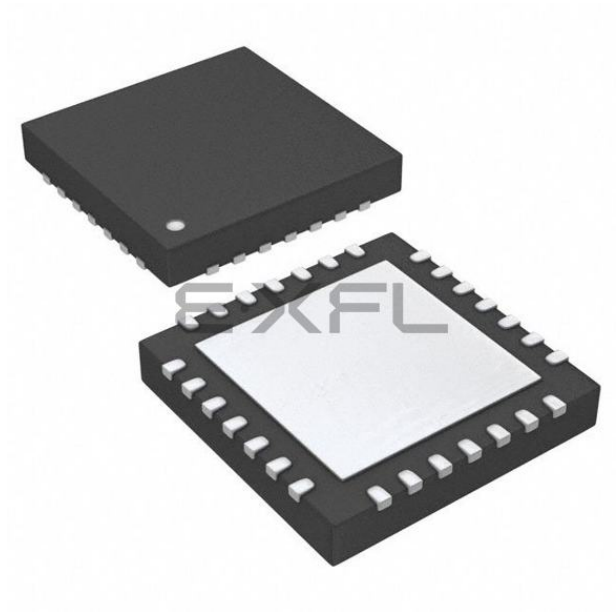


Welcome to E-XFL.COM

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	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18854-i-ml

Table of Contents

TABLE 1-2: PIC16F18854 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
OUT(2)	CWG3C	—	CMOS/OD	Complementary Waveform Generator 3 output C.
	CWG3D	—	CMOS/OD	Complementary Waveform Generator 3 output D.
	CLC1OUT	—	CMOS/OD	Configurable Logic Cell 1 output.
	CLC2OUT	—	CMOS/OD	Configurable Logic Cell 2 output.
	CLC3OUT	—	CMOS/OD	Configurable Logic Cell 3 output.
	CLC4OUT	—	CMOS/OD	Configurable Logic Cell 4 output.
	NCO1	—	CMOS/OD	Numerically Controller Oscillator output.
	CLKR	—	CMOS/OD	Clock Reference module output.

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 13-1 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 13-3.

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.

2.1 Automatic Interrupt Context Saving

During interrupts, certain registers are automatically saved in shadow registers and restored when returning from the interrupt. This saves stack space and user code. See **Section 7.5 "Automatic Context Saving"** for more information.

2.2 16-Level Stack with Overflow and Underflow

These devices have a hardware stack memory 15 bits wide and 16 words deep. A Stack Overflow or Underflow will set the appropriate bit (STKOVF or STKUNF) in the PCON register, and if enabled, will cause a software Reset. See **Section 3.4 "Stack"** for more details.

2.3 File Select Registers

There are two 16-bit File Select Registers (FSR). FSRs can access all file registers and program memory, which allows one Data Pointer for all memory. When an FSR points to program memory, there is one additional instruction cycle in instructions using INDF to allow the data to be fetched. General purpose memory can now also be addressed linearly, providing the ability to access contiguous data larger than 80 bytes. There are also new instructions to support the FSRs. See **Section 3.5 "Indirect Addressing"** for more details.

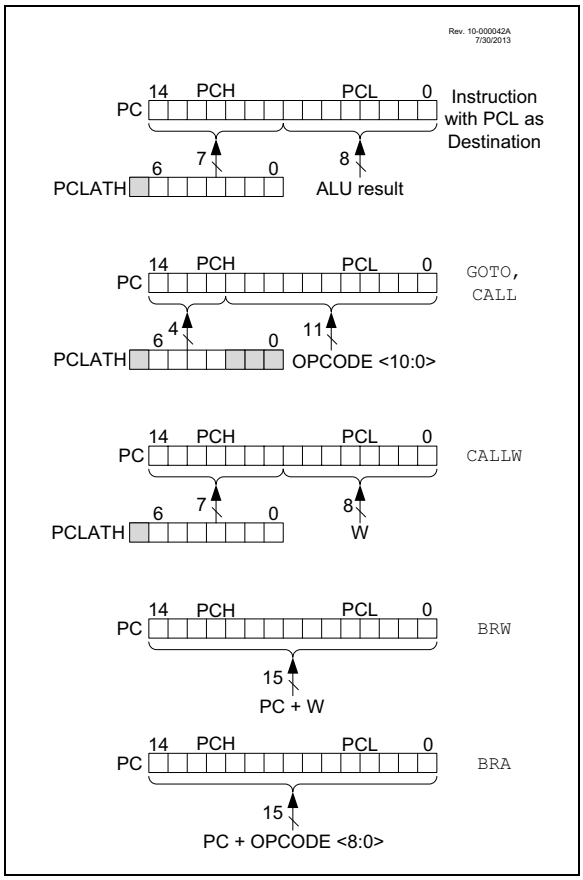
2.4 Instruction Set

There are 49 instructions for the enhanced mid-range CPU to support the features of the CPU. See **Section 36.0 "Instruction Set Summary"** for more details.

3.3 PCL and PCLATH

The Program Counter (PC) is 15 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<14:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 3-3 shows the five situations for the loading of the PC.

FIGURE 3-3: LOADING OF PC IN DIFFERENT SITUATIONS



3.3.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<14:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper seven bits to the PCLATH register. When the lower eight bits are written to the PCL register, all 15 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

3.3.2 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to Application Note AN556, *"Implementing a Table Read"* (DS00556).

3.3.3 COMPUTED FUNCTION CALLS

A computed function CALL allows programs to maintain tables of functions and provide another way to execute state machines or look-up tables. When performing a table read using a computed function CALL, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block).

If using the CALL instruction, the PCH<2:0> and PCL registers are loaded with the operand of the CALL instruction. PCH<6:3> is loaded with PCLATH<6:3>.

