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	12
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1614-i-sl

TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 4											
20Ch	WPUA	—	—	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	--11 1111	--11 1111
20Dh	WPUB(4)	WPUB7	WPUB6	WPUB5	WPUB4	—	—	—	—	1111 ----	1111 ----
20Eh	WPUC	WPUC7(4)	WPUC6(4)	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	1111 1111	111 1111
20Fh	—	Unimplemented								—	—
210h	—	Unimplemented								—	—
211h	SSP1BUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	xxxx xxxx
212h	SSP1ADD	ADD<7:0>								0000 0000	0000 0000
213h	SSP1MSK	MSK<7:0>								1111 1111	1111 1111
214h	SSP1STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
215h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000
216h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
217h	SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
218h to 21Fh	—	Unimplemented								—	—

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

Note 1: PIC16F1614/8 only.

2: Unimplemented, read as '1'.

3: PIC16(L)F1614 only.

4: PIC16(L)F1618 only.

TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 17											
88Ch	AT1CLK	—	—	—	—	—	—	—	CS0	---- ---0	---- ---0
88Dh	AT1SIG	—	—	—	—	—	SSEL<2:0>			---- -000	---- -000
88Eh	AT1CSEL1	—	—	—	—	—	CP1S<2:0>			---- -000	---- -000
88Fh	AT1CC1L	CC1<7:0>								0000 0000	0000 0000
890h	AT1CC1H	—	—	—	—	—	—	CC1<9:8>		---- -000	---- -000
891h	AT1CCON1	CC1EN	—	—	CC1POL	CAP1P	—	—	CC1MODE	0--0 0--0	0--0 0--0
892h	AT1CSEL2	—	—	—	—	—	CP2S<2:0>			---- -000	---- -000
893h	AT1CC2L	CC2<7:0>								0000 0000	0000 0000
894h	AT1CC2H	—	—	—	—	—	—	CC2<9:8>		---- -000	---- -000
895h	AT1CCON2	CC2EN	—	—	CC2POL	CAP2P	—	—	CC2MODE	0--0 0--0	0--0 0--0
896h	AT1CSEL3	—	—	—	—	—	CP3S<2:0>			---- -000	---- -000
897h	AT1CC1L	CC3<7:0>								0000 0000	0000 0000
898h	AT1CC1H	—	—	—	—	—	—	CC3<9:8>		---- -000	---- -000
899h	AT1CCON1	CC3EN	—	—	CC3POL	CAP3P	—	—	CC3MODE	0--0 0--0	0--0 0--0
89Ah to 89Fh	—	Unimplemented								—	—
Bank 18-26											
x0Ch/ x8Ch — x1Fh/ x9Fh	—	Unimplemented								—	—

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

Note 1: PIC16F1614/8 only.

2: Unimplemented, read as '1'.

3: PIC16(L)F1614 only.

4: PIC16(L)F1618 only.

TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Banks 27											
D80h to D8Bh	—	Unimplemented								—	—
D8Ch	SMT1TMRL	SMT1TMR<7:0>								0000 0000	0000 0000
D8Dh	SMT1TMRH	SMT1TMR<15:8>								0000 0000	0000 0000
D8Eh	SMT1TMRU	SMT1TMR<23:16>								0000 0000	0000 0000
D8Fh	SMT1CPRL	SMT1CPR<7:0>								xxxx xxxx	xxxx xxxx
D90h	SMT1CPRH	SMT1CPR<15:8>								xxxx xxxx	xxxx xxxx
D91h	SMT1CPRU	SMT1CPR<23:16>								xxxx xxxx	xxxx xxxx
D92h	SMT1CPWL	SMT1CPW<7:0>								xxxx xxxx	xxxx xxxx
D93h	SMT1CPWH	SMT1CPW<15:8>								xxxx xxxx	xxxx xxxx
D94h	SMT1CPWU	SMT1CPW<23:16>								xxxx xxxx	xxxx xxxx
D95h	SMT1PRL	SMT1PR<7:0>								xxxx xxxx	xxxx xxxx
D96h	SMT1PRH	SMT1PR<15:8>								xxxx xxxx	xxxx xxxx
D97h	SMT1PRU	SMT1PR<23:16>								xxxx xxxx	xxxx xxxx
D98h	SMT1CON0	EN	—	STP	WPOL	SPOL	CPOL	SMT1PS<1:0>		0-00 0000	0-00 0000
D99h	SMT1CON1	SMT1GO	REPEAT	—	—	MODE<3:0>				00-- 0000	00-- 0000
D9Ah	SMT1STAT	CPRUP	CPWUP	RST	—	—	TS	WS	AS	000- -000	000- -000
D9Bh	SMT1CLK	—	—	—	—	—	CSEL<2:0>			---- -000	---- -000
D9Ch	SMT1SIG	—	—	—	SSEL<4:0>					---0 0000	---0 0000
D9Dh	SMT1WIN	—	—	—	WSEL<4:0>					---0 0000	---0 0000
D9Eh	SMT2TMRL	SMT2TMR<7:0>								0000 0000	0000 0000
D9Fh	SMT2TMRH	SMT2TMR<15:8>								0000 0000	0000 0000
DA0h	SMT2TMRU	SMT2TMR<23:16>								0000 0000	0000 0000
DA1h	SMT2CPRL	SMT2CPR<7:0>								xxxx xxxx	xxxx xxxx
DA2h	SMT2CPRH	SMT2CPR<15:8>								xxxx xxxx	xxxx xxxx
DA3h	SMT2CPRU	SMT2CPR<23:16>								xxxx xxxx	xxxx xxxx
DA4h	SMT2CPWL	SMT2CPW<7:0>								xxxx xxxx	xxxx xxxx

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

Note 1: PIC16F1614/8 only.
2: Unimplemented, read as '1'.
3: PIC16(L)F1614 only.
4: PIC16(L)F1618 only.

TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Banks 28											
E0Ch to E0Eh	—	Unimplemented								—	—
E0Fh	PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	---- ---0	---- ---0
E10h	INTPPS	—	—	—	INTPPS<4:0>					---0 0010	---0 0010
E11h	T0CKIPPS	—	—	—	T0CKIPPS<4:0>					---0 0010	---0 0010
E12h	T1CKIPPS	—	—	—	T1CKIPPS<4:0>					---0 0101	---0 0101
E13h	T1GPPS	—	—	—	T1GPPS<4:0>					---0 0100	---0 0100
E14h	CCP1PPS	—	—	—	CCP1PPS<4:0>					---1 0101	---1 0101
E15h	CCP2PPS	—	—	—	CCP2PPS<4:0>					---1 0011	---1 0011
E16h	ATINPPS	—	—	—	ATINPPS<4:0>					---1 0101	---1 0101
E17h	CWGINPPS	—	—	—	CWGINPPS<4:0>					---0 0010	---0 0010
E18h	T2PPS	—	—	—	T2PPS<4:0>					---0 0101	---0 0101
E19h	T3CKIPPS	—	—	—	T3CKIPPS<4:0>					---1 0101	---1 0101
E1Ah	T3GPPS	—	—	—	T3GPPS<4:0>					---1 0100	---1 0100
E1Bh	T4PPS	—	—	—	T4PPS<4:0>					---1 0001	---1 0001
E1Ch	T5CKIPPS	—	—	—	T5CKIPPS<4:0>					---1 0000	---1 0000
E1Dh	T5GPPS	—	—	—	T5GPPS<4:0>					---1 0011	---1 0011
E1Eh	T6PPS	—	—	—	T6PPS<4:0>					---0 0011	---0 0011
E1Fh	ATCC1PPS	—	—	—	ATCC1PPS<4:0>					---1 0011	---1 0011
E20h	SSPCLKPPS(3)	—	—	—	SSPCLKPPS<4:0>					---1 0000	---1 0000
E20h	SSPCLKPPS(4)	—	—	—	SSPCLKPPS<4:0>					---1 0000	---0 1110
E21h	SSPDATPPS(3)	—	—	—	SSPDATPPS<4:0>					---1 0001	---1 0001
E21h	SSPDATPPS(4)	—	—	—	SSPDATPPS<4:0>					---1 0001	---0 1100
E22h	SSPSSPPS(3)	—	—	—	SSPSSPPS<4:0>					---1 0011	---1 0011
E22h	SSPSSPPS(4)	—	—	—	SSPSSPPS<4:0>					---1 0110	---1 0110
E23h	ATCC2PPS	—	—	—	ATCC2PPS<4:0>					---1 0100	---1 0100

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

Note 1: PIC16F1614/8 only.

2: Unimplemented, read as '1'.

3: PIC16(L)F1614 only.

4: PIC16(L)F1618 only.

4.6 Device ID and Revision ID

The 14-bit Device ID word is located at 8006h and the 14-bit Revision ID is located at 8005h. These locations are read-only and cannot be erased or modified. See **Section 10.4 "User ID, Device ID and Configuration Word Access"** for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

4.7 Register Definitions: Device ID

REGISTER 4-4: DEVID: DEVICE ID REGISTER

R	R	R	R	R	R
DEV<13:8>					
bit 13					bit 8

R	R	R	R	R	R	R	R
DEV<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit		
'1' = Bit is set	'0' = Bit is cleared	

bit 13-0 **DEV<13:0>:** Device ID bits

Device	DEVID<13:0> Values
PIC16F1614	11 0000 0111 1000 (3078h)
PIC16LF1614	11 0000 0111 1010 (307Ah)
PIC16F1618	11 0000 0111 1001 (3079h)
PIC16LF1618	11 0000 0111 1011 (307Bh)

REGISTER 4-5: REVID: REVISION ID REGISTER

R	R	R	R	R	R
REV<13:8>					
bit 13					bit 8

R	R	R	R	R	R	R	R
REV<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit		
'1' = Bit is set	'0' = Bit is cleared	

bit 13-0 **REV<13:0>:** Revision ID bits

6.1 Power-On Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

6.1.1 POWER-UP TIMER (PWRT)

The Power-up Timer provides a nominal 64 ms time-out on POR or Brown-out Reset.

The device is held in Reset as long as PWRT is active. The PWRT delay allows additional time for the VDD to rise to an acceptable level. The Power-up Timer is enabled by clearing the PWRTE bit in Configuration Words.

The Power-up Timer starts after the release of the POR and BOR.

For additional information, refer to Application Note AN607, *"Power-up Trouble Shooting"* (DS00607).

6.2 Brown-Out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- BOR is always on
- BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 6-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV bit in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Figure 6-2 for more information.

TABLE 6-1: BOR OPERATING MODES

BOREN<1:0>	SBOREN	Device Mode	BOR Mode	Instruction Execution upon: Release of POR or Wake-up from Sleep
11	X	X	Active	Waits for BOR ready(1) (BORRDY = 1)
10	X	Awake	Active	Waits for BOR ready (BORRDY = 1)
		Sleep	Disabled	
01	1	X	Active	Waits for BOR ready(1) (BORRDY = 1)
	0	X	Disabled	Begins immediately (BORRDY = x)
00	X	X	Disabled	

Note 1: In these specific cases, "release of POR" and "wake-up from Sleep," there is no delay in start-up. The BOR ready flag, (BORRDY = 1), will be set before the CPU is ready to execute instructions because the BOR circuit is forced on by the BOREN<1:0> bits.

6.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

6.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

6.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

BOR protection is unchanged by Sleep.

REGISTER 11-14: SCANHADRH: SCAN HIGH ADDRESS HIGH BYTE 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
HADR<15:8>(1, 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 **HADR<15:8>**: Scan End Address bits(1, 2)
Most Significant bits of the address at the end of the designated scan

Note 1: Registers SCANHADRH/L form a 16-bit value, but are not guarded for atomic or asynchronous access; registers should only be read or written while SCANGO = 0 (SCANCON0 register).
2: While SCANGO = 1 (SCANCON0 register), writing to this register is ignored.

REGISTER 11-15: SCANHADRL: SCAN HIGH ADDRESS LOW BYTE 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
HADR<7:0>(1, 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 **HADR<7:0>**: Scan End Address bits(1, 2)
Least Significant bits of the address at the end of the designated scan

Note 1: Registers SCANHADRH/L form a 16-bit value, but are not guarded for atomic or asynchronous access; registers should only be read or written while SCANGO = 0 (SCANCON0 register).
2: While SCANGO = 1 (SCANCON0 register), writing to this register is ignored.

