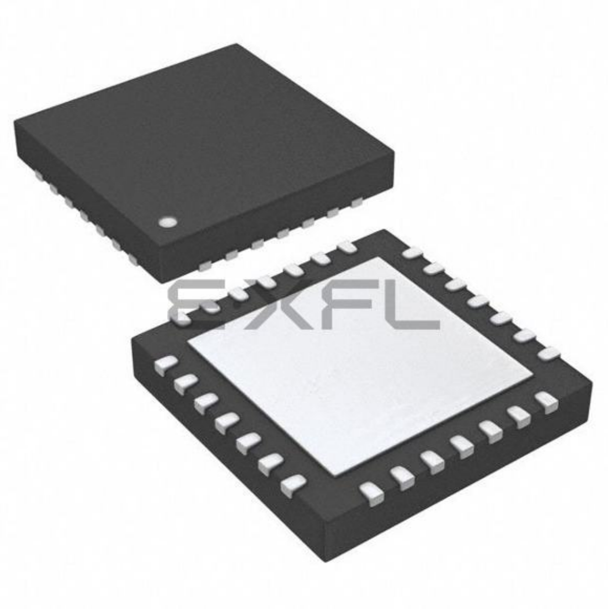


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 17x10b; D/A 1x5b, 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-UFQFN Exposed Pad
Supplier Device Package	28-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1713-e-mv

3.3.1 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation. The core registers occupy the first 12 addresses of every data memory bank (addresses x00h/x08h through x0Bh/x8Bh). These registers are listed below in Table 3-2. For detailed information, see Table 3-10.

TABLE 3-2: CORE REGISTERS

Addresses	**BANKx**
x00h or x80h	INDF0
x01h or x81h	INDF1
x02h or x82h	PCL
x03h or x83h	STATUS
x04h or x84h	FSR0L
x05h or x85h	FSR0H
x06h or x86h	FSR1L
x07h or x87h	FSR1H
x08h or x88h	BSR
x09h or x89h	WREG
x0Ah or x8Ah	PCLATH
x0Bh or x8Bh	INTCON

TABLE 3-11: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 14-27											
x0Ch/ x8Ch — x1Fh/ x9Fh	—	Unimplemented								—	—
Bank 28											
E0Ch — E0Eh	—	Unimplemented								—	—
E0Fh	PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	`---- ---0`	`---- ---0`
E10h	INTPPS	—	—	—	INTPPS<4:0>					`---0 1000`	`---u uuuu`
E11h	T0CKIPPS	—	—	—	T0CKIPPS<4:0>					`---0 0100`	`---u uuuu`
E12h	T1CKIPPS	—	—	—	T1CKIPPS<4:0>					`---1 0000`	`---u uuuu`
E13h	T1GPPS	—	—	—	T1GPPS<4:0>					`---0 1101`	`---u uuuu`
E14h	CCP1PPS	—	—	—	CCP1PPS<4:0>					`---1 0010`	`---u uuuu`
E15h	CCP2PPS	—	—	—	CCP2PPS<4:0>					`---1 0001`	`---u uuuu`
E16h	—	Unimplemented								—	—
E17h	COGINPPS	—	—	—	COGINPPS<4:0>					`---0 1000`	`---u uuuu`
E18h	—	Unimplemented								—	—
E19h	—	Unimplemented								—	—
E1Ah E1FH	—	Unimplemented								—	—
E20h	SSPCLKPPS	—	—	—	SSPCLKPPS<4:0>					`---1 0011`	`---u uuuu`
E21h	SSPDATPPS	—	—	—	SSPDATPPS<4:0>					`---1 0100`	`---u uuuu`
E22h	SSPSSPPS	—	—	—	SSPSSPPS<4:0>					`---0 0101`	`---u uuuu`
E23h	—	Unimplemented								—	—
E24h	RXPPS	—	—	—	RXPPS<4:0>					`---1 0111`	`---u uuuu`
E25h	CKPPS	—	—	—	CKPPS<4:0>					`---1 0110`	`---u uuuu`
E26h	—	Unimplemented								—	—
E27h	—	Unimplemented								—	—
E28h	CLCIN0PPS	—	—	—	CLCIN0PPS<4:0>					`---0 0000`	`---u uuuu`
E29h	CLCIN1PPS	—	—	—	CLCIN1PPS<4:0>					`---0 0001`	`---u uuuu`
E2Ah	CLCIN2PPS	—	—	—	CLCIN2PPS<4:0>					`---0 1110`	`---u uuuu`
E2Bh	CLCIN3PPS	—	—	—	CLCIN3PPS<4:0>					`---0 1111`	`---u uuuu`
E2Ch to E6Fh	—	Unimplemented								—	—

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

Note 1: Unimplemented, read as '1'.

2: Unimplemented on PIC16(L)F1713/6.

© 2013-2016 Microchip Technology Inc.

REGISTER 4-2: CONFIG2: CONFIGURATION WORD 2

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
LVP[1]	$\overline{\text{DEBUG}}$[2]	$\overline{\text{LPBOR}}$	BORV[3]	STVREN	PLLEN
bit 13					bit 8

R/P-1	U-1	U-1	U-1	U-1	R/P-1	R/P-1	R/P-1
ZCDDIS	—	—	—	—	PPS1WAY	WRT<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘1’
‘0’ = Bit is cleared	‘1’ = Bit is set	-n = Value when blank or after Bulk Erase

bit 13 **LVP:** Low-Voltage Programming Enable bit[1]
1 = Low-voltage programming enabled
0 = High-voltage on $\overline{\text{MCLR}}$ must be used for programming

bit 12 **$\overline{\text{DEBUG}}$:** In-Circuit Debugger Mode bit[2]
1 = In-Circuit Debugger disabled, ICSPCLK and ICSPDAT are general purpose I/O pins
0 = In-Circuit Debugger enabled, ICSPCLK and ICSPDAT are dedicated to the debugger

bit 11 **$\overline{\text{LPBOR}}$:** Low-Power BOR Enable bit
1 = Low-Power Brown-out Reset is disabled
0 = Low-Power Brown-out Reset is enabled

bit 10 **BORV:** Brown-out Reset Voltage Selection bit[3]
1 = Brown-out Reset voltage (V_{BOR}), low trip point selected.
0 = Brown-out Reset voltage (V_{BOR}), high trip point selected.

bit 9 **STVREN:** Stack Overflow/Underflow Reset Enable bit
1 = Stack Overflow or Underflow will cause a Reset
0 = Stack Overflow or Underflow will not cause a Reset

bit 8 **PLLEN:** PLL Enable bit
1 = 4xPLL enabled
0 = 4xPLL disabled

bit 7 **ZCDDIS:** ZCD Disable bit
1 = ZCD disabled. ZCD can be enabled by setting the ZCDSEN bit of ZCDCON
0 = ZCD always enabled

bit 6-3 **Unimplemented:** Read as ‘1’

bit 2 **PPS1WAY**: PPSLOCK Bit One-Way Set Enable bit
1 = The PPSLOCK bit can only be set once after an unlocking sequence is executed; once PPSLOCK is set, all future changes to PPS registers are prevented
0 = The PPSLOCK bit can be set and cleared as needed (provided an unlocking sequence is executed)

bit 1-0 **WRT<1:0>:** Flash Memory Self-Write Protection bits
4 kW Flash memory
11 = Write protection off
10 = 000h to 1FFh write protected, 200h to FFFh may be modified by PMCON control
01 = 000h to 7FFh write protected, 800h to FFFh may be modified by PMCON control
00 = 000h to FFFh write protected, no addresses may be modified by PMCON control

Note 1: The LVP bit cannot be programmed to ‘0’ when Programming mode is entered via LVP.
2: The $\overline{\text{DEBUG}}$ bit in Configuration Words is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a ‘1’.
3: See V_{BOR} parameter for specific trip point voltages.