The CALLW instruction enables computed calls by combining PCLATH and W to form the destination address. A computed CALLW is accomplished by loading the W register with the desired address and executing CALLW. The PCL register is loaded with the value of W and PCH is loaded with PCLATH.

3.3.4 BRANCHING

The branching instructions add an offset to the PC. This allows relocatable code and code that crosses page boundaries. There are two forms of branching, BRW and BRA. The PC will have incremented to fetch the next instruction in both cases. When using either branching instruction, a PCL memory boundary may be crossed.

If using BRW, load the W register with the desired unsigned address and execute BRW. The entire PC will be loaded with the address PC + 1 + W.

If using BRA, the entire PC will be loaded with PC + 1, the signed value of the operand of the BRA instruction.

3.4 Stack

All devices have a 16-level x 15-bit wide hardware stack (refer to Figure 3-4 through Figure 3-7). The stack space is not part of either program or data space. The PC is PUSHed onto the stack when CALL or CALLW instructions are executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer if the STVREN bit is programmed to '0' (Configuration Words). This means that after the stack has been PUSHed sixteen times, the seventeenth PUSH overwrites the value that was stored from the first PUSH. The eighteenth PUSH overwrites the second PUSH (and so on). The STKOVF and STKUNF flag bits will be set on an Overflow/Underflow, regardless of whether the Reset is enabled.

Note 1: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, CALLW, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

3.4.1 ACCESSING THE STACK

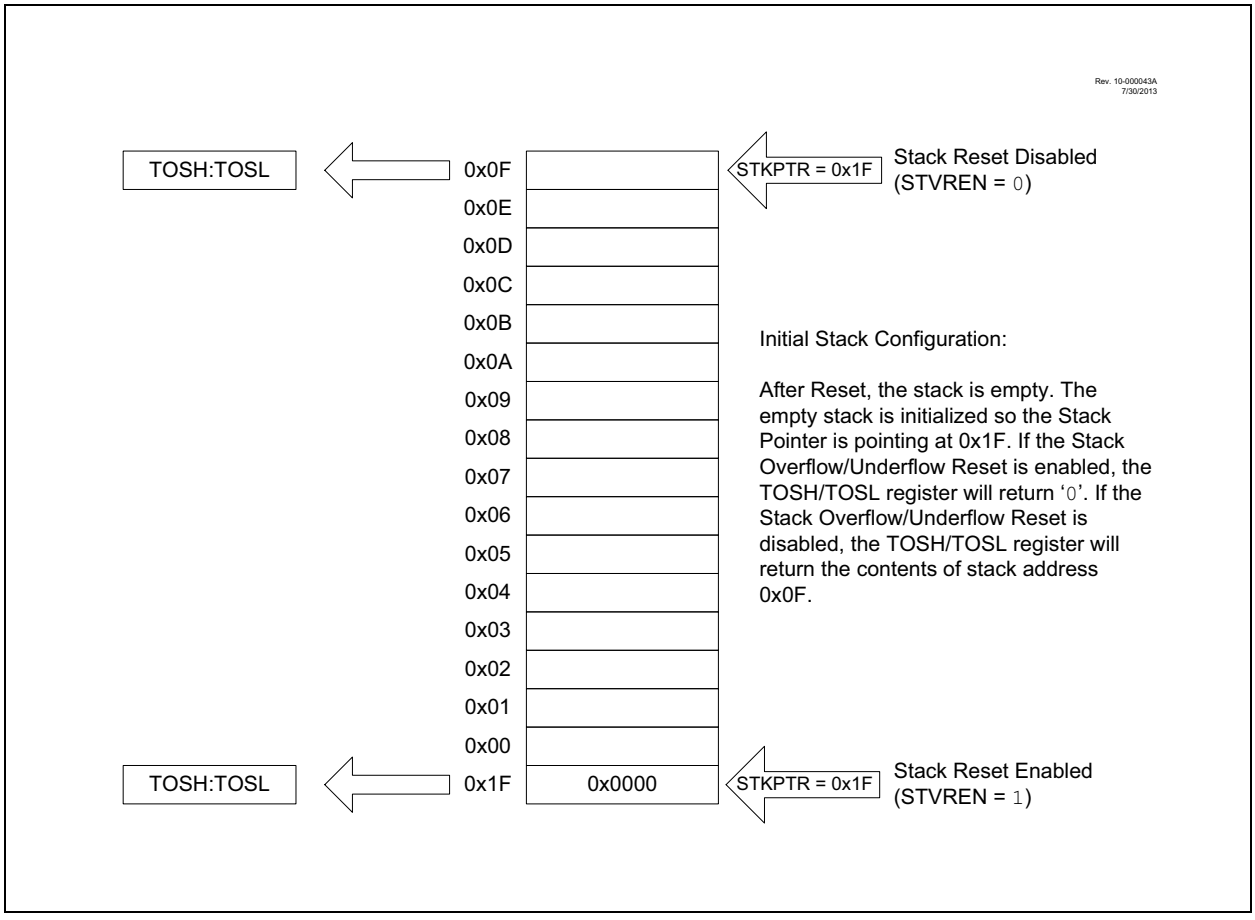
The stack is available through the TOSH, TOSL and STKPTR registers. STKPTR is the current value of the Stack Pointer. TOSH:TOSL register pair points to the TOP of the stack. Both registers are read/writable. TOS is split into TOSH and TOSL due to the 15-bit size of the PC. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL. STKPTR is five bits to allow detection of overflow and underflow.

Note: Care should be taken when modifying the STKPTR while interrupts are enabled.

