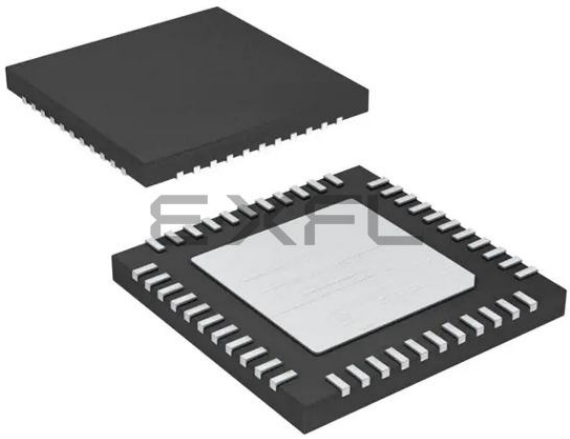


Welcome to E-XFL.COM

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	36
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 35x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18876-i-ml

TABLE 1-2: PIC16F18856 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC4/ANC4/SDA1(3,4)/SDI1(1)/IOCC4	RC4	TTL/ST	CMOS/OD	General purpose I/O.
	ANC4	AN	—	ADC Channel C4 input.
	SDA1(3,4)	I2C/ SMBus	OD	MSSP1 I2C serial data input/output.
	SDI1(1)	TTL/ST	—	MSSP1 SPI serial data input.
	IOCC4	TTL/ST	—	Interrupt-on-change input.
RC5/ANC5/T4IN(1)/IOCC5	RC5	TTL/ST	CMOS/OD	General purpose I/O.
	ANC5	AN	—	ADC Channel C5 input.
	T4IN(1)	TTL/ST	—	Timer4 external input.
	IOCC5	TTL/ST	—	Interrupt-on-change input.
RC6/ANC6/CK(3)/IOCC6	RC6	TTL/ST	CMOS/OD	General purpose I/O.
	ANC6	AN	—	ADC Channel C6 input.
	CK(3)	TTL/ST	CMOS/OD	EUSART synchronous mode clock input/output.
	IOCC6	TTL/ST	—	Interrupt-on-change input.
RC7/ANC7/RX(1)/DT(3)/IOCC7	RC7	TTL/ST	CMOS/OD	General purpose I/O.
	ANC7	AN	—	ADC Channel C7 input.
	RX(1)	TTL/ST	—	EUSART Asynchronous mode receiver data input.
	DT(3)	TTL/ST	CMOS/OD	EUSART Synchronous mode data input/output.
	IOCC7	TTL/ST	—	Interrupt-on-change input.
RE3/IOCE3/$\overline{MCLR}$/VPP	RE3	TTL/ST	—	General purpose input only (when $\overline{MCLR}$ is disabled by the Configuration bit).
	IOCE3	TTL/ST	—	Interrupt-on-change input.
	$\overline{MCLR}$	ST	—	Master clear input with internal weak pull up resistor.
	VPP	HV	—	ICSP™ High-Voltage Programming mode entry input.
VDD	VDD	Power	—	Positive supply voltage 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 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.

TABLE 1-2: PIC16F18856 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.

TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18856	PIC16(L)F18876	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 29													
CPU CORE REGISTERS; see Table 3-2 for specifics													
E8Ch	—	—		Unimplemented								—	—
E8Dh	—	—		Unimplemented								—	—
E8Eh	—	—		Unimplemented								—	—
E8Fh	PPSLOCK			—	—	—	—	—	—	—	PPSLOCKED	---- ---0	---- ---0
E90h	INTPPS			—	—	—	—	INTPPS<3:0>				---- 1000	---- uuuu
E91h	T0CKIPPS			—	—	—	—	T0CKIPPS<3:0>				---- 0100	---- uuuu
E92h	T1CKIPPS			—	—	—	T1CKIPPS<4:0>					---1 0000	---u uuuu
E93h	T1GPPS			—	—	—	T1GPPS<4:0>					---0 1101	---u uuuu
E94h	T3CKIPPS			—	—	—	T3CKIPPS<4:0>					---1 0000	---u uuuu
E95h	T3GPPS			—	—	—	T3GPPS<4:0>					---1 0000	---u uuuu
E96h	T5CKIPPS			—	—	—	T5CKIPPS<4:0>					---1 0000	---u uuuu
E97h	T5GPPS			—	—	—	T5GPPS<4:0>					---0 1100	---u uuuu
E98h	—	—		Unimplemented								—	—
E99h	—	—		Unimplemented								—	—
E9Ah	—	—		Unimplemented								—	—
E9Bh	—	—		Unimplemented								—	—
E9Ch	T2AINPPS			—	—	—	T2AINPPS<4:0>					---1 0011	---u uuuu
E9Dh	T4AINPPS			—	—	—	T4AINPPS<4:0>					---1 0101	---u uuuu
E9Eh	T6AINPPS			—	—	—	T6AINPPS<4:0>					---0 1111	---u uuuu
E9Fh	—	—		Unimplemented								—	—
EA0h	—	—		Unimplemented								—	—
EA1h	CCP1PPS			—	—	—	CCP1PPS<4:0>					---1 0010	---u uuuu

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

Note 1: Register present on PIC16F18855/75 devices only.

2: Unimplemented, read as '1'.

