



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	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	11
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	14-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1554-i-st

TABLE 1-3: PIC16LF1559 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/ICSPDAT/ICDDAT	RA0	TTL	CMOS	General Purpose I/O
	AN0	AN	—	ADC Channel Input
	ICSPDAT	ST	CMOS	ICSP™ Data I/O
	ICDDAT	ST	CMOS	In-Circuit Debug Data
RA1/AN1/VREF+/ICSPCLK/ICDCLK	RA1	TTL	CMOS	General Purpose I/O
	AN1	AN	—	ADC Channel Input
	VREF+	AN	—	ADC Positive Voltage Reference Input
	ICSPCLK	ST	CMOS	ICSP Programming Clock
	ICDCLK	ST	CMOS	In-Circuit Debug Clock
RA2/AN2/TOCKI/INT	RA2	TTL	CMOS	General Purpose I/O
	AN2	AN	—	ADC Channel Input
	TOCKI	ST	—	Timer0 Clock Input
	INT	ST	—	External Interrupt
RA3/VPP/ $\overline{SS}^{(1)}$ /SDA ⁽¹⁾ /SDI ⁽¹⁾ / \overline{MCLR}	RA3	TTL	CMOS	General Purpose Input with IOC and WPU
	VPP	HV	—	Programming Voltage
	\overline{SS}	ST	—	Slave Select Input
	SDA	I ² C	OD	I ² C Data Input/Output
	SDI	CMOS	—	SPI Data Input
	\overline{MCLR}	ST	—	Master Clear with Internal Pull-up
RA4/AN10/ADTRIG/CLKOUT/T1G	RA4	TTL	CMOS	General Purpose I/O
	AN10	AN	—	ADC Channel Input
	ADTRIG	ST	—	ADC Conversion Trigger Input
	CLKOUT	—	CMOS	Fosc/4 Output
	T1G	ST	—	Timer1 Gate input.
RA5/AN20/CLKIN/T1CKI	RA5	TTL	CMOS	General Purpose I/O
	AN20	AN	—	ADC Channel Input
	CLKIN	CMOS	—	External Clock Input (EC mode)
	T1CKI	ST	—	Timer1 clock Input
RB4/AN26/SDA ⁽¹⁾ /SDI ⁽¹⁾	RB4	TTL	CMOS	General Purpose I/O
	AN26	AN	—	ADC Channel Input
	SDA	I ² C	OD	I ² C Data Input/Output
	SDI	CMOS	—	SPI Data 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 I²C = Schmitt Trigger input with I²C levels
HV = High Voltage XTAL = Crystal

Note 1: Alternate pin function selected with the APFCON (Register 11-1) register.

PIC16LF1554/1559

TABLE 3-9: SPECIAL FUNCTION REGISTER SUMMARY

Address	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 0											
000h	INDF0 ⁽¹⁾	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
001h	INDF1 ⁽¹⁾	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
002h	PCL ⁽¹⁾	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
003h	STATUS ⁽¹⁾	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	---1 1000	---q quuu
004h	FSR0L ⁽¹⁾	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
005h	FSR0H ⁽¹⁾	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
006h	FSR1L ⁽¹⁾	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
007h	FSR1H ⁽¹⁾	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
008h	BSR ⁽¹⁾	—	—	—	BSR<4:0>					---0 0000	---0 0000
009h	WREG ⁽¹⁾	Working Register								0000 0000	uuuu uuuu
00Ah	PCLATH ⁽¹⁾	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
00Bh	INTCON ⁽¹⁾	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
00Ch	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--xx xxxx
00Dh	PORTB ⁽²⁾	Unimplemented								—	—
	PORTB ⁽³⁾	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	xxxx ----
00Eh	PORTC ⁽²⁾	—	—	RC5	RC4	RC3	RC2	RC1	RC0	--xx xxxx	--xx xxxx
	PORTC ⁽³⁾	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	xxxx xxxx
011h	PIR1	TMR1GIF	AD1IF	RCIF	TXIF	SSP1IF	—	TMR2IF	TMR1IF	0000 0-00	0000 0-00
012h	PIR2	—	AD2IF	—	—	BCLIF	—	—	—	-0-- 0---	-0-- 0---
015h	TMR0	Timer0 Module Register								xxxx xxxx	uuuu uuuu
016h	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Count								xxxx xxxx	uuuu uuuu
017h	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Count								xxxx xxxx	uuuu uuuu
018h	T1CON	TMR1CS<1:0>		T1CKPS<1:0>		—	$\overline{\text{T1SYNC}}$	—	TMR1ON	0000 -0-0	uuuu -u-u
019h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	$\overline{\text{T1GGO/DONE}}$	T1GVAL	—	T1GSS	0000 0x-0	uuuu ux-u
01Ah	TMR2	Timer 2 Module Register								0000 0000	0000 0000
01Bh	PR2	Timer 2 Period Register								1111 1111	1111 1111
01Ch	T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		-000 0000	-000 0000
01Dh	—	Unimplemented								—	—
01Eh	—	Unimplemented								—	—
01Fh	—	Unimplemented								—	—

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

Note 1: These registers can be accessed from any bank.