During normal program operation, CALL, CALLW and interrupts will increment STKPTR while RETLW, RETURN, and RETFIE will decrement STKPTR. At any time, STKPTR can be inspected to see how much stack is left. The STKPTR always points at the currently used place on the stack. Therefore, a CALL or CALLW will increment the STKPTR and then write the PC, and a return will unload the PC and then decrement the STKPTR.

Reference Figure 3-4 through Figure 3-7 for examples of accessing the stack.

FIGURE 3-4: ACCESSING THE STACK EXAMPLE 1



5.13 Register Definitions: Power Control

REGISTER 5-2: PCON0: POWER CONTROL REGISTER 0

R/W/HS-0/q	R/W/HS-0/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
bit 7							bit 0

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

bit 7 **STKOVF:** Stack Overflow Flag bit
1 = A Stack Overflow occurred
0 = A Stack Overflow has not occurred or cleared by firmware

bit 6 **STKUNF:** Stack Underflow Flag bit
1 = A Stack Underflow occurred
0 = A Stack Underflow has not occurred or cleared by firmware

bit 5 **$\overline{\text{WDTWV}}$:** WDT Window Violation Flag bit
1 = A WDT Window Violation Reset has not occurred or set by firmware
0 = A WDT Window Violation Reset has occurred (a CLRWDT instruction was executed either without arming the window or outside the window (cleared by hardware)

bit 4 **$\overline{\text{RWDT}}$:** Watchdog Timer Reset Flag bit
1 = A Watchdog Timer Reset has not occurred or set to '1' by firmware
0 = A Watchdog Timer Reset has occurred (cleared by hardware)

bit 3 **$\overline{\text{RMCLR}}$:** $\overline{\text{MCLR}}$ Reset Flag bit
1 = A $\overline{\text{MCLR}}$ Reset has not occurred or set to '1' by firmware
0 = A $\overline{\text{MCLR}}$ Reset has occurred (cleared by hardware)

bit 2 **$\overline{\text{RI}}$:** RESET Instruction Flag bit
1 = A RESET instruction has not been executed or set to '1' by firmware
0 = A RESET instruction has been executed (cleared by hardware)

bit 1 **$\overline{\text{POR}}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{\text{BOR}}$:** Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

TABLE 5-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BORCON	SBOREN	—	—	—	—	—	—	BORRDY	85
PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	90
STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	26
WDTCON0	—	—	WDTPS<4:0>					SWDTEN	146

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

REGISTER 8-2: CPUDOZE: DOZE AND IDLE REGISTER

R/W-0/u	R/W/HC/HS-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
IDLEN	DOZEN(1,2)	ROI	DOE	—	DOZE<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	

bit 7 **IDLEN:** Idle Enable bit
1 = A SLEEP instruction inhibits the CPU clock, but not the peripheral clock(s)
0 = A SLEEP instruction places the device into full Sleep mode

bit 6 **DOZEN:** Doze Enable bit(1,2)
1 = The CPU executes instruction cycles according to DOZE setting
0 = The CPU executes all instruction cycles (fastest, highest power operation)

bit 5 **ROI:** Recover-on-Interrupt bit
1 = Entering the Interrupt Service Routine (ISR) makes DOZEN = 0 bit, bringing the CPU to full-speed operation.
0 = Interrupt entry does not change DOZEN

bit 4 **DOE:** Doze on Exit bit
1 = Executing RETFIE makes DOZEN = 1, bringing the CPU to reduced speed operation.
0 = RETFIE does not change DOZEN

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

bit 2-0 **DOZE<2:0>:** Ratio of CPU Instruction Cycles to Peripheral Instruction Cycles
111 = 1:256
110 = 1:128
101 = 1:64
100 = 1:32
011 = 1:16
010 = 1:8
001 = 1:4
000 = 1:2

Note 1: When ROI = 1 or DOE = 1, DOZEN is changed by hardware interrupt entry and/or exit.
2: Entering ICD overrides DOZEN, returning the CPU to full execution speed; this bit is not affected.

TABLE 9-3: SUMMARY OF REGISTERS ASSOCIATED WITH 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>				102
OSCCON2	—	COSC<2:0>			CDIV<3:0>				102
OSCCON3	CSWHOLD	SOSCPWR	—	ORDY	NOSCR	—	—	—	103
PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	90
STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	26
WDTCON0	—	—	WDTPS<4:0>					SEN	146
WDTCON1	—	WDTCS<2:0>			—	WINDOW<2:0>			146
WDTPSL	PSCNT<7:0>								146
WDTPSH	PSCNT<15:8>								146
WDTTMR	—	WDTTMR<4:0>				STATE	PSCNT<17:16>		146

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

TABLE 9-4: 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	—	—	$\overline{\text{CLKOUTEN}}$	74
	7:0	—	RSTOSC<2:0>			—	FEXTOSC<2:0>			