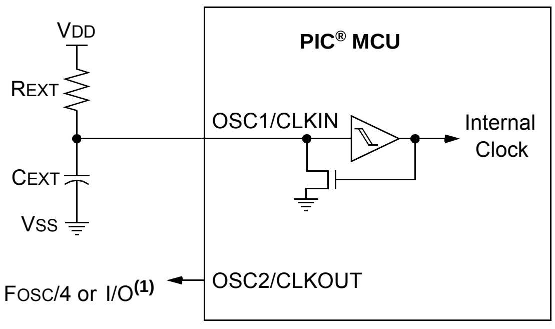
6.2.1.6 External RC Mode

The external Resistor-Capacitor (EXTRC) mode supports the use of an external RC circuit. This allows the designer maximum flexibility in frequency choice while keeping costs to a minimum when clock accuracy is not required.

The RC circuit connects to OSC1. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. The function of the OSC2/CLKOUT pin is determined by the $\overline{\text{CLKOUTEN}}$ bit in Configuration Words.

Figure 6-6 shows the external RC mode connections.

FIGURE 6-6: EXTERNAL RC MODES



Recommended values: 10 kΩ ≤ REXT ≤ 100 kΩ, <3V
3 kΩ ≤ REXT ≤ 100 kΩ, 3-5V
CEXT > 20 pF, 2-5V

Note 1: Output depends upon $\overline{\text{CLKOUTEN}}$ bit of the Configuration Words.

The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. Other factors affecting the oscillator frequency are:

- threshold voltage variation
- component tolerances
- packaging variations in capacitance

The user also needs to take into account variation due to tolerance of external RC components used.

6.2.2 INTERNAL CLOCK SOURCES

The device may be configured to use the internal oscillator block as the system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in Configuration Words to select the INTOSC clock source, which will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to the internal oscillator during run-time. See **Section 6.3 “Clock Switching”** for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT is available for general purpose I/O or CLKOUT.

The function of the OSC2/CLKOUT pin is determined by the $\overline{\text{CLKOUTEN}}$ bit in Configuration Words.

The internal oscillator block has two independent oscillators and a dedicated Phase-Lock Loop, HFPLL that can produce one of three internal system clock sources.

1. The **HFINTOSC** (High-Frequency Internal Oscillator) is factory calibrated and operates at 16 MHz. The HFINTOSC source is generated from the 500 kHz MFINTOSC source and the dedicated Phase-Lock Loop, HFPLL. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 6-3).
2. The **MFINTOSC** (Medium Frequency Internal Oscillator) is factory calibrated and operates at 500 kHz. The frequency of the MFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 6-3).
3. The **LFINTOSC** (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

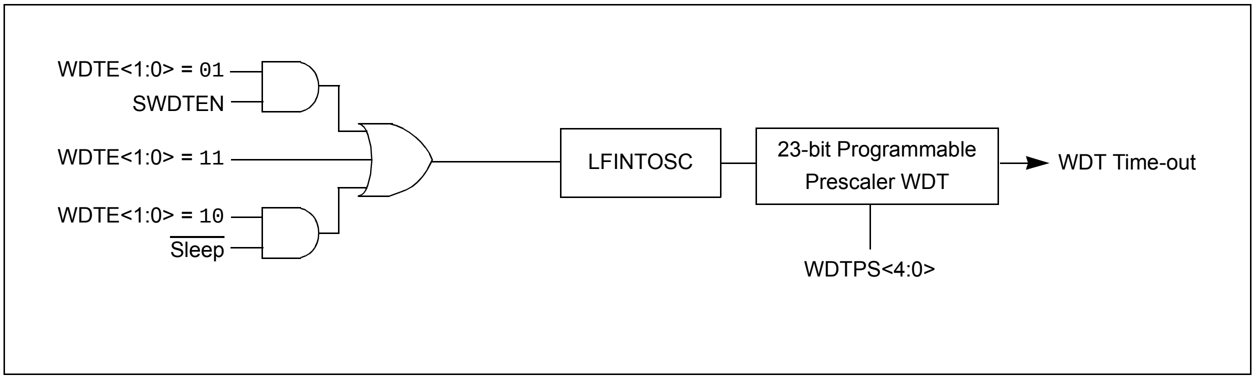
9.0 WATCHDOG TIMER (WDT)

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:

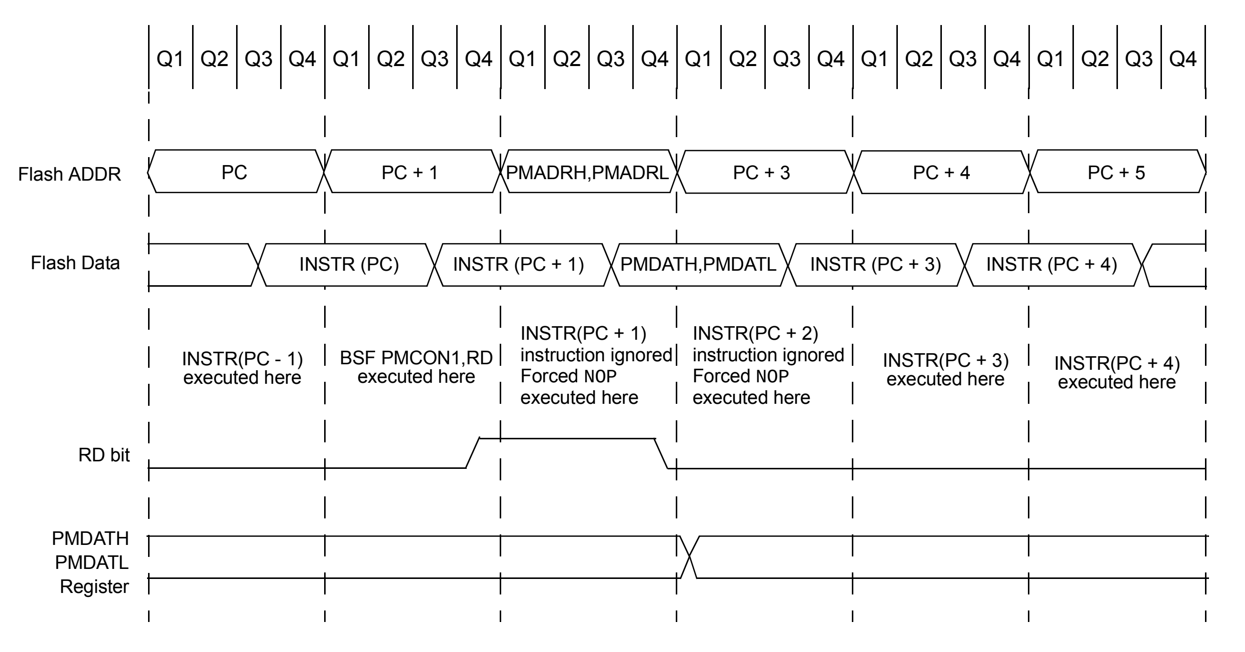
- Independent clock source
- Multiple operating modes
  - WDT is always on
  - WDT is off when in Sleep
  - WDT is controlled by software
  - WDT is always off
- Configurable time-out period is from 1 ms to 256 seconds (nominal)
- Multiple Reset conditions
- Operation during Sleep

FIGURE 9-1: WATCHDOG TIMER BLOCK DIAGRAM



© 2013-2016 Microchip Technology Inc.