TABLE 13-2: AVAILABLE PORTS FOR OUTPUT BY PERIPHERAL(2)

RxyPPS<4:0>	Output Signal	PIC16(L)F1618			PIC16(L)F1614	
		PORTA	PORTB	PORTC	PORTA	PORTC
11xxx	Reserved	●	●	●	●	●
10111	Reserved	●	●	●	●	●
10110	Reserved	●	●	●	●	●
10101	Reserved	●	●	●	●	●
10100	Reserved	●	●	●	●	●
10011	DT	●	●	●	●	●
10010	TX/CK	●	●	●	●	●
10001	SDO/SDA(1)	●	●	●	●	●
10000	SCK/SCL(1)	●	●	●	●	●
01111	PWM4_out	●	●	●	●	●
01110	PWM3_out	●	●	●	●	●
01101	CCP2_out	●	●	●	●	●
01100	CCP1_out	●	●	●	●	●
01011	CWG1OUTD(1)	●	●	●	●	●
01010	CWG1OUTC(1)	●	●	●	●	●
01001	CWG1OUTB(1)	●	●	●	●	●
01000	CWG1OUTA(1)	●	●	●	●	●
00111	LC4_out	●	●	●	●	●
00110	LC3_out	●	●	●	●	●
00101	LC2_out	●	●	●	●	●
00100	LC1_out	●	●	●	●	●
00011	ZCD1_out	●	●	●	●	●
00010	sync_C2OUT	●	●	●	●	●
00001	sync_C1OUT	●	●	●	●	●
00000	LATxy	●	●	●	●	●

Note 1: TRIS control is overridden by the peripheral as required.

2: Unsupported peripherals will output a '0'.

19.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The analog comparator module includes the following features:

- Independent comparator control
- Programmable input selection
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- Programmable Speed/Power optimization
- PWM shutdown
- Programmable and Fixed Voltage Reference

19.1 Comparator Overview

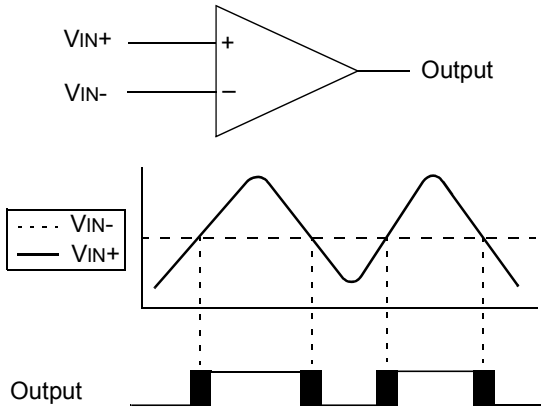
A single comparator is shown in Figure 19-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

The comparators available for this device are located in Table 19-1.

TABLE 19-1: COMPARATOR AVAILABILITY PER DEVICE

Device	C1	C2
PIC16(L)F1618	●	●
PIC16(L)F1614	●	●

FIGURE 19-1: SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

23.0 TIMER2/4/6 MODULE

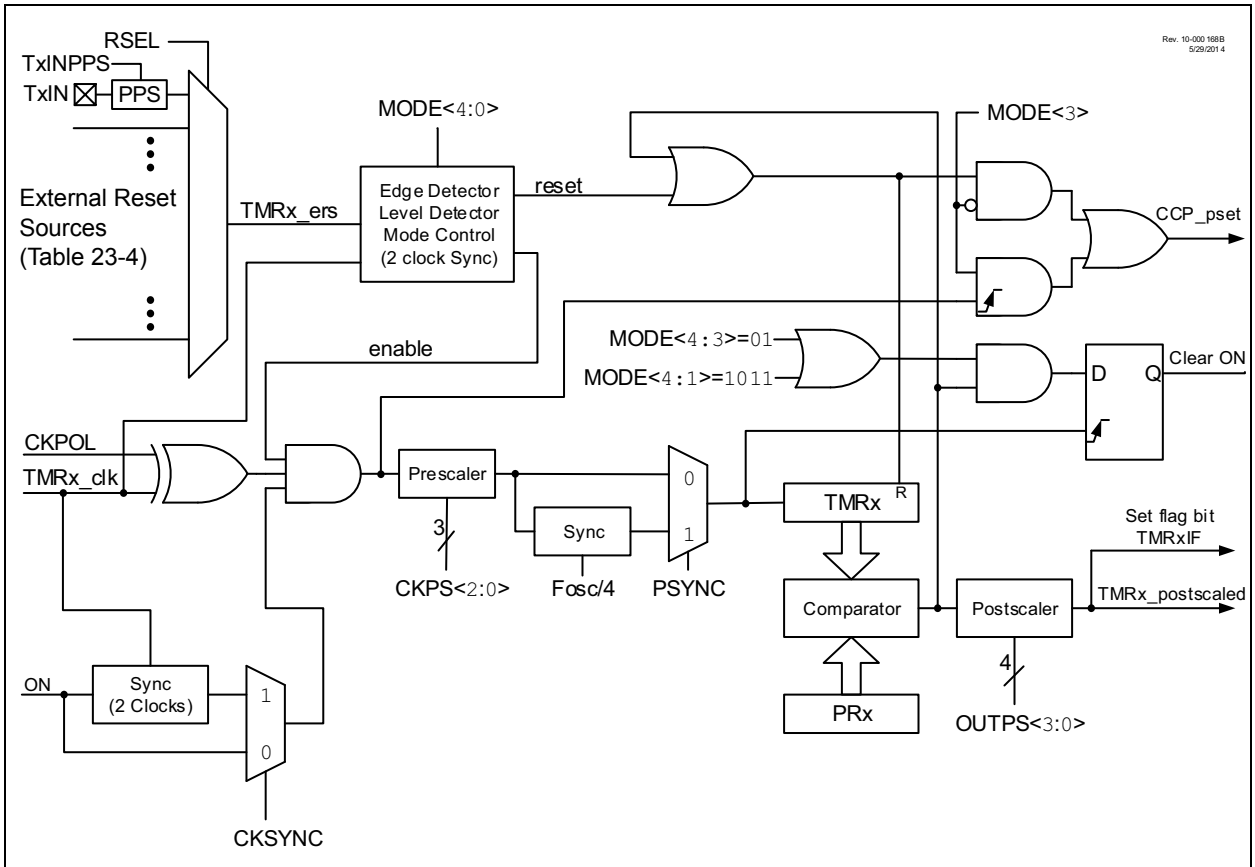
The Timer2/4/6 modules are 8-bit timers that can operate as free-running period counters or in conjunction with external signals that control start, run, freeze, and reset operation in One-Shot and Monostable modes of operation. Sophisticated waveform control such as pulse density modulation are possible 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 23-1 for a block diagram of Timer2. See Figure 23-2 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 23-1: TIMER2 BLOCK DIAGRAM