FIGURE 3-5: ACCESSING THE STACK EXAMPLE 2

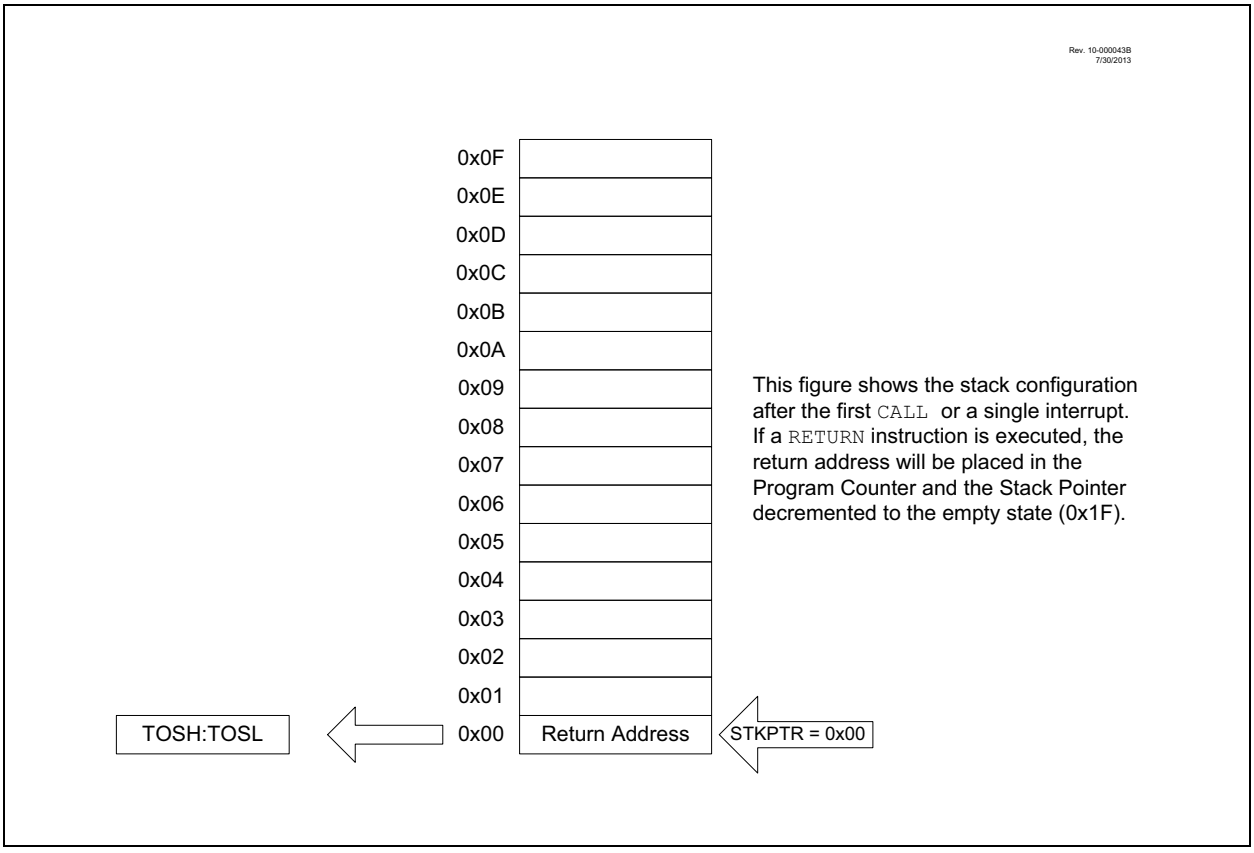
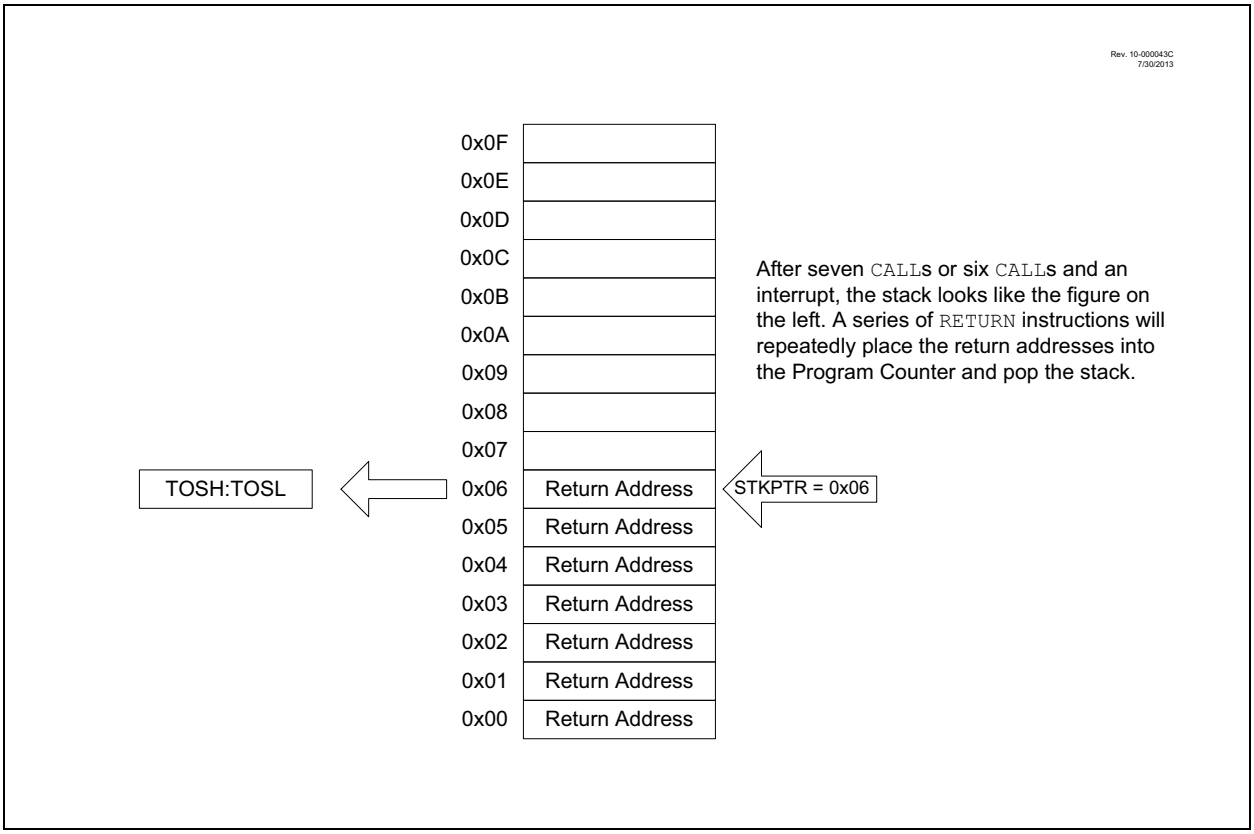


FIGURE 3-6: ACCESSING THE STACK EXAMPLE 3



10.4.2 NVM UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the NVM from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- PFM Row Erase
- Load of PFM write latches
- Write of PFM write latches to PFM memory
- Write of PFM write latches to User IDs
- Write to EEPROM

The unlock sequence consists of the following steps and must be completed in order:

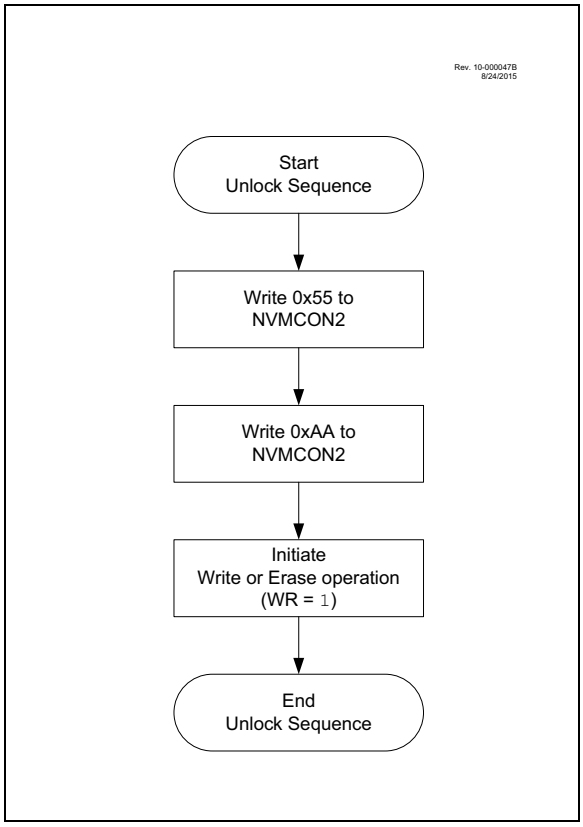
- Write 55h to NVMCON2
- Write AAh to NMVCON2
- Set the WR bit of NVMCON1

Once the WR bit is set, the processor will stall internal operations until the operation is complete and then resume with the next instruction.

Note:	The two NOP instructions after setting the WR bit that were required in previous devices are not required for PIC16(L)F18856/76 devices. See Figure 10-2.

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

FIGURE 10-2: NVM UNLOCK SEQUENCE FLOWCHART



EXAMPLE 10-2: NVM UNLOCK SEQUENCE

```
BCF      INTCON, GIE      ; Recommended so sequence is not interrupted
BANKSEL  NVMCON1          ;
BSF      NVMCON1, WREN    ; Enable write/erase
MOVLW    55h              ; Load 55h


MOVWF    NVMCON2          ; Step 1: Load 55h into NVMCON2
MOVLW    AAh              ; Step 2: Load W with AAh
MOVWF    NVMCON2          ; Step 3: Load AAH into NVMCON2
BSF      NVMCON1, WR      ; Step 4: Set WR bit to begin write/erase
BSF      INTCON, GIE      ; Re-enable interrupts
```

Note 1: Sequence begins when NVMCON2 is written; steps 1-4 must occur in the cycle-accurate order shown.

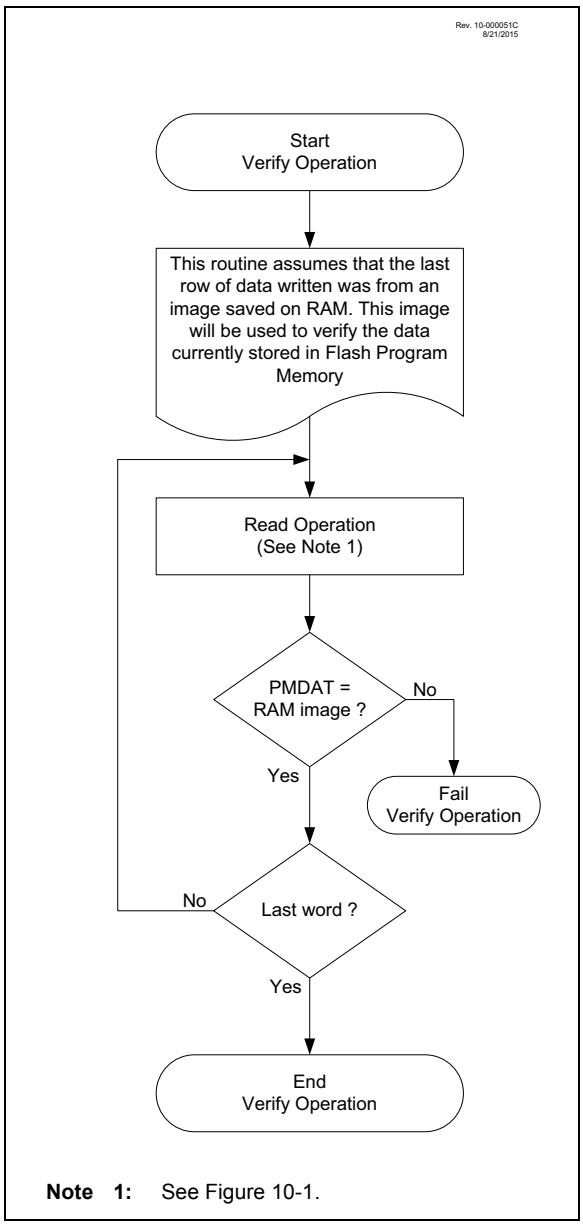
2: Opcodes shown are illustrative; any instruction that has the indicated effect may be used.

10.4.7 NVMREG DATA EEPROM MEMORY, USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS

Instead of accessing Program Flash Memory (PFM), the Data EEPROM Memory, the User ID's, Device ID/ Revision ID and Configuration Words can be accessed when NVMREGS = 1 in the NVMCON1 register. This is the region that would be pointed to by PC<15> = 1, but not all addresses are accessible. Different access may exist for reads and writes. Refer to Table 10-3.

When read access is initiated on an address outside the parameters listed in Table 10-3, the NVMDATH: NVMDATL register pair is cleared, reading back '0's.

FIGURE 10-7: FLASH PROGRAM MEMORY MODIFY FLOWCHART



Note 1: See Figure 10-1.

TABLE 10-3: EEPROM, USER ID, DEV/REV ID AND CONFIGURATION WORD ACCESS (NVMREGS = 1)

Address	Function	Read Access	Write Access
8000h-8003h	User IDs	Yes	Yes
8005h-8006h	Device ID/Revision ID	Yes	No
8007h-800Bh	Configuration Words 1-5	Yes	No
F000h-F0FFh	EEPROM	Yes	Yes

REGISTER 12-53: CCDPE: CURRENT CONTROL DRIVE NEGATIVE PORTE REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	CCDPE2	CCDPE1	CCDPE0
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 7-3 **Unimplemented:** Read as '0'

bit 2-0 **CCDPE<2:0>:** RE<2:0> Current Control Drive Positive Control bits(1)

1 = Current control source enabled

0 = Current control source disabled

Note 1: If CCDPEy is set, when CCDEN = 0 (Register 12-1), operation of the pin is undefined.

REGISTER 12-54: CCDNE: CURRENT CONTROL DRIVE NEGATIVE PORTE REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	CCDNE2	CCDNE1	CCDNE0
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 7-3 **Unimplemented:** Read as '0'

bit 2-0 **CCDNE<2:0>:** RE<2:0> Current Control Drive Negative Control bits(1)

1 = Current control source enabled

0 = Current control source disabled

Note 1: If CCDNEy is set, when CCDEN = 0 (Register 12-1), operation of the pin is undefined.

