

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

-XF

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, 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	224 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf15354-i-ml

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

TABLE 4	I-10: SPEC	IAL FUNCTION	REGISTER	SUMMARY	BANKS 0-	63					
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	V <u>alue o</u> n: MCLR
Bank 0											
	CPU CORE REGISTERS; see Table 4-9 for specifics										
00Ch	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	uuuu uuuu
00Dh	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	XXXX XXXX	uuuu uuuu
00Eh	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
00Fh	—				Unimple	mented					
010h	PORTE	_	—	—	—	RE3	—	—	—	x	u
011h	—				Unimple	mented					
012h	TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
013h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
014h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
015h	—				Unimple	mented					
016h	TRISE	—	—	—	—	_(1)	—	—	_	1	1
017h	—				Unimple	mented					
018h	LATA	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx xxxx	uuuu uuuu
019h	LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx xxxx	uuuu uuuu
01Ah	LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	uuuu uuuu
01Bh	—		Unimplemented								
01Ch	—				Unimple	mented					
01Dh					Unimple	mented					
01Eh					Unimple	mented					
01Fh					Unimple	mented					
01Eh					Unimple	mented					
01Fh	_				Unimple	mented					

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

Note 1: Unimplemented, read as '1'.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	V <u>alue o</u> n: MCLR
Bank 60											
				CPU COF	RE REGISTERS;	see Table 4-3 for	r specifics				
1E0Ch	_				Unimpler	mented					
1E0Dh	_				Unimpler	mented					
1E0Eh	_				Unimpler	mented					
1E0Fh	CLCDATA	_		_		MLC4OUT	MLC3OUT	MLC2OUT	MLC1OUT	xxxx	uuuu
1E10h	CLCCON	LC1EN	_	LC10UT	LC1INTP	LC1INTN		LC1MODE<2:	0>	0-00 0000	0-00 0000
1E11h	CLC1POL	LC1POL		_		LC1G4POL	LC1G3POL	LC1G2POL	LC1G1POL	0 xxxx	0 uuuu
1E12h	CLC1SEL0	—	—			LC1	D1S<5:0>			xx xxxx	uu uuuu
1E13h	CLC1SEL1	_	—			LC1	02S<5:0>			xx xxxx	uu uuuu
1E14h	CLC1SEL2	_	—			LC1	03S<5:0>			xx xxxx	uu uuuu
1E15h	CLC1SEL3	_	—			LC1	04S<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	_	—			LC2	D1S<5:0>			xx xxxx	uu uuuu
1E1Dh	CLC2SEL1	_	—			LC2	02S<5:0>			xx xxxx	uu uuuu
1E1Eh	CLC2SEL2	_	—			LC2	03S<5:0>			xx xxxx	uu uuuu
1E1Fh	CLC2SEL3	_	—			LC2	04S<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	_	—			LC3	D1S<5:0>			xx xxxx	uu uuuu
1E27h	CLC3SEL1					LC3	02S<5:0>			xx xxxx	uu uuuu
1E28h	CLC3SEL2	_	—			LC3	03S<5:0>			xx xxxx	uu uuuu
1E29h	CLC3SEL3	_				LC3	04S<5:0>			xx xxxx	uu uuuu
1E2Ah	CLC3GLS0	LC3G1D4T	LC3G4D3N	LC3G1D3T	LC3G1D3N	LC3G1D2T	LC3G1D2N	LC3G1D1T	LC3G1D1N	xxxx xxxx	uuuu uuuu

TABLE 4-10: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

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

6.1 Microchip Unique identifier (MUI)

The PIC16(L)F15354/55 devices are individually encoded during final manufacturing with a Microchip Unique Identifier, or MUI. The MUI cannot be erased by a Bulk Erase command or any other user-accessible means. This feature allows for manufacturing traceability of Microchip Technology devices in applications where this is a required. It may also be used by the application manufacturer for a number of functions that require unverified unique identification, such as:

- Tracking the device
- Unique serial number

The MUI consists of nine program words. When taken together, these fields form a unique identifier. The MUI is stored in nine read-only locations, located between 8100h to 8109h in the DIA space. Table 6-1 lists the addresses of the identifier words.