FIGURE 10-2: FLASH PROGRAM MEMORY READ CYCLE EXECUTION



EXAMPLE 10-1: FLASH PROGRAM MEMORY READ

```
* This code block will read 1 word of program
* memory at the memory address:
   PROG_ADDR_HI : PROG_ADDR_LO
*   data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

    BANKSEL  PMADRL           ; Select Bank for PMCON registers
    MOVLW    PROG_ADDR_LO     ;
    MOVWF    PMADRL           ; Store LSB of address
    MOVLW    PROG_ADDR_HI     ;
    MOVWF    PMADRH           ; Store MSB of address

    BCF      PMCON1,CFGS      ; Do not select Configuration Space
    BSF      PMCON1,RD        ; Initiate read
    NOP                       ; Ignored (Figure 10-1)
    NOP                       ; Ignored (Figure 10-1)

    MOVF     PMDATL,W         ; Get LSB of word
    MOVWF    PROG_DATA_LO     ; Store in user location
    MOVF     PMDATH,W         ; Get MSB of word
    MOVWF    PROG_DATA_HI     ; Store in user location
```

© 2013-2016 Microchip Technology Inc.

11.3 PORTB Registers

PORTB is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISB (Register 11-10). Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., enable the output driver and put 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 11-9) 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).

11.3.1 DIRECTION CONTROL

The TRISB register (Register 11-10) 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 inputs always read ‘0’.

11.3.2 OPEN-DRAIN CONTROL

The ODCONB register (Register 11-14) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONB bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONB bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

11.3.3 SLEW RATE CONTROL

The SLRCONB register (Register 11-15) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONB bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONB bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

11.3.4 INPUT THRESHOLD CONTROL

The INLVLB register (Register 11-16) controls the input voltage threshold for each of the available PORTB input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTB register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 34-4: I/O Ports for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

11.3.5 ANALOG CONTROL

The ANSELB register (Register 11-12) 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 ANSELB 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 ANSEL bits must be initialized to ‘0’ by user software.

11.3.6 PORTB FUNCTIONS AND OUTPUT PRIORITIES

Each pin defaults to the PORT latch data after reset. Other functions are selected with the peripheral pin select logic. See **Section 12.0 “Peripheral Pin Select (PPS) Module”** for more information. Analog input functions, such as ADC and Op Amp inputs, are not shown in the peripheral pin select lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELB register. Digital output functions continue to may continue to control the pin when it is in Analog mode.

© 2013-2016 Microchip Technology Inc.

19.1.2 DATA GATING

Outputs from the input multiplexers are directed to the desired logic function input through the data gating stage. Each data gate can direct any combination of the four selected inputs.

Note: Data gating is undefined at power-up.

The gate stage is more than just signal direction. The gate can be configured to direct each input signal as inverted or non-inverted data. Directed signals are ANDed together in each gate. The output of each gate can be inverted before going on to the logic function stage.

The gating is in essence a 1-to-4 input AND/NAND/OR/NOR gate. When every input is inverted and the output is inverted, the gate is an OR of all enabled data inputs. When the inputs and output are not inverted, the gate is an AND or all enabled inputs.

Table 19-2 summarizes the basic logic that can be obtained in gate 1 by using the gate logic select bits. The table shows the logic of four input variables, but each gate can be configured to use less than four. If no inputs are selected, the output will be zero or one, depending on the gate output polarity bit.

TABLE 19-2: DATA GATING LOGIC

CLCxGLS0	LCxG1POL	Gate Logic
0x55	1	AND
0x55	0	NAND
0xAA	1	NOR
0xAA	0	OR
0x00	0	Logic 0
0x00	1	Logic 1

It is possible (but not recommended) to select both the true and negated values of an input. When this is done, the gate output is zero, regardless of the other inputs, but may emit logic glitches (transient-induced pulses). If the output of the channel must be zero or one, the recommended method is to set all gate bits to zero and use the gate polarity bit to set the desired level.

Data gating is configured with the logic gate select registers as follows:

- Gate 1: CLCxGLS0 (Register 19-7)
- Gate 2: CLCxGLS1 (Register 19-8)
- Gate 3: CLCxGLS2 (Register 19-9)
- Gate 4: CLCxGLS3 (Register 19-10)

Register number suffixes are different than the gate numbers because other variations of this module have multiple gate selections in the same register.

Data gating is indicated in the right side of Figure 19-2. Only one gate is shown in detail. The remaining three gates are configured identically with the exception that the data enables correspond to the enables for that gate.

19.1.3 LOGIC FUNCTION

There are eight available logic functions including:

- AND-OR
- OR-XOR
- AND
- S-R Latch
- D Flip-Flop with Set and Reset
- D Flip-Flop with Reset
- J-K Flip-Flop with Reset
- Transparent Latch with Set and Reset

Logic functions are shown in Figure 19-3. Each logic function has four inputs and one output. The four inputs are the four data gate outputs of the previous stage. The output is fed to the inversion stage and from there to other peripherals, an output pin, and back to the CLCx itself.

19.1.4 OUTPUT POLARITY

The last stage in the configurable logic cell is the output polarity. Setting the LCxPOL bit of the CLCxCON register inverts the output signal from the logic stage. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

REGISTER 19-2: CLCxPOL: SIGNAL POLARITY CONTROL REGISTER

R/W-0/0	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LCxPOL	—	—	—	LCxG4POL	LCxG3POL	LCxG2POL	LCxG1POL
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 **LCxPOL:** LCOUT Polarity Control bit
1 = The output of the logic cell is inverted
0 = The output of the logic cell is not inverted

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **LCxG4POL:** Gate 4 Output Polarity Control bit
1 = The output of gate 4 is inverted when applied to the logic cell
0 = The output of gate 4 is not inverted

bit 2 **LCxG3POL:** Gate 3 Output Polarity Control bit
1 = The output of gate 3 is inverted when applied to the logic cell
0 = The output of gate 3 is not inverted

bit 1 **LCxG2POL:** Gate 2 Output Polarity Control bit
1 = The output of gate 2 is inverted when applied to the logic cell
0 = The output of gate 2 is not inverted

bit 0 **LCxG1POL:** Gate 1 Output Polarity Control bit
1 = The output of gate 1 is inverted when applied to the logic cell
0 = The output of gate 1 is not inverted

© 2013-2016 Microchip Technology Inc.