Note 1: Signal to the CCP to trigger the PWM pulse
2: See Section 22.5 for description of CCP interaction in the different TMR modes

24.4.5 START CONDITION

The I^2C specification defines a Start condition as a transition of SDA from a high to a low state while SCL line is high. A Start condition is always generated by the master and signifies the transition of the bus from an Idle to an Active state. Figure 24-12 shows wave forms for Start and Stop conditions.

A bus collision can occur on a Start condition if the module samples the SDA line low before asserting it low. This does not conform to the I^2C Specification that states no bus collision can occur on a Start.

24.4.6 STOP CONDITION

A Stop condition is a transition of the SDA line from low-to-high state while the SCL line is high.

Note: At least one SCL low time must appear before a Stop is valid, therefore, if the SDA line goes low then high again while the SCL line stays high, only the Start condition is detected.

24.4.7 RESTART CONDITION

A Restart is valid any time that a Stop would be valid. A master can issue a Restart if it wishes to hold the bus after terminating the current transfer. A Restart has the same effect on the slave that a Start would, resetting all slave logic and preparing it to clock in an address. The master may want to address the same or another slave. Figure 24-13 shows the wave form for a Restart condition.

In 10-bit Addressing Slave mode a Restart is required for the master to clock data out of the addressed slave. Once a slave has been fully addressed, matching both high and low address bytes, the master can issue a Restart and the high address byte with the R/$\overline{W}$ bit set. The slave logic will then hold the clock and prepare to clock out data.

After a full match with R/$\overline{W}$ clear in 10-bit mode, a prior match flag is set and maintained until a Stop condition, a high address with R/$\overline{W}$ clear, or high address match fails.

24.4.8 START/STOP CONDITION INTERRUPT MASKING

The SCIE and PCIE bits of the SSPxCON3 register can enable the generation of an interrupt in Slave modes that do not typically support this function. Slave modes where interrupt on Start and Stop detect are already enabled, these bits will have no effect.

FIGURE 24-12: I^2C START AND STOP CONDITIONS

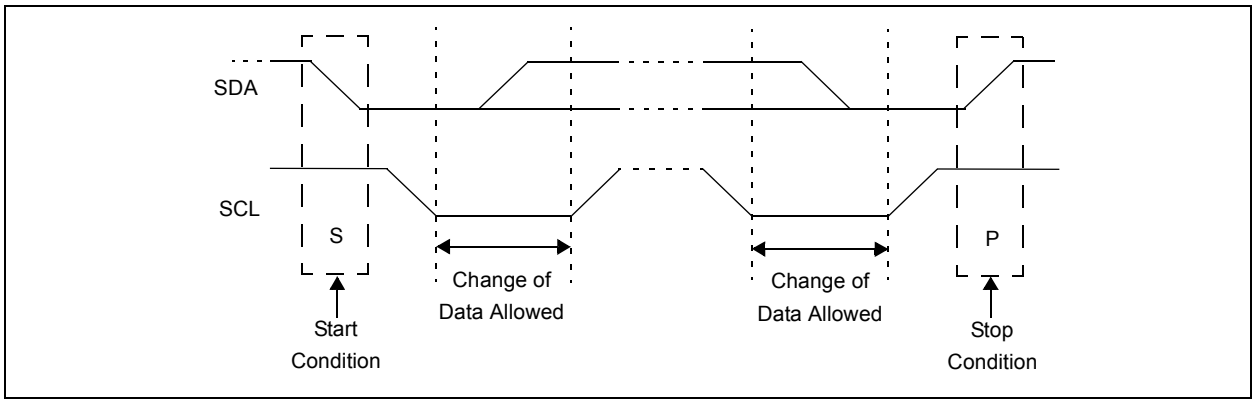


FIGURE 24-13: I^2C RESTART CONDITION

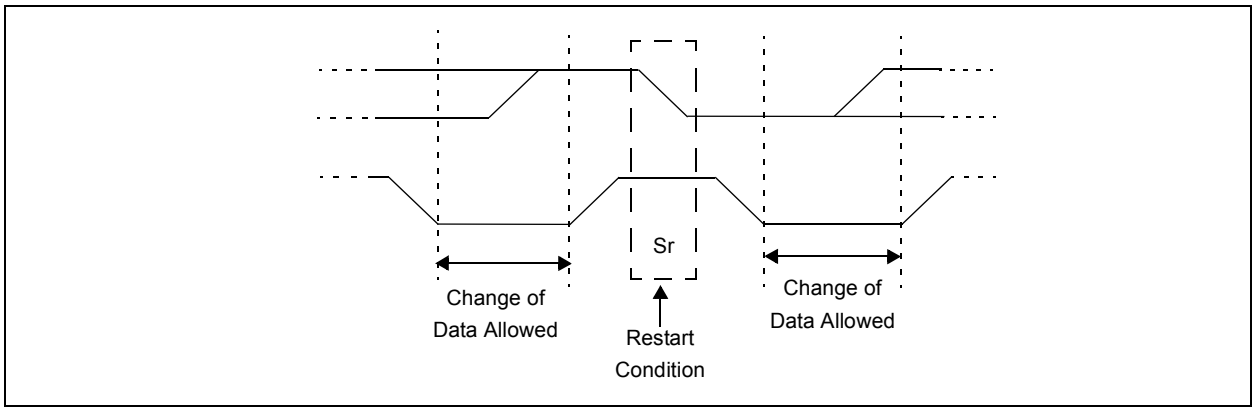


TABLE 26-1: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6

TABLE 26-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale	16	4	1	1	1	1
PR2 Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

26.4.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section5.0 “Oscillator Module”** for additional details.

