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	12
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18426-e-sl

Pin Diagrams

1 14/16-Pin Diagrams

Figure 1. 14-Pin PDIP, SOIC, TSSOP

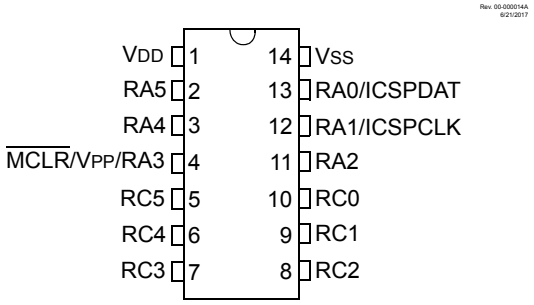
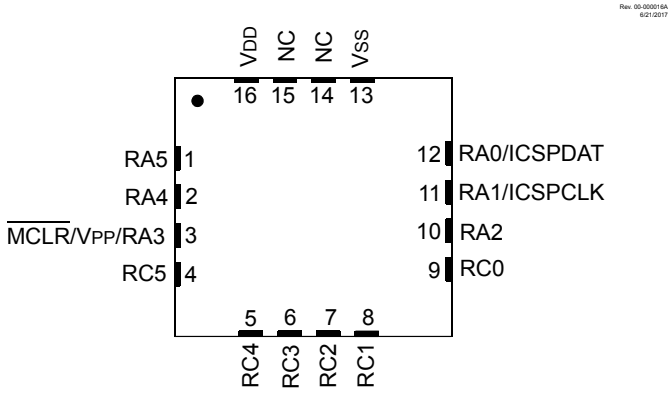


Figure 2. 16-Pin UQFN (4x4)



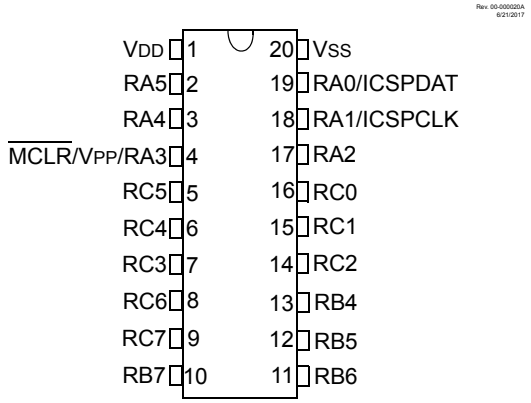
Note: It is recommended that the exposed bottom pad be connected to V_{SS}.

Related Links

14/16-Pin Allocation Table

2 20-Pin Diagrams

Figure 3. 20-Pin PDIP, SOIC, SSOP



I/O	14-pin PDIP/SOIC/TSSOP	16-pin UQFN	ADC	Reference	Comparator	NCO	DAC	DSM	Timers	CCP	PWM	CWG	MSSP	ZCD	EUSART	CLC	CLKR	Interrupts	Pull-up	Basic
RC0	10	9	ANC0	—	C2IN0+	—	—	—	T5CKI(1)	—	—	—	SCK1(1) SCL1(1,3,4)	—	—	—	—	IOCC0	Y	—
RC1	9	8	ANC1	—	C1IN1- C2IN1-	—	—	—	T4IN(1)	CCP4IN(1)	—	—	SDI1(1) SDA1(1,3,4)	—	—	CLCIN2(1)	—	IOCC1	Y	—
RC2	8	7	ANC2 ADACT(1)	—	C1IN2- C2IN2-	—	—	MDCARL(1)	—	—	—	—	—	—	—	—	—	IOCC2	Y	—
RC3	7	6	ANC3	—	C1IN3- C2IN3-	—	—	—	T5G(1)	CCP2IN(1)	—	—	$\overline{SS1}$(1)	—	—	CLCIN0(1)	—	IOCC3	Y	—
RC4	6	5	ANC4	—	—	—	—	—	T3G(1)	—	—	—	SCK2(1,5) SCL2(1,3,4,5)	—	CK1(1,3)	CLCIN1(1)	—	IOCC4	Y	—
RC5	5	4	ANC5	—	—	—	—	MDCARH(1)	T3CKI(1)	CCP1IN(1)	—	—	SDI2(1,5) SDA2(1,3,4,5)	—	RX1(1) DT1(1,3)	—	—	IOCC5	Y	—
V_{DD}	1	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{DD}
V_{SS}	14	13	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{SS}
OUT(2)	—	—	ADCGRDA	—	C1OUT	NCO1OUT	—	DSM1OUT	TMR0OUT	CCP1OUT	PWM6OUT	CWG1A CWG2A	SDO1 SDO2	—	DT1(3)	CLC1OUT	CLKR	—	—	—
	—	—	ADCGRDB	—	C2OUT	—	—	—	—	CCP2OUT	PWM7OUT	CWG1B CWG2B	SCK1 SCK2	—	CK1(3)	CLC2OUT	—	—	—	—
	—	—	—	—	—	—	—	—	—	CCP3OUT	—	CWG1C CWG2C	SCL1(3) SCL2(3)	—	TX1	CLC3OUT	—	—	—	—
	—	—	—	—	—	—	—	—	—	CCP4OUT	—	CWG1D CWG2D	SDA1(3) SDA2(3)	—	—	CLC4OUT	—	—	—	—