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

EXAMPLE 10-5: DEVICE ID ACCESS

```
; This write routine assumes the following:
; 1.  64 bytes of data are loaded, starting at the address in DATA_ADDR
; 2.  Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
; stored in little endian format
; 3.  A valid starting address (the least significant bits = 00000) is loaded in ADDRH:ADDRL
; 4.  ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)
; 5.  NVM interrupts are not taken into account

          BANKSEL             NVMADRH
          MOVF                ADDRH,W
          MOVWF               NVMADRH               ; Load initial address
          MOVF                ADDRL,W
          MOVWF               NVMADRL
          MOVLW               LOW DATA_ADDR         ; Load initial data address
          MOVWF               FSR0L
          MOVLW               HIGH DATA_ADDR
          MOVWF               FSR0H
          BCF                 NVMCON1,NVMREGS       ; Set PFM as write location
          BSF                 NVMCON1,WREN          ; Enable writes
          BSF                 NVMCON1,LWLO          ; Load only write latches

LOOP
          MOVIW               FSR0++
          MOVWF               NVMDATL               ; Load first data byte
          MOVIW               FSR0++
          MOVWF               NVMDATH               ; Load second data byte

          MOVF                NVMADRL,W
          XORLW               0x1F                  ; Check if lower bits of address are 00000
          ANDLW               0x1F                  ; and if on last of 32 addresses
          BTFSC               STATUS,Z              ; Last of 32 words?
          GOTO                START_WRITE           ; If so, go write latches into memory

          CALL                UNLOCK_SEQ            ; If not, go load latch
          INCF                NVMADRL,F             ; Increment address
          GOTO                LOOP

START_WRITE
          BCF                 NVMCON1,LWLO          ; Latch writes complete, now write memory
          CALL                UNLOCK_SEQ            ; Perform required unlock sequence
          BCF                 NVMCON1,WREN          ; Disable writes

UNLOCK_SEQ
          MOVLW               55h
          BCF                 INTCON,GIE            ; Disable interrupts
          MOVWF               NVMCON2               ; Begin unlock sequence
          MOVLW               AAh
          MOVWF               NVMCON2
          BSF                 NVMCON1,WR
          BSF                 INTCON,GIE            ; Unlock sequence complete, re-enable interrupts
          return
```


TABLE 13-4: SUMMARY OF REGISTERS ASSOCIATED WITH THE PPS MODULE (CONTINUED)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
RA4PPS	—	—	RA4PPS<5:0>						215
RA5PPS	—	—	RA5PPS<5:0>						215
RA6PPS	—	—	RA6PPS<5:0>						215
RA7PPS	—	—	RA7PPS<5:0>						215
RB0PPS	—	—	RB0PPS<5:0>						215
RB1PPS	—	—	RB1PPS<5:0>						215
RB2PPS	—	—	RB2PPS<5:0>						215
RB3PPS	—	—	RB3PPS<5:0>						215
RB4PPS	—	—	RB4PPS<5:0>						215
RB5PPS	—	—	RB5PPS<5:0>						215
RB6PPS	—	—	RB6PPS<5:0>						215
RB7PPS	—	—	RB7PPS<5:0>						215
RC0PPS	—	—	RC0PPS<5:0>						215
RC1PPS	—	—	RC1PPS<5:0>						215
RC2PPS	—	—	RC2PPS<5:0>						215
RC3PPS	—	—	RC3PPS<5:0>						215
RC4PPS	—	—	RC4PPS<5:0>						215
RC5PPS	—	—	RC5PPS<5:0>						215
RC6PPS	—	—	RC6PPS<5:0>						215
RC7PPS	—	—	RC7PPS<5:0>						215

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the PPS module.

REGISTER 15-7: IOCCP: INTERRUPT-ON-CHANGE PORTC POSITIVE EDGE 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
IOCCP7	IOCCP6	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0
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 **IOCCP<7:0>:** Interrupt-on-Change PORTC Positive Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a positive-going edge. IOCCFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin

REGISTER 15-8: IOCCN: INTERRUPT-ON-CHANGE PORTC NEGATIVE EDGE 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
IOCCN7	IOCCN6	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0
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 **IOCCN<7:0>:** Interrupt-on-Change PORTC Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative-going edge. IOCCFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin

REGISTER 15-9: IOCCF: INTERRUPT-ON-CHANGE PORTC FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCCF7	IOCCF6	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0
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 - Bit is set in hardware

bit 7-0 **IOCCF<7:0>:** Interrupt-on-Change PORTC Flag bits
1 = An enabled change was detected on the associated pin
Set when IOCCPx = 1 and a rising edge was detected on RCx, or when IOCCNx = 1 and a falling edge was detected on RCx.
0 = No change was detected, or the user cleared the detected change

16.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- ADC input channel
- ADC positive reference
- Comparator positive input
- Digital-to-Analog Converter (DAC)

The FVR can be enabled by setting the FVREN bit of the FVRCON register.

Note:	Fixed Voltage Reference output cannot exceed VDD.

16.1 Independent Gain Amplifiers

The output of the FVR, which is connected to the ADC, comparators, and DAC, is routed through two independent programmable gain amplifiers. Each amplifier can be programmed for a gain of 1x, 2x or 4x, to produce the three possible voltage levels.