26.4.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

26.4.7 PWM OUTPUT

The output of the CCP in PWM mode is the PWM signal generated by the module and described above. This output is available as an input signal to the CWG, as an auto-conversion trigger for the ADC, as an external Reset signal for the TMR2 modules, as a window input to the SMT, and as an input to the CLC module. In addition, the CCPx pin output can be mapped to output pins through the use of PPS (see **Section13.2 “PPS Outputs”**).

27.0 PULSE-WIDTH MODULATION (PWM) MODULE

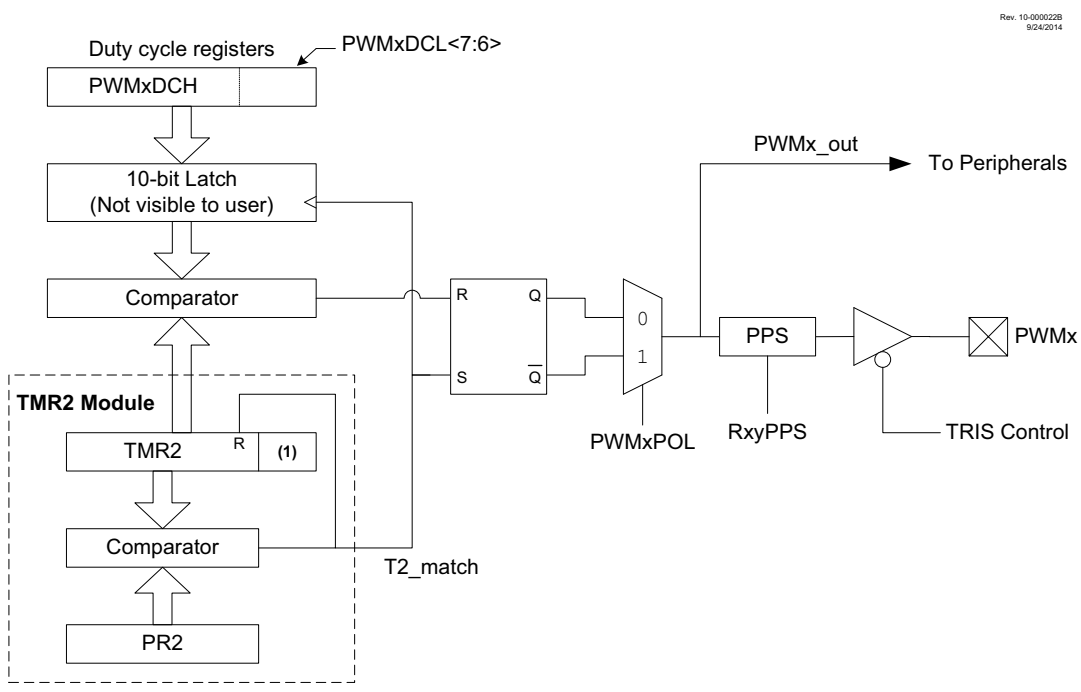
The PWM module generates a Pulse-Width Modulated signal determined by the duty cycle, period, and resolution that are configured by the following registers:

- PR2
- T2CON
- PWMxDCH
- PWMxDCL
- PWMxCON

Figure 27-1 shows a simplified block diagram of PWM operation.

For a step-by-step procedure on how to set up this module for PWM operation, refer to **Section 27.1.9 "Setup for PWM Operation using PWMx Pins"**.

FIGURE 27-1: SIMPLIFIED PWM BLOCK DIAGRAM



Note 1: 8-bit timer is concatenated with two bits generated by Fosc or two bits of the internal prescaler to create 10-bit time-base.