Note:

1. This is a PPS re-mappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins.
2. All digital output signals shown in these rows are PPS re-mappable. These signals may be mapped to output onto one of several PORTx pin options.
3. This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.
4. These pins may be configured for I^2C logic levels. PPS assignments to the other pins will operate, but input logic levels will be standard TTL/ST as selected by the INLVL register, instead of the I^2C specific or SMBUS input buffer thresholds.
5. MSSP2 is not available on the PIC16(L)F18424 or PIC16(L)F18444 devices.

Device ID, EEPROM, and Configuration Words" section for more information on accessing these memory locations. For more information on checksum calculation, see the *"PIC16(L)F184XX Memory Programming Specification"*, (DS40001970).

Related Links

NVMREG Access to Device Information Area, Device Configuration Area, User ID, Device ID, EEPROM, and Configuration Words

4.5 Device ID and Revision ID

The 14-bit device ID word is located at 0x8006 and the 14-bit revision ID is located at 0x8005. These locations are read-only and cannot be erased or modified.

Development tools, such as device programmers and debuggers, may be used to read the Device ID, Revision ID and Configuration Words. Refer to the *"Nonvolatile Memory (NVM) Control"* section for more information on accessing these locations.

Related Links

(NVM) Nonvolatile Memory Control

4.6 Register Summary - Configuration Words

Offset	Name	Bit Pos.								
0x8007	CONFIG1	7:0		RSTOSC[2:0]				FEXTOSC[2:0]		
		13:8			FCMEN		CSWEN			CLKOUTEN
0x8008	CONFIG2	7:0	BOREN		LPBOREN			PWRTS[1:0]		MCLRE
		13:8			DEBUG	STVREN	PPS1WAY	ZCDDIS	BORV	
0x8009	CONFIG3	7:0		WDTE[1:0]		WDTCPS[4:0]				
		13:8			WDTCCS[2:0]			WDTCWS[2:0]		
0x800A	CONFIG4	7:0	WRTAPP			SAFEN	BBEN	BBSIZE[2:0]		
		13:8			LVP		WRTSAF	WRTD	WRTC	WRTB
0x800B	CONFIG5	7:0								CP
		13:8								

4.7 Register Definitions: Configuration Words

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

The purpose of the temperature indicator module is to provide a temperature-dependent voltage that can be measured by an analog module. The *"Temperature Indicator Module"* chapter 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, V_{TSENSE} vs. Temperature curve.

- TSLR: Address 8112h to 8114h store the measurements for the low range setting of the temperature sensor at V_{DD} = 3V.
- TSHR: Address 8115h to 8117h store the measurements for the high range setting of the temperature sensor at V_{DD} = 3V.

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

Related Links

Temperature Indicator Module

5.4 Fixed Voltage Reference Data

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of V_{DD}, 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 the *"Fixed Voltage Reference (FVR)"* chapter (see related links).

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 1x 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)

Related Links

(FVR) Fixed Voltage Reference

9.6.7 OSCFRQ

Name: OSCFRQ
Address: 0x893

HFINTOSC Frequency Selection Register

Bit	7	6	5	4	3	2	1	0
						HFFRQ[2:0]		
Access						R/W	R/W	R/W
Reset						q	q	q

Bits 2:0 – HFFRQ[2:0] HFINTOSC Frequency Selection bits

FRQ<2:0>	Nominal Frequency (MHz) (NOSC = 110)	2x PLL Frequency (MHz) (NOSC = 001)
111	Reserved	Reserved
110	32	
101	16	32
100	12	24
011	8	16
010	4	Reserved
001	2	
000	1	

Note:

1. When RSTOSC = 110 (HFINTOSC 1 MHz), the FRQ bits will default to '010' upon Reset; when RSTOSC = 001 (HFINTOSC 32 MHz), the FRQ bits will default to '101' upon Reset.

- Current draw from pins with internal weak pull-ups
- Modules using any oscillator

I/O pins that are high-impedance inputs should be pulled to V_{DD} or V_{SS} externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include modules such as the DAC and FVR modules.

Related Links

Low-Power Sleep Mode
STATUS
(FVR) Fixed Voltage Reference
(DAC) 5-Bit Digital-to-Analog Converter Module

11.2.1 Wake-up from Sleep

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{\text{MCLR}}$ pin, if enabled.
2. BOR Reset, if enabled.
3. POR Reset.
4. Windowed Watchdog Timer, if enabled.
5. Any external interrupt.
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information).

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to the *"Memory Execution Violation"* section.

When the `SLEEP` instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the `SLEEP` instruction. If the GIE bit is enabled, the device executes the instruction after the `SLEEP` instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

The WDT is cleared when the device wakes-up from Sleep, regardless of the source of wake-up.