TABLE 19-3: SUMMARY OF REGISTERS ASSOCIATED WITH CLCx

Name	Bit7	Bit6	Bit5	Bit4	BIt3	Bit2	Bit1	Bit0	Register on Page
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	120
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	126
ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	—	—	131
CLC1CON	LC1EN	—	LC1OUT	LC1INTP	LC1INTN	LC1MODE<2:0>			209
CLC2CON	LC2EN	—	LC2OUT	LC2INTP	LC2INTN	LC2MODE<2:0>			209
CLC3CON	LC3EN	—	LC3OUT	LC3INTP	LC3INTN	LC3MODE<2:0>			209
CLCDATA	—	—	—	—	MCL4OUT	MLC3OUT	MLC2OUT	MLC1OUT	217
CLC1GLS0	LC1G1D4T	LC1G1D4N	LC1G1D3T	LC1G1D3N	LC1G1D2T	LC1G1D2N	LC1G1D1T	LC1G1D1N	213
CLC1GLS1	LC1G2D4T	LC1G2D4N	LC1G2D3T	LC1G2D3N	LC1G2D2T	LC1G2D2N	LC1G2D1T	LC1G2D1N	214
CLC1GLS2	LC1G3D4T	LC1G3D4N	LC1G3D3T	LC1G3D3N	LC1G3D2T	LC1G3D2N	LC1G3D1T	LC1G3D1N	215
CLC1GLS3	LC1G4D4T	LC1G4D4N	LC1G4D3T	LC1G4D3N	LC1G4D2T	LC1G4D2N	LC1G4D1T	LC1G4D1N	216
CLC1POL	LC1POL	—	—	—	LC1G4POL	LC1G3POL	LC1G2POL	LC1G1POL	210
CLC1SEL0	—	—	—	LC1D1S<4:0>					210
CLC1SEL1	—	—	—	LC1D2S<4:0>					211
CLC1SEL2	—	—	—	LC1D3S<4:0>					211
CLC1SEL3	—	—	—	LC1D4S<4:0>					212
CLC2GLS0	LC2G1D4T	LC2G1D4N	LC2G1D3T	LC2G1D3N	LC2G1D2T	LC2G1D2N	LC2G1D1T	LC2G1D1N	213
CLC2GLS1	LC2G2D4T	LC2G2D4N	LC2G2D3T	LC2G2D3N	LC2G2D2T	LC2G2D2N	LC2G2D1T	LC2G2D1N	214
CLC2GLS2	LC2G3D4T	LC2G3D4N	LC2G3D3T	LC2G3D3N	LC2G3D2T	LC2G3D2N	LC2G3D1T	LC2G3D1N	215
CLC2GLS3	LC2G4D4T	LC2G4D4N	LC2G4D3T	LC2G4D3N	LC2G4D2T	LC2G4D2N	LC2G4D1T	LC2G4D1N	216
CLC2POL	LC2POL	—	—	—	LC2G4POL	LC2G3POL	LC2G2POL	LC2G1POL	210
CLC2SEL0	—	—	—	LC2D1S<4:0>					211
CLC2SEL1	—	—	—	LC2D2S<4:0>					211
CLC2SEL2	—	—	—	LC2D3S<4:0>					211
CLC2SEL3	—	—	—	LC2D4S<4:0>					212
CLC3GLS0	LC3G1D4T	LC3G1D4N	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	213
CLC3GLS1	LC3G2D4T	LC3G2D4N	LC3G2D3T	LC3G2D3N	LC3G2D2T	LC3G2D2N	LC3G2D1T	LC3G2D1N	214
CLC3GLS2	LC3G3D4T	LC3G3D4N	LC3G3D3T	LC3G3D3N	LC3G3D2T	LC3G3D2N	LC3G3D1T	LC3G3D1N	215
CLC3GLS3	LC3G4D4T	LC3G4D4N	LC3G4D3T	LC3G4D3N	LC3G4D2T	LC3G4D2N	LC3G4D1T	LC3G4D1N	216
CLC3POL	LC3POL	—	—	—	LC3G4POL	LC3G3POL	LC3G2POL	LC3G1POL	210
CLC3SEL0	—	—	—	LC3D1S<4:0>					211
CLC3SEL1	—	—	—	LC3D2S<4:0>					211
CLC3SEL2	—	—	—	LC3D3S<4:0>					211
CLC3SEL3	—	—	—	LC3D4S<4:0>					212
CLC4GLS0	LC4G1D4T	LC4G1D4N	LC4G1D3T	LC4G1D3N	LC4G1D2T	LC4G1D2N	LC4G1D1T	LC4G1D1N	213
CLC4GLS1	LC4G2D4T	LC4G2D4N	LC4G2D3T	LC4G2D3N	LC4G2D2T	LC4G2D2N	LC4G2D1T	LC4G2D1N	214
CLC4GLS2	LC4G3D4T	LC4G3D4N	LC4G3D3T	LC4G3D3N	LC4G3D2T	LC4G3D2N	LC4G3D1T	LC4G3D1N	215
CLC4GLS3	LC4G4D4T	LC4G4D4N	LC4G4D3T	LC4G4D3N	LC4G4D2T	LC4G4D2N	LC4G4D1T	LC4G4D1N	216
CLC4POL	LC4POL	—	—	—	LC4G4POL	LC4G3POL	LC4G2POL	LC4G1POL	210
CLC4SEL0	—	—	—	LC4D1S<4:0>					211
CLC4SEL1	—	—	—	LC4D2S<4:0>					211
CLC4SEL2	—	—	—	LC4D3S<4:0>					211
CLC4SEL3	—	—	—	LC4D4S<4:0>					212
CLCxPPS	—	—	—	CLCxPPS<4:0>					136
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	83

Legend: — = unimplemented read as ‘0’. Shaded cells are not used for CLC module.

© 2013-2016 Microchip Technology Inc.

21.1.5 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC Interrupt Flag is the ADIF bit in the PIR1 register. The ADC Interrupt Enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

Note 1: The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.

2: The ADC operates during Sleep only when the FRC oscillator is selected.

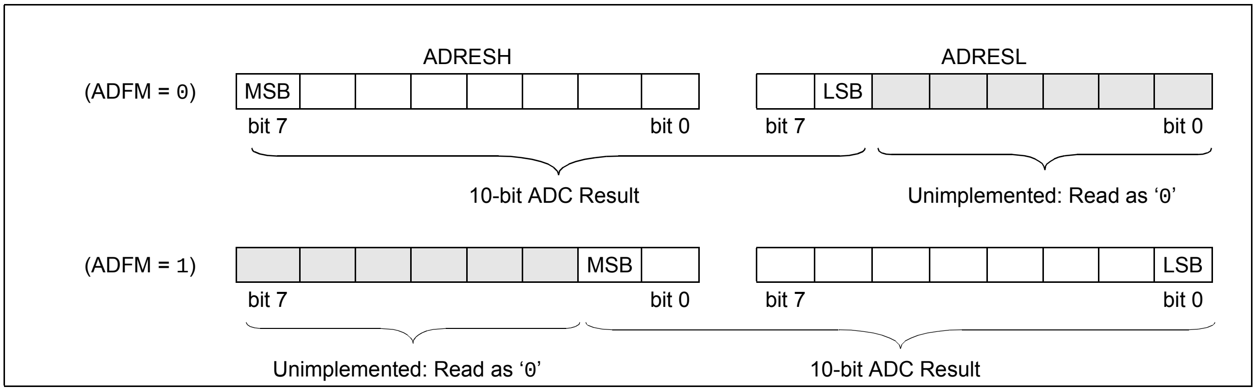
This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the `SLEEP` instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the ADIE bit of the PIE1 register and the PEIE bit of the INTCON register must both be set and the GIE bit of the INTCON register must be cleared. If all three of these bits are set, the execution will switch to the Interrupt Service Routine.