Note:	For applications that require verified unique
	identification, contact your Microchip Tech-
	nology sales office to create a Serialized
	Quick Turn Programming option.

6.2 External Unique Identifier (EUI)

The EUI data is stored at locations 810Ah to 8111h in the program memory region. This region is an optional space for placing application specific information. The data is coded per customer requirements during manufacturing. The EUI cannot be erased by a Bulk erase command.

Note: Data is stored in this address range on receiving a request from the customer. The customer may contact the local sales representative, or Field Applications Engineer, and provide them the unique identifier information that is required to be stored in this region.

6.3 Analog-to-Digital Conversion Data of the Temperature Sensor

The purpose of the temperature sensor module is to provide a temperature-dependent voltage that can be measured by an analog module. **Section 19.0 "Temperature Indicator Module**" explains the operation of the Temperature Indicator module and defines terms such as the low range and high range settings of the sensor.

The DIA table contains the internal ADC measurement values of the temperature sensor for low and high range at fixed points of reference. The values are measured during test and are unique to each device. The right-justified ADC readings are stored in the DIA memory region. The calibration data can be used to plot the approximate sensor output voltage, VTSENSE vs. Temperature curve without having to make calibration measurements in the application.

- **TSLR<3:1>**: Address 8112h to 8114h store the measurements for the low range setting of the temperature sensor at VDD = 3V.
- TSHR<3:1>: Address 8115h to 8117h store the measurements for the high range setting of the temperature sensor at VDD = 3V.

The stored measurements are made by the device ADC using the internal $V_{REF} = 2.048V$.

6.4 Fixed Voltage Reference Data

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- · ADC input channel
- ADC positive reference
- Comparator positive input
- Digital-to-Analog Converter

For more information on the FVR, refer to **Section 18.0 "Fixed Voltage Reference (FVR)"**.

The DIA stores measured FVR voltages for this device in mV for the different buffer settings of 1x, 2x or 4x at program memory locations 8118h to 811Dh.

- FVRA1X stores the value of ADC FVR1 Output voltage for 1x setting (in mV)
- FVRA2X stores the value of ADC FVR1 Output Voltage for 2x setting (in mV)
- FVRA4X stores the value of ADC FVR1 Output Voltage for 4x setting (in mV)
- FVRC1X stores the value of Comparator FVR2 output voltage for 2x setting (in mV)
- FVRC2X stores the value of Comparator FVR2 output voltage for 2x setting (in mV)
- FVRC4X stores the value of Comparator FVR2 output voltage for 4x setting (in mV)

7.0 DEVICE CONFIGURATION INFORMATION

The Device Configuration Information (DCI) is a dedicated region in the Program Flash Memory mapped from 8200h to 821Fh. The data stored in the DCI memory is hard-coded into the device during manufacturing.

Refer to Table 7-1: Device Configuration Information for PIC16(L)F15354/55 Devices for the complete DCI table address and description. The DCI holds information about the device which is useful for programming and bootloader applications. These locations are read-only and cannot be erased or modified.

	Nomo	DESCRIPTION	VAI		
ADDRESS Name		DESCRIPTION	PIC16(L)F15354	PIC16(L)F15355	
8200h	ERSIZ	Erase Row Size	32	32	Words
8201h	WLSIZ	Number of write latches	32	32	Latches
8202h	URSIZ	Number of User Rows	128	256	Rows
8203h	EESIZ	EE Data memory size	0	0	Bytes
8204h	PCNT	Pin Count	28	28	Pins

TABLE 7-1: DEVICE CONFIGURATION INFORMATION FOR PIC16(L)F15354/55 DEVICES

7.1 DIA and DCI Access

The DIA and DCI data are read-only and cannot be erased or modified. See **13.3.6** "NVMREG Access to Device Information Area, Device Configuration Area, User ID, Device ID and Configuration Words" for more information on accessing these memory locations.

