	—	—	—	SDI(1)	—	—	IOC	Y	—
RB5	12	9	AN11	—	—	—	—	—	—	—	RX(1,3)	—	—	—	—	—	IOC	Y	—
RB6	11	8	—	—	—	—	—	—	—	—	—	—	—	SCK(1,3)	—	—	IOC	Y	—
RB7	10	7	—	—	—	—	—	—	—	—	CK(1)	—	—	—	—	—	IOC	Y	—
RC0	16	13	AN4	—	C2IN+	T5CKI(1)	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
RC1	15	12	AN5	—	C1IN1- C2IN1-	T4IN(1)	—	—	—	CLCIN2(2)	—	SMTSIG2(1)	—	—	—	—	IOC	Y	—
RC2	14	11	AN6	—	C1IN2- C2IN2-	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
RC3	7	4	AN7	—	C1IN3- C2IN3-	T5G(1)	CCP2(1)	—	—	CLCIN0(1)	—	—	ATCC(1)	—	—	—	IOC	Y	—
RC4	6	3	—	—	—	T3G(1)	—	—	—	CLCIN1(1)	—	—	—	—	—	HIC4	IOC	Y	—
RC5	5	2	—	—	—	T3CKI(1)	CCP1(1)	—	—	—	—	—	ATIN(1)	—	—	HIC5	IOC	Y	—
RC6	8	5	AN8	—	—	—	—	—	—	—	—	—	—	$\overline{\text{SS}}$(1)	—	—	IOC	Y	—
RC7	9	6	AN9	—	—	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	—
VDD	1	18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VSS	20	17	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Note 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection registers.
2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers.
3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

6.12 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON registers are updated to indicate the cause of the Reset. Table 6-3 and Table 6-4 show the Reset conditions of these registers.

TABLE 6-3: RESET STATUS BITS AND THEIR SIGNIFICANCE

STKOVF	STKUNF	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Condition
0	0	1	1	1	0	x	1	1	Power-on Reset
0	0	1	1	1	0	x	0	x	Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$
0	0	1	1	1	0	x	x	0	Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$
0	0	u	1	1	u	0	1	1	Brown-out Reset
u	u	0	u	u	u	u	0	u	WDT Reset
u	u	u	u	u	u	u	0	0	WDT Wake-up from Sleep
u	u	u	u	u	u	u	1	0	Interrupt Wake-up from Sleep
u	u	u	0	u	u	u	u	u	$\overline{\text{MCLR}}$ Reset during normal operation
u	u	u	0	u	u	u	1	0	$\overline{\text{MCLR}}$ Reset during Sleep
u	u	u	u	0	u	u	u	u	RESET Instruction Executed
1	u	u	u	u	u	u	u	u	Stack Overflow Reset (STVREN = 1)
u	1	u	u	u	u	u	u	u	Stack Underflow Reset (STVREN = 1)

TABLE 6-4: RESET CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	0000h	---1 1000	0011 110x
$\overline{\text{MCLR}}$ Reset during normal operation	0000h	---u uuuu	uuuu 0uuu
$\overline{\text{MCLR}}$ Reset during Sleep	0000h	---1 0uuu	uuuu 0uuu
WDT Reset	0000h	---0 uuuu	uuu0 uuuu
WDT Wake-up from Sleep	PC + 1	---0 0uuu	uuuu uuuu
Brown-out Reset	0000h	---1 1uuu	00uu 11u0
Interrupt Wake-up from Sleep	PC + 1(1)	---1 0uuu	uuuu uuuu
RESET Instruction Executed	0000h	---u uuuu	uuuu u0uu
Stack Overflow Reset (STVREN = 1)	0000h	---u uuuu	1uuu uuuu
Stack Underflow Reset (STVREN = 1)	0000h	---u uuuu	u1uu uuuu
WDT Window Violation	0000h	---1 uuuu	uu0u uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

Note 1: When the wake-up is due to an interrupt and the Global Interrupt Enable bit (GIE) is set, the return address is pushed on the stack and PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

REGISTER 7-8: PIR2: PERIPHERAL INTERRUPT REQUEST REGISTER 2

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
OSFIF	C2IF	C1IF	—	BCL1IF	TMR6IF	TMR4IF	CCP2IF
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 **OSFIF**: Oscillator Fail Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 6 **C2IF**: Comparator C2 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 5 **C1IF**: Comparator C1 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

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

bit 3 **BCL1IF**: MSSP Bus Collision Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 2 **TMR6IF:** Timer6 to PR6 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 1 **TMR4IF:** Timer4 to PR4 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 0 **CCP2IF**: CCP2 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

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.

10.2.4 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the address in PMADRH:PMADRL of the row to be programmed.
2. Load each write latch with data.
3. Initiate a programming operation.
4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 10-5 (row writes to program memory with 32 write latches) for more details.

The write latches are aligned to the Flash row address boundary defined by the upper 11 bits of PMADRH:PMADRL, (PMADRH<6:0>:PMADRL<7:4>) with the lower four bits of PMADRL, (PMADRL<3:0>) determining the write latch being loaded. Write operations do not cross these boundaries. At the completion of a program memory write operation, the data in the write latches is reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a row of program memory. These steps are divided into two parts. First, each write latch is loaded with data from the PMDATH:PMDATL using the unlock sequence with LWLO = 1. When the last word to be loaded into the write latch is ready, the LWLO bit is cleared and the unlock sequence executed. This initiates the programming operation, writing all the latches into Flash program memory.

Note:	The special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. If the unlock sequence is interrupted, writing to the latches or program memory will not be initiated.

1. Set the WREN bit of the PMCON1 register.
2. Clear the CFGS bit of the PMCON1 register.
3. Set the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash program memory.
4. Load the PMADRH:PMADRL register pair with the address of the location to be written.
5. Load the PMDATH:PMDATL register pair with the program memory data to be written.
6. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The write latch is now loaded.
7. Increment the PMADRH:PMADRL register pair to point to the next location.
8. Repeat steps 5 through 7 until all but the last write latch has been loaded.
9. Clear the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '0', the write sequence will initiate the write to Flash program memory.
10. Load the PMDATH:PMDATL register pair with the program memory data to be written.
11. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The entire program memory latch content is now written to Flash program memory.

