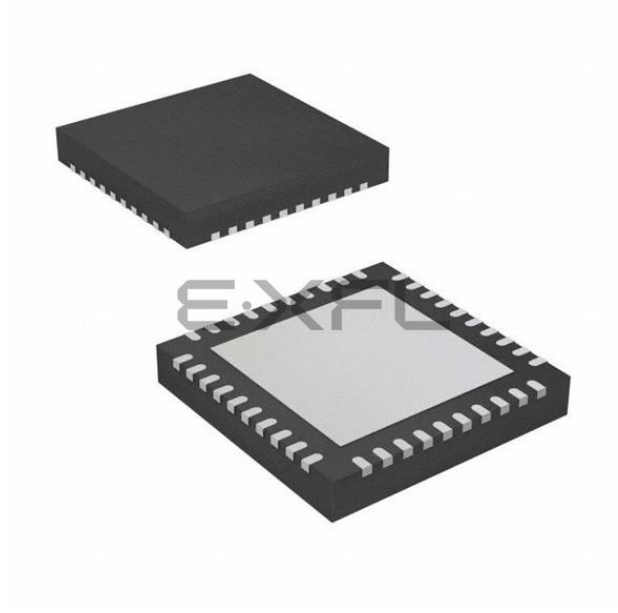


Welcome to E-XFL.COM

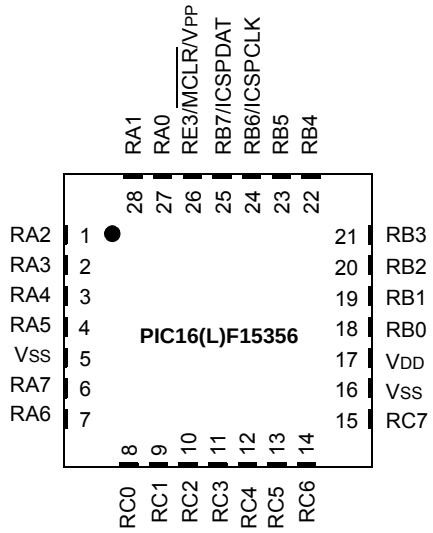
What is "Embedded - Microcontrollers"?

"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	224 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	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf15376-i-mv

28-PIN UQFN (4x4), UQFN (6x6)

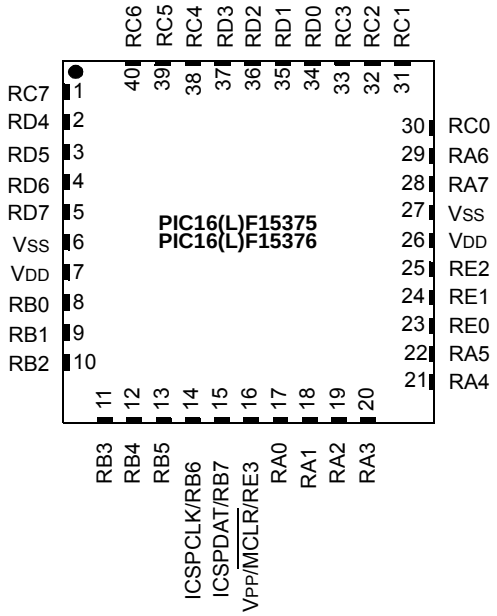


Note 1: See Table 3 for location of all peripheral functions.

2: All VDD and all VSS pins must be connected at the circuit board level. Allowing one or more VSS or VDD pins to float may result in degraded electrical performance or non-functionality.

3: The bottom pad of the QFN/UQFN package should be connected to VSS at the circuit board level.

40-PIN UQFN (5x5)



Note 1: See Table 4 for the pin allocation tables.

2: The bottom pad of the QFN/UQFN package should be connected to VSS at the circuit board level.

4.3.3 SPECIAL FUNCTION REGISTER

The Special Function Registers are registers used by the application to control the desired operation of peripheral functions in the device. The Special Function Registers occupy the 20 bytes of the data banks 0-59 and 100 bytes of the data banks 60-63, after the core registers.

The SFRs associated with the operation of the peripherals are described in the appropriate peripheral chapter of this data sheet.

4.3.4 GENERAL PURPOSE RAM

There are up to 80 bytes of GPR in each data memory bank.

4.3.4.1 Linear Access to GPR

The general purpose RAM can be accessed in a non-banked method via the FSRs. This can simplify access to large memory structures. See **Section 4.6.2 "Linear Data Memory"** for more information.

4.3.5 COMMON RAM

There are 16 bytes of common RAM accessible from all banks.

4.3.6 DEVICE MEMORY MAPS

The memory maps are as shown in Table 4-4 through Table 4-11.