REGISTER 14-6: PMD5 – PMD CONTROL REGISTER 5

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	R/W-0/0
SMT2MD	SMT1MD	—	CLC4MD	CLC3MD	CLC2MD	CLC1MD	DSMMD
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 **SMT2MD:** Disable Signal Measurement Timer2 bit
1 = SMT2 module disabled
0 = SMT2 module enabled

bit 6 **SMT1MD:** Disable Signal Measurement Timer1 bit
1 = SMT1 module disabled
0 = SMT1 module enabled

bit 5 **Unimplemented:** Read as ‘0’

bit 4 **CLC4MD:** Disable CLC4 bit
1 = CLC4 module disabled
0 = CLC4 module enabled

bit 3 **CLC3MD:** Disable CLC3 bit
1 = CLC3 module disabled
0 = CLC3 module enabled

bit 2 **CLC2MD:** Disable CLC2 bit
1 = CLC2 module disabled
0 = CLC2 module enabled

bit 1 **CLC1MD:** Disable CLC bit
1 = CLC1 module disabled
0 = CLC1 module enabled

bit 0 **DSMMD:** Disable Data Signal Modulator bit
1 = DSM module disabled
0 = DSM module enabled

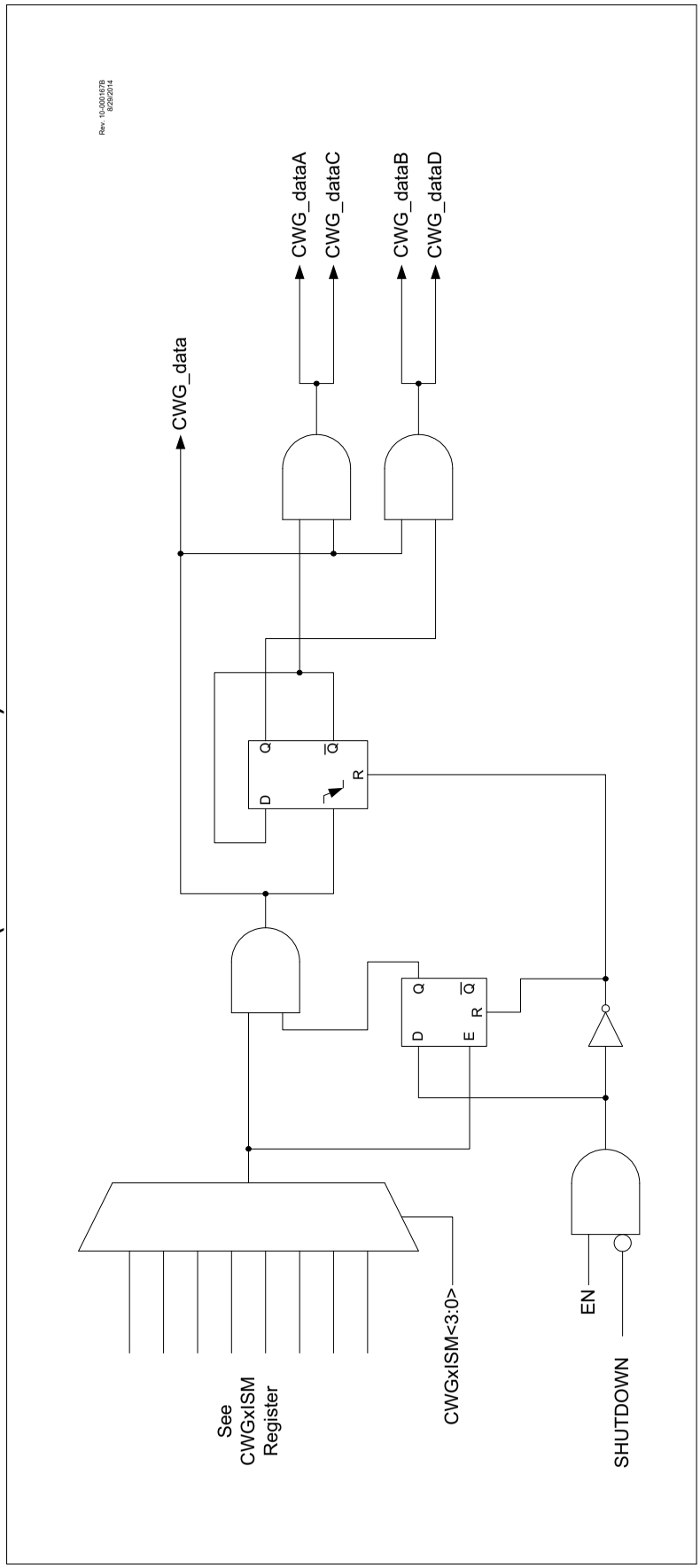


FIGURE 20-2: SIMPLIFIED CWG BLOCK DIAGRAM (PUSH-PULL MODE)

21.6 Handling VPEAK variations

If the peak amplitude of the external voltage is expected to vary, the series resistor must be selected to keep the ZCD current source and sink below the design maximum range of ± 600 μA and above a reasonable minimum range. A general rule of thumb is that the maximum peak voltage can be no more than six times the minimum peak voltage. To ensure that the maximum current does not exceed ± 600 μA and the minimum is at least ± 100 μA, compute the series resistance as shown in Equation 21-5. The compensating pull-up for this series resistance can be determined with Equation 21-4 because the pull-up value is independent from the peak voltage.

EQUATION 21-5: SERIES R FOR V RANGE

$$R_{SERIES} = \frac{V_{MAXPEAK} + V_{MINPEAK}}{7 \times 10^{-4}}$$

21.7 Operation During Sleep

The ZCD current sources and interrupts are unaffected by Sleep.

21.8 Effects of a Reset

The ZCD circuit can be configured to default to the active or inactive state on Power-On-Reset (POR). When the ZCDDIS Configuration bit is cleared, the ZCD circuit will be active at POR. When the $\overline{ZCD}$ Configuration bit is set, the EN bit of the ZCDxCON register must be set to enable the ZCD module.

21.9 Disabling the ZCD Module

The ZCD module can be disabled in two ways:

1. Configuration Word 2H has the $\overline{ZCD}$ bit, which disables the ZCD module when set, but it can be enabled using the EN bit of the ZCDCON register (Register 21-1). If the $\overline{ZCD}$ bit is clear, the ZCD is always enabled.
2. The ZCD can also be disabled using the ZCDMD bit of the PMD2 register (Register 14-3) this is subject to the status of the $\overline{ZCD}$ bit.