Note:	The program memory write latches are reset to the Blank state (0x3FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state.

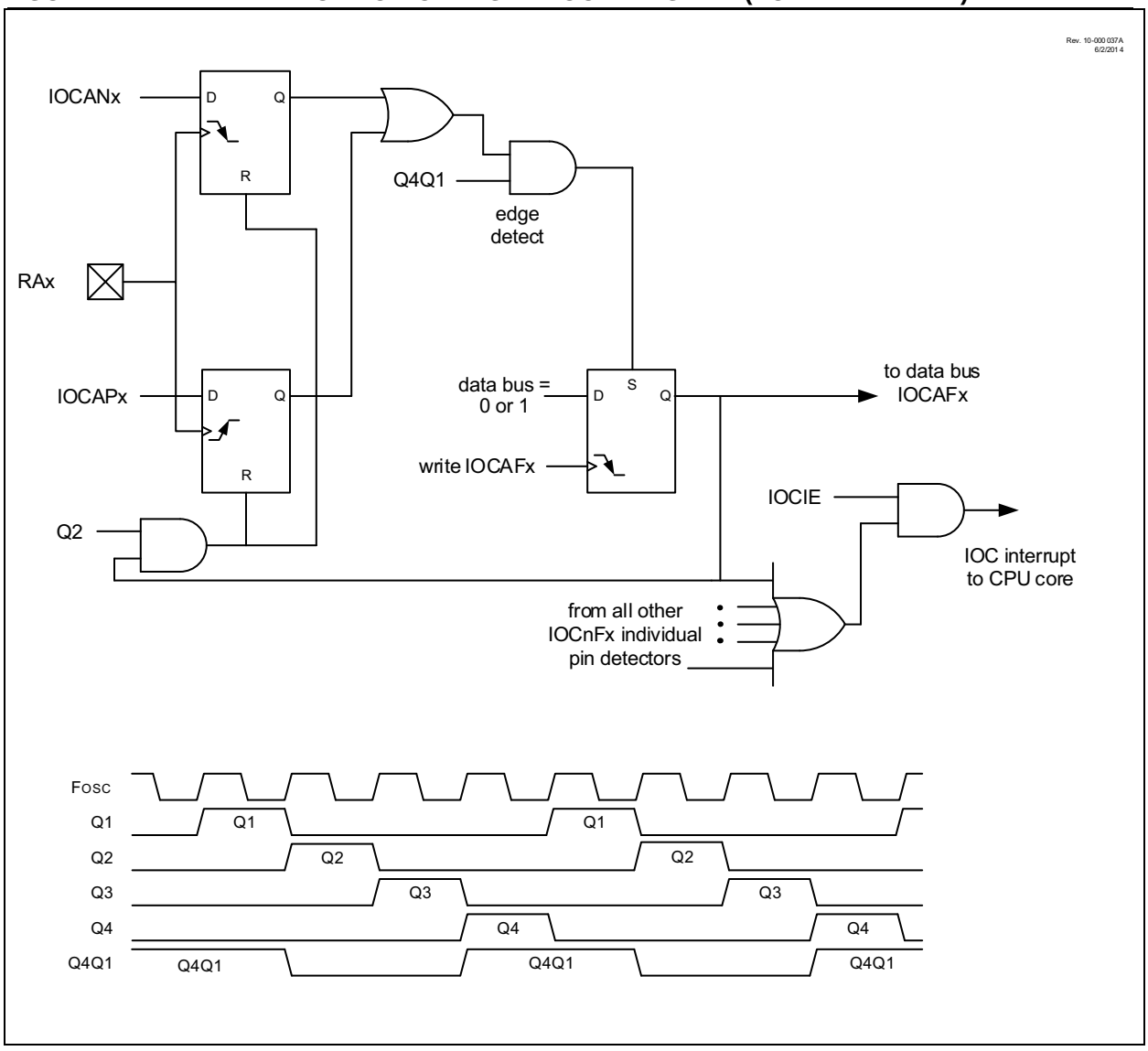
An example of the complete write sequence is shown in Example 10-3. The initial address is loaded into the PMADRH:PMADRL register pair; the data is loaded using indirect addressing.

TABLE 12-5: SUMMARY OF CONFIGURATION WORD WITH PORTB

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	IESO	$\overline{\text{CLKOUTEN}}$	BOREN<1:0>		—	69
	7:0	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE<1:0>		FOSC<2:0>			

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

FIGURE 14-1: INTERRUPT-ON-CHANGE BLOCK DIAGRAM (PORTA EXAMPLE)



REGISTER 17-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 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<9:2>							
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<9:2>**: ADC Result Register bits
Upper eight bits of 10-bit conversion result