21.1.6 RESULT FORMATTING

The 10-bit ADC conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON1 register controls the output format.

Figure 21-3 shows the two output formats.

FIGURE 21-3: 10-BIT ADC CONVERSION RESULT FORMAT



© 2013-2016 Microchip Technology Inc.

FIGURE 30-16: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 0, AHEN = 1, DHEN = 1)

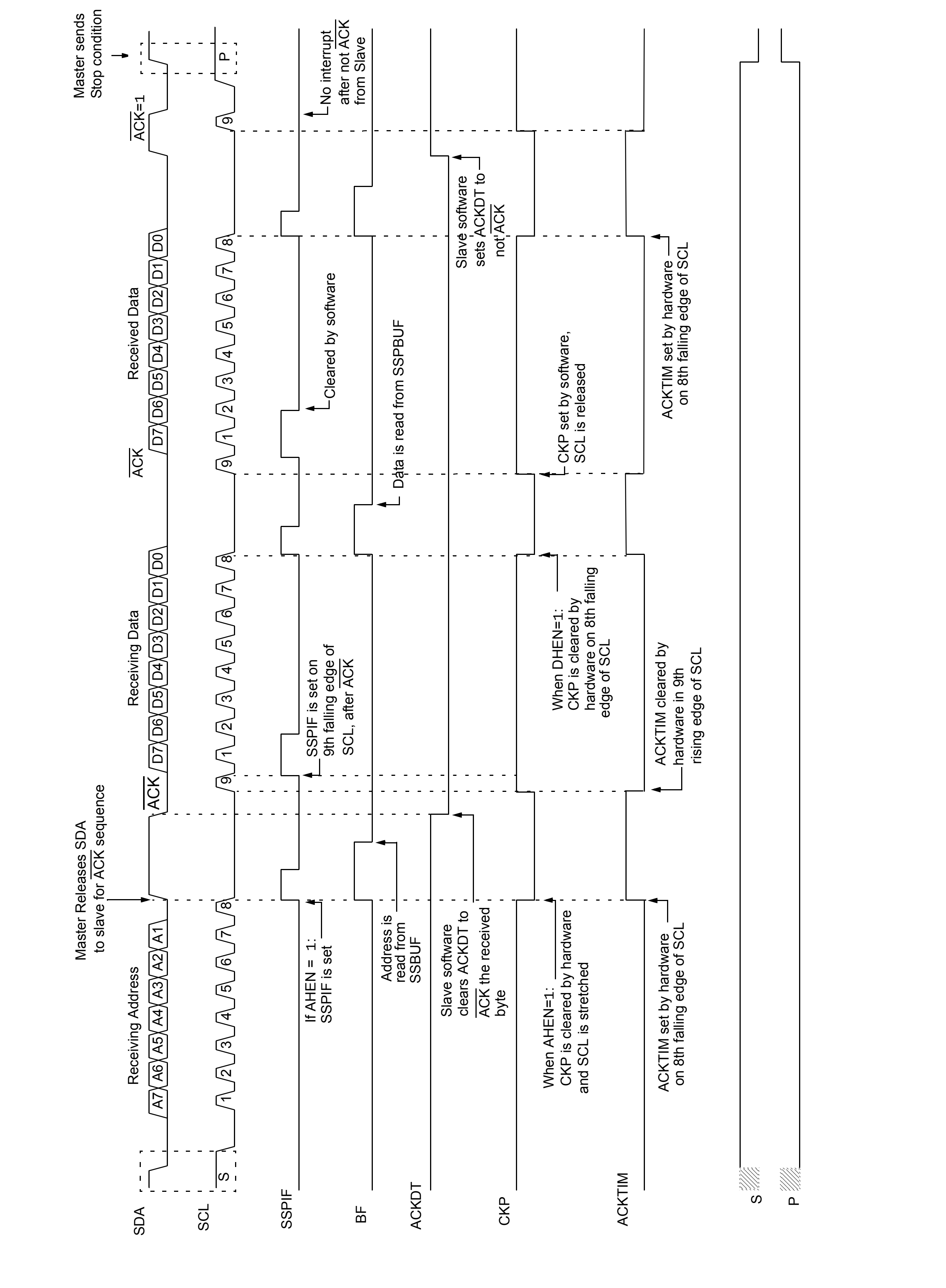
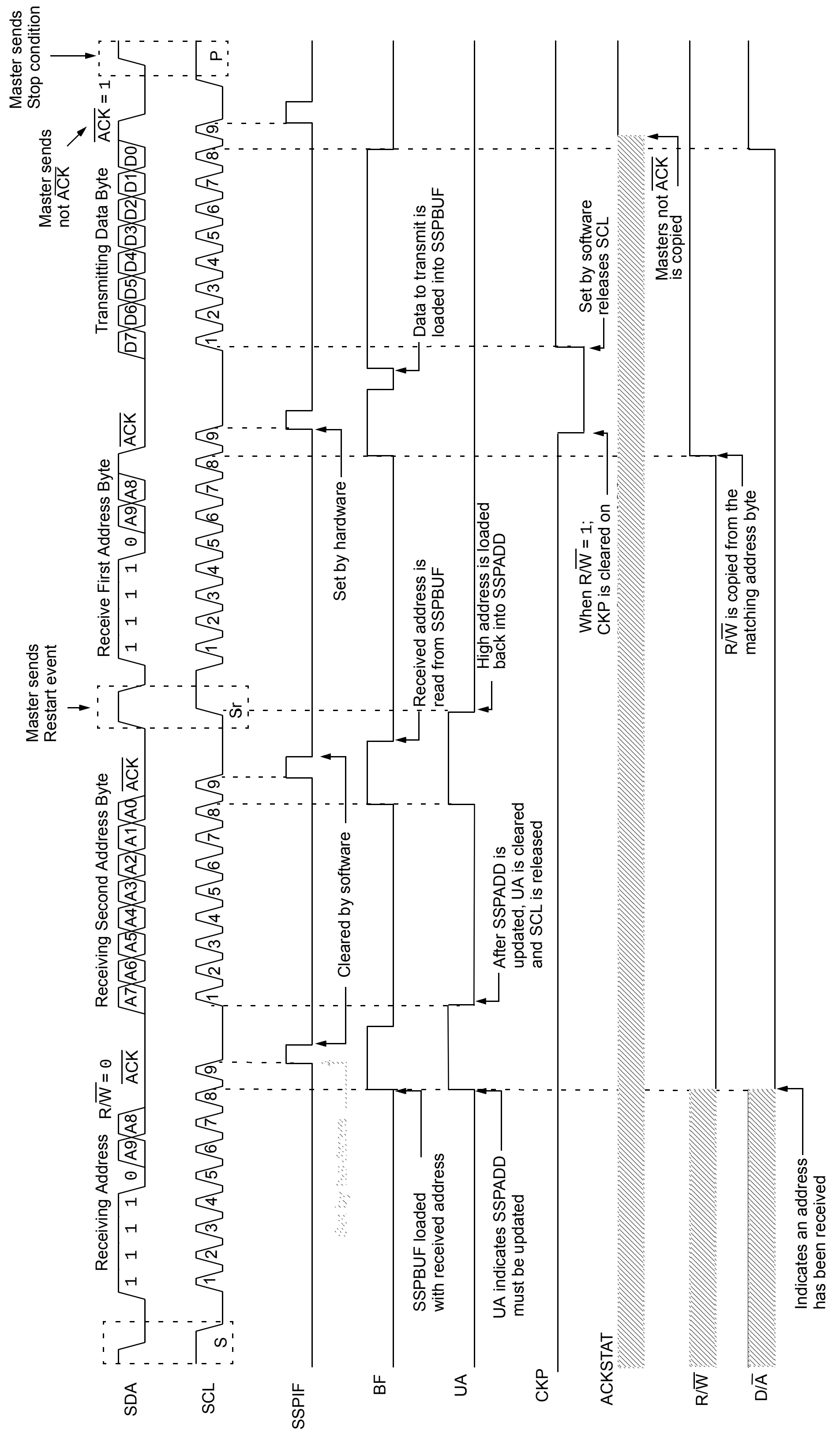