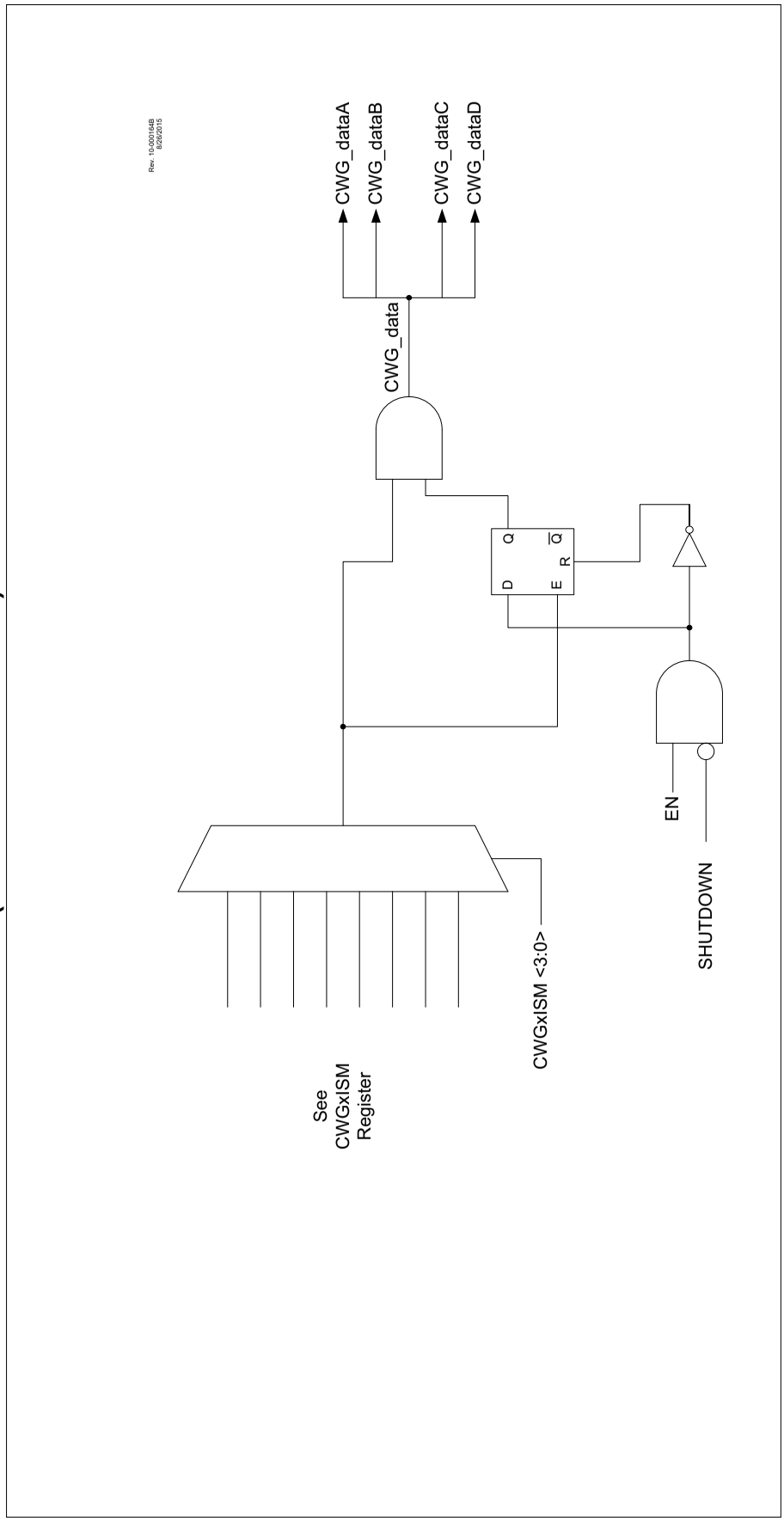


FIGURE 28-4: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)

FIGURE 28-13: SHUTDOWN FUNCTIONALITY, AUTO-RESTART DISABLED (REN = 0, LSAC = 01, LSBD = 01)

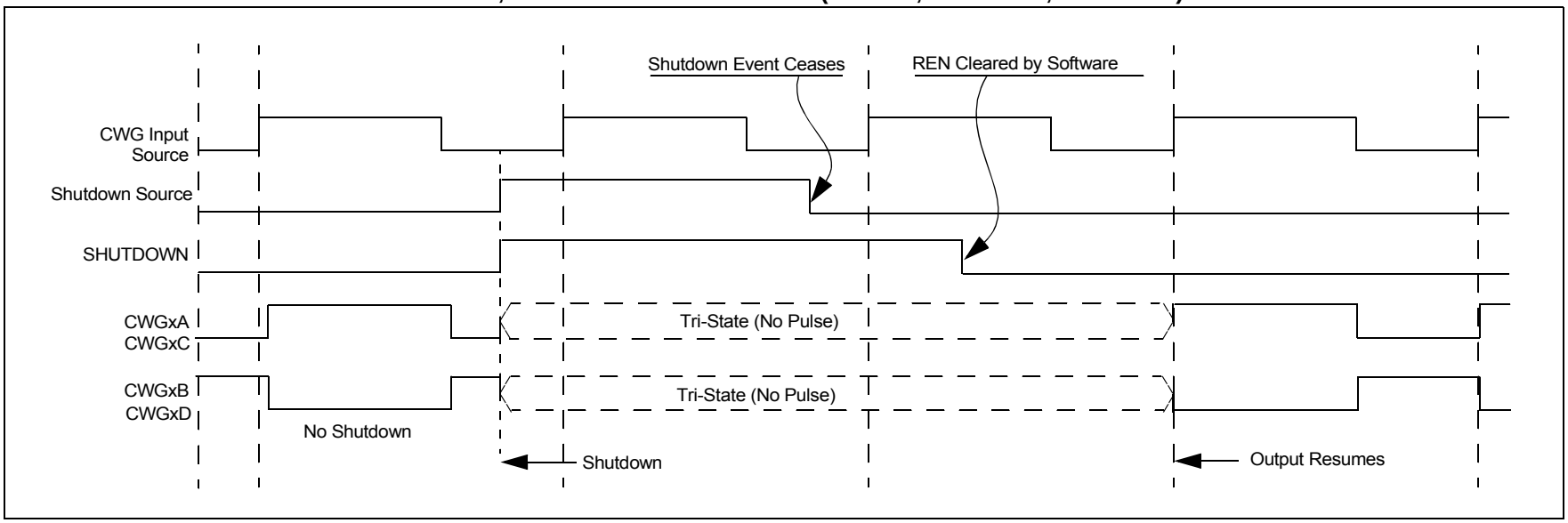


FIGURE 28-14: SHUTDOWN FUNCTIONALITY, AUTO-RESTART ENABLED (REN = 1, LSAC = 01, LSBD = 01)