REGISTER 17-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

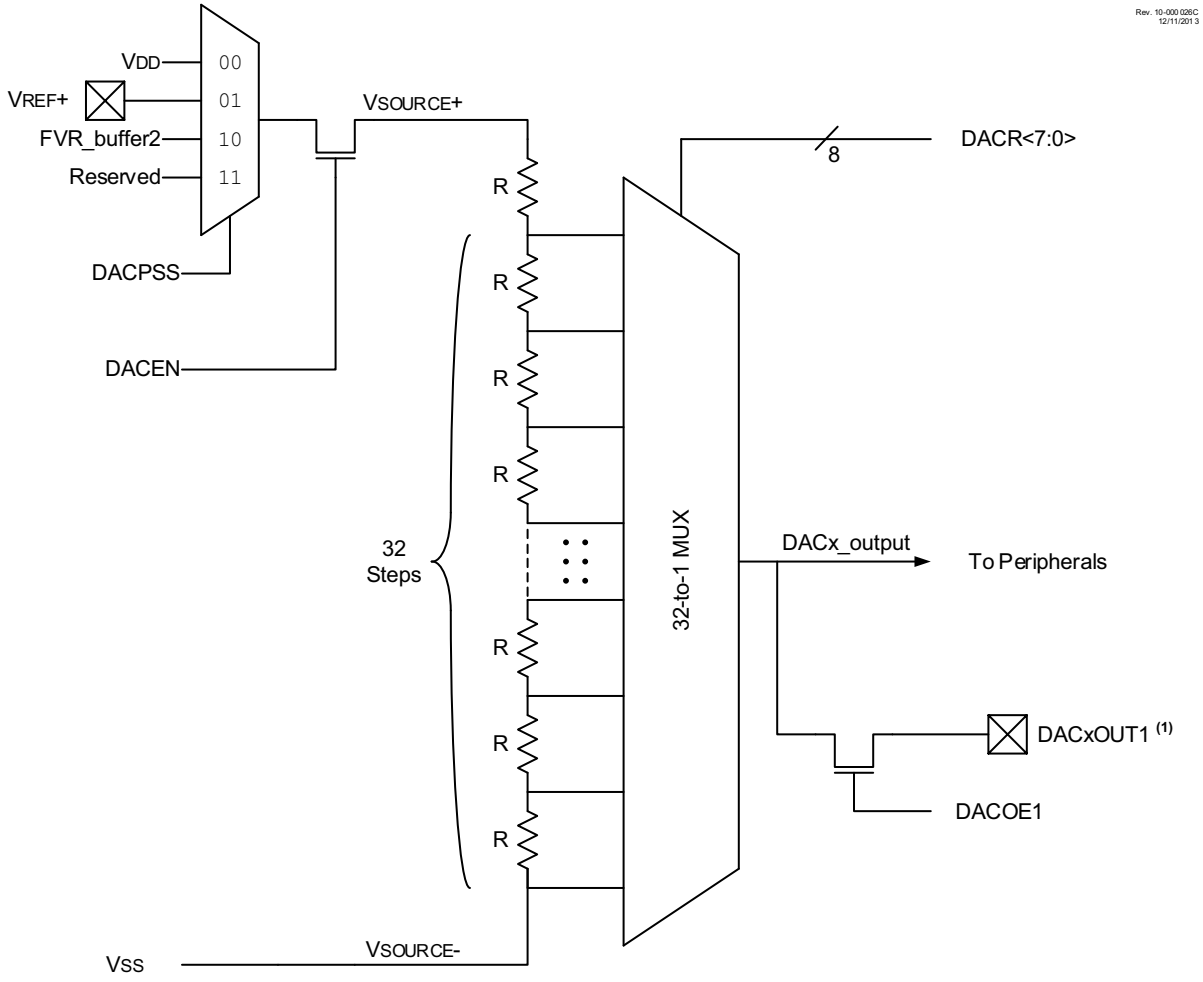
R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<1:0>		—	—	—	—	—	—
bit 7							bit 0

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 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower two bits of 10-bit conversion result

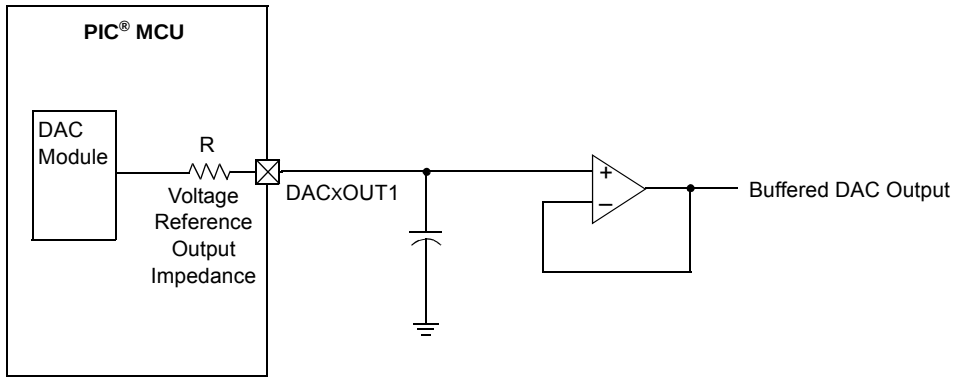
bit 5-0 **Reserved**: Do not use.

FIGURE 18-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM



Note 1: The unbuffered DACx_output is provided on the DACxOUT pin(s).

FIGURE 18-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE



20.0 ZERO-CROSS DETECTION (ZCD) MODULE

The ZCD module detects when an A/C signal crosses through the ground potential. The actual zero crossing threshold is the zero crossing reference voltage, VCPINV, which is typically 0.75V above ground.

The connection to the signal to be detected is through a series current limiting resistor. The module applies a current source or sink to the ZCD pin to maintain a constant voltage on the pin, thereby preventing the pin voltage from forward biasing the ESD protection diodes. When the applied voltage is greater than the reference voltage, the module sinks current. When the applied voltage is less than the reference voltage, the module sources current. The current source and sink action keeps the pin voltage constant over the full range of the applied voltage. The ZCD module is shown in the simplified block diagram Figure 20-2.

The ZCD module is useful when monitoring an A/C waveform for, but not limited to, the following purposes:

- A/C period measurement
- Accurate long term time measurement
- Dimmer phase delayed drive
- Low EMI cycle switching

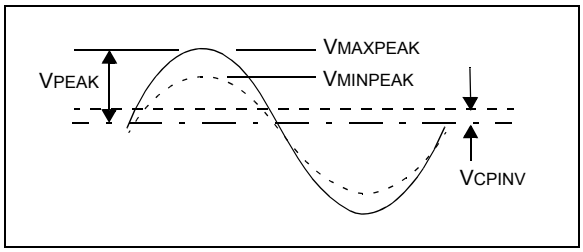
20.1 External Resistor Selection

The ZCD module requires a current limiting resistor in series with the external voltage source. The impedance and rating of this resistor depends on the external source peak voltage. Select a resistor value that will drop all of the peak voltage when the current through the resistor is nominally 300 μA. Refer to Equation 20-1 and Figure 20-1. Make sure that the ZCD I/O pin internal weak pull-up is disabled so it does not interfere with the current source and sink.