TABLE 4-11: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 60											
CPU CORE REGISTERS; see Table 4-3 for specifics											
1E0Ch	—	Unimplemented								—	—
1E0Dh	—	Unimplemented								—	—
1E0Eh	—	Unimplemented								—	—
1E0Fh	CLCDATA	—	—	—	—	MLC4OUT	MLC3OUT	MLC2OUT	MLC1OUT	---- xxxx	---- uuuu
1E10h	CLCCON	LC1EN	—	LC1OUT	LC1INTP	LC1INTN	LC1MODE<2:0>			0-00 0000	0-00 0000
1E11h	CLC1POL	LC1POL	—	—	—	LC1G4POL	LC1G3POL	LC1G2POL	LC1G1POL	0--- xxxx	0--- uuuu
1E12h	CLC1SEL0	—	—	LC1D1S<5:0>						--xx xxxx	--uu uuuu
1E13h	CLC1SEL1	—	—	LC1D2S<5:0>						--xx xxxx	--uu uuuu
1E14h	CLC1SEL2	—	—	LC1D3S<5:0>						--xx xxxx	--uu uuuu
1E15h	CLC1SEL3	—	—	LC1D4S<5:0>						--xx xxxx	--uu uuuu
1E16h	CLC1GLS0	LC1G1D4T	LC1G4D3N	LC1G1D3T	LC1G1D3N	LC1G1D2T	LC1G1D2N	LC1G1D1T	LC1G1D1N	xxxx xxxx	uuuu uuuu
1E17h	CLC1GLS1	LC1G2D4T	LC1G4D3N	LC1G2D3T	LC1G2D3N	LC1G2D2T	LC1G2D2N	LC1G2D1T	LC1G2D1N	xxxx xxxx	uuuu uuuu
1E18h	CLC1GLS2	LC1G3D4T	LC1G4D3N	LC1G3D3T	LC1G3D3N	LC1G3D2T	LC1G3D2N	LC1G3D1T	LC1G3D1N	xxxx xxxx	uuuu uuuu
1E19h	CLC1GLS3	LC1G4D4T	LC1G4D3N	LC1G4D3T	LC1G4D3N	LC1G4D2T	LC1G4D2N	LC1G4D1T	LC1G4D1N	xxxx xxxx	uuuu uuuu
1E1Ah	CLC2CON	LC2EN	—	LC2OUT	LC2INTP	LC2INTN	LC2MODE<2:0>			0-00 0000	0-00 0000
1E1Bh	CLC2POL	LC2POL	—	—	—	LC2G4POL	LC2G3POL	LC2G2POL	LC2G1POL	0--- xxxx	0--- uuuu
1E1Ch	CLC2SEL0	—	—	LC2D1S<5:0>						--xx xxxx	--uu uuuu
1E1Dh	CLC2SEL1	—	—	LC2D2S<5:0>						--xx xxxx	--uu uuuu
1E1Eh	CLC2SEL2	—	—	LC2D3S<5:0>						--xx xxxx	--uu uuuu
1E1Fh	CLC2SEL3	—	—	LC2D4S<5:0>						--xx xxxx	--uu uuuu
1E20h	CLC2GLS0	LC2G1D4T	LC2G4D3N	LC2G1D3T	LC2G1D3N	LC2G1D2T	LC2G1D2N	LC2G1D1T	LC2G1D1N	xxxx xxxx	uuuu uuuu
1E21h	CLC2GLS1	LC2G2D4T	LC2G4D3N	LC2G2D3T	LC2G2D3N	LC2G2D2T	LC2G2D2N	LC2G2D1T	LC2G2D1N	xxxx xxxx	uuuu uuuu
1E22h	CLC2GLS2	LC2G3D4T	LC2G4D3N	LC2G3D3T	LC2G3D3N	LC2G3D2T	LC2G3D2N	LC2G3D1T	LC2G3D1N	xxxx xxxx	uuuu uuuu
1E23h	CLC2GLS3	LC2G4D4T	LC2G4D3N	LC2G4D3T	LC2G4D3N	LC2G4D2T	LC2G4D2N	LC2G4D1T	LC2G4D1N	xxxx xxxx	uuuu uuuu
1E24h	CLC3CON	LC3EN	—	LC3OUT	LC3INTP	LC3INTN	LC3MODE			0-00 0000	0-00 0000
1E25h	CLC3POL	LC3POL	—	—	—	LC3G4POL	LC3G3POL	LC3G2POL	LC3G1POL	0--- xxxx	0--- uuuu
1E26h	CLC3SEL0	—	—	LC3D1S<5:0>						--xx xxxx	--uu uuuu
1E27h	CLC3SEL1	—	—	LC3D2S<5:0>						--xx xxxx	--uu uuuu
1E28h	CLC3SEL2	—	—	LC3D3S<5:0>						--xx xxxx	--uu uuuu
1E29h	CLC3SEL3	—	—	LC3D4S<5:0>						--xx xxxx	--uu uuuu
1E2Ah	CLC3GLS0	LC3G1D4T	LC3G4D3N	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	xxxx xxxx	uuuu uuuu

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

8.1 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.

8.2 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 8-1 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 8-2 for more information.

TABLE 8-1: 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	Wait for release of BOR(1) (BORRDY = 1)
10	X	Awake	Active	Waits for release of BOR (BORRDY = 1) Waits for BOR Reset release
		Sleep	Disabled	
01	1	X	Active	Waits for BOR Reset release (BORRDY = 1)
	0	X	Disabled	Begins immediately (BORRDY = x)
00	X	X	Disabled	

Note 1: In this specific case, “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.

8.2.1 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.

8.2.2 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.

REGISTER 9-5: OSCEN: OSCILLATOR MANUAL ENABLE 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	U-0	U-0
EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	—	—
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 **EXTOEN:** External Oscillator Manual Request Enable bit(1)
1 = EXTOSC is explicitly enabled, operating as specified by FEXTOSC
0 = EXTOSC could be enabled by some modules

bit 6 **HFOEN:** HFINTOSC Oscillator Manual Request Enable bit
1 = HFINTOSC is explicitly enabled, operating as specified by OSCFRQ
0 = HFINTOSC could be enabled by another module

bit 5 **MFOEN:** MFINTOSC Oscillator Manual Request Enable bit
1 = MFINTOSC is explicitly enabled
0 = MFINTOSC could be enabled by another module

bit 4 **LFOEN:** LFINTOSC (31 kHz) Oscillator Manual Request Enable bit
1 = LFINTOSC is explicitly enabled
0 = LFINTOSC could be enabled by another module

bit 3 **SOSCEN:** Secondary (Timer1) Oscillator Manual Request bit
1 = Secondary oscillator is explicitly enabled, operating as specified by SOSCPWR
0 = Secondary oscillator could be enabled by another module

bit 2 **ADOEN:** FRC Oscillator Manual Request Enable bit
1 = FRC is explicitly enabled
0 = FRC could be enabled by another module

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

11.4 Register Definitions: Voltage Regulator and DOZE Control

REGISTER 11-1: VREGCON: VOLTAGE REGULATOR CONTROL REGISTER (1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	U-0
—	—	—	—	—	—	VREGPM	—
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 **Unimplemented:** Read as '0'