SDA
SCL
SSPIF
BF
ACKDT
CKP
ACKTIM
S
P
Receiving Address
A7 A6 A5 A4 A3 A2 A1
Master Releases SDA to slave for $\overline{\text{ACK}}$ sequence
$\overline{\text{ACK}}$
Receiving Data
D7 D6 D5 D4 D3 D2 D1 D0
$\overline{\text{ACK}}$
Received Data
D7 D6 D5 D4 D3 D2 D1 D0
$\overline{\text{ACK}}$=1
Master sends Stop condition
S
1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
P
If AHEN = 1: SSPIF is set
SSPIF is set on 9th falling edge of SCL, after $\overline{\text{ACK}}$
Cleared by software
No interrupt after not $\overline{\text{ACK}}$ from Slave
Address is read from SSBUF
Data is read from SSPBUF
Slave software clears ACKDT to $\overline{\text{ACK}}$ the received byte
Slave software sets ACKDT to not $\overline{\text{ACK}}$
When AHEN=1: CKP is cleared by hardware and SCL is stretched
When DHEN=1: CKP is cleared by hardware on 8th falling edge of SCL
CKP set by software, SCL is released
ACKTIM set by hardware on 8th falling edge of SCL
ACKTIM cleared by hardware in 9th rising edge of SCL
ACKTIM set by hardware on 8th falling edge of SCL

FIGURE 30-22: I^{2}C SLAVE, 10-BIT ADDRESS, TRANSMISSION (SEN = 0, AHEN = 0, DHEN = 0)


SDA
SCL
SSPIF
BF
UA
CKP
ACKSTAT
R/$\overline{W}$
D/$\overline{A}$
S
Receiving Address
R/$\overline{W}$ = 0
1 1 1 1 0 A9 A8
$\overline{ACK}$
1 2 3 4 5 6 7 8 9
Receiving Second Address Byte
A7 A6 A5 A4 A3 A2 A1 A0 $\overline{ACK}$
1 2 3 4 5 6 7 8 9
Master sends Restart event
Sr
Receive First Address Byte
1 1 1 1 0 A9 A8
$\overline{ACK}$
1 2 3 4 5 6 7 8 9
Transmitting Data Byte
D7 D6 D5 D4 D3 D2 D1 D0
1 2 3 4 5 6 7 8 9
Master sends not $\overline{ACK}$
$\overline{ACK}$ = 1
Master sends Stop condition
P
SSPBUF loaded with received address
UA indicates SSPADD must be updated
Indicates an address has been received
Cleared by software
After SSPADD is updated, UA is cleared and SCL is released
Received address is read from SSPBUF
High address is loaded back into SSPADD
Set by hardware
When R/$\overline{W}$ = 1; CKP is cleared on
R/$\overline{W}$ is copied from the matching address byte
Data to transmit is loaded into SSPBUF
Set by software releases SCL
Masters not $\overline{ACK}$ is copied

© 2013-2016 Microchip Technology Inc.

30.6.7 I^{2}C MASTER MODE RECEPTION

Master mode reception (Figure 30-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSP1CON2 register.

> **Note:** The MSSP module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag bit is set, the SSPIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSP is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPCON2 register.

30.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when the SSPBUF register is read.

30.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when eight bits are received into the SSPSR and the BF flag bit is already set from a previous reception.

30.6.7.3 WCOL Status Flag

If the user writes the SSPBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

30.6.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit of the SSPCON2 register.
2. SSPIF is set by hardware on completion of the Start.
3. SSPIF is cleared by software.
4. User writes SSPBUF with the slave address to transmit and the R/$\overline{\text{W}}$ bit set.
5. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPBUF is written to.
6. The MSSP module shifts in the $\overline{\text{ACK}}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPCON2 register.
7. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
8. User sets the RCEN bit of the SSPCON2 register and the master clocks in a byte from the slave.
9. After the eighth falling edge of SCL, SSPIF and BF are set.
10. Master clears SSPIF and reads the received byte from SSPUF, clears BF.
11. Master sets $\overline{\text{ACK}}$ value sent to slave in ACKDT bit of the SSPCON2 register and initiates the $\overline{\text{ACK}}$ by setting the ACKEN bit.
12. Master's $\overline{\text{ACK}}$ is clocked out to the slave and SSPIF is set.
13. User clears SSPIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{\text{ACK}}$ or Stop to end communication.

© 2013-2016 Microchip Technology Inc.

31.1.2 EUSART ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 31-2. The data is received on the RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all eight or nine bits of the character have been shifted in, they are immediately transferred to a two character First-In-First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the EUSART receiver. The FIFO and RSR registers are not directly accessible by software. Access to the received data is via the RCREG register.

31.1.2.1 Enabling the Receiver

The EUSART receiver is enabled for asynchronous operation by configuring the following three control bits:

- CREN = `1`
- SYNC = `0`
- SPEN = `1`

All other EUSART control bits are assumed to be in their default state.

Setting the CREN bit of the RCSTA register enables the receiver circuitry of the EUSART. Clearing the SYNC bit of the TXSTA register configures the EUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the EUSART. The programmer must set the corresponding TRIS bit to configure the RX/DT I/O pin as an input.

Note: If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

31.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting ‘`0`’ or ‘`1`’ is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always a ‘`1`’. If the data recovery circuit samples a ‘`0`’ in the Stop bit position then a framing error is set for this character, otherwise the framing error is cleared for this character. See **Section 31.1.2.4 “Receive Framing Error”** for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the EUSART receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RCREG register.