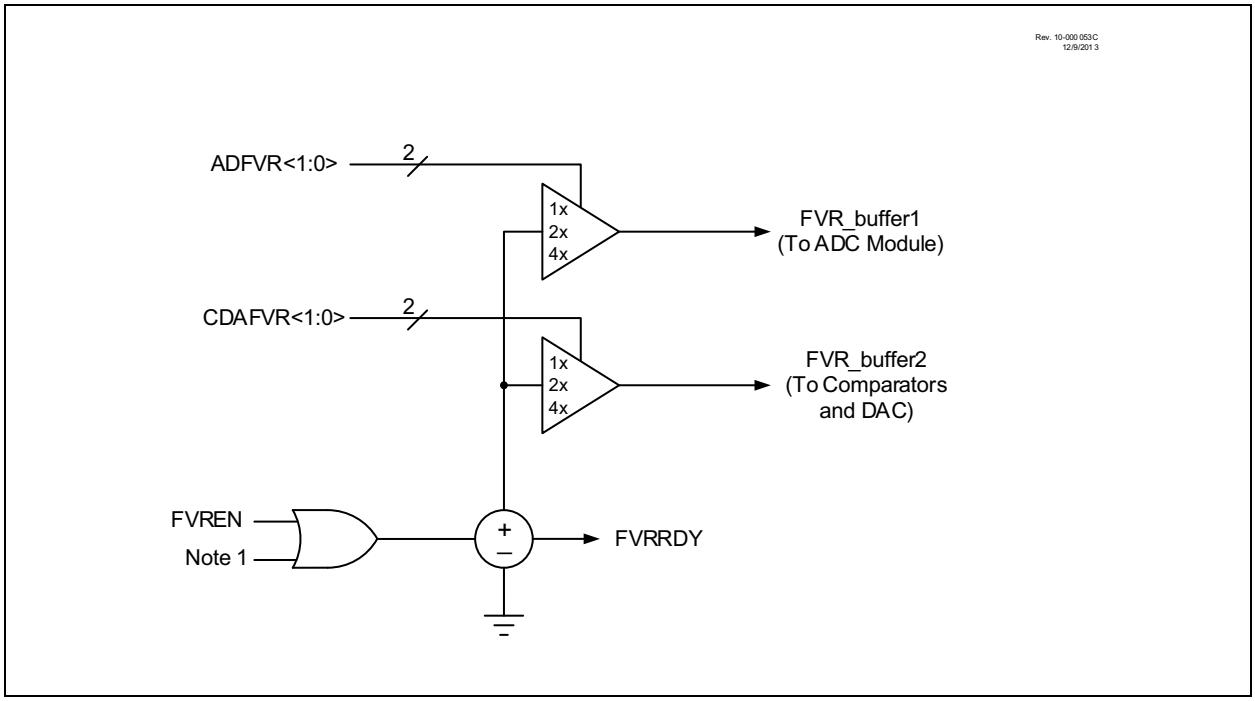
The ADFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the ADC module. Reference **Section 23.0 "Analog-to-Digital Converter With Computation (ADC2) Module"** for additional information.

The CDAFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the DAC and comparator module. Reference **Section 25.0 "5-Bit Digital-to-Analog Converter (DAC1) Module"** and **Section 18.0 "Comparator Module"** for additional information.

16.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference and amplifier circuits to stabilize. Once the circuits stabilize and are ready for use, the FVRRDY bit of the FVRCON register will be set.

FIGURE 16-1: VOLTAGE REFERENCE BLOCK DIAGRAM



17.4 ADC Acquisition Time

To ensure accurate temperature measurements, the user must wait at least 200 μs after the ADC input multiplexer is connected to the temperature indicator output before the conversion is performed. In addition, the user must wait 200 μs between consecutive conversions of the temperature indicator output.

TABLE 17-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE TEMPERATURE INDICATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDFVR<1:0>		ADFVR<1:0>		234

Legend: Shaded cells are unused by the Temperature Indicator module.

Using the Auto-conversion Trigger does not assure proper ADC timing. It is the user's responsibility to ensure that the ADC timing requirements are met. See Table 23-2 for auto-conversion sources.

TABLE 23-2: ADC AUTO-CONVERSION TABLE

ADACT Value	Sour0x1Dce Peripher0x1Dal	Description
0x00	Disabled	External Trigger Disabled
0x01	ADACTPPS	Pin selected by ADACTPPS
0x02	TMR0	Timer0 overflow condition
0x03	TMR1	Timer1 overflow condition
0x04	TMR2	Match between Timer2 postscaled value and PR2
0x05	TMR3	Timer3 overflow condition
0x06	TMR4	Match between Timer4 postscaled value and PR4
0x07	TMR5	Timer5 overflow condition
0x08	TMR6	Match between Timer6 postscaled value and PR6
0x09	SMT1	Match between SMT1 and SMT1PR
0x0A	SMT2	Match between SMT2 and SMT2PR
0x0B	CCP1	CCP1 output
0x0C	CCP2	CCP2 output
0x0D	CCP3	CCP3 output
0x0E	CCP4	CCP4 output
0x0F	CCP5	CCP5 output
0x10	PWM6	PWM6 output
0x11	PWM7	PWM7 output
0x12	C1	Comparator C1 output
0x13	C2	Comparator C2 output
0x14	IOC	Interrupt-on-change interrupt trigger
0x15	CLC1	CLC1 output
0x16	CLC2	CLC2 output
0x17	CLC3	CLC3 output
0x18	CLC4	CLC4 output
0x19-0x1B	Reserved	Reserved, do not use
0x1C	ADERR	Read of ADERR register
0x1D	ADRESH	Read of ADRESH register
0x1E	Reserved	Reserved, do not use
0x1F	ADPCH	Read of ADPCH register

FIGURE 26-5: CARRIER LOW SYNCHRONIZATION (MDSHSYNC = 0, MDCLSYNC = 1)