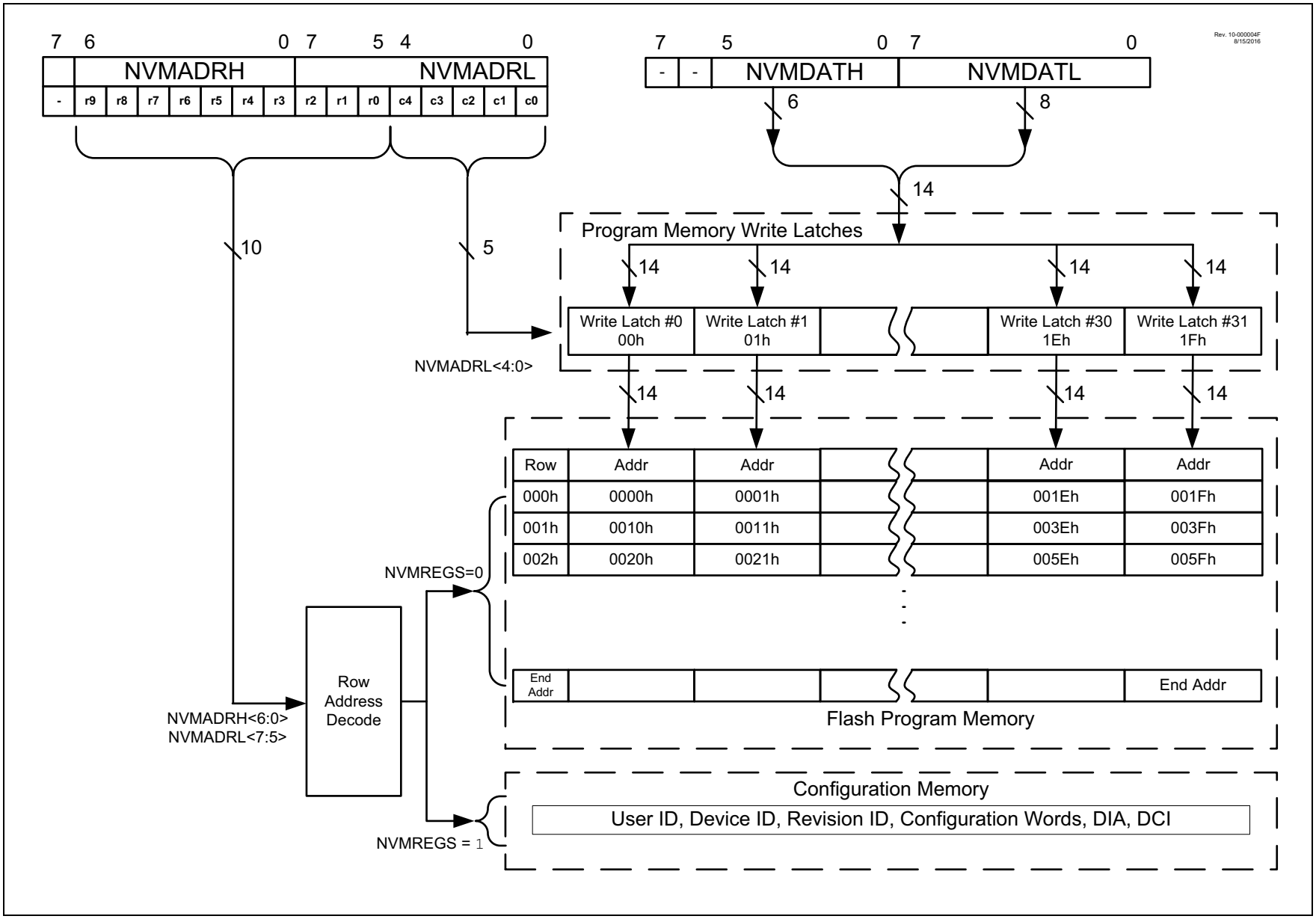
bit 1 **VREGPM**: Voltage Regulator Power Mode Selection bit
- 1 = Low-Power Sleep mode enabled in Sleep(2)
  Draws lowest current in Sleep, slower wake-up
- 0 = Normal Power mode enabled in Sleep(2)
  Draws higher current in Sleep, faster wake-up

bit 0 **Unimplemented:** Read as '1'. Maintain this bit set

Note 1: PIC16F15356/75/76/85/86 only.

2: See **Section 37.0 "Electrical Specifications"**.

FIGURE 13-4: NVMREGS WRITES TO PROGRAM FLASH MEMORY WITH 32 WRITE LATCHES



13.3.8 WRERR BIT

The WRERR bit can be used to determine if a write error occurred.

WRERR will be set if one of the following conditions occurs:

- If WR is set while the NVMADRH:NMVADRL points to a write-protected address
- A Reset occurs while a self-write operation was in progress
- An unlock sequence was interrupted

The WRERR bit is normally set by hardware, but can be set by the user for test purposes. Once set, WRERR must be cleared in software.

TABLE 13-4: ACTIONS FOR PFM WHEN WR = 1

Free	LWLO	Actions for PFM when WR = 1	Comments
1	x	Erase the 32-word row of NVMADRH:NVMADRL location. See **Section 13.3.3 "NVMREG Erase of PFM"**	• If WP is enabled, WR is cleared and WRERR is set • All 32 words are erased • NVMDATH:NVMDATL is ignored
0	1	Copy NVMDATH:NVMDATL to the write latch corresponding to NVMADR LSBs. See **Section 13.3.3 "NVMREG Erase of PFM"**	• Write protection is ignored • No memory access occurs
0	0	Write the write-latch data to PFM row. See **Section 13.3.3 "NVMREG Erase of PFM"**	• If WP is enabled, WR is cleared and WRERR is set • Write latches are reset to 3FFh • NVMDATH:NVMDATL is ignored

REGISTER 14-3: LATA: PORTA DATA LATCH REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-1/1	R/W-x/u	R/W-x/u	R/W-x/u
LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0
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 **LATA<7:0>**: RA<7:0> Output Latch Value bits(1)

Note 1: Writes to PORTA are actually written to corresponding LATA register. Reads from PORTA register returns actual I/O pin values.