EQUATION 20-1: EXTERNAL RESISTOR

$$R_{SERIES} = \frac{V_{PEAK}}{3\times10^{-4}}$$

FIGURE 20-1: EXTERNAL VOLTAGE



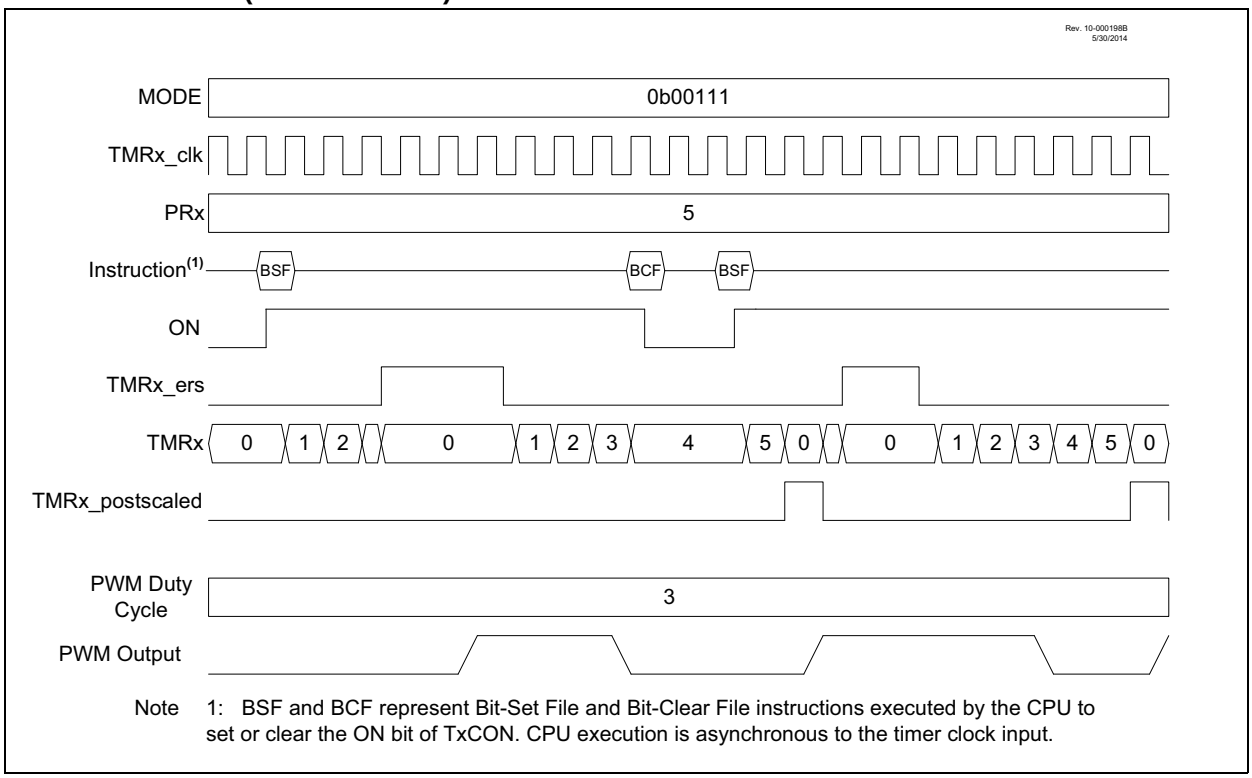
23.5.4 LEVEL-TRIGGERED HARDWARE LIMIT MODE

In the Level-Triggered Hardware Limit Timer modes the counter is reset by high or low levels of the external signal TMRx_ers, as shown in Figure 23-7. Selecting MODE<4:0> = 00110 will cause the timer to reset on a low level external signal. Selecting MODE<4:0> = 00111 will cause the timer to reset on a high level external signal. In the example, the counter is reset while TMRx_ers = 1. ON is controlled by BSF and BCF instructions. When ON = 0 the external signal is ignored.

When the CCP uses the timer as the PWM time base then the PWM output will be set high when the timer starts counting and then set low only when the timer count matches the CCPRx value. The timer is reset when either the timer count matches the PRx value or two clock periods after the external Reset signal goes true and stays true.