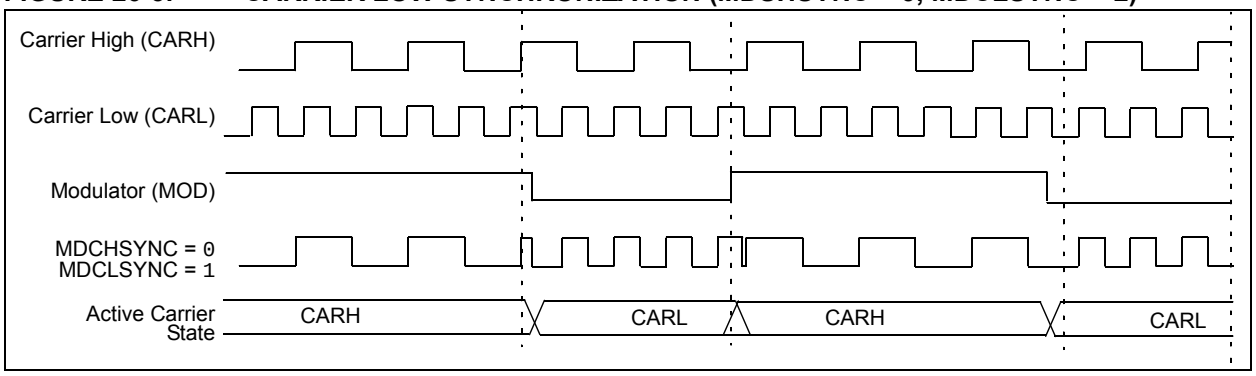
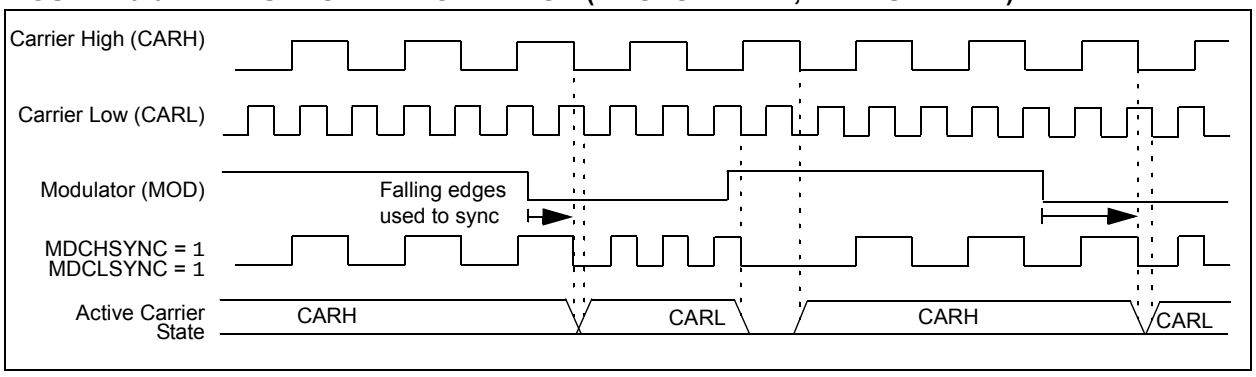


FIGURE 26-6: FULL SYNCHRONIZATION (MDSHSYNC = 1, MDCLSYNC = 1)



29.0 TIMER2/4/6 MODULE

The Timer2/4/6 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 such as 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 29-1 for a block diagram of Timer2. See Figure 29-2 for the clock source block diagram.

Note:	Three identical Timer2 modules are implemented on this device. The timers are named Timer2, Timer4, and Timer6. All references to Timer2 apply as well to Timer4 and Timer6. All references to T2PR apply as well to T4PR and T6PR.