REGISTER 14-4: ANSELA: PORTA ANALOG SELECT REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
ANSA7	ANSA6	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0
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 **ANSA<7:0>**: Analog Select between Analog or Digital Function on pins RA<7:0>, respectively
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.
0 = Digital I/O. Pin is assigned to port or digital special function.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

TABLE 20-1: ADC CLOCK PERIOD (TAD) VS. DEVICE OPERATING FREQUENCIES

ADC Clock Period (TAD)		Device Frequency (FOSC)					
ADC Clock Source	ADCS<2:0>	32 MHz	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
FOSC/2	000	62.5ns(2)	100 ns(2)	125 ns(2)	250 ns(2)	500 ns(2)	2.0 μs
FOSC/4	100	125 ns(2)	200 ns(2)	250 ns(2)	500 ns(2)	1.0 μs	4.0 μs
FOSC/8	001	0.5 μs(2)	400 ns(2)	0.5 μs(2)	1.0 μs	2.0 μs	8.0 μs(3)
FOSC/16	101	800 ns	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs(3)
FOSC/32	010	1.0 μs	1.6 μs	2.0 μs	4.0 μs	8.0 μs(3)	32.0 μs(2)
FOSC/64	110	2.0 μs	3.2 μs	4.0 μs	8.0 μs(3)	16.0 μs(2)	64.0 μs(2)
ADCRC	x11	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)

Legend: Shaded cells are outside of recommended range.

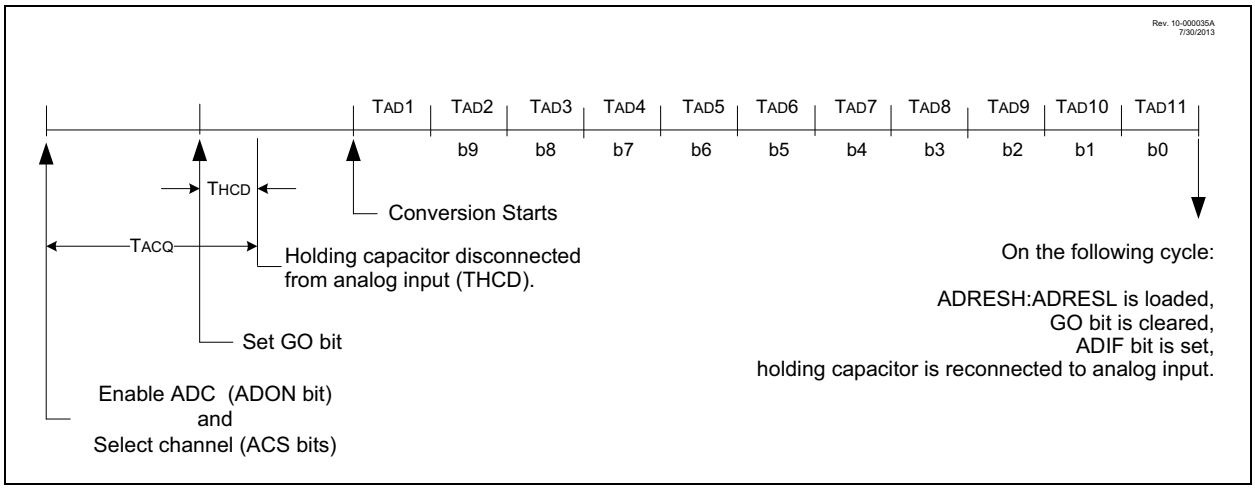
Note 1: See TAD parameter for ADCRC source typical TAD value.

2: These values violate the required TAD time.

3: Outside the recommended TAD time.

4: The ADC clock period (TAD) and total ADC conversion time can be minimized when the ADC clock is derived from the system clock FOSC. However, the ADCRC oscillator source must be used when conversions are to be performed with the device in Sleep mode.

FIGURE 20-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES



20.2 ADC Operation

20.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note:	The GO/$\overline{\text{DONE}}$ bit will not be set in the same instruction that turns on the ADC. Refer to **Section 20.2.6 "ADC Conversion Procedure"**.

20.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF Interrupt Flag bit
- Update the ADRESH and ADRESL registers with new conversion result

20.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRESH and ADRESL registers will be updated with the partially complete Analog-to-Digital conversion sample. Incomplete bits will match the last bit converted.

Note:	A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

20.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the ADCRC option. When the ADCRC oscillator source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than ADCRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

20.2.5 AUTO-CONVERSION TRIGGER

The Auto-conversion Trigger allows periodic ADC measurements without software intervention. When a rising edge of the selected source occurs, the GO/$\overline{\text{DONE}}$ bit is set by hardware.

The Auto-conversion Trigger source is selected with the ADACT<3:0> bits of the ADACT register.

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 20-2 for auto-conversion sources.

TABLE 20-2: ADC AUTO-CONVERSION TABLE

ADACT VALUE	SOURCE/ PERIPHERAL	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	CCP1	CCP1 output
0x06	CCP2	CCP2 output
0x07	PWM3	PWM3 output
0x08	PWM4	PWM4 output
0x09	PWM5	PWM5 output
0x0A	PWM6	PWM6 output
0x0B	NCO1	NCO1 output
0x0C	C1OUT	Comparator C1 output
0x0D	C2OUT	Comparator C2 output
0x0E	IOCIF	Interrupt-on change flag trigger
0x0F	CLC1	CLC1 output
0x10	CLC2	CLC2 output
0x11	CLC3	CLC3 output
0x12	CLC4	CLC4 output
0x13-0xFF	Reserved	Reserved, do not use

REGISTER 28-4: CCPRxH REGISTER: CCPx REGISTER HIGH BYTE

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x
CCPRx<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 Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0

CCPxMODE = Capture mode
CCPRxH<7:0>: Captured value of TMR1H
CCPxMODE = Compare mode
CCPRxH<7:0>: MS Byte compared to TMR1H
CCPxMODE = PWM modes when CCPxFMT = 0:
CCPRxH<7:2>: Not used
CCPRxH<1:0>: Pulse-width Most Significant two bits
CCPxMODE = PWM modes when CCPxFMT = 1:
CCPRxH<7:0>: Pulse-width Most Significant eight bits

29.1.1 PWM CLOCK SELECTION

The PIC16(L)F15356/75/76/85/86 allows each individual CCP and PWM module to select the timer source that controls the module. Each module has an independent selection.