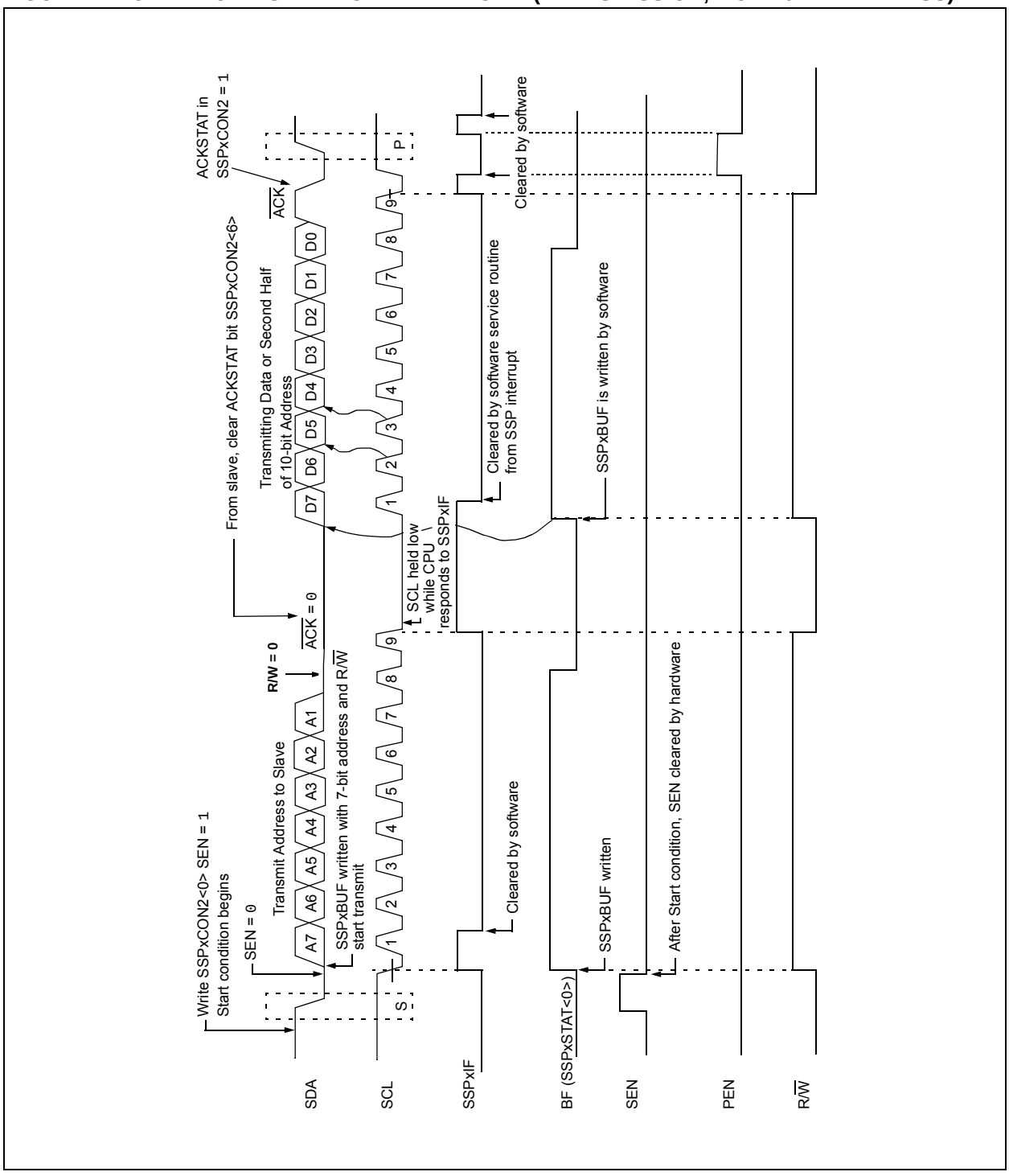
The timer starts counting, and the PWM output is set high, on either the clock following the PRx match or two clocks after the external Reset signal relinquishes the Reset. The PWM output will remain high until the timer counts up to match the CCPRx pulse width value. If the external Reset signal goes true while the PWM output is high then the PWM output will remain high until the Reset signal is released allowing the timer to count up to match the CCPRx value.

FIGURE 23-7: LEVEL-TRIGGERED HARDWARE LIMIT MODE TIMING DIAGRAM (MODE = 00111)



Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

FIGURE 24-28: I²C MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



27.1.9 SETUP FOR PWM OPERATION USING PWMx PINS

The following steps should be taken when configuring the module for PWM operation using the PWMx pins:

1. Disable the PWMx pin output driver(s) by setting the associated TRIS bit(s).
2. Clear the PWMxCON register.
3. Load the PR2 register with the PWM period value.
4. Clear the PWMxDCH register and bits <7:6> of the PWMxDCL register.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIR1 register. See note below.
   - Configure the CKPS bits of the T2CON register with the Timer2 prescale value.
   - Enable Timer2 by setting the ON bit of the T2CON register.
6. Enable PWM output pin and wait until Timer2 overflows, TMR2IF bit of the PIR1 register is set. See note below.
7. Enable the PWMx pin output driver(s) by clearing the associated TRIS bit(s) and setting the PWMxOE bit of the PWMxCON register.
8. Configure the PWM module by loading the PWMxCON register with the appropriate values.

Note 1: In order to send a complete duty cycle and period on the first PWM output, the above steps must be followed in the order given. If it is not critical to start with a complete PWM signal, then move Step 8 to replace Step 4.

2: For operation with other peripherals only, disable PWMx pin outputs.

REGISTER 28-7: CWGxOCON0: CWGx STEERING CONTROL REGISTER(1)

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
OVRD	OVRC	OVRB	OVRA	STRD(2)	STRC(2)	STRB(2)	STRA(2)
bit 7							bit 0

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

bit 7 **OVRD:** Steering Data D bit

bit 6 **OVRC:** Steering Data C bit

bit 5 **OVRB:** Steering Data B bit

bit 4 **OVRA:** Steering Data A bit

bit 3 **STRD:** Steering Enable D bit(2)
- 1 = CWGxD output has the CWGx_data waveform with polarity control from POLD bit
- 0 = CWGxD output is assigned the value of OVRD bit

bit 2 **STRC:** Steering Enable C bit(2)
- 1 = CWGxC output has the CWGx_data waveform with polarity control from POLC bit
- 0 = CWGxC output is assigned the value of OVRC bit

bit 1 **STRB:** Steering Enable B bit(2)
- 1 = CWGxB output has the CWGx_data waveform with polarity control from POLB bit
- 0 = CWGxB output is assigned the value of OVRB bit

bit 0 **STRA:** Steering Enable A bit(2)
- 1 = CWGxA output has the CWGx_data waveform with polarity control from POLA bit
- 0 = CWGxA output is assigned the value of OVRA bit

Note 1: The bits in this register apply only when MODE<2:0> = 00x.

2: This bit is effectively double-buffered when MODE<2:0> = 001.

30.0 SIGNAL MEASUREMENT TIMER (SMT)

The SMT is a 24-bit counter with advanced clock and gating logic, which can be configured for measuring a variety of digital signal parameters such as pulse width, frequency and duty cycle, and the time difference between edges on two signals.