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

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	R/W/HS-0/0
_	_	NVMIF	NCO1IF	_	_	_	CWG1IF
bit 7					•		bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpler	mented bit, read	as '0'	
u = Bit is uncha	anged	x = Bit is unkr	nown	-n/n = Value	at POR and BO	R/Value at all o	ther Resets
'1' = Bit is set		'0' = Bit is clea	ared	HS = Hardwa	are set		
bit 7-6	Unimplemen	ted: Read as '	0'				
bit 5	NVMIF: Nonv	olatile Memory	(NVM) Interru	upt Flag bit			
	1 = The reque 0 = NVM inter	ested NVM ope rrupt not assert	ration has cor	npleted			
bit 4	NCO1IF: Num	nerically Contro	olled Oscillator	r (NCO) Interru	upt Flag bit		
	1 = The NCO	has rolled ove	r				
	0 = No NCO interrupt event has occurred						
bit 3-1	bit 3-1 Unimplemented: Read as '0'						
bit 0	bit 0 CWG1IF: CWG1 Interrupt Flag bit						
	1 = CWG1 has gone into shutdown						
	0 = CWG1 is	operating norm	nally, or interru	pt cleared			

REGISTER 10-17: PIR7: PERIPHERAL INTERRUPT REQUEST REGISTER 7

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

11.2.3.2 Peripheral Usage in Sleep

Some peripherals that can operate in Sleep mode will not operate properly with the Low-Power Sleep mode selected. The Low-Power Sleep mode is intended for use with these peripherals:

- Brown-out Reset (BOR)
- Watchdog Timer (WDT)
- External interrupt pin/interrupt-on-change pins
- Timer1 (with external clock source)

It is the responsibility of the end user to determine what is acceptable for their application when setting the VREGPM settings in order to ensure operation in Sleep.

Note:	The PIC16LF15354/55 does not have a
	configurable Low-Power Sleep mode.
	PIC16LF15354/55 is an unregulated
	device and is always in the lowest power
	state when in Sleep, with no wake-up time
	penalty. This device has a lower maximum
	VDD and I/O voltage than the
	PIC16F15354/55. See Section 37.0
	"Electrical Specifications" for more
	information.

11.3 IDLE Mode

When the Idle Enable (IDLEN) bit is clear (IDLEN = 0), the SLEEP instruction will put the device into full Sleep mode (see **Section 11.2 "Sleep Mode**"). When IDLEN is set (IDLEN = 1), the SLEEP instruction will put the device into IDLE mode. In IDLE mode, the CPU and memory operations are halted, but the peripheral clocks continue to run. This mode is similar to DOZE mode, except that in IDLE both the CPU and PFM are shut off.

Note:	Peripherals using FOSC will continue
	running while in Idle (but not in Sleep).
	Peripherals using HFINTOSC,
	LFINTOSC, or SOSC will continue
	running in both Idle and Sleep.

Note: If CLKOUT is enabled (CLKOUT = 0, Configuration Word 1), the output will continue operating while in Idle.

11.3.0.1 Idle and Interrupts

IDLE mode ends when an interrupt occurs (even if GIE = 0), but IDLEN is not changed. The device can re-enter IDLE by executing the SLEEP instruction.

If Recover-on-Interrupt is enabled (ROI = 1), the interrupt that brings the device out of Idle also restores full-speed CPU execution when doze is also enabled.

11.3.0.2 Idle and WDT

When in IDLE, the WDT Reset is blocked and will instead wake the device. The WDT wake-up is not an interrupt, therefore ROI does not apply.

Note: The WDT can bring the device out of IDLE, in the same way it brings the device out of Sleep. The DOZEN bit is not affected.

13.3.4 NVMREG WRITE TO PFM

Program memory is programmed using the following steps:

- 1. Load the address of the row to be programmed into NVMADRH:NVMADRL.
- 2. Load each write latch with data.
- 3. Initiate a programming operation.
- 4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 13-4 (row writes to program memory with 32 write latches) for more details.