2: PIC16LF1554.

3: PIC16LF1559.

4: These registers/bits are available at two address locations, in Bank 1 and Bank 14.

PIC16LF1554/1559

TABLE 3-9: SPECIAL FUNCTION REGISTER SUMMARY (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 all other Resets
Bank 14											
700h	INDF0 ⁽¹⁾	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
701h	INDF1 ⁽¹⁾	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
702h	PCL ⁽¹⁾	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
703h	STATUS ⁽¹⁾	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
704h	FSR0L ⁽¹⁾	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
705h	FSR0H ⁽¹⁾	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
706h	FSR1L ⁽¹⁾	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
707h	FSR1H ⁽¹⁾	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
708h	BSR ⁽¹⁾	—	—	—	BSR<4:0>					---0 0000	---0 0000
709h	WREG ⁽¹⁾	Working Register								0000 0000	uuuu uuuu
70Ah	PCLATH ⁽¹⁾	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
70Bh	INTCON ⁽¹⁾	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
70Ch	—	Unimplemented								—	—
70Dh	—	Unimplemented								—	—
70Eh	—	Unimplemented								—	—
70Fh	—	Unimplemented								—	—
710h	—	Unimplemented								—	—
711h	AD1CON0/ AAD1CON0 ⁽⁴⁾	—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE1	AD1ON	-000 0000	-000 0000
712h	AADCON1/ ADCOMCON ⁽⁴⁾	ADFM	ADCS<2:0>			—	GO/ DONE_ALL	ADPREF<1:0>		0000 -000	0000 -000
713h	AD1CON2/ AAD1CON2 ⁽⁴⁾	—	TRIGSEL<2:0>			—	—	—	—	-000 ----	-000 ----
714h	AAD1CON3	AD1EPPOL	AD1IPPOL	—	—	—	—	AD1IPEN	AD1DSEN	00-- --00	00-- --00
715h	AADSTAT	—	AD2CONV	AD2STG<1:0>		—	AD1CONV	AD1STG<1:0>		-000 -000	-000 -000
716h	AAD1PRE	—	AD1PRE<6:0>							-000 0000	-000 0000
717h	AAD1ACQ	—	AAD1ACQ<6:0>							-000 0000	-000 0000
718h	AAD1GRD	GRD1BOE	GRD1AOE	GRD1POL	—	—	—	—	—	000- ----	000- ----
719h	AAD1CAP	—	—	—	—	ADD1CAP<3:0>				---- 0000	---- 0000
71Ah	AD1RES0L/ AAD1RES0L ⁽⁴⁾	ADC Result 0 Register Low								xxxx xxxx	uuuu uuuu
71Bh	AD1RES0H/ AAD1RES0H ⁽⁴⁾	ADC Result 0 Register High								xxxx xxxx	uuuu uuuu
71Ch	AD1RES1L/ AAD1RES1L	ADC Result 1 Register Low								xxxx xxxx	uuuu uuuu
71Dh	AD1RES1H/ AAD1RES1H	ADC Result 1 Register High								xxxx xxxx	uuuu uuuu
71Eh	AAD1CH ⁽²⁾	—	—	—	—	CH13	CH12	CH11	CH10	---- 0000	---- 0000
	AAD1CH ⁽³⁾	—	CH16	CH15	CH14	CH13	CH12	CH11	CH10	-000 0000	-000 0000
71Fh	—	Unimplemented								—	—

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

- Note**
- 1: These registers can be accessed from any bank.
 - 2: PIC16LF1554.
 - 3: PIC16LF1559.
 - 4: These registers/bits are available at two address locations, in Bank 1 and Bank 14.