Features of the SMT include:

- 24-bit timer/counter
  - Four 8-bit registers (SMTxTMRL/H/U)
  - Readable and writable
  - Optional 16-bit operating mode
- Two 24-bit measurement capture registers
- One 24-bit period match register
- Multi-mode operation, including relative timing measurement
- Interrupt on period match
- Multiple clock, gate and signal sources
- Interrupt on acquisition complete
- Ability to read current input values

Note:	These devices implement two SMT modules. All references to SMTx apply to SMT1 and SMT2.

FIGURE 30-8: HIGH AND LOW MEASURE MODE REPEAT ACQUISITION TIMING DIAGRAM

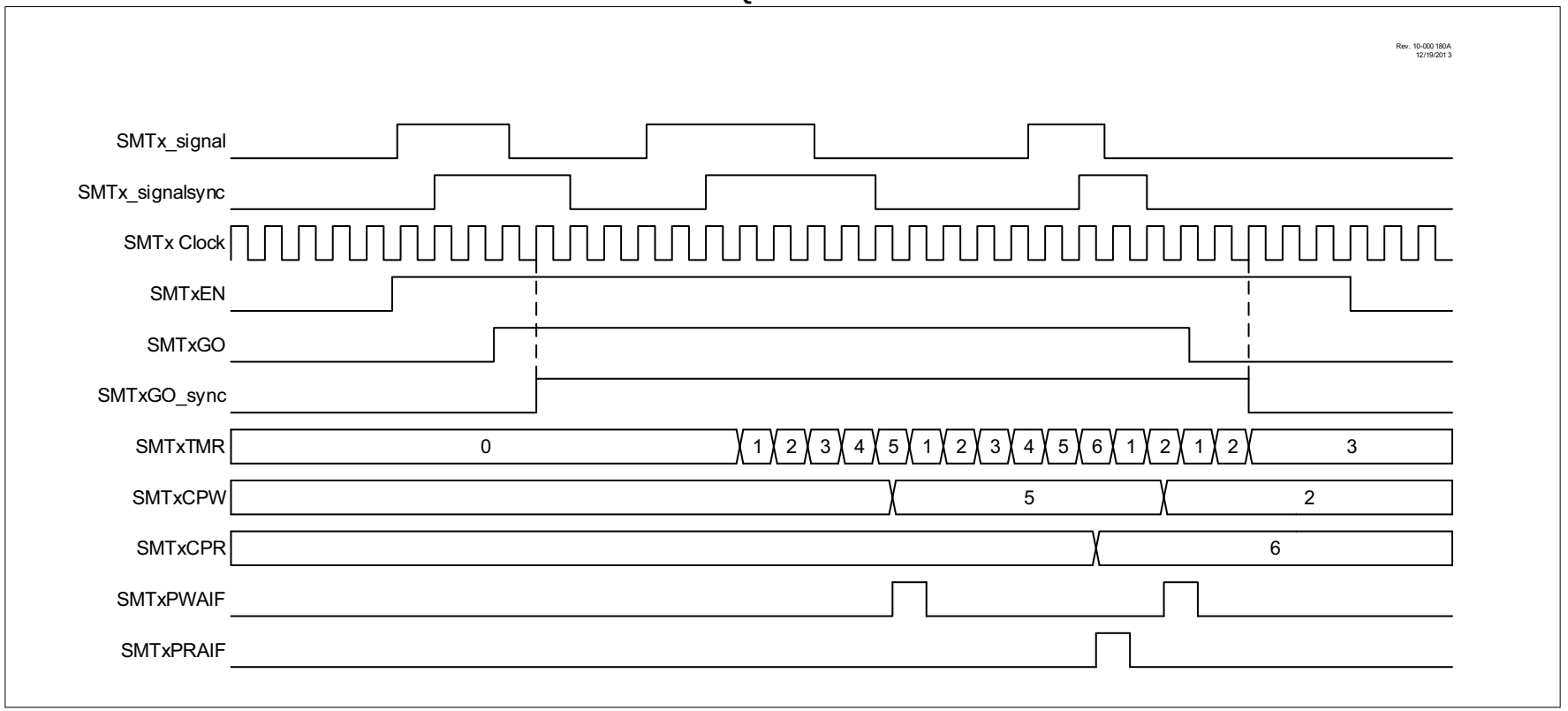
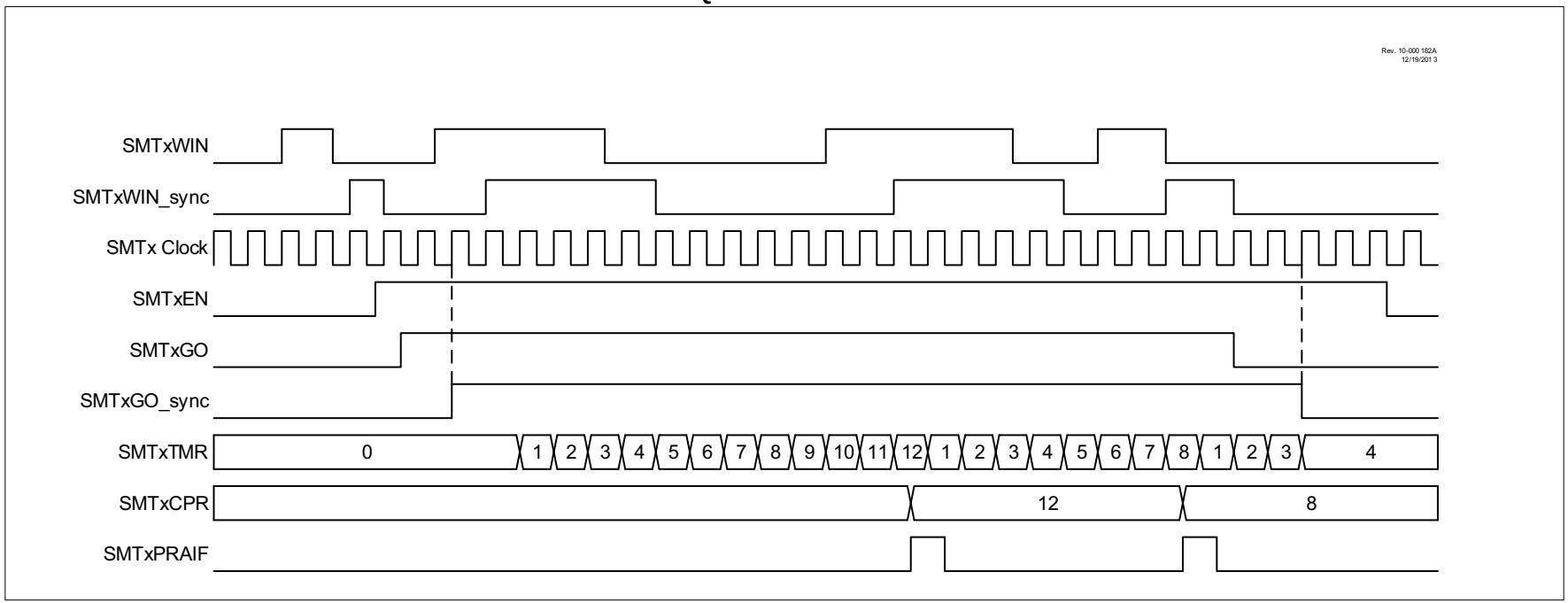