29.1.2 USING THE TMR2 WITH THE PWM MODULE

This device has a newer version of the TMR2 module that has many new modes, which allow for greater customization and control of the PWM signals than on older parts. Refer to **Section 27.5 "Operation Examples"** for examples of PWM signal generation using the different modes of Timer2.

Note:	PWM operation requires that the timer used as the PWM time base has the FOSC/4 clock source selected.

29.1.3 PWM PERIOD

Referring to Figure 29-1, the PWM output has a period and a pulse width. The frequency of the PWM is the inverse of the period (1/period).

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

EQUATION 29-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

Note 1: TOSC = 1/FOSC

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The PWMx pin is set (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM pulse width is latched from PWMxDC.

Note:	If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

29.1.4 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to the PWMxDC register. The PWMxDCH contains the eight MSbs and the PWMxDCL<7:6> bits contain the two LSbs.

The PWMDC register is double-buffered and can be updated at any time. This double buffering is essential for glitch-free PWM operation. New values take effect when TMR2 = PR2. Note that PWMDC is left-justified.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

Equation 29-2 is used to calculate the PWM pulse width.

Equation 29-3 is used to calculate the PWM duty cycle ratio.

EQUATION 29-2: PULSE WIDTH

$$Pulse\ Width = (PWMxDC) \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

EQUATION 29-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(PWMxDC)}{4(PR2 + 1)}$$

29.1.5 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 29-4.

EQUATION 29-4: PWM RESOLUTION

$$Resolution = \frac{\log[4(PR2 + 1)]}{\log(2)}\ bits$$

31.7 Register Definitions: CLC Control

REGISTER 31-1: CLCxCON: CONFIGURABLE LOGIC CELL CONTROL REGISTER

R/W-0/0	U-0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
LCxEN	—	LCxOUT	LCxINTP	LCxINTN	LCxMODE<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 **LCxEN:** Configurable Logic Cell Enable bit
1 = Configurable logic cell is enabled and mixing input signals
0 = Configurable logic cell is disabled and has logic zero output

bit 6 **Unimplemented:** Read as ‘0’

bit 5 **LCxOUT:** Configurable Logic Cell Data Output bit
Read-only: logic cell output data, after LCPOL; sampled from CLCxOUT

bit 4 **LCxINTP:** Configurable Logic Cell Positive Edge Going Interrupt Enable bit
1 = CLCxIF will be set when a rising edge occurs on CLCxOUT
0 = CLCxIF will not be set

bit 3 **LCxINTN:** Configurable Logic Cell Negative Edge Going Interrupt Enable bit
1 = CLCxIF will be set when a falling edge occurs on CLCxOUT
0 = CLCxIF will not be set

bit 2-0 **LCxMODE<2:0>:** Configurable Logic Cell Functional Mode bits
111 = Cell is 1-input transparent latch with S and R
110 = Cell is J-K flip-flop with R
101 = Cell is 2-input D flip-flop with R
100 = Cell is 1-input D flip-flop with S and R
011 = Cell is S-R latch
010 = Cell is 4-input AND
001 = Cell is OR-XOR
000 = Cell is AND-OR

TABLE 31-4: SUMMARY OF REGISTERS ASSOCIATED WITH CLCx

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	146
PIR5	CLC4IF	CLC3IF	CLC2IF	CLC1IF	—	—	—	TMR1GIF	160
PIE5	CLC4IE	CLC4IE	CLC2IE	CLC1IE	—	—	—	TMR1GIE	152
CLC1CON	LC1EN	—	LC1OUT	LC1INTP	LC1INTN	LC1MODE<2:0>			410
CLC1POL	LC1POL	—	—	—	LC1G4POL	LC1G3POL	LC1G2POL	LC1G1POL	411
CLC1SEL0	—	—	LC1D1S<5:0>						412
CLC1SEL1	—	—	LC1D2S<5:0>						412
CLC1SEL2	—	—	LC1D3S<5:0>						412
CLC1SEL3	—	—	LC1D4S<5:0>						412
CLC1GLS0	—	—	LC1G1D3T	LC1G1D3N	LC1G1D2T	LC1G1D2N	LC1G1D1T	LC1G1D1N	413
CLC1GLS1	—	—	LC1G2D3T	LC1G2D3N	LC1G2D2T	LC1G2D2N	LC1G2D1T	LC1G2D1N	414
CLC1GLS2	—	—	LC1G3D3T	LC1G3D3N	LC1G3D2T	LC1G3D2N	LC1G3D1T	LC1G3D1N	415
CLC1GLS3	—	—	LC1G4D3T	LC1G4D3N	LC1G4D2T	LC1G4D2N	LC1G4D1T	LC1G4D1N	416
CLC2CON	LC2EN	—	LC2OUT	LC2INTP	LC2INTN	LC2MODE<2:0>			410
CLC2POL	LC2POL	—	—	—	LC2G4POL	LC2G3POL	LC2G2POL	LC2G1POL	411
CLC2SEL0	—	—	LC2D1S<5:0>						412
CLC2SEL1	—	—	LC2D2S<5:0>						412
CLC2SEL2	—	—	LC2D3S<5:0>						412
CLC2SEL3	—	—	LC2D4S<5:0>						412
CLC2GLS0	—	—	LC2G1D3T	LC2G1D3N	LC2G1D2T	LC2G1D2N	LC2G1D1T	LC2G1D1N	413
CLC2GLS1	—	—	LC2G2D3T	LC2G2D3N	LC2G2D2T	LC2G2D2N	LC2G2D1T	LC2G2D1N	414
CLC2GLS2	—	—	LC2G3D3T	LC2G3D3N	LC2G3D2T	LC2G3D2N	LC2G3D1T	LC2G3D1N	415
CLC2GLS3	—	—	LC2G4D3T	LC2G4D3N	LC2G4D2T	LC2G4D2N	LC2G4D1T	LC2G4D1N	416
CLC3CON	LC3EN	—	LC3OUT	LC3INTP	LC3INTN	LC3MODE<2:0>			410
CLC3POL	LC3POL	—	—	—	LC3G4POL	LC3G3POL	LC3G2POL	LC3G1POL	411
CLC3SEL0	—	—	LC3D1S<5:0>						412
CLC3SEL1	—	—	LC3D2S<5:0>						412
CLC3SEL2	—	—	LC3D3S<5:0>						412
CLC3SEL3	—	—	LC3D4S<5:0>						412
CLC3GLS0	—	—	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	413
CLC3GLS1	—	—	LC3G2D3T	LC3G2D3N	LC3G2D2T	LC3G2D2N	LC3G2D1T	LC3G2D1N	414
CLC3GLS2	—	—	LC3G3D3T	LC3G3D3N	LC3G3D2T	LC3G3D2N	LC3G3D1T	LC3G3D1N	415
CLC3GLS3	—	—	LC3G4D3T	LC3G4D3N	LC3G4D2T	LC3G4D2N	LC3G4D1T	LC3G4D1N	416
CLC4CON	LC4EN	—	LC4OUT	LC4INTP	LC4INTN	LC4MODE<2:0>			410
CLC4POL	LC4POL	—	—	—	LC4G4POL	LC4G3POL	LC4G2POL	LC4G1POL	411
CLC4SEL0	—	—	LC4D1S<5:0>						412
CLC4SEL1	—	—	LC4D2S<5:0>						412
CLC4SEL2	—	—	LC4D3S<5:0>						412
CLC4SEL3	—	—	LC4D4S<5:0>						412
CLC4GLS0	—	—	LC4G1D3T	LC4G1D3N	LC4G1D2T	LC4G1D2N	LC4G1D1T	LC4G1D1N	413