The write latches are aligned to the Flash row address boundary defined by the upper ten bits of NVMADRH:NVMADRL, (NVMADRH<6:0>:NVMADRL<7:5>) with the lower five bits of NVMADRL, (NVMADRL<7:5>) determining the write latch being loaded. Write operations do not cross these boundaries. At the completion of a program memory write operation, the data in the write latches is reset to contain 0x3FFF. The following steps should be completed to load the write latches and program a row of program memory. These steps are divided into two parts. First, each write latch is loaded with data from the NVMDATH:NVMDATL using the unlock sequence with LWLO = 1. When the last word to be loaded into the write latch is ready, the LWLO bit is cleared and the unlock sequence executed. This initiates the programming operation, writing all the latches into Flash program memory.

- Note: The special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. If the unlock sequence is interrupted, writing to the latches or program memory will not be initiated.
- 1. Set the WREN bit of the NVMCON1 register.
- Clear the NVMREGS bit of the NVMCON1 register.
- Set the LWLO bit of the NVMCON1 register. When the LWLO bit of the NVMCON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash program memory.
- 4. Load the NVMADRH:NVMADRL register pair with the address of the location to be written.
- 5. Load the NVMDATH:NVMDATL register pair with the program memory data to be written.
- Execute the unlock sequence (Section 13.3.2 "NVM Unlock Sequence"). The write latch is now loaded.
- 7. Increment the NVMADRH:NVMADRL register pair to point to the next location.
- 8. Repeat steps 5 through 7 until all but the last write latch has been loaded.
- Clear the LWLO bit of the NVMCON1 register. When the LWLO bit of the NVMCON1 register is '0', the write sequence will initiate the write to Flash program memory.
- 10. Load the NVMDATH:NVMDATL register pair with the program memory data to be written.
- Execute the unlock sequence (Section 13.3.2 "NVM Unlock Sequence"). The entire program memory latch content is now written to Flash program memory.
- **Note:** The program memory write latches are reset to the blank state (0x3FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state.

An example of the complete write sequence is shown in Example 13-4. The initial address is loaded into the NVMADRH:NVMADRL register pair; the data is loaded using indirect addressing.

19.2.1 CALIBRATION

Single-Point Calibration

Single-point calibration is performed by application software using Equation 19-1 and the assumed Mt. A reading of VTSENSE at a known temperature is taken, and the theoretical temperature is calculated by temporarily setting TOFFSET = 0. Then TOFFSET is computed as the difference of the actual and calculated temperatures. Finally, TOFFSET is stored in nonvolatile memory within the device, and is applied to future readings to gain a more accurate measurement.

The magnitude of error in a typical single-point calibration is approximately 3-4°C.

- Note 1: The TOFFSET value may be determined by the user with a temperature test, or it can be based on the Microchip-supplied data from the DIA table. Please refer to Section 6.0 "Device Information Area" for more information.
 - 2: Although the measurement range is -40°C to +125 °C, due to the variations in the value of Mv, the single-point calculated TSENSE value may indicate a temperature from -140°C to +225°C, before the calibration offset is applied.

Higher-Order Calibration

If the application requires more precise temperature measurement, additional calibrations steps will be necessary. For these applications, two-point or threepoint calibration is recommended.

19.2.2 TEMPERATURE RESOLUTION

The resolution of the ADC reading, Ma (°C/count), depends on both the ADC resolution N and the reference voltage used for conversion, as shown in Equation 19-2. It is recommended to use the smallest VREF value, such as 2.048 FVR reference voltage, instead of VDD.

Note:	Refer	to	Sec	tion	37.0	"Electrical
	Specifi	icatio	ns"	for	FVR	reference
	voltage	accu	iracy.			

EQUATION 19-2: TEMPERATURE RESOLUTION (°C/LSb)

$$Ma = \frac{V_{REF}}{2^N} \times Mt$$

$$Ma = \frac{\frac{V_{REF}}{2^{N}}}{Mv}$$

Where:

Mv = sensor voltage sensitivity (V/°C)

VREF = Reference voltage of the ADC module (in Volts)

N = Resolution of the ADC

EXAMPLE 19-1: TEMPERATURE RESOLUTION

Using VREF = 2.048V and a 10-bit ADC provides 2 mV/LSb measurements.