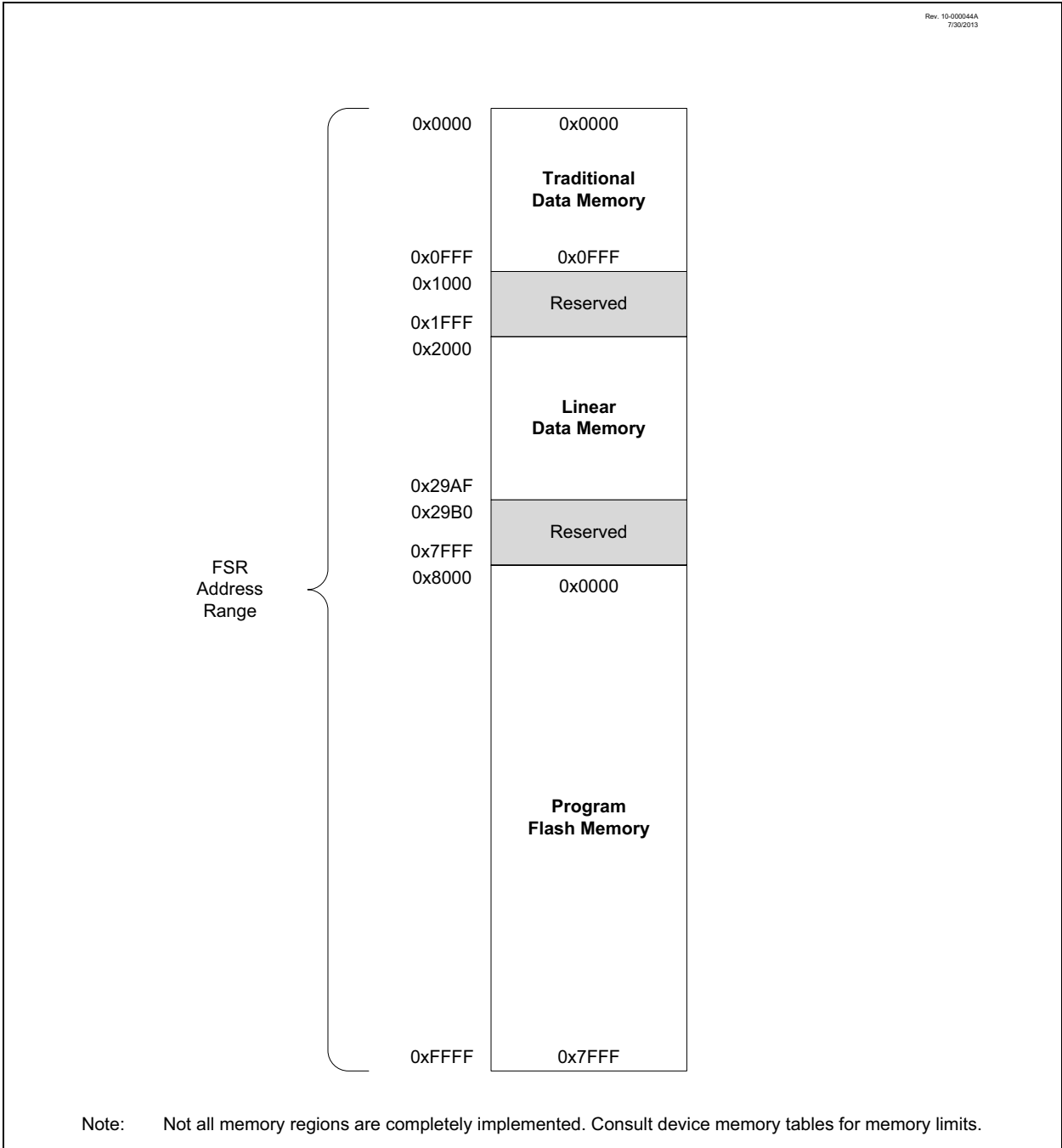
3.5 Indirect Addressing

The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the File Select Registers (FSR). If the FSRn address specifies one of the two INDFn registers, the read will return '0' and the write will not occur (though Status bits may be affected). The FSRn register value is created by the pair FSRnH and FSRnL.

The FSR registers form a 16-bit address that allows an addressing space with 65536 locations. These locations are divided into three memory regions:

- Traditional Data Memory
- Linear Data Memory
- Program Flash Memory

FIGURE 3-9: INDIRECT ADDRESSING



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 PWRT 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" (DS00000607).

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

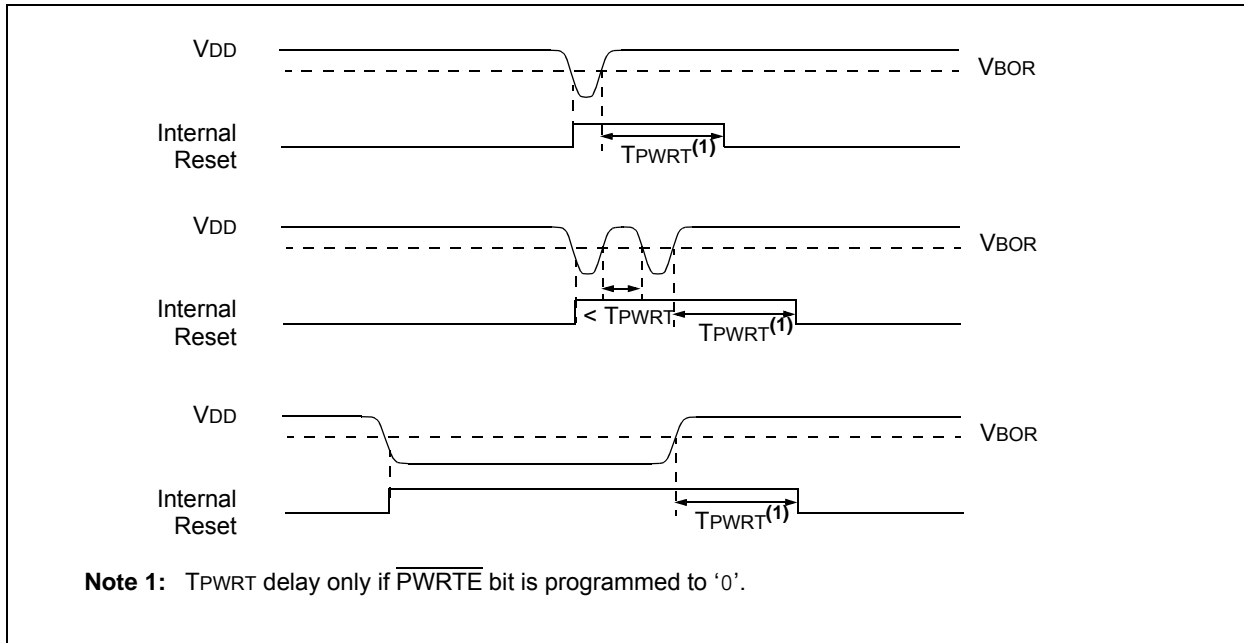
When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

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.