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

32.2 SPI Mode Overview

The Serial Peripheral Interface (SPI) bus is a synchronous serial data communication bus that operates in Full-Duplex mode. Devices communicate in a master/slave environment where the master device initiates the communication. A slave device is controlled through a Chip Select known as Slave Select.

The SPI bus specifies four signal connections:

- Serial Clock (SCK)
- Serial Data Out (SDO)
- Serial Data In (SDI)
- Slave Select ($\overline{SS}$)

Figure 32-1 shows the block diagram of the MSSP module when operating in SPI mode.

The SPI bus operates with a single master device and one or more slave devices. When multiple slave devices are used, an independent Slave Select connection is required from the master device to each slave device.

Figure 32-4 shows a typical connection between a master device and multiple slave devices.

The master selects only one slave at a time. Most slave devices have tri-state outputs so their output signal appears disconnected from the bus when they are not selected.

Transmissions involve two shift registers, eight bits in size, one in the master and one in the slave. Data is always shifted out one bit at a time, with the Most Significant bit (MSb) shifted out first. At the same time, a new Least Significant bit (LSb) is shifted into the same register.

Figure 32-5 shows a typical connection between two processors configured as master and slave devices.

Data is shifted out of both shift registers on the programmed clock edge and latched on the opposite edge of the clock.

The master device transmits information out on its SDO output pin which is connected to, and received by, the slave's SDI input pin. The slave device transmits information out on its SDO output pin, which is connected to, and received by, the master's SDI input pin.

To begin communication, the master device first sends out the clock signal. Both the master and the slave devices should be configured for the same clock polarity.

The master device starts a transmission by sending out the MSb from its shift register. The slave device reads this bit from that same line and saves it into the LSb position of its shift register.

During each SPI clock cycle, a full-duplex data transmission occurs. This means that while the master device is sending out the MSb from its shift register (on its SDO pin) and the slave device is reading this bit and saving it as the LSb of its shift register, that the slave device is also sending out the MSb from its shift register (on its SDO pin) and the master device is reading this bit and saving it as the LSb of its shift register.

After eight bits have been shifted out, the master and slave have exchanged register values.

If there is more data to exchange, the shift registers are loaded with new data and the process repeats itself.

Whether the data is meaningful or not (dummy data), depends on the application software. This leads to three scenarios for data transmission:

- Master sends useful data and slave sends dummy data.
- Master sends useful data and slave sends useful data.
- Master sends dummy data and slave sends useful data.

Transmissions may involve any number of clock cycles. When there is no more data to be transmitted, the master stops sending the clock signal and it deselects the slave.

Every slave device connected to the bus that has not been selected through its slave select line must disregard the clock and transmission signals and must not transmit out any data of its own.

32.6.13.2 Bus Collision During a Repeated Start Condition

During a Repeated Start condition, a bus collision occurs if:

a) A low level is sampled on SDA when SCL goes from low level to high level (Case 1).
b) SCL goes low before SDA is asserted low, indicating that another master is attempting to transmit a data '1' (Case 2).

When the user releases SDA and the pin is allowed to float high, the BRG is loaded with SSPxADD and counts down to zero. The SCL pin is then deasserted and when sampled high, the SDA pin is sampled.

If SDA is low, a bus collision has occurred (i.e., another master is attempting to transmit a data '0', Figure 32-36). If SDA is sampled high, the BRG is reloaded and begins counting. If SDA goes from high-to-low before the BRG times out, no bus collision occurs because no two masters can assert SDA at exactly the same time.

If SCL goes from high-to-low before the BRG times out and SDA has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition, see Figure 32-37.

If, at the end of the BRG time-out, both SCL and SDA are still high, the SDA pin is driven low and the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCL pin, the SCL pin is driven low and the Repeated Start condition is complete.