FIGURE 30-10: WINDOWED MEASURE MODE REPEAT ACQUISITION TIMING DIAGRAM



30.6.9 COUNTER MODE

This mode increments the timer on each pulse of the SMTx_signal input. This mode is asynchronous to the SMT clock and uses the SMTx_signal as a time source. The SMTxCPW register will be updated with the current SMTxTMR value on the falling edge of the SMTxWIN input. See Figure 30-18.

FIGURE 30-21: WINDOWED COUNTER MODE REPEAT ACQUISITION TIMING DIAGRAM

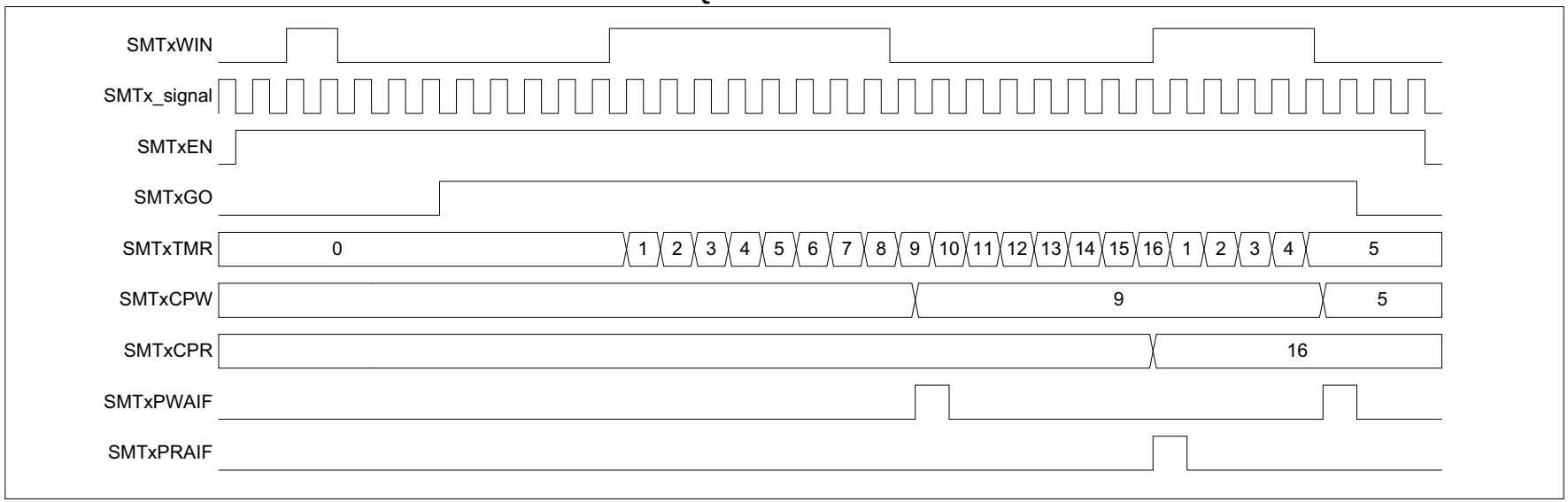


FIGURE 30-22: WINDOWED COUNTER MODE SINGLE ACQUISITION TIMING DIAGRAM

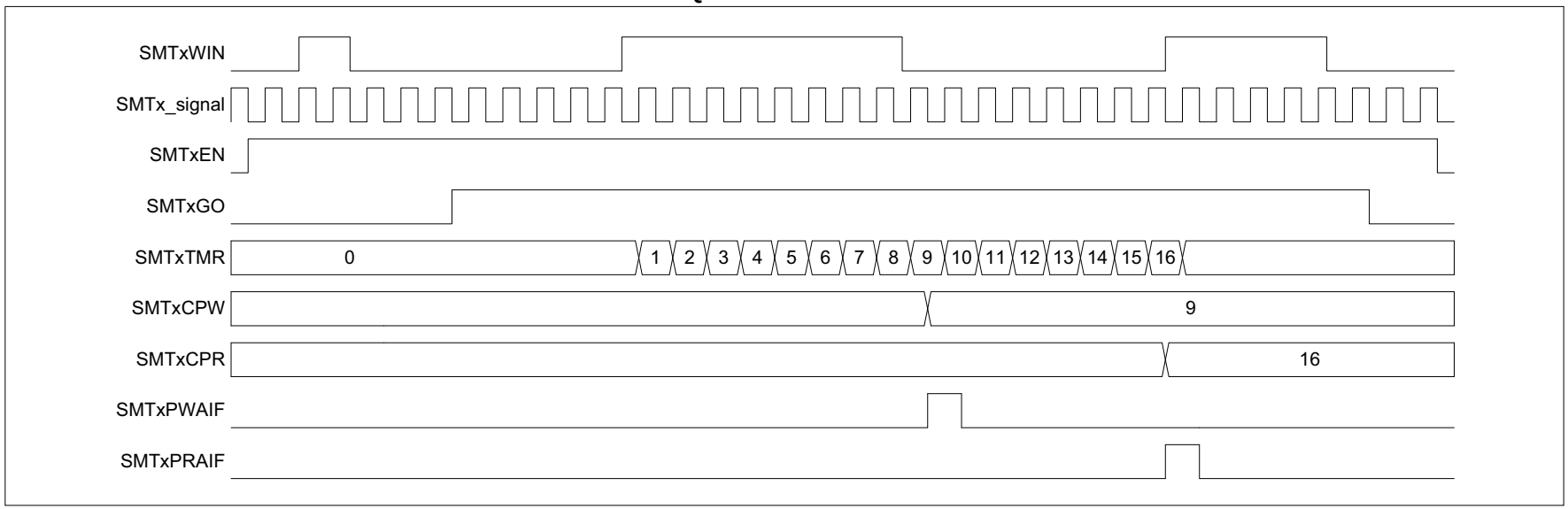
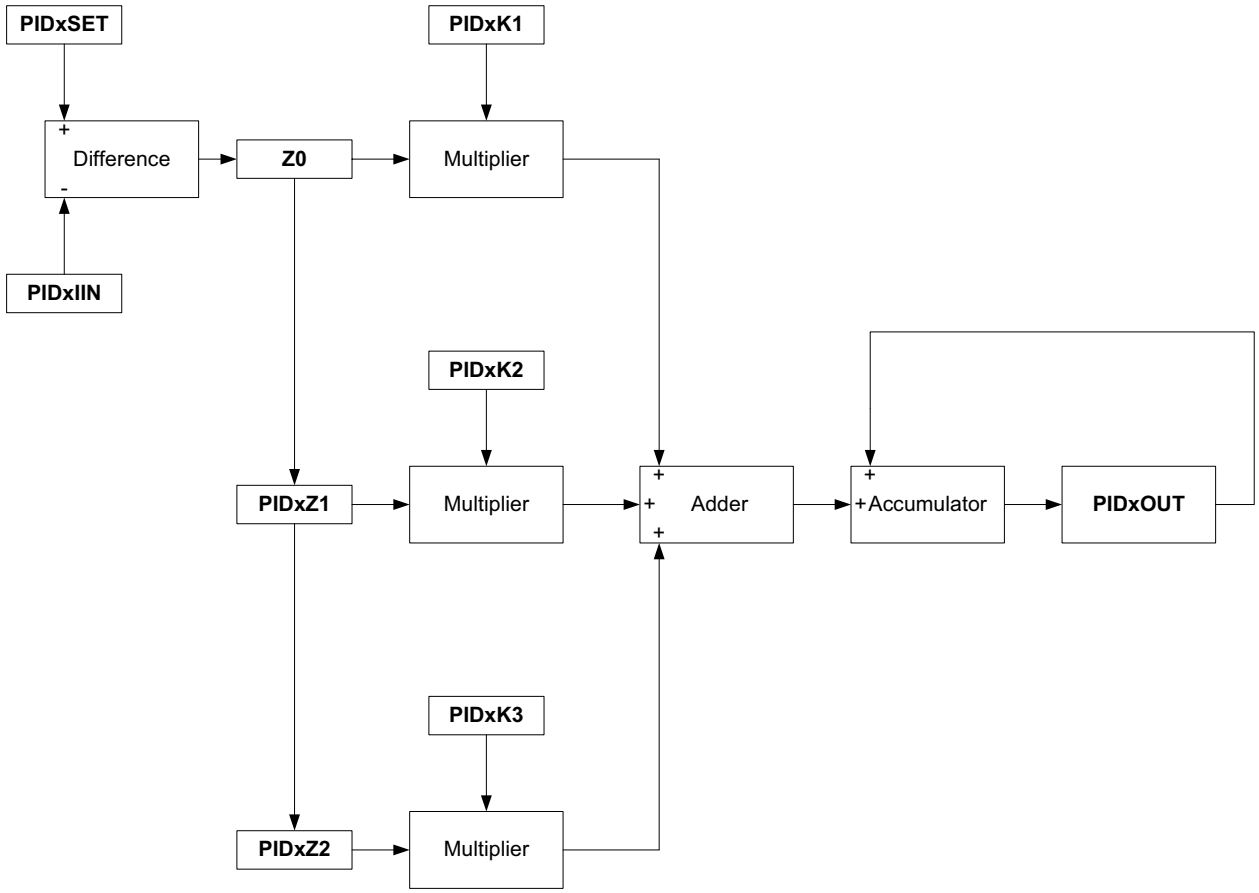


FIGURE 32-1: PID MODULE BASIC DATA FLOW BLOCK DIAGRAM, PID MODES



Note 1: After the results of PIDxZ2 are multiplied by PIDxK3 and the result is added to the accumulator, the current value from PIDxZ1 is loaded into PIDxZ2. The same is true for PIDxZ1 and the current SET-IN value.

REGISTER 32-26: PIDxACCLH: PID ACCUMULATOR LOW HIGH REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ACC<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	q = Value depends on condition

bit 7-0 **ACC<15:8>:** Bits <15:8> of ACC. ACC is the accumulator register in which all of the multiplier results for the PID are accumulated before being written to the output.

REGISTER 32-27: PIDxACCLL: PID ACCUMULATOR LOW LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ACC<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	q = Value depends on condition

bit 7-0 **ACC<7:0>:** Bits <7:0> of ACC. ACC is the accumulator register in which all of the multiplier results for the PID are accumulated before being written to the output.