Note: If the receive FIFO is overrun, no additional characters will be received until the overrun condition is cleared. See **Section 31.1.2.5 “Receive Overrun Error”** for more information on overrun errors.

31.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR1 register is set whenever the EUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE, Interrupt Enable bit of the PIE1 register
- PEIE, Peripheral Interrupt Enable bit of the INTCON register
- GIE, Global Interrupt Enable bit of the INTCON register

The RCIF interrupt flag bit will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

© 2013-2016 Microchip Technology Inc.

31.1.2.8 Asynchronous Reception Setup:

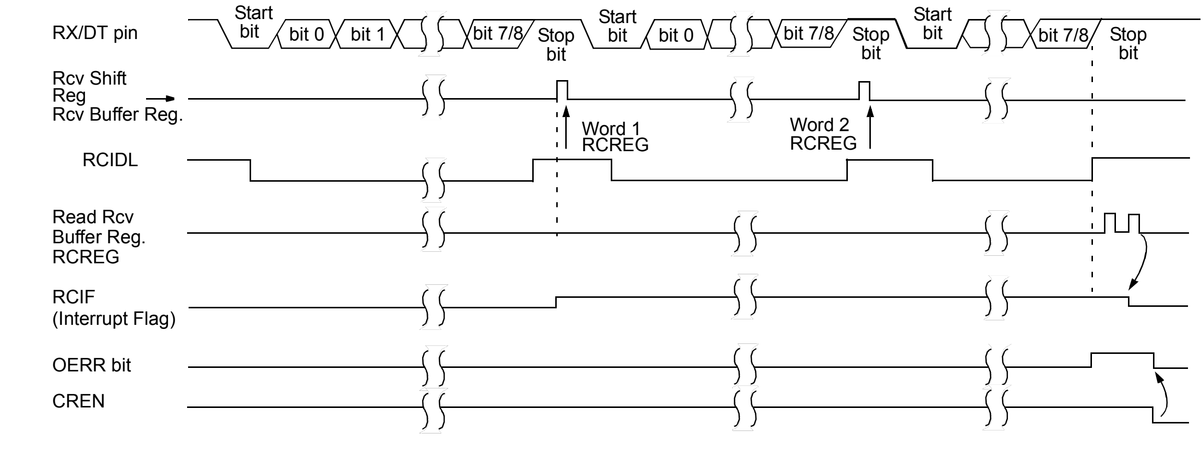
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.4 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

31.1.2.9 9-bit Address Detection Mode Setup

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.4 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
10. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

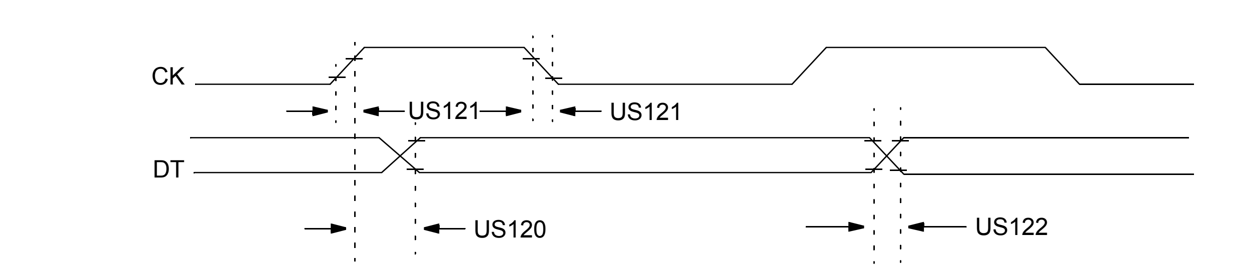
FIGURE 31-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

© 2013-2016 Microchip Technology Inc.

FIGURE 34-15: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

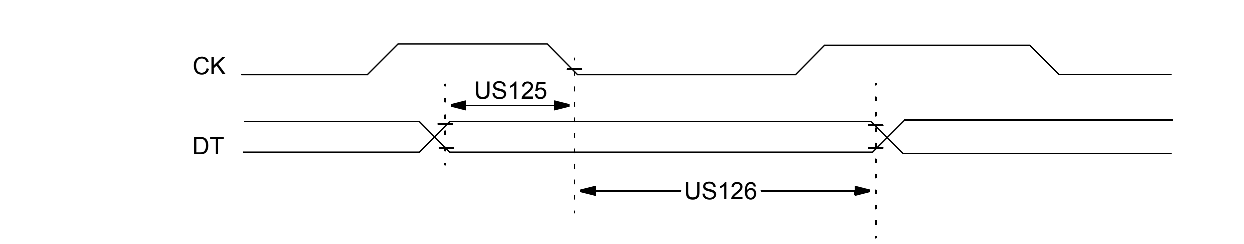


Note: Refer to Figure 34-4 for load conditions.

TABLE 34-22: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US120	TCKH2DTV	SYNC XMIT (Master and Slave) Clock high to data-out valid	—	80	ns	3.0V ≤ VDD ≤ 5.5V
			—	100	ns	1.8V ≤ VDD ≤ 5.5V
US121	TCKRF	Clock out rise time and fall time (Master mode)	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V
US122	TDTRF	Data-out rise time and fall time	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V

FIGURE 34-16: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



Note: Refer to Figure 34-4 for load conditions.

TABLE 34-23: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US125	TDTV2CKL	SYNC RCV (Master and Slave) Data-setup before CK ↓ (DT hold time)	10	—	ns	
US126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	—	ns	

© 2013-2016 Microchip Technology Inc.

Note: Unless otherwise noted, V_{IN} = 5V, F_{OSC} = 300 kHz, C_{IN} = 0.1 µF, T_A = 25°C.

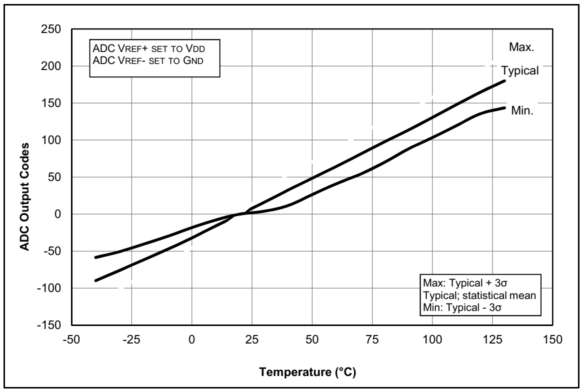


FIGURE 35-91: *Temp. Indicator Slope Normalized to 20°C, Low Range, V_{DD} = 1.8V, PIC16LF1713/6 Only.*

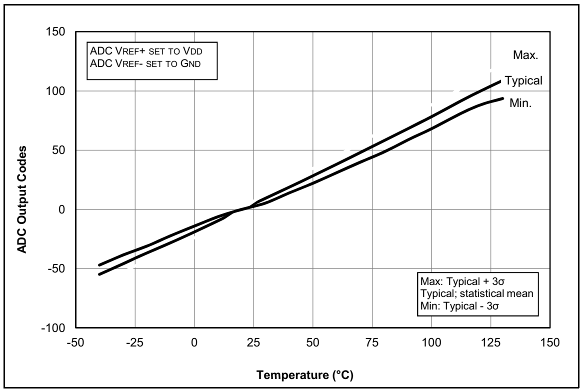


FIGURE 35-92: *Temp. Indicator Slope Normalized to 20°C, Low Range, V_{DD} = 3.0V, PIC16LF1713/6 Only.*