Because Mv can vary from -2.40 to -2.65 mV/°C, the range of Ma = 0.75 to 0.83 °C/LSb.

19.3 ADC Acquisition Time

To ensure accurate temperature measurements, the user must wait a fixed amount of time for the ADC value to settle, after the ADC input multiplexer is connected to the temperature indicator output, before the conversion is performed. This specification is provided in **Section 37.0 "Electrical Specifications"**.



27.5.8 LEVEL RESET, EDGE-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

In Level -Triggered One-Shot mode the timer count is reset on the external signal level and starts counting on the rising/falling edge of the transition from Reset level to the active level while the ON bit is set. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = 01110)
- High Reset level (MODE<4:0> = 01111)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control a new external signal edge is required after the ON bit is set to start the counter.

When Level-Triggered Reset One-Shot mode is used in conjunction with the CCP PWM operation the PWM drive goes active with the external signal edge that starts the timer. The PWM drive goes inactive when the timer count equals the CCPRx pulse width count. The PWM drive does not go active when the timer count clears at the PRx period count match.



FIGURE 30-1: SIMPLIFIED CWG BLOCK DIAGRAM (HALF-BRIDGE MODE)

PIC16(L)F15354/55

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 (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.

32.2.3 SPI MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK line. The master determines when the slave (Processor 2, Figure 32-5) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPxBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPxSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPxBUF register as if a normal received byte (interrupts and Status bits appropriately set). The clock polarity is selected by appropriately programming the CKP bit of the SSPxCON1 register and the CKE bit of the SSPxSTAT register. This then, would give waveforms for SPI communication as shown in Figure 32-6, Figure 32-8, Figure 32-9 and Figure 32-10, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- Fosc/4 (or Tcy)
- Fosc/16 (or 4 * Tcy)
- Fosc/64 (or 16 * Tcy)
- Timer2 output/2
- Fosc/(4 * (SSPxADD + 1))

Figure 32-6 shows the waveforms for Master mode.

When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPxBUF is loaded with the received data is shown.



FIGURE 32-6: SPI MODE WAVEFORM (MASTER MODE)

33.3 EUSART Baud Rate Generator (BRG)

The Baud Rate Generator (BRG) is an 8-bit or 16-bit timer that is dedicated to the support of both the asynchronous and synchronous EUSART operation. By default, the BRG operates in 8-bit mode. Setting the BRG16 bit of the BAUDxCON register selects 16-bit mode.

The SPxBRGH, SPxBRGL register pair determines the period of the free running baud rate timer. In Asynchronous mode the multiplier of the baud rate period is determined by both the BRGH bit of the TXxSTA register and the BRG16 bit of the BAUDxCON register. In Synchronous mode, the BRGH bit is ignored.

Table 33-1 contains the formulas for determining the baud rate. Example 33-1 provides a sample calculation for determining the baud rate and baud rate error.

Typical baud rates and error values for various Asynchronous modes have been computed for your convenience and are shown in Table 33-3. It may be advantageous to use the high baud rate (BRGH = 1), or the 16-bit BRG (BRG16 = 1) to reduce the baud rate error. The 16-bit BRG mode is used to achieve slow baud rates for fast oscillator frequencies.

Writing a new value to the SPxBRGH, SPxBRGL register pair causes the BRG timer to be reset (or cleared). This ensures that the BRG does not wait for a timer overflow before outputting the new baud rate.

If the system clock is changed during an active receive operation, a receive error or data loss may result. To avoid this problem, check the status of the RCIDL bit to make sure that the receive operation is idle before changing the system clock.