Related Links

Memory Execution Violation

11.2.2 Wake-up Using Interrupts

When global interrupts are disabled (GIE cleared) and any interrupt source, with the exception of the clock switch interrupt, has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs before the execution of a `SLEEP` instruction
  - `SLEEP` instruction will execute as a `NOP`
  - WDT and WDT prescaler will not be cleared
  - $\overline{\text{TO}}$ bit of the STATUS register will not be set

The Low-Power Sleep mode is beneficial for applications that stay in Sleep mode for long periods of time. The Normal mode is beneficial for applications that need to wake from Sleep quickly and frequently.

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)
- Windowed Watchdog Timer (WWDT)
- 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.

Important: The LF devices do not have a configurable Low-Power Sleep mode. LFs are unregulated devices and are always in the lowest power state when in Sleep, with no wake-up time penalty. These devices have a lower maximum V_{DD} and I/O voltage than the F devices.

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. 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 program memory are shut off.

Important: Peripherals using F_{OSC} will continue running while in Idle (but not in Sleep). Peripherals using HFINTOSC:LFINTOSC will continue running in both Idle and Sleep.

Important: If $\overline{\text{CLKOUTEN}}$ is enabled ($\overline{\text{CLKOUTEN}}$ = 0, Configuration Word 1), the output will continue operating while in Idle.

11.3.1 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.

11.3.2 Idle and WWDT

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

the allowed window of the WDT period. Any CLRWDT instruction that occurs outside of this window will trigger a window violation and will cause a WWDT Reset, similar to a WWDT time out. See Figure 12-2 for an example.

The window size is controlled by the WINDOW Configuration bits, or the WINDOW bits, if WDTCWS = 111.

The five Most Significant bits of the WDTTMR register are used to determine whether the window is open, as defined by the WINDOW bits.

In the event of a window violation, a Reset will be generated and the $\overline{\text{WDTWV}}$ bit of the PCON0 register will be cleared. This bit is set by a POR or can be set in firmware.

Related Links

PCON0

12.5 Clearing the WWDT

The WWDT is cleared when any of the following conditions occur:

- Any Reset
- Valid CLRWDT instruction is executed
- Device enters Sleep
- Exit Sleep by Interrupt
- WWDT is disabled
- Oscillator Start-up Timer (OST) is running
- Any write to the WDTCON0 or WDTCON1 registers

12.5.1 CLRWDT Considerations (Windowed Mode)

When in Windowed mode, the WWDT must be armed before a CLRWDT instruction will clear the timer. This is performed by reading the WDTCON0 register. Executing a CLRWDT instruction without performing such an arming action will trigger a window violation regardless of whether the window is open or not.

See Table 12-2 for more information.

12.6 Operation During Sleep

When the device enters Sleep, the WWDT is cleared. If the WWDT is enabled during Sleep, the WWDT resumes counting. When the device exits Sleep, the WWDT is cleared again.

The WWDT remains clear until the Oscillator Start-up Timer (OST) completes, if enabled.

When a WWDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits in the STATUS register are changed to indicate the event. The $\overline{\text{RWDT}}$ bit in the PCON0 register can also be used.

Table 12-2. WWDT Clearing Conditions

Conditions	WWDT
WDTE = 00	Cleared
WDTE = 01 and SEN = 0	

16. (PMD) Peripheral Module Disable

This module provides the ability to selectively enable or disable a peripheral. Disabling a peripheral places it in its lowest possible power state. The user can disable unused modules to reduce the overall power consumption.

The PIC16(L)F18426/46 devices address this requirement by allowing peripheral modules to be selectively enabled or disabled. Disabling a peripheral places it in the lowest possible power mode.

Important: All modules are ON by default following any system Reset.

16.1 Disabling a Module

A peripheral can be disabled by setting the corresponding peripheral disable bit in the PMDx register. Disabling a module has the following effects:

- The module is held in Reset and does not function.
- All the SFRs pertaining to that peripheral become “unimplemented”
  - Writing is disabled
  - Reading returns 0x00
- Module outputs are disabled

Related Links

PPSLOCK

16.2 Enabling a Module

Clearing the corresponding module disable bit in the PMDx register, re-enables the module and the SFRs will reflect the Power-on Reset values.

Important: There should be no reads/writes to the module SFRs for at least two instruction cycles after it has been re-enabled.

16.3 System Clock Disable

Setting SYSCMD disables the system clock (F_{OSC}) distribution network to the peripherals. Not all peripherals make use of SYSCLK, so not all peripherals are affected. Refer to the specific peripheral description to see if it will be affected by this bit.

Related Links

PMD0

22.9.4 NCOxINC

Name: NCOxINC
Address: 0x058F

NCO Increment Register

Bit	23	22	21	20	19	18	17	16
					INCU[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
	INCH[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	INCL[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 19:16 – INCU[3:0] NCO Increment – Upper Byte(1)