FIGURE 29-1: TIMER2 BLOCK DIAGRAM

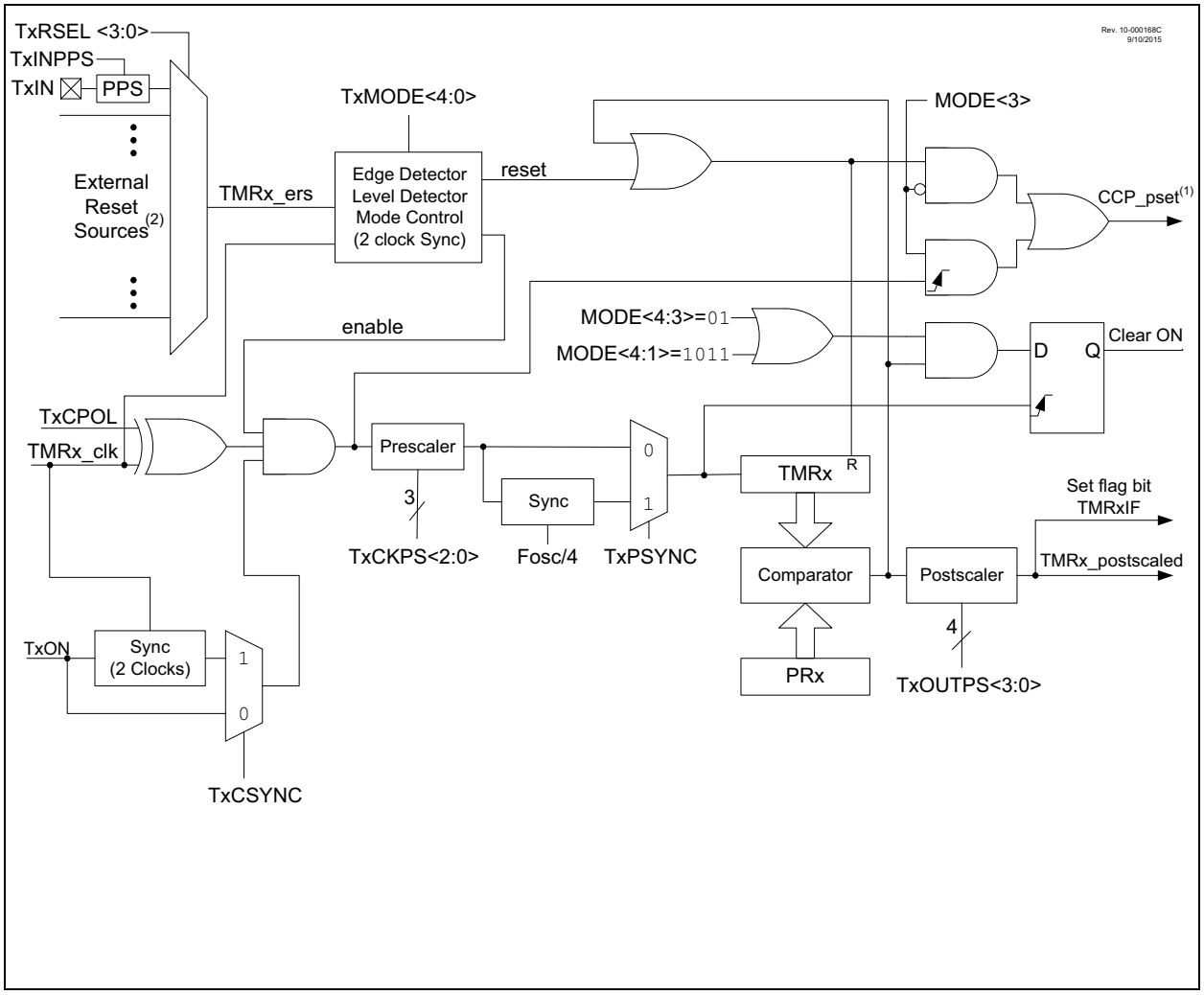


FIGURE 32-12: GATED WINDOWED MEASURE MODE REPEAT ACQUISITION TIMING DIAGRAM

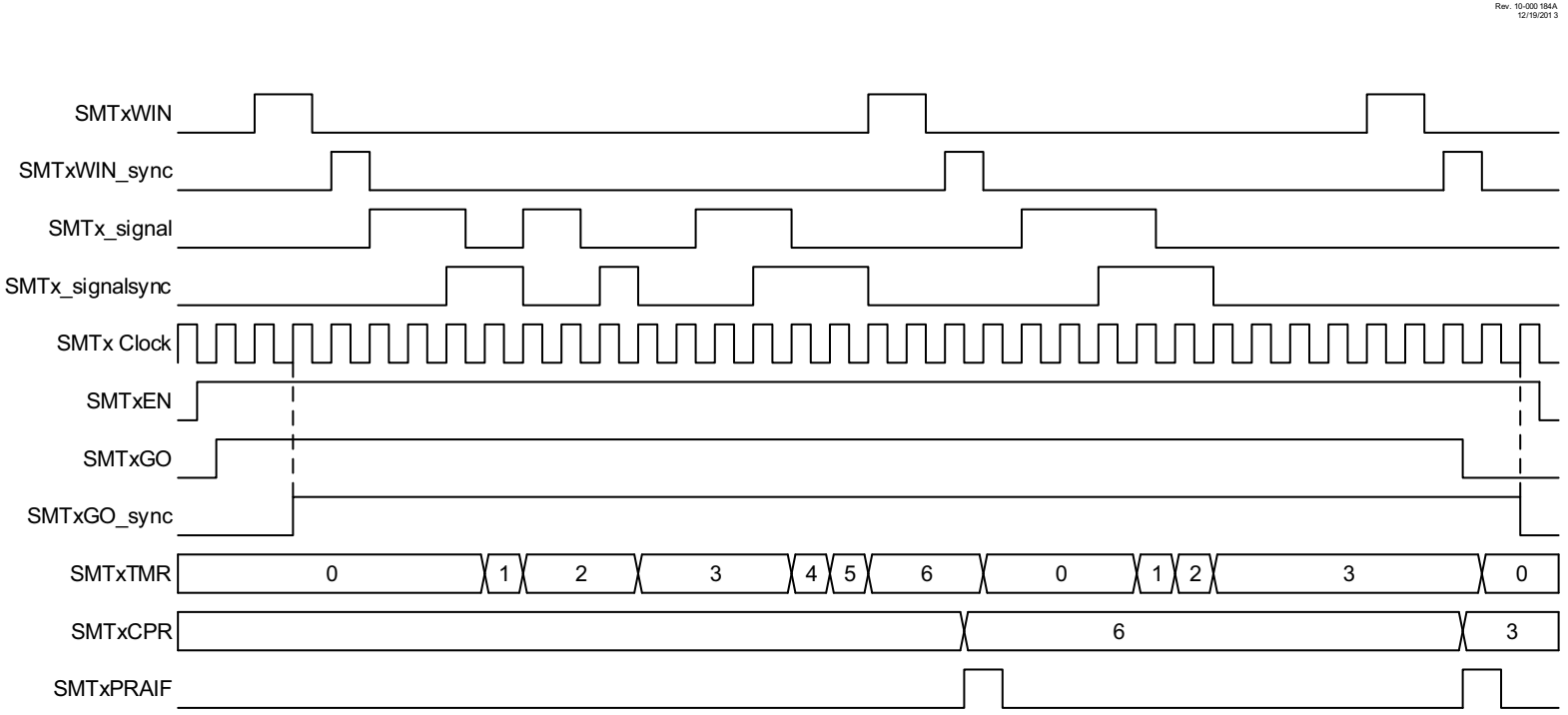


FIGURE 32-17: CAPTURE MODE SINGLE ACQUISITION TIMING DIAGRAM

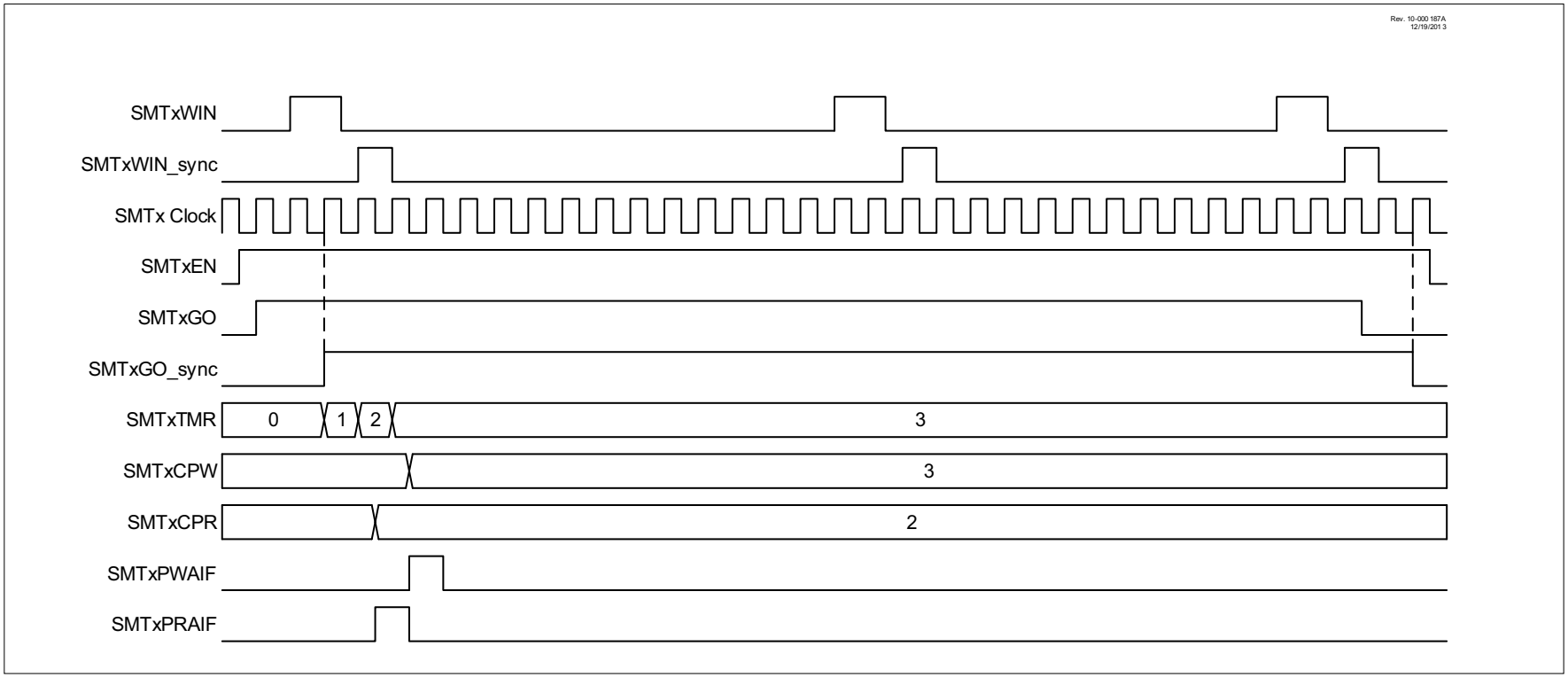


TABLE 32-3: SUMMARY OF REGISTERS ASSOCIATED WITH SMTx

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PIE8	—	—	SMT2PWAIE	SMT2PRAIE	SMT2IE	SMT1PWAIE	SMT1PRAIE	SMT1IE	123
PIR8	—	—	SMT2PWAIF	SMT2PRAIF	SMT2IF	SMT1PWAIF	SMT1PRAIF	SMT1IF	133
SMT1TMRL	SMT1TMR<7:0>								515
SMT1TMRH	SMT1TMR<15:8>								515
SMT1TMRU	SMT1TMR<23:16>								515
SMT1CPRL	SMT1CPR<7:0>								516
SMT1CPRH	SMT1CPR<15:8>								516
SMT1CPRU	SMT1CPR<23:16>								516
SMT1CPWL	SMT1CPW<7:0>								517
SMT1CPWH	SMT1CPW<15:8>								517
SMT1CPWU	SMT1CPW<23:16>								517
SMT1PRL	SMT1PR<7:0>								518
SMT1PRH	SMT1PR<15:8>								518
SMT1PRU	SMT1PR<23:16>								518
SMT1CON0	EN	—	STP	WPOL	SPOL	CPOL	SMT1PS<1:0>		509
SMT1CON1	SMT1GO	REPEAT	—	—	MODE<3:0>				510
SMT1STAT	CPRUP	CPWUP	RST	—	—	TS	WS	AS	511