REGISTER 23-16: ADRESH: ADC RESULT REGISTER HIGH, ADFRM=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
Most Significant eight bits of 10-bit conversion result.

REGISTER 23-17: ADRESL: ADC RESULT REGISTER LOW, ADFRM=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. Least Significant two bits of 10-bit conversion result.
bit 5-0 **Reserved**: Do not use.

REGISTER 23-18: ADRESH: ADC RESULT REGISTER HIGH, ADFRM=1

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<9:8>	
bit 7							bit 0

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

bit 7-2 **Reserved**: Do not use.
bit 1-0 **ADRES<9:8>**: ADC Sample Result bits. Most Significant two bits of 10-bit conversion result.

REGISTER 28-2: TxGCON: TIMER1/3/5 GATE CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	U-0	U-0
GE	GPOL	GTM	GSPM	GGO/$\overline{\text{DONE}}$	GVAL	—	—
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 **GE:** Timer1 Gate Enable bit

If ON = 0:
This bit is ignored
If ON = 1:
1 = Timer1 counting is controlled by the Timer1 gate function
0 = Timer1 is always counting

bit 6 **GPOL:** Timer1 Gate Polarity bit

1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 5 **GTM:** Timer1 Gate Toggle Mode bit

1 = Timer1 Gate Toggle mode is enabled
0 = Timer1 Gate Toggle mode is disabled and toggle flip-flop is cleared
Timer1 gate flip-flop toggles on every rising edge.

bit 4 **GSPM:** Timer1 Gate Single-Pulse Mode bit

1 = Timer1 Gate Single-Pulse mode is enabled and is controlling Timer1 gate
0 = Timer1 Gate Single-Pulse mode is disabled

bit 3 **GGO/$\overline{\text{DONE}}$:** Timer1 Gate Single-Pulse Acquisition Status bit

1 = Timer1 gate single-pulse acquisition is ready, waiting for an edge
0 = Timer1 gate single-pulse acquisition has completed or has not been started
This bit is automatically cleared when GSPM is cleared

bit 2 **GVAL:** Timer1 Gate Value Status bit

Indicates the current state of the Timer1 gate that could be provided to TMR1H:TMR1L
Unaffected by Timer1 Gate Enable (GE)

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

29.5 Operation Examples

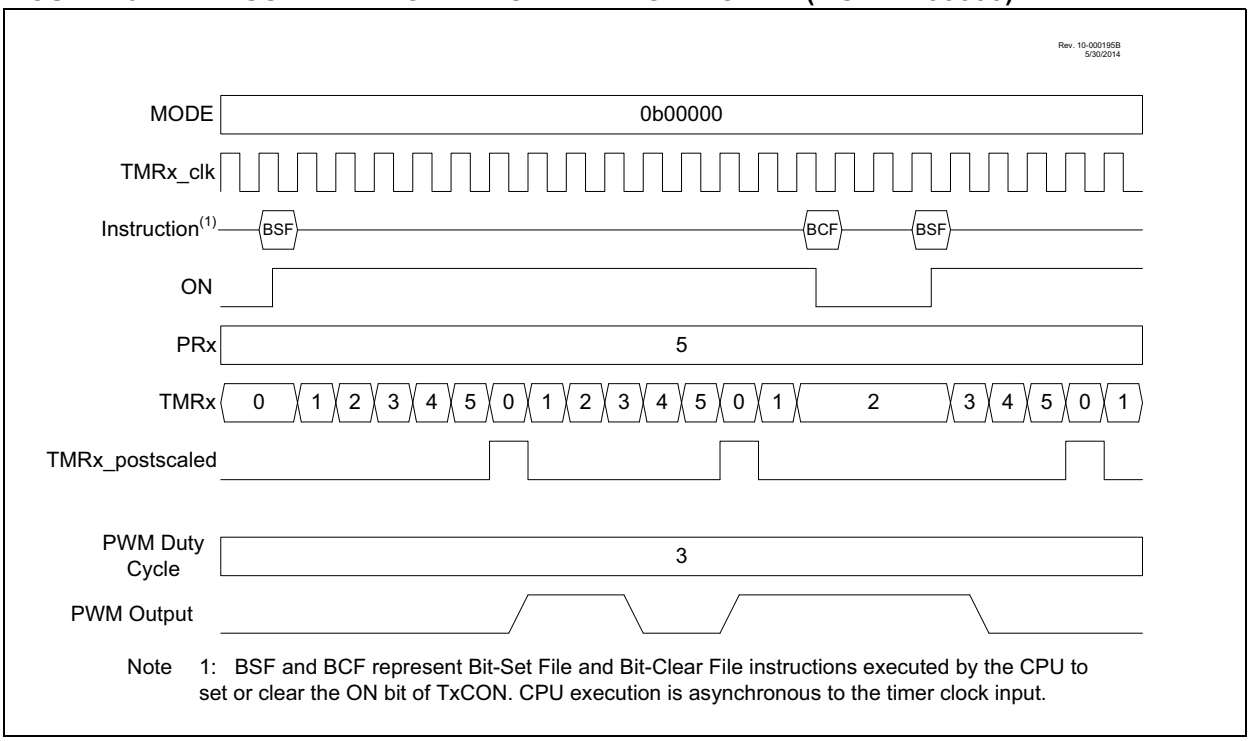
Unless otherwise specified, the following notes apply to the following timing diagrams:

- Both the prescaler and postscaler are set to 1:1 (both the CKPS and OUTPS bits in the TxCON register are cleared).
- The diagrams illustrate any clock except Fosc/4 and show clock-sync delays of at least two full cycles for both ON and Timer2_ers. When using Fosc/4, the clock-sync delay is at least one instruction period for Timer2_ers; ON applies in the next instruction period.
- The PWM Duty Cycle and PWM output are illustrated assuming that the timer is used for the PWM function of the CCP module as described in **Section 30.0 "Capture/Compare/PWM Modules"**. The signals are not a part of the Timer2 module.

29.5.1 SOFTWARE GATE MODE

This mode corresponds to legacy Timer2 operation. The timer increments with each clock input when ON = 1 and does not increment when ON = 0. When the TMRx count equals the PRx period count the timer resets on the next clock and continues counting from 0. Operation with the ON bit software controlled is illustrated in Figure 29-4. With PRx = 5, the counter advances until TMRx = 5, and goes to zero with the next clock.

FIGURE 29-4: SOFTWARE GATE MODE TIMING DIAGRAM (MODE = 00000)



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.