FIGURE 32-36: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

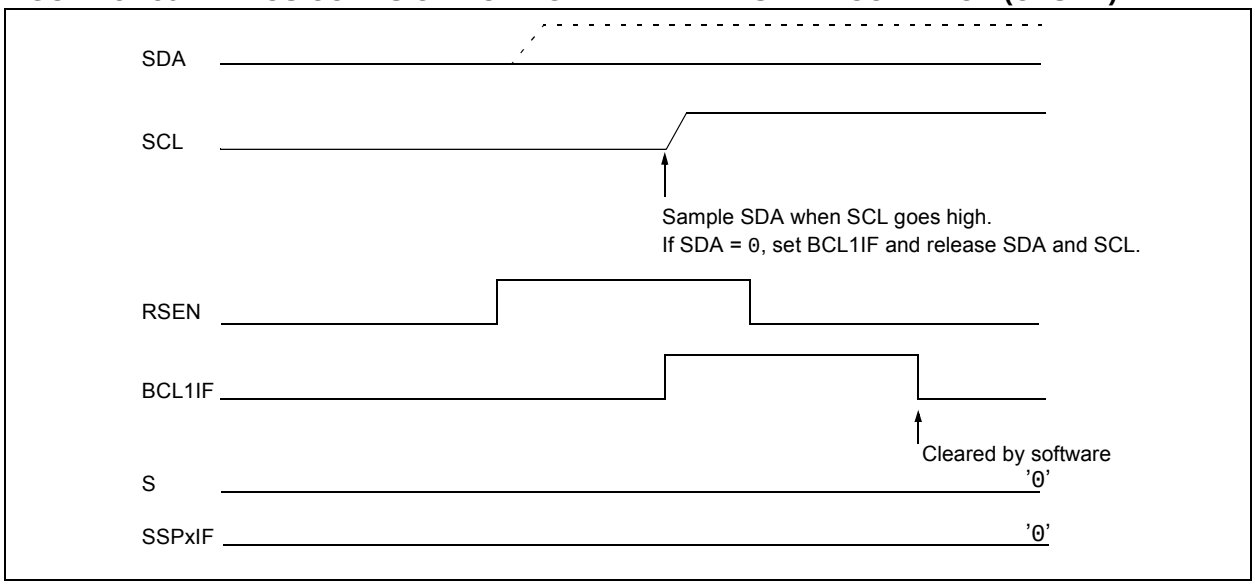
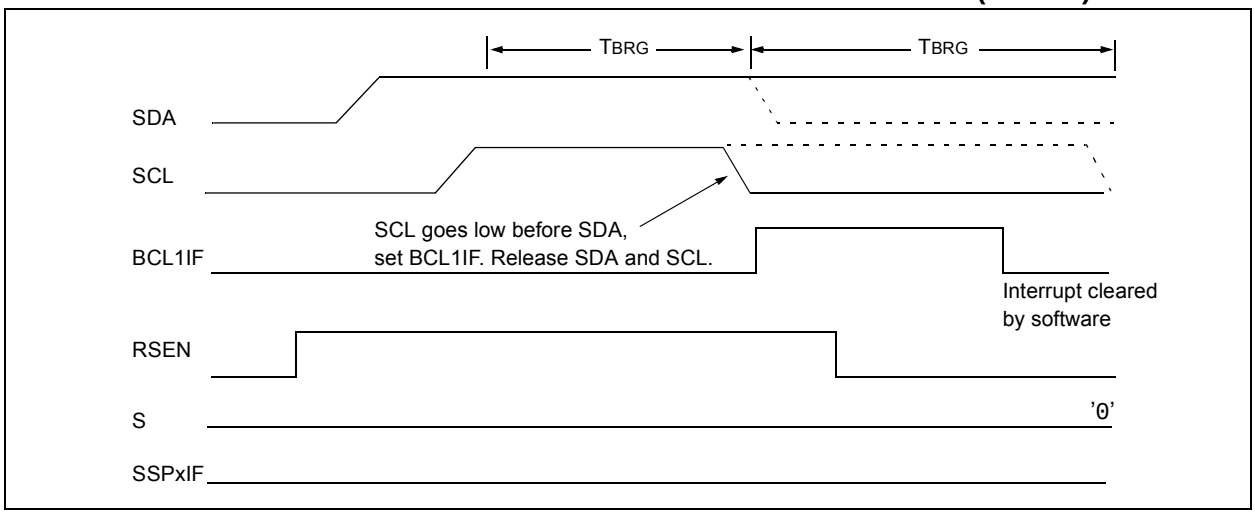


FIGURE 32-37: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



TRIS	Load TRIS Register with W
Syntax:	[*label*] TRIS f
Operands:	$5 \le f \le 7$
Operation:	(W) → TRIS register ‘f’
Status Affected:	None
Description:	Move data from W register to TRIS register. When ‘f’ = 5, TRISA is loaded. When ‘f’ = 6, TRISB is loaded. When ‘f’ = 7, TRISC is loaded.

XORLW	Exclusive OR literal with W
Syntax:	[*label*] XORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .XOR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are XOR’ed with the 8-bit literal ‘k’. The result is placed in the W register.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .XOR. (f) → (destination)
Status Affected:	Z
Description:	Exclusive OR the contents of the W register with register ‘f’. If ‘d’ is ‘0’, the result is stored in the W register. If ‘d’ is ‘1’, the result is stored back in register ‘f’.