SMT1CLK	—	—	—	—	—	CSEL<2:0>			512
SMT1SIG	—	—	—	SSEL<4:0>					514
SMT1WIN	—	—	—	WSEL<4:0>					513
SMT2TMRL	SMT2TMR<7:0>								515
SMT2TMRH	SMT2TMR<15:8>								515
SMT2TMRU	SMT2TMR<23:16>								515
SMT2CPRL	SMT2CPR<7:0>								516
SMT2CPRH	SMT2CPR<15:8>								516
SMT2CPRU	SMT2CPR<23:16>								516
SMT2CPWL	SMT2CPW<7:0>								517
SMT2CPWH	SMT2CPW<15:8>								517
SMT2CPWU	SMT2CPW<23:16>								517
SMT2PRL	SMT2PR<7:0>								518
SMT2PRH	SMT2PR<15:8>								518
SMT2PRU	SMT2PR<23:16>								518
SMT2CON0	EN	—	STP	WPOL	SPOL	CPOL	SMT2PS<1:0>		509
SMT2CON1	SMT2GO	REPEAT	—	—	MODE<3:0>				510
SMT2STAT	CPRUP	CPWUP	RST	—	—	TS	WS	AS	511
SMT2CLK	—	—	—	—	—	CSEL<2:0>			512
SMT2SIG	—	—	—	SSEL<4:0>					514
SMT2WIN	—	—	—	WSEL<4:0>					513

Legend: – = unimplemented read as '0'. Shaded cells are not used for SMTx module.

THE MICROCHIP WEBSITE

Microchip provides online support via our website at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip website at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the website at: http://www.microchip.com/support