31.2.4 SPI SLAVE MODE

In Slave mode, the data is transmitted and received as external clock pulses appear on SCK. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCK pin. The Idle state is determined by the CKP bit of the SSPxCON1 register.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCK pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake-up from Sleep.

31.2.4.1 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisy-chain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole chain acts as one large communication shift register. The daisy-chain feature only requires a single Slave Select line from the master device.

Figure 31-7 shows the block diagram of a typical daisy-chain connection when operating in SPI mode.

In a daisy-chain configuration, only the most recent byte on the bus is required by the slave. Setting the BOEN bit of the SSPxCON3 register will enable writes to the SSPxBUF register, even if the previous byte has not been read. This allows the software to ignore data that may not apply to it.

31.2.5 SLAVE SELECT SYNCHRONIZATION

The Slave Select can also be used to synchronize communication. The Slave Select line is held high until the master device is ready to communicate. When the Slave Select line is pulled low, the slave knows that a new transmission is starting.

If the slave fails to receive the communication properly, it will be reset at the end of the transmission, when the Slave Select line returns to a high state. The slave is then ready to receive a new transmission when the Slave Select line is pulled low again. If the Slave Select line is not used, there is a risk that the slave will eventually become out of sync with the master. If the slave misses a bit, it will always be one bit off in future transmissions. Use of the Slave Select line allows the slave and master to align themselves at the beginning of each transmission.

The $\overline{SS}$ pin allows a Synchronous Slave mode. The SPI must be in Slave mode with $\overline{SS}$ pin control enabled (SSPxCON1<3:0> = 0100).

When the $\overline{SS}$ pin is low, transmission and reception are enabled and the SDO pin is driven.

When the $\overline{SS}$ pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

Note 1: When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPxCON1<3:0> = 0100), the SPI module will reset if the $\overline{SS}$ pin is set to VDD.

2: When the SPI is used in Slave mode with CKE set; the user must enable $\overline{SS}$ pin control.

3: While operated in SPI Slave mode the SMP bit of the SSPxSTAT register must remain clear.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the $\overline{SS}$ pin to a high level or clearing the SSPEN bit.

31.4.5 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 31-12 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.

31.4.6 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.

31.4.7 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 31-13 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.

31.4.8 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 31-12: I^2C START AND STOP CONDITIONS

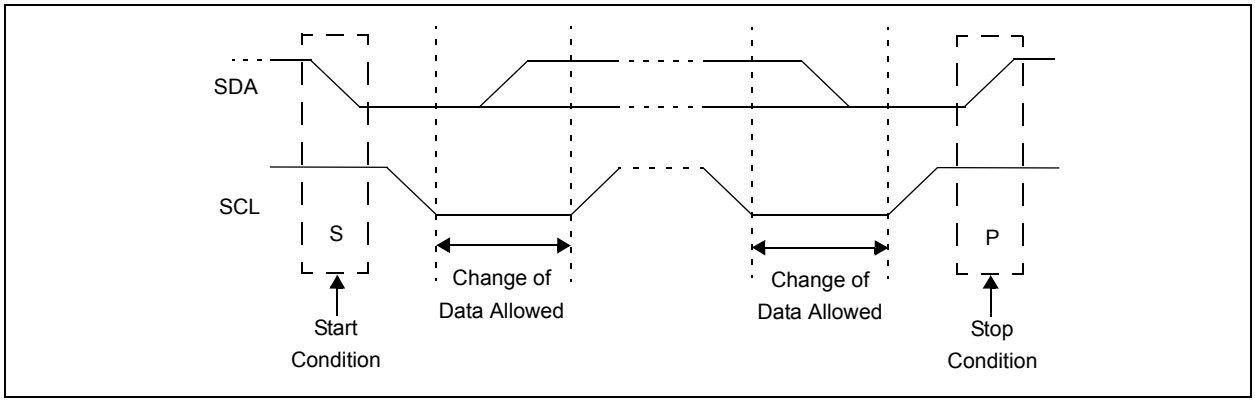


FIGURE 31-13: I^2C RESTART CONDITION

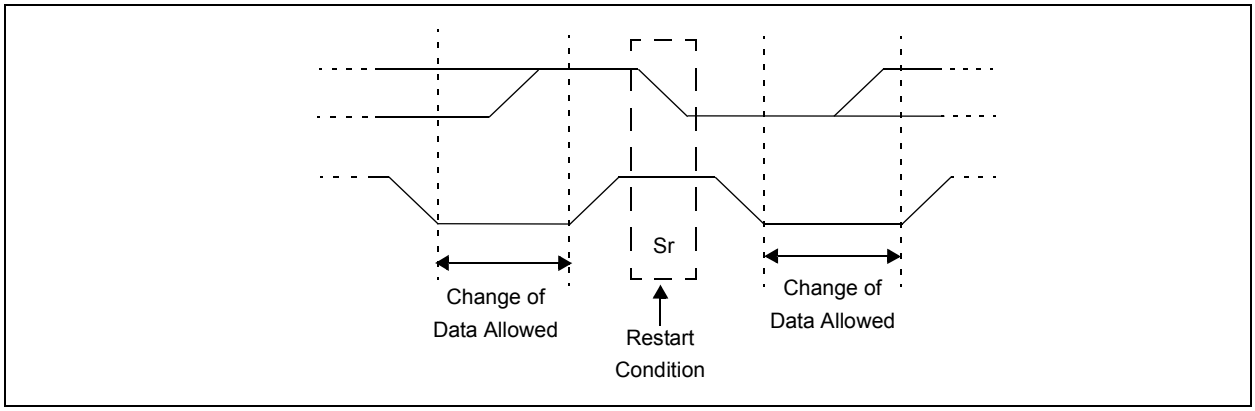


FIGURE 31-34: BUS COLLISION DURING START CONDITION (SCL = 0)