TABLE 37-17: ZERO CROSS DETECT (ZCD) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C							
Param. No.	**Sym.**	**Characteristics**	**Min.**	**Typ†**	**Max.**	**Units**	**Comments**
ZC01	V_{PINZC}	Voltage on Zero Cross Pin	—	0.75	—	V	
ZC02	I_{ZCD_MAX}	Maximum source or sink current	—	—	600	µA	
ZC03	T_{RESPH}	Response Time, Rising Edge	—	1	—	µs	
	T_{RESPL}	Response Time, Falling Edge	—	1	—	µs	

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 37-12: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

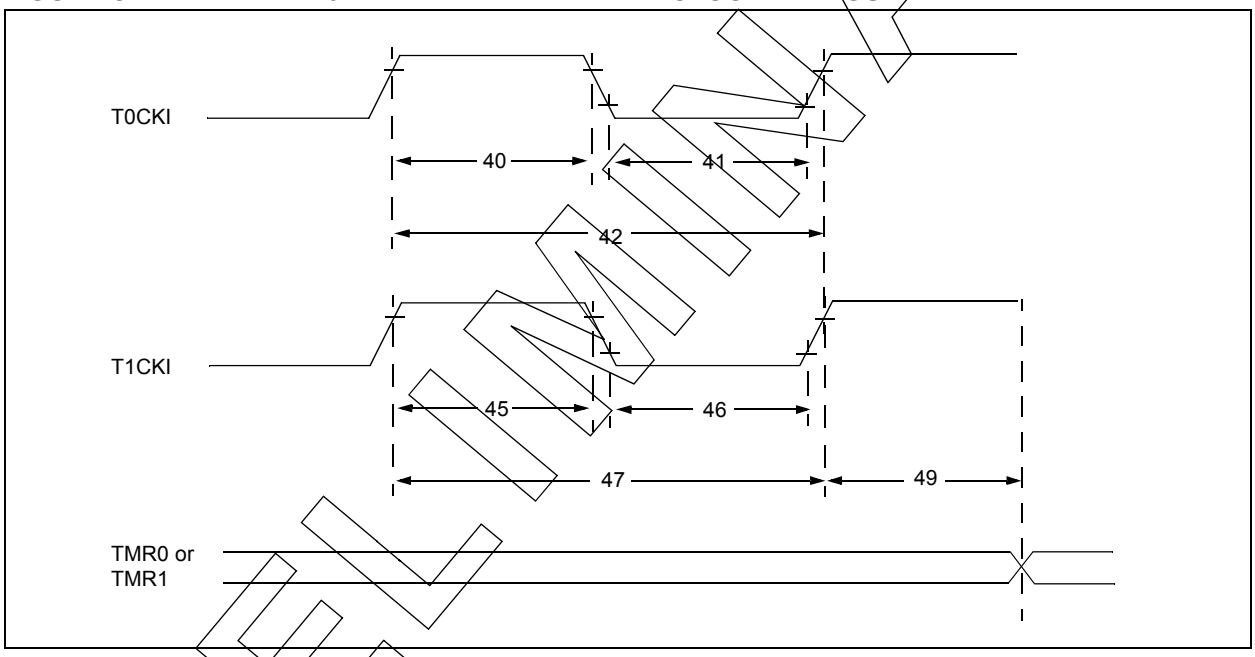


FIGURE 37-19: SPI SLAVE MODE TIMING (CKE = 0)

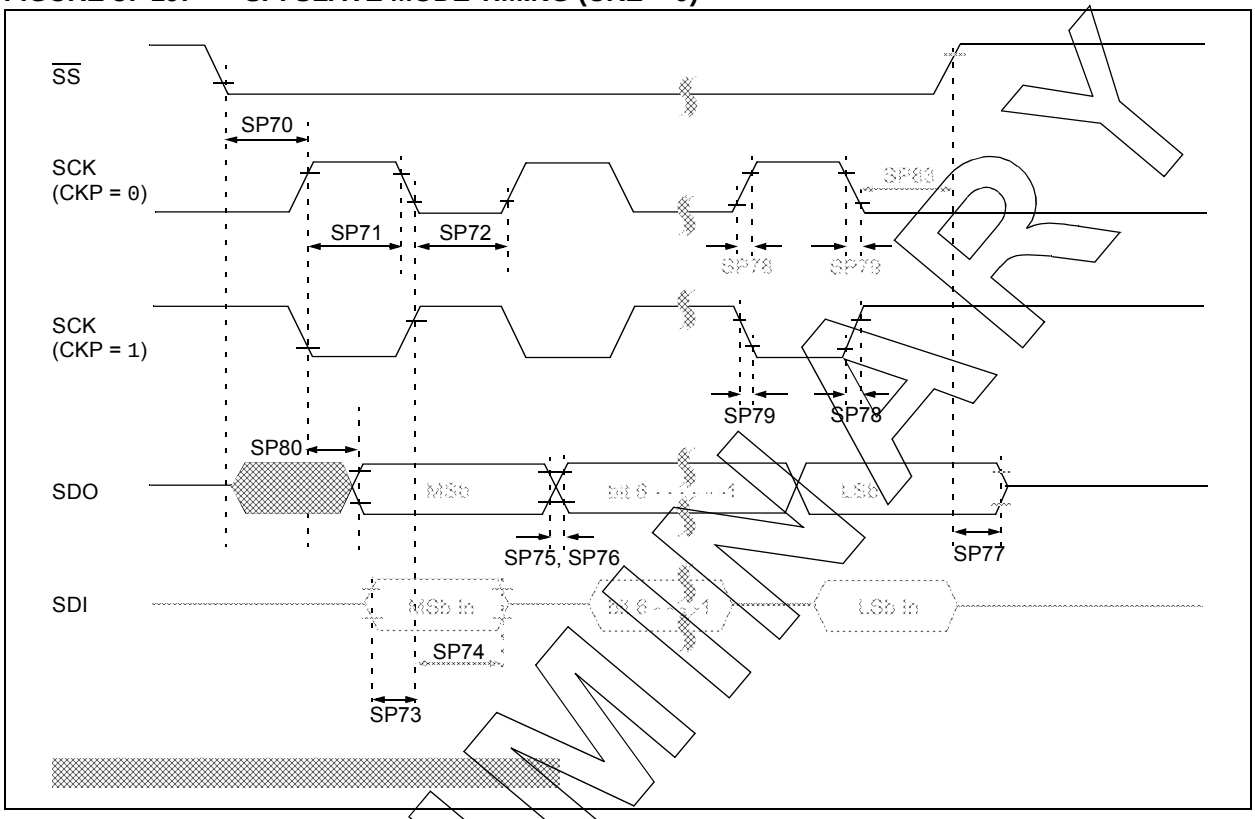


FIGURE 37-20: SPI SLAVE MODE TIMING (CKE = 1)