EXAMPLE 33-1: CALCULATING BAUD RATE ERROR

For a device with Fosc of 16 MHz, desired baud rate of 9600, Asynchronous mode, 8-bit BRG:

Desired Baud Rate = $\frac{FOSC}{64([SPBRGH:SPBRGL] + 1)}$

Solving for SPxBRGH:SPxBRGL:

$X = \frac{Fosc}{\frac{Desired Baud Rate}{64} - 1}$
$=\frac{\frac{16000000}{9600}}{64}-1$
= [25.042] = 25
Calculated Baud Rate = $\frac{16000000}{64(25+1)}$
= 9615
Error = $\frac{Calc. Baud Rate - Desired Baud Rate}{Desired Baud Rate}$
$=\frac{(9615-9600)}{9600} = 0.16\%$

SLEEP	Enter Sleep mode
Syntax:	[label] SLEEP
Operands:	None
Operation:	$\begin{array}{l} 00h \rightarrow WDT, \\ 0 \rightarrow WDT \mbox{ prescaler}, \\ 1 \rightarrow \overline{TO}, \\ 0 \rightarrow \overline{PD} \end{array}$
Status Affected:	TO, PD
Description:	The power-down Status bit, PD is cleared. Time-out Status bit, TO is set. Watchdog Timer and its prescaler are cleared. See Section 11.2 "Sleep Mode" for more information.

SUBWF	Subtract W from f		
Syntax:	[label] SU	JBWF f,d	
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$		
Operation:	(f) - (W) \rightarrow (destination)		
Status Affected:	C, DC, Z		
Description:	Subtract (2's complement method) W register from 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.		
	C = 0	W > f	
	C = 1	$W \leq f$	

DC = 0

	DC = 1	$W<3:0> \le f<3:0>$
SUBWFB	Subtract	t W from f with Borrow
Syntax:	SUBWFE	3 f {,d}
Operands:	$0 \le f \le 12$	7
	d ∈ [0,1]	
Operation:	(f) – (W) –	$-(\overline{B}) \rightarrow dest$
Status Affected:	C, DC, Z	
Description:	Subtract V (CARRY) compleme result is si	N and the BORROW flag from register 'f' (2's ent method). If 'd' is '0', the tored in W. If 'd' is '1', the

result is stored back in register 'f'.

W<3:0> > f<3:0>

SUBLW	Subtract W from literal			
Syntax:	[<i>label</i>] SUBLW k			
Operands:	$0 \le k \le 255$			
Operation:	$k - (W) \to (W)$			
Status Affected:	C, DC, Z			
Description:	The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.			
	C = 0 W > k			

U = 0	VV > K
C = 1	$W \leq k$
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SWAPF	Swap Nibbles in f		
Syntax:	[label] SWAPF f,d		
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$		
Operation:	$(f<3:0>) \rightarrow (destination<7:4>),$ $(f<7:4>) \rightarrow (destination<3:0>)$		
Status Affected:	None		
Description:	The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.		







FIGURE 37-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP





28-Lead Plastic Shrink Small Outline (SS) – 5.30 mm Body [SSOP]

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



	Units		MILLIMETERS		
Dimension	n Limits	MIN	NOM	MAX	
Number of Pins	Ν	28			
Pitch	е	0.65 BSC			
Overall Height	Α	-	-	2.00	
Molded Package Thickness	A2	1.65	1.75	1.85	
Standoff	A1	0.05	-	-	
Overall Width	Е	7.40	7.80	8.20	
Molded Package Width	E1	5.00	5.30	5.60	
Overall Length	D	9.90	10.20	10.50	
Foot Length	L	0.55	0.75	0.95	
Footprint	L1	1.25 REF			
Lead Thickness	с	0.09	-	0.25	
Foot Angle	φ	0°	4°	8°	
Lead Width	b	0.22	_	0.38	

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
 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 C04-073B

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

'Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEEL0Q® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV = ISO/TS 16949=

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, KeeLoq logo, Kleer, LANCheck, LINK MD, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC32 logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, ETHERSYNCH, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and QUIET-WIRE are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, RightTouch logo, REAL ICE, Ripple Blocker, Serial Quad I/O, SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2016, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-0987-8