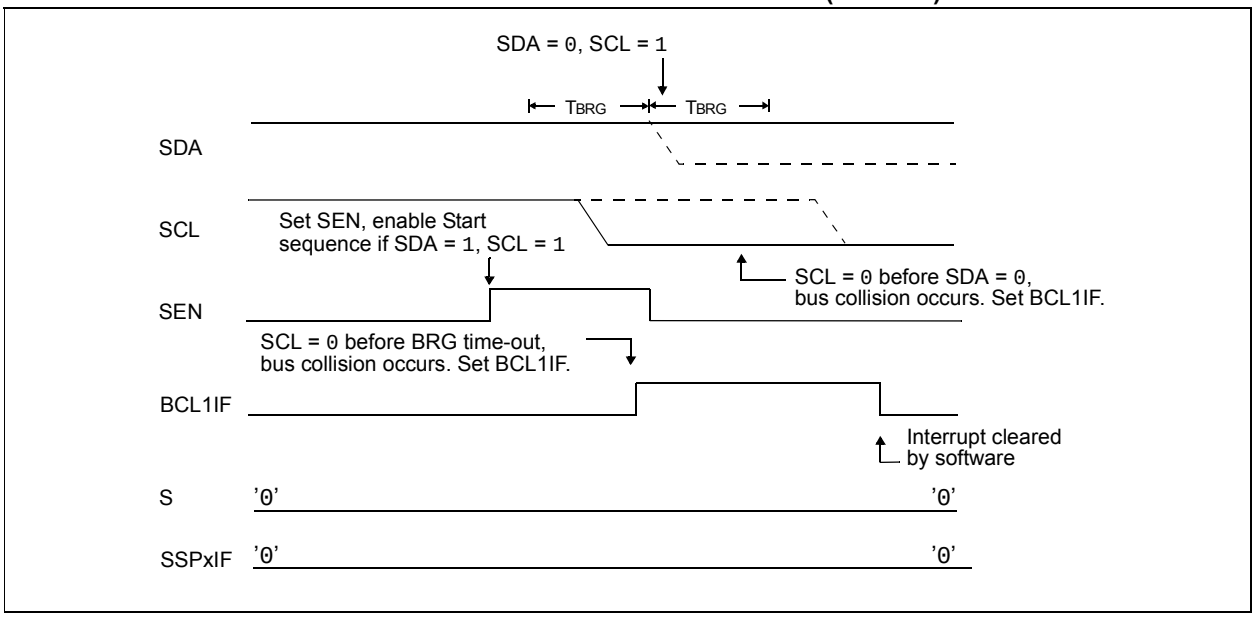
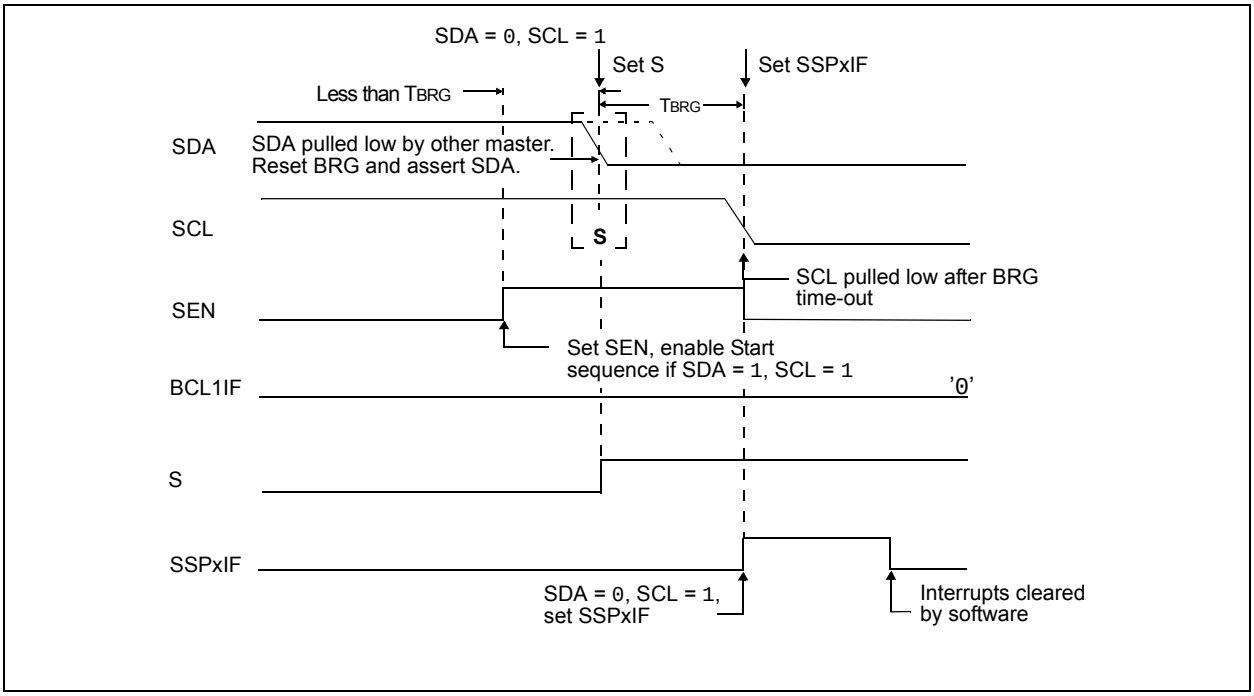


FIGURE 31-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION



32.6.4 HIGH AND LOW MEASURE MODE

This mode measures the high and low pulse time of the SMTSIGx relative to the SMT clock. It begins incrementing the SMTxTMR on a rising edge on the SMTSIGx input, then updates the SMTxCPW register with the value and resets the SMTxTMR on a falling edge, starting to increment again. Upon observing another rising edge, it updates the SMTxCPR register with its current value and once again resets the SMTxTMR value and begins incrementing again. See Figure 32-8 and Figure 32-9.

REGISTER 34-1: CLKRCON: REFERENCE CLOCK CONTROL REGISTER

R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CLKREN	—	—	CLKRDC<1:0>		CLKRDIV<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 **CLKREN:** Reference Clock Module Enable bit
- 1 = Reference Clock module enabled
- 0 = Reference Clock module is disabled

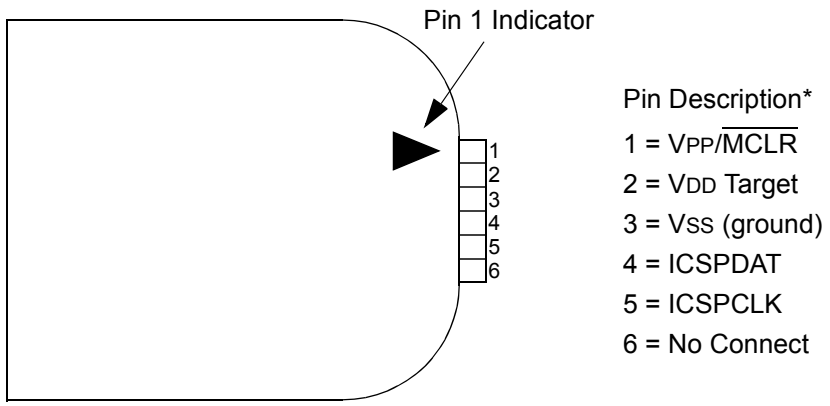
bit 6-5 **Unimplemented:** Read as '0'

bit 4-3 **CLKRDC<1:0>:** Reference Clock Duty Cycle bits (1)
- 11 = Clock outputs duty cycle of 75%
- 10 = Clock outputs duty cycle of 50%
- 01 = Clock outputs duty cycle of 25%
- 00 = Clock outputs duty cycle of 0%

bit 2-0 **CLKRDIV<2:0>:** Reference Clock Divider bits
- 111 = Input clock divided by 128
- 110 = Input clock divided by 64
- 101 = Input clock divided by 32
- 100 = Input clock divided by 16
- 011 = Input clock divided by 8
- 010 = Input clock divided by 4
- 001 = Input clock divided by 2
- 000 = Input clock