PIC16LF1554/1559

FIGURE 6-2: BROWN-OUT SITUATIONS



6.3 Register Definitions: BOR Control

REGISTER 6-1: BORCON: BROWN-OUT RESET CONTROL REGISTER

R/W-1/u	R/W-0/u	U-0	U-0	U-0	U-0	U-0	R-q/u
SBOREN	BORFS	—	—	—	—	—	BORRDY
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 **SBOREN:** Software Brown-Out Reset Enable bit

If BOREN <1:0> in Configuration Words = 01:

1 = BOR Enabled

0 = BOR Disabled

If BOREN <1:0> in Configuration Words ≠ 01:

SBOREN is read/write, but has no effect on the BOR

bit 6 **BORFS:** Brown-Out Reset Fast Start bit⁽¹⁾

If BOREN <1:0> = 10 (Disabled in Sleep) or BOREN <1:0> = 01 (Under software control):

1 = Band gap is forced on always (covers sleep/wake-up/operating cases)

0 = Band gap operates normally, and may turn off

If BOREN <1:0> = 11 (Always on) or BOREN <1:0> = 00 (Always off)

BORFS is Read/Write, but has no effect.

bit 5-1 **Unimplemented:** Read as '0'

bit 0 **BORRDY:** Brown-Out Reset Circuit Ready Status bit

1 = The Brown-out Reset circuit is active

0 = The Brown-out Reset circuit is inactive

Note 1: BOREN<1:0> bits are located in Configuration Words.

PIC16LF1554/1559

6.13 Power Control (PCON) Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-On Reset ($\overline{\text{POR}}$)
- Brown-Out Reset ($\overline{\text{BOR}}$)
- Reset Instruction Reset ($\overline{\text{RI}}$)
- MCLR Reset ($\overline{\text{RMCLR}}$)
- Watchdog Timer Reset ($\overline{\text{RWDT}}$)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)

The PCON register bits are shown in Register 6-2.

6.14 Register Definitions: Power Control

REGISTER 6-2: PCON: POWER CONTROL REGISTER

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

Legend:

HC = Bit is cleared by hardware

HS = Bit is set by hardware

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-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 **STKOVF:** Stack Overflow Flag bit

1 = A Stack Overflow occurred

0 = A Stack Overflow has not occurred or cleared by firmware

bit 6 **STKUNF:** Stack Underflow Flag bit

1 = A Stack Underflow occurred

0 = A Stack Underflow has not occurred or cleared by firmware

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

bit 4 **$\overline{\text{RWDT}}$:** Watchdog Timer Reset Flag bit

1 = A Watchdog Timer Reset has not occurred or set by firmware

0 = A Watchdog Timer Reset has occurred (cleared by hardware)

bit 3 **$\overline{\text{RMCLR}}$:** MCLR Reset Flag bit

1 = A $\overline{\text{MCLR}}$ Reset has not occurred or set by firmware

0 = A $\overline{\text{MCLR}}$ Reset has occurred (cleared by hardware)

bit 2 **$\overline{\text{RI}}$:** RESET Instruction Flag bit

1 = A RESET instruction has not been executed or set by firmware

0 = A RESET instruction has been executed (cleared by hardware)

bit 1 **$\overline{\text{POR}}$:** Power-On Reset Status bit

1 = No Power-on Reset occurred

0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{\text{BOR}}$:** Brown-Out Reset Status bit

1 = No Brown-out Reset occurred

0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

TABLE 6-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS⁽¹⁾

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BORCON	SBOREN	BORFS	—	—	—	—	—	BORRDY	66
PCON	STKOVF	STKUNF	—	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	70
STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	22
WDTCON	—	—	WDTPS<4:0>					SWDTEN	86

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

Note 1: Other (non Power-up) Resets include $\overline{\text{MCLR}}$ Reset and Watchdog Timer Reset during normal operation.