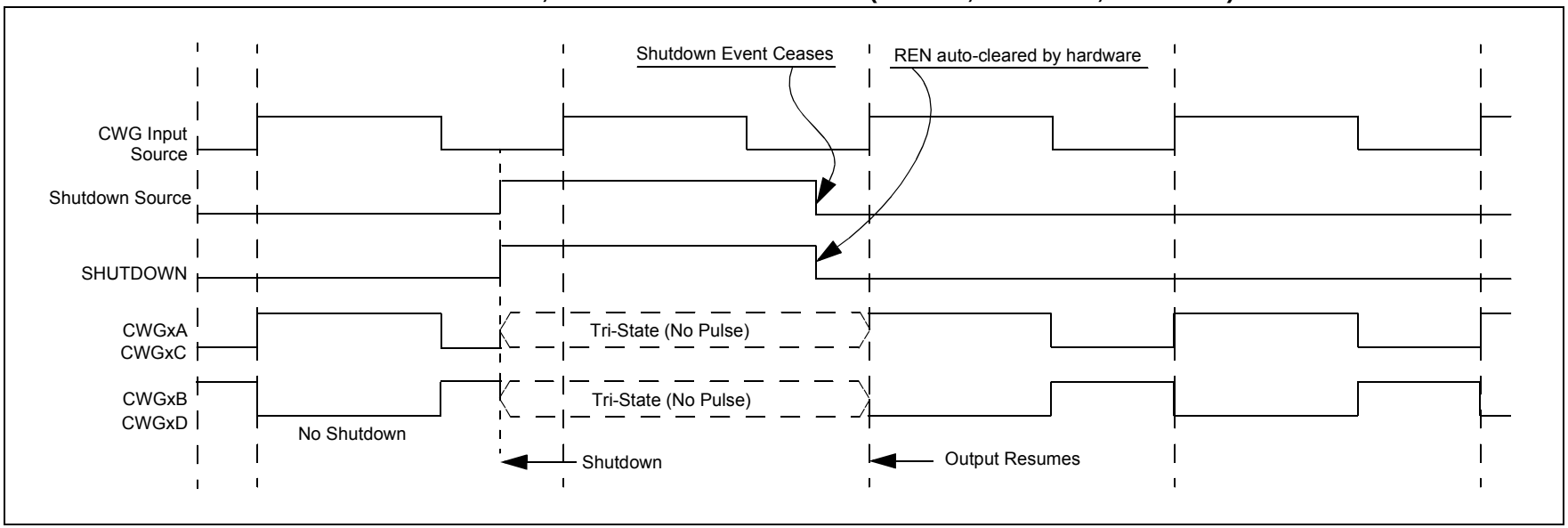
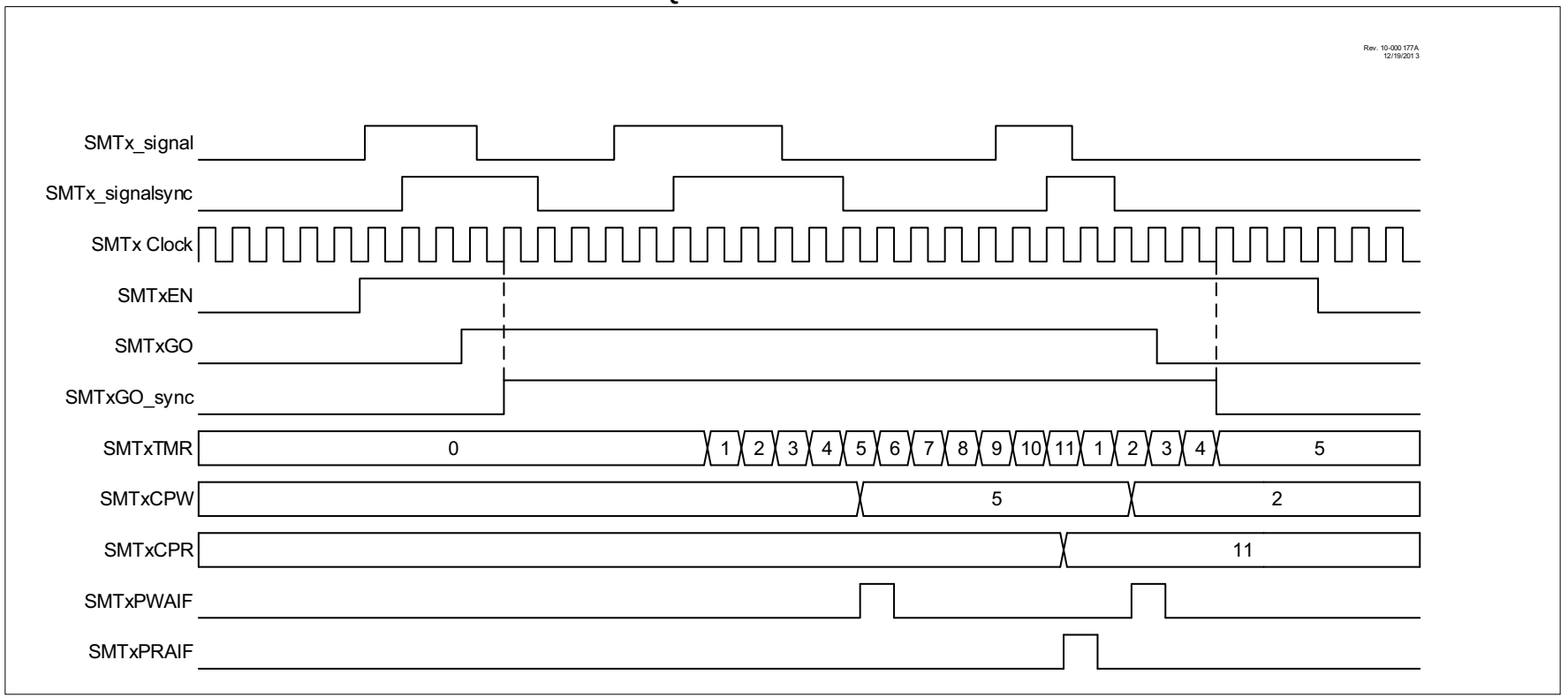


FIGURE 30-6: PERIOD AND DUTY-CYCLE REPEAT ACQUISITION MODE TIMING DIAGRAM



30.6.7 TIME OF FLIGHT MEASURE MODE

This mode measures the time interval between a rising edge on the SMTWINx input and a rising edge on the SMTx_signal input, beginning to increment the timer upon observing a rising edge on the SMTWINx input, while updating the SMTxCPR register and resetting the timer upon observing a rising edge on the SMTx_signal input. In the event of two SMTWINx rising edges without an SMTx_signal rising edge, it will update the SMTxCPW register with the current value of the timer and reset the timer value. See Figure 30-14 and Figure 30-15.

REGISTER 30-8: SMT2SIG: SMT2 SIGNAL INPUT SELECT REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	SSEL<4:0>				
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 7-5 **Unimplemented**: Read as ‘0’

bit 4-0 **SSEL<4:0>:** SMT2 Signal Selection bits
11111 = Reserved
•
•
•
10101 = Reserved
10100 = PWM4_out
10011 = PWM3_out
10010 = CCP2_out
10001 = CCP1_out
10000 = TMR0_overflow
01111 = Reserved
01110 = SMT1_match
01101 = TMR5_overflow
01100 = TMR3_overflow
01011 = TMR1_overflow
01010 = Reserved
01001 = Reserved
01000 = LC2_out
00111 = LC1_out
00110 = TMR6_postscaled
00101 = TMR4_postscaled
00100 = TMR2_postscaled
00011 = ZCD1_out
00010 = C2OUT_sync
00001 = C1OUT_sync
00000 = SMTxSIG pin

MOVF **Move f**

Syntax: [*label*] MOVF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) → (dest)

Status Affected: Z

Description: The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example:

```
MOVF    FSR, 0
```

After Instruction
W = value in FSR register
Z = 1

MOVIW **Move INDFn to W**

Syntax:
[*label*] MOVIW ++FSRn
[*label*] MOVIW --FSRn
[*label*] MOVIW FSRn++
[*label*] MOVIW FSRn--
[*label*] MOVIW k[FSRn]

Operands: $n \in [0,1]$
$mm \in [00,01, 10, 11]$
$-32 \le k \le 31$

Operation: INDFn → W
Effective address is determined by
- FSR + 1 (preincrement)
- FSR - 1 (predecrement)
- FSR + k (relative offset)

After the Move, the FSR value will be either:
- FSR + 1 (all increments)
- FSR - 1 (all decrements)
- Unchanged

Status Affected: Z

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	--FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn--	11

Description: This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

Note: The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h - FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.