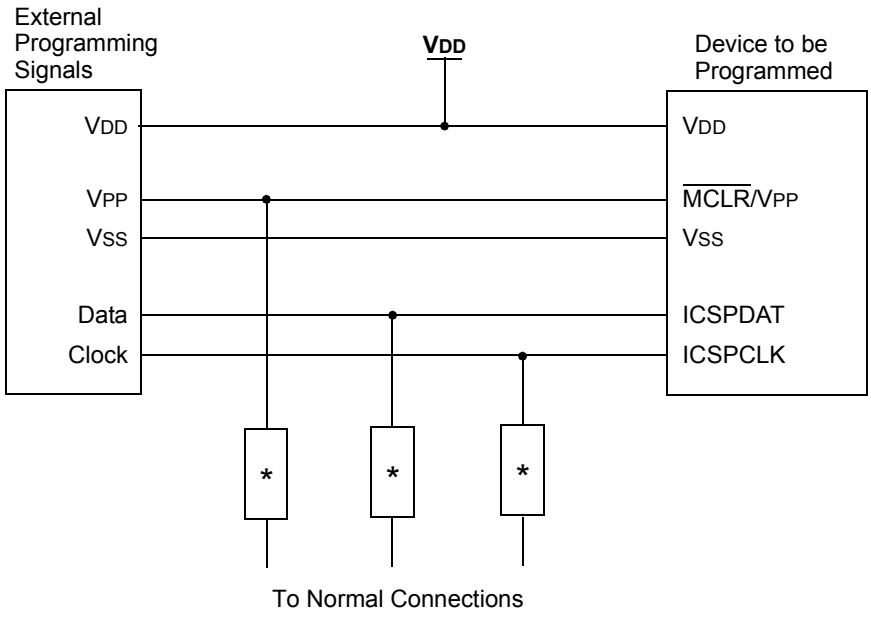
Note 1: Bits are valid for Reference Clock divider values of two or larger, the base clock cannot be further divided.

FIGURE 35-2: PICkit™ PROGRAMMER STYLE CONNECTOR INTERFACE



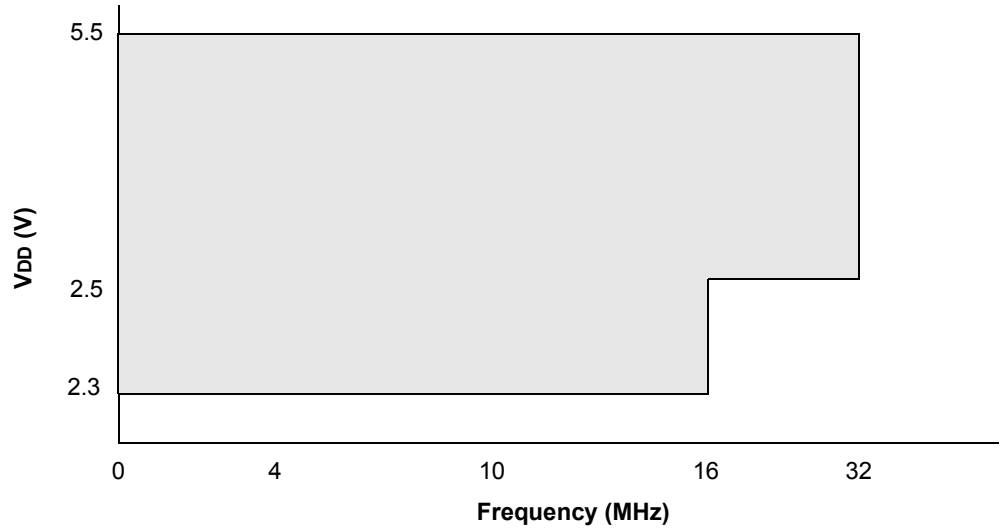
* The 6-pin header (0.100" spacing) accepts 0.025" square pins.

FIGURE 35-3: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



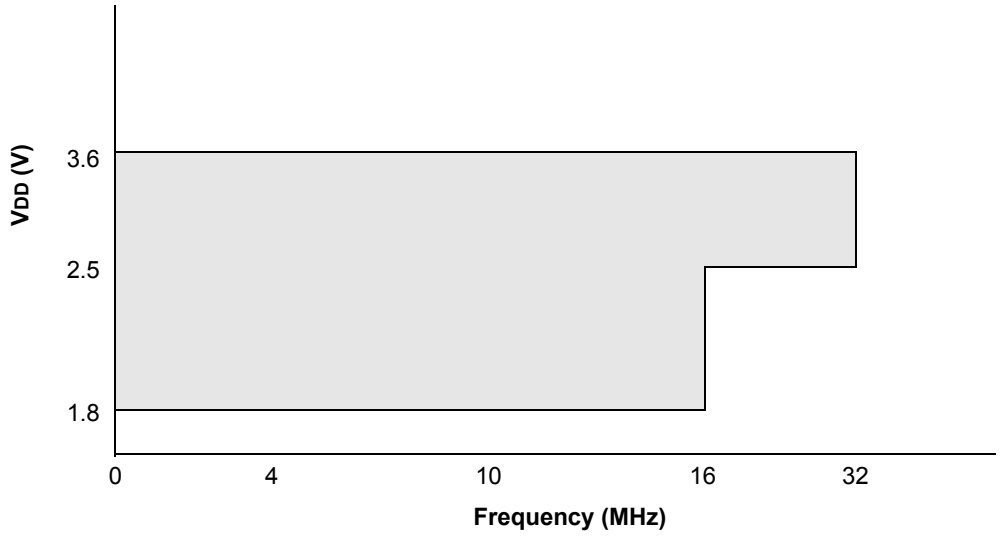
* Isolation devices (as required).

FIGURE 37-1: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16F18856/76 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
2: Refer to Table for each Oscillator mode's supported frequencies.

FIGURE 37-2: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16LF18856/76 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
2: Refer to Table for each Oscillator mode's supported frequencies.