TABLE 6-6: SUMMARY OF CONFIGURATION WORD WITH RESETS

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG1	13:8	—	—	—	—	CLKOUTEN	BOREN<1:0>		—	53
	7:0	CP	MCLRE	PWRTE	WDTE<1:0>		—	FOSC<1:0>		
CONFIG2	13:8	—	—	LVP	DEBUG	LPBOR	BORV	STVREN	—	54
	7:0	—	—	—	—	—	—	WRT<1:0>		

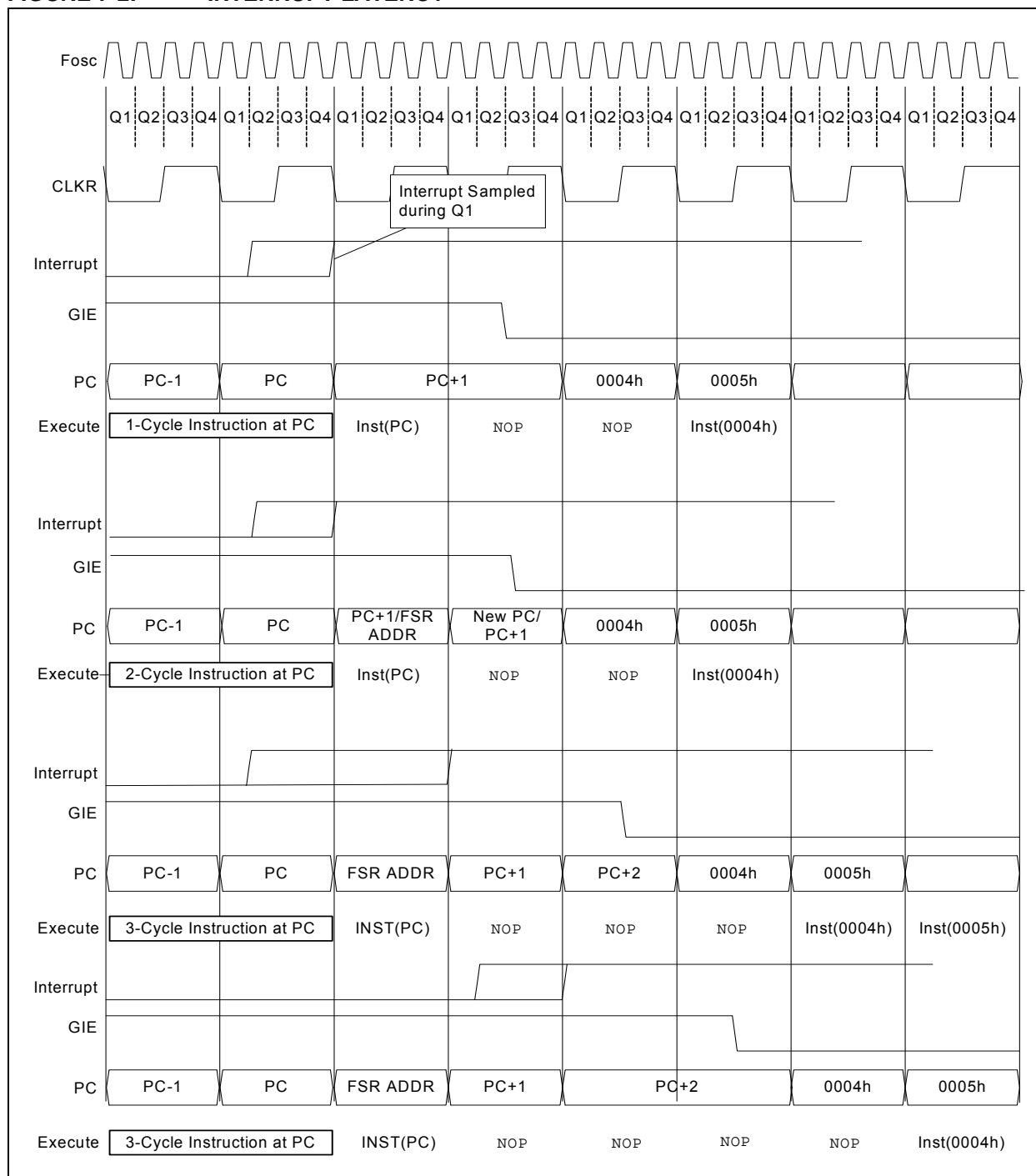
Legend: — = unimplemented location, read as '0'. Shaded cells are not used by Resets.

PIC16LF1554/1559

7.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The latency for synchronous interrupts is three or four instruction cycles. For asynchronous interrupts, the latency is three to five instruction cycles, depending on when the interrupt occurs. See Figure 7-2 and Figure 7-3 for more details.

FIGURE 7-2: INTERRUPT LATENCY



PIC16LF1554/1559

REGISTER 7-4: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
TMR1GIF	AD1IF	RCIF	TXIF	SSP1IF	—	TMR2IF	TMR1IF
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	TMR1GIF: Timer1 Gate Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 6	AD1IF: ADC 1 Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 5	RCIF: USART Receive Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 4	TXIF: USART Transmit Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 3	SSP1IF: Synchronous Serial Port (MSSP) Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 2	Unimplemented: Read as '0'
bit 1	TMR2IF: Timer2 to PR2 Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending
bit 0	TMR1IF: Timer1 Overflow Interrupt Flag bit 1 = Interrupt is pending 0 = Interrupt is not pending

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

PIC16LF1554/1559

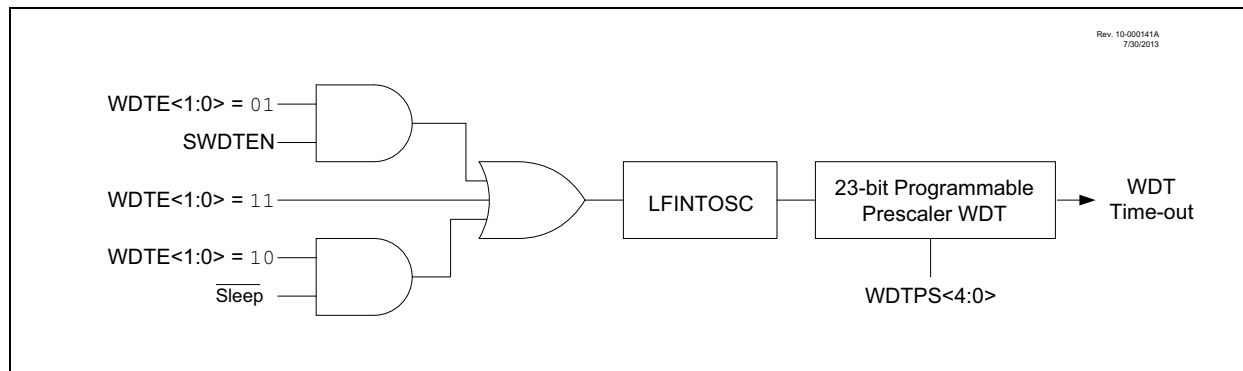
9.0 WATCHDOG TIMER (WDT)

The Watchdog Timer is a system timer that generates a Reset if the firmware does not issue a `CLRWDT` instruction within the time-out period. The Watchdog Timer is typically used to recover the system from unexpected events.

The WDT has the following features:

- Independent clock source
- Multiple operating modes
 - WDT is always on
 - WDT is off when in Sleep
 - WDT is controlled by software
 - WDT is always off
- Configurable time-out period is from 1 ms to 256 seconds (nominal)
- Multiple Reset conditions
- Operation during Sleep

FIGURE 9-1: WATCHDOG TIMER BLOCK DIAGRAM



EXAMPLE 10-2: ERASING ONE ROW OF PROGRAM MEMORY

```

; This row erase routine assumes the following:
; 1. A valid address within the erase row is loaded in ADDRHL:ADDRL
; 2. ADDRHL and ADDRL are located in shared data memory 0x70 - 0x7F (common RAM)

        BCF      INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL  PMADRL
        MOVF     ADDRL,W         ; Load lower 8 bits of erase address boundary
        MOVWF    PMADRL
        MOVF     ADDRHL,W        ; Load upper 6 bits of erase address boundary
        MOVWF    PMADRH
        BCF      PMCON1,CFGSR    ; Not configuration space
        BSF      PMCON1,FREER    ; Specify an erase operation
        BSF      PMCON1,WREN     ; Enable writes

        MOVLW    55h             ; Start of required sequence to initiate erase
        MOVWF    PMCON2          ; Write 55h
        MOVLW    0AAh           ;
        MOVWF    PMCON2          ; Write AAh
        BSF      PMCON1,WR       ; Set WR bit to begin erase
        NOP                      ; NOP instructions are forced as processor starts
        NOP                      ; row erase of program memory.
        NOP                      ;
        NOP                      ; The processor stalls until the erase process is complete
        NOP                      ; after erase processor continues with 3rd instruction

        BCF      PMCON1,WREN     ; Disable writes
        BSF      INTCON,GIE      ; Enable interrupts

```

Required
Sequence

11.5 PORTB Registers (PIC16LF1559 Only)

11.5.1 DATA REGISTER

PORTB is a 4-bit wide, bidirectional port. The corresponding data direction register is TRISB (Register 11-8). Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., disable the output driver). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 11-1 shows how to initialize an I/O port.

Reading the PORTB register (Register 11-7) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATB).

11.5.2 DIRECTION CONTROL

The TRISB register (Register 11-8) controls the PORTB pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISB register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

11.5.3 ANALOG CONTROL

The ANSELB register (Register 11-10) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELB bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELB bits has no effect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELB bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSELx bits must be initialized to '0' by user software.

11.5.4 PORTB FUNCTIONS AND OUTPUT PRIORITIES

Each PORTB pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 11-5.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input functions, such as ADC and comparator inputs, are not shown in the priority lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELx registers. Digital output functions may control the pin when it is in Analog mode with the priority shown below in Table 11-5.

TABLE 11-5: PORTB OUTPUT PRIORITY

Pin Name	Function Priority ⁽¹⁾
RB4	SDA RB4
RB5	RB5
RB6	SCL SCK RB6
RB7	TX RB7

Note 1: Priority listed from highest to lowest.

PIC16LF1554/1559

18.5.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

18.5.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

18.5.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 18-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note: Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation.

18.5.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/DONE bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/DONE bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/DONE bit is once again set in software. See Figure 18-5 for timing details.

If the Single Pulse Gate mode is disabled by clearing the T1GSPM bit in the T1GCON register, the T1GGO/DONE bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See Figure 18-6 for timing details.

18.5.5 TIMER1 GATE VALUE STATUS

When Timer1 Gate Value Status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

18.5.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 Gate Event Interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

20.5.3 SLAVE TRANSMISSION

When the $\overline{R/W}$ bit of the incoming address byte is set and an address match occurs, the $\overline{R/W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register, and an \overline{ACK} pulse is sent by the slave on the ninth bit.

Following the \overline{ACK} , slave hardware clears the CKP bit and the SCL pin is held low (see **Section 20.5.6 “Clock Stretching”** for more detail). By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data.

The transmit data must be loaded into the SSPBUF register which also loads the SSPSR register. Then the SCL pin should be released by setting the CKP bit of the SSPCON1 register. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time.

The \overline{ACK} pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. This \overline{ACK} value is copied to the ACKSTAT bit of the SSPCON2 register. If ACKSTAT is set (not \overline{ACK}), then the data transfer is complete. In this case, when the not \overline{ACK} is latched by the slave, the slave goes idle and waits for another occurrence of the Start bit. If the SDA line was low (\overline{ACK}), the next transmit data must be loaded into the SSPBUF register. Again, the SCL pin must be released by setting bit CKP.

An MSSP interrupt is generated for each data transfer byte. The SSP1IF bit must be cleared by software and the SSPSTAT register is used to determine the status of the byte. The SSP1IF bit is set on the falling edge of the ninth clock pulse.

20.5.3.1 Slave Mode Bus Collision

A slave receives a Read request and begins shifting data out on the SDA line. If a bus collision is detected and the SBCDE bit of the SSPCON3 register is set, the BCLIF bit of the PIRx register is set. Once a bus collision is detected, the slave goes Idle and waits to be addressed again. User software can use the BCLIF bit to handle a slave bus collision.

20.5.3.2 7-Bit Transmission

A master device can transmit a read request to a slave, and then clock data out of the slave. The list below outlines what software for a slave will need to do to accomplish a standard transmission. Figure 20-18 can be used as a reference to this list.

1. Master sends a Start condition on SDA and SCL.
2. S bit of SSPSTAT is set; SSP1IF is set if interrupt on Start detect is enabled.
3. Matching address with $\overline{R/W}$ bit set is received by the Slave setting SSP1IF bit.
4. Slave hardware generates an \overline{ACK} and sets SSP1IF.
5. SSP1IF bit is cleared by user.
6. Software reads the received address from SSPBUF, clearing BF.
7. $\overline{R/W}$ is set so CKP was automatically cleared after the \overline{ACK} .
8. The slave software loads the transmit data into SSPBUF.
9. CKP bit is set releasing SCL, allowing the master to clock the data out of the slave.
10. SSP1IF is set after the \overline{ACK} response from the master is loaded into the ACKSTAT register.
11. SSP1IF bit is cleared.
12. The slave software checks the ACKSTAT bit to see if the master wants to clock out more data.

Note 1: If the master \overline{ACK} s the clock will be stretched.

2: ACKSTAT is the only bit updated on the rising edge of SCL (9th) rather than the falling.

13. Steps 9-13 are repeated for each transmitted byte.
14. If the master sends a not \overline{ACK} ; the clock is not held, but SSP1IF is still set.
15. The master sends a Restart condition or a Stop.
16. The slave is no longer addressed.

FIGURE 20-19: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 1)



PIC16LF1554/1559

FIGURE 21-12: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)

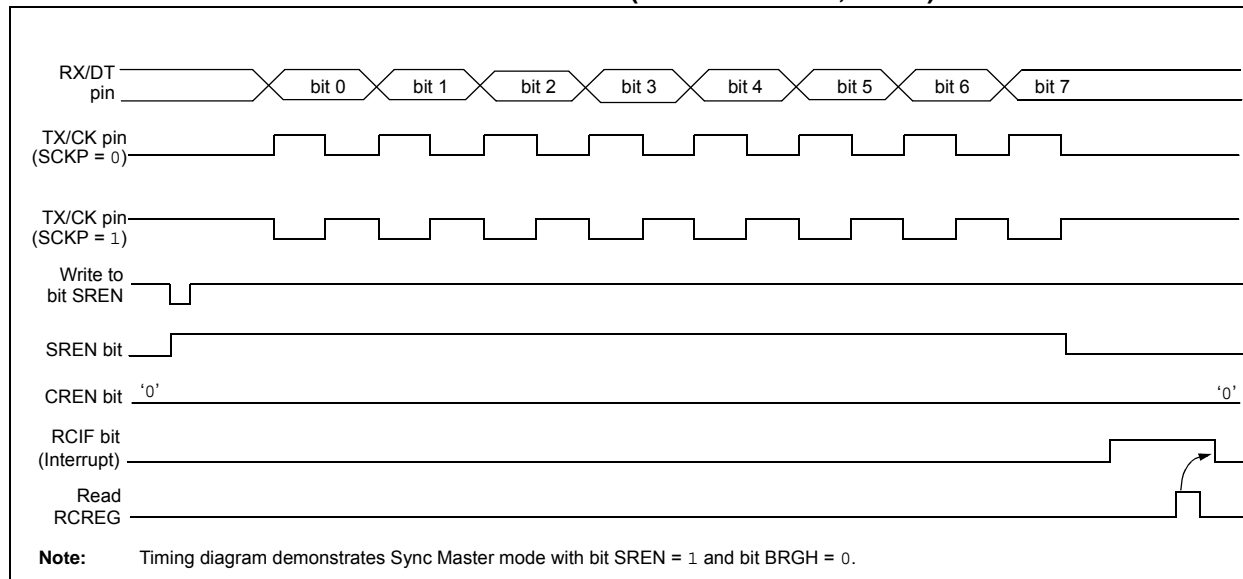


TABLE 21-8: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	245
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	77
PIE1	TMR1GIE	AD1IE	RCIE	TXIE	SSP1IE	—	TMR2IE	TMR1IE	78
PIR1	TMR1GIF	AD1IF	RCIF	TXIF	SSP1IF	—	TMR2IF	TMR1IF	80
RCREG	EUSART Receive Data Register								239*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	244
SPBRGL	BRG<7:0>								246*
SPBRGH	BRG<15:8>								246*
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	112
TXSTA	CSRC	TX9	TXEN	SYNC	SENDER	BRGH	TRMT	TX9D	243

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for synchronous master reception.

* Page provides register information.

27.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

27.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

27.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

27.9 PICkit 3 In-Circuit Debugger/Programmer

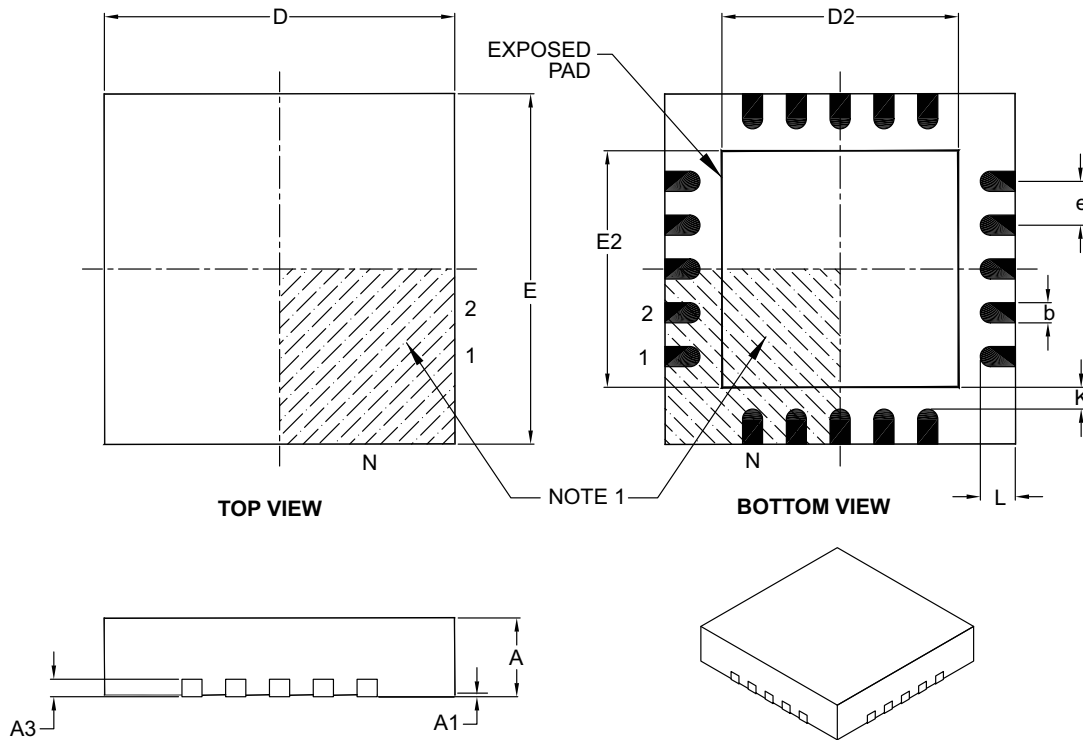
The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

27.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

20-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	20		
Pitch	e	0.50 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	4.00 BSC		
Exposed Pad Width	E2	2.60	2.70	2.80
Overall Length	D	4.00 BSC		
Exposed Pad Length	D2	2.60	2.70	2.80
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-126B

PIC16LF1554/1559

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

<u>PART NO.</u>	<u>[X]⁽¹⁾</u>	<u>X</u>	<u>/XX</u>	<u>XXX</u>
Device	Tape and Reel Option	Temperature Range	Package	Pattern
Device:	PIC16LF1554, PIC16LF1559.			
Tape and Reel Option:	Blank = Standard packaging (tube or tray) T = Tape and Reel ⁽¹⁾			
Package:⁽²⁾	P = Plastic DIP SL = SOIC ST = TSSOP ML = QFN SS = SSOP GZ = UQFN			
Pattern:	QTP, SQTP, Code or Special Requirements (blank otherwise)			

Examples:

- a) PIC16LF1559T/SS
Tape and Reel, SOIC package
- b) PIC16LF1554/P
PDIP package
- c) PIC16LF1559/ML 298
QFN package

Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

2: For other small form-factor package availability and marking information, please visit www.microchip.com/packaging or contact your local sales office.