MOVLB **Move literal to BSR**

Syntax: [*label*] MOVLB k

Operands: $0 \le k \le 31$

Operation: k → BSR

Status Affected: None

Description: The 5-bit literal ‘k’ is loaded into the Bank Select Register (BSR).

Note: Unless otherwise noted, VIN = 5V, FOSC = 500 kHz, CIN = 0.1 µF, TA = 25°C.

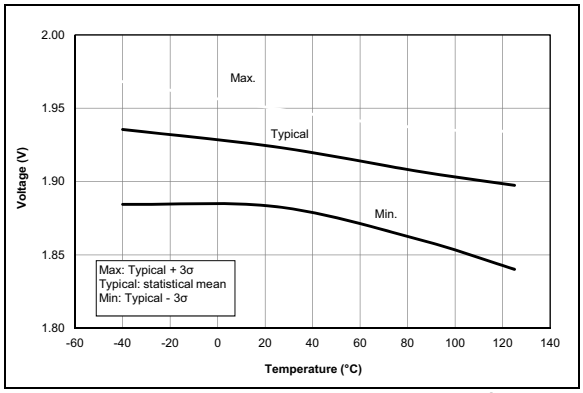


FIGURE 36-49: *Brown-Out Reset Voltage, Low Trip Point (BORV = 1), PIC16LF1614/8 Only.*

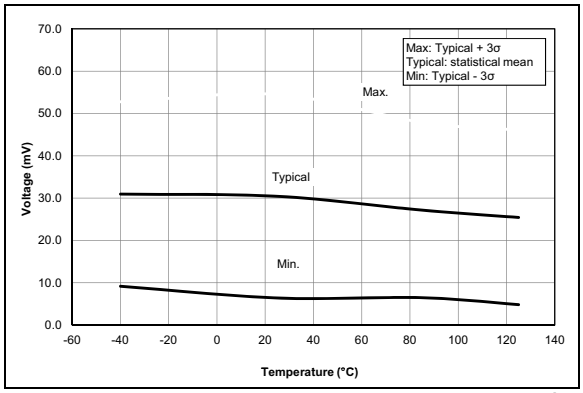


FIGURE 36-52: *Brown-Out Reset Hysteresis, Low Trip Point (BORV = 1), PIC16F1614/8 Only.*

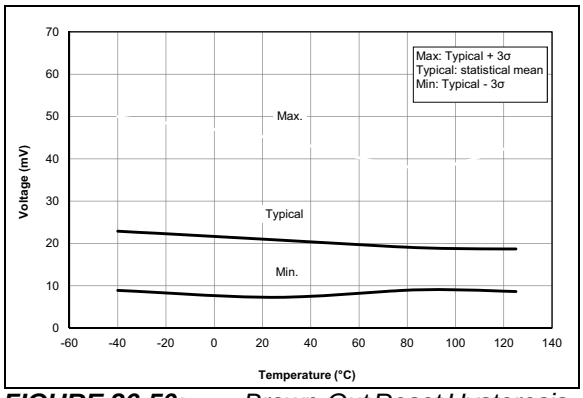


FIGURE 36-50: *Brown-Out Reset Hysteresis, Low Trip Point (BORV = 1), PIC16LF1614/8 Only.*

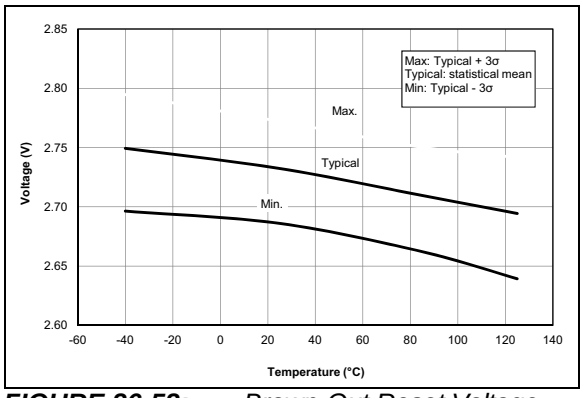


FIGURE 36-53: *Brown-Out Reset Voltage, High Trip Point (BORV = 0).*

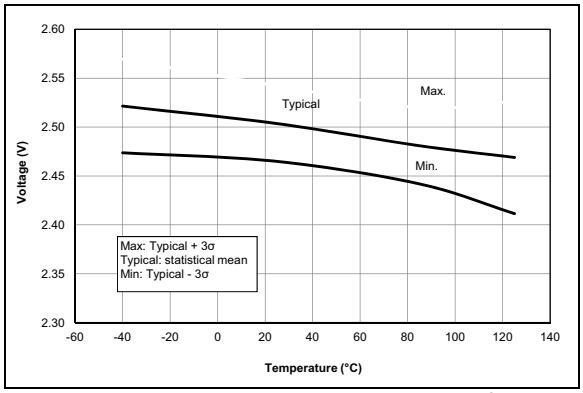


FIGURE 36-51: *Brown-Out Reset Voltage, Low Trip Point (BORV = 1), PIC16F1614/8 Only.*

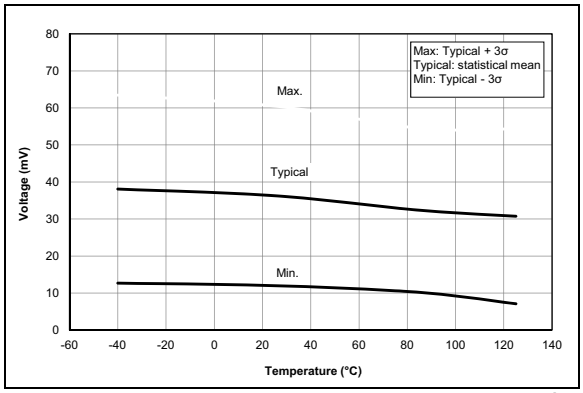


FIGURE 36-54: *Brown-Out Reset Hysteresis, High Trip Point (BORV = 0).*