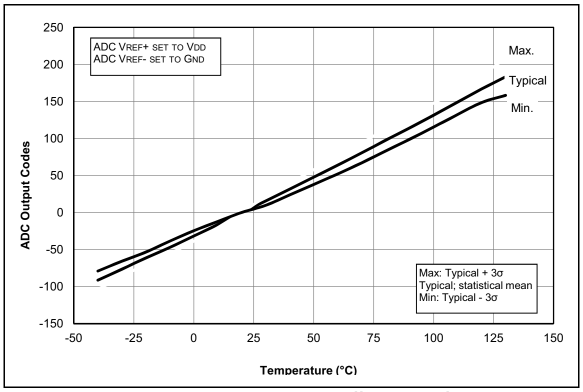


FIGURE 35-93: *Temp. Indicator Slope Normalized to 20°C, High Range, V_{DD} = 3.6V, PIC16LF1713/6 Only.*

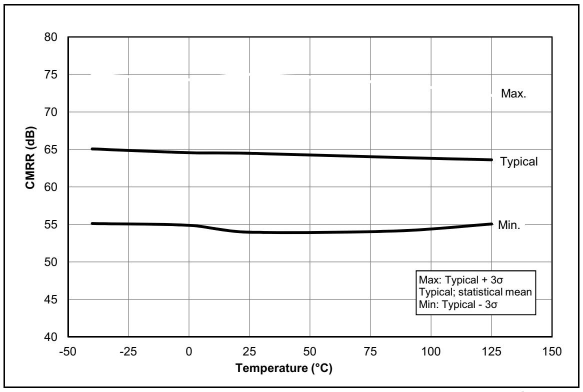


FIGURE 35-94: *Op Amp, Common Mode Rejection Ratio (CMRR), V_{DD} = 3.0V.*

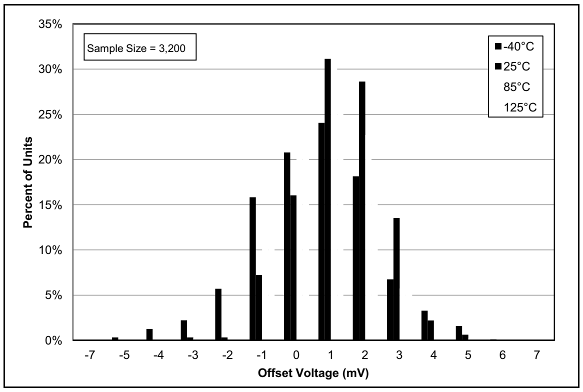


FIGURE 35-95: *Op Amp, Input Offset Voltage Histogram, V_{DD} = 3.0V, V_{CM} = V_{DD}/2.*

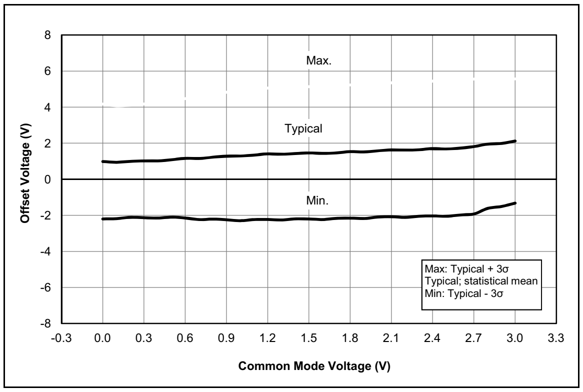
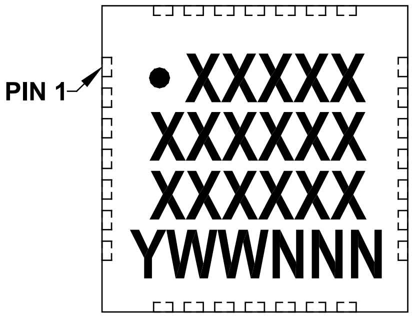


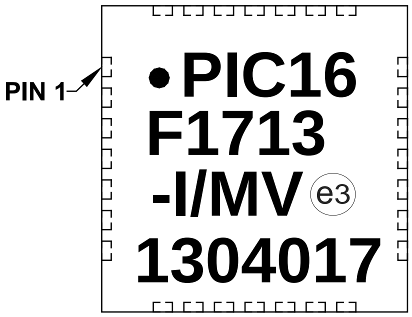
FIGURE 35-96: *Op Amp, Offset Over Common Mode Voltage, V_{DD} = 3.0V, Temp. = 25°C.*

Package Marking Information (Continued)

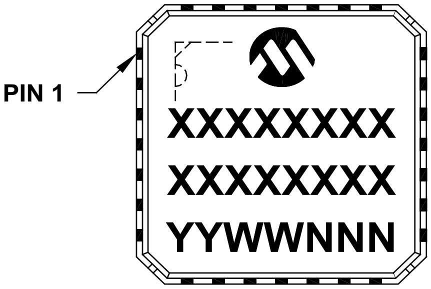
28-Lead UQFN (4x4x0.5 mm)



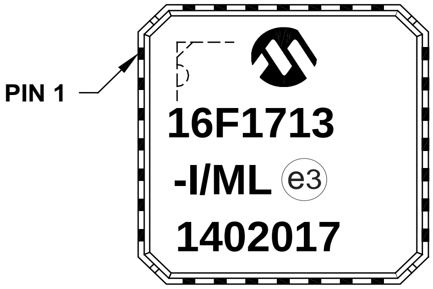
Example



28-Lead QFN (6x6x0.9 mm)



Example



Legend:		
	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	e3	Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (11/2013)

Initial release.

Revision B (01/2014)

Updated the Pin allocation table; Updated Tables 1-2, 3-9 and 12-1; Updated Registers 11-20, 18-6, 18-7 and 21-1; Updated Register summaries; Added Registers 13-10 to 13-12; Added Section 24; Updated the ZCD section; Removed the HFINTOSC graphs; Added 28 QFN package; Other minor corrections.

Revision C (01/2016)

Updated first page, under Memory information. Updated PIC16(L)F1713/6 Family Types Table.

Added Sections 3.2: High Endurance Flash and 6.3.5: Clock Switching Before Sleep. Added Table 3-4 and 3-6.

Removed Sections 18.1.1 and 24.4. Updated new Section 18.1.1.

Updated Examples 3-2 and 21-1. Updated Figures 18-2, 18-3, 18-4, 18-5, 18-6, 21-1, 22-1, and 23-1. Updated Register 21-1 and 22-1. Updated Sections 8.2.2, 18.12, 20.0, 21.1.3, 21.2.6, 22.0, 22.1, 22.1.1, 31.1, 31.4.2, and 35.0. Updated Tables 3-1, 3-9, 6-1, 34-1, 34-2, 34-3, 34-4, 34-7, 34-8, 34-10, 34-11 and 34-24.

© 2013-2016 Microchip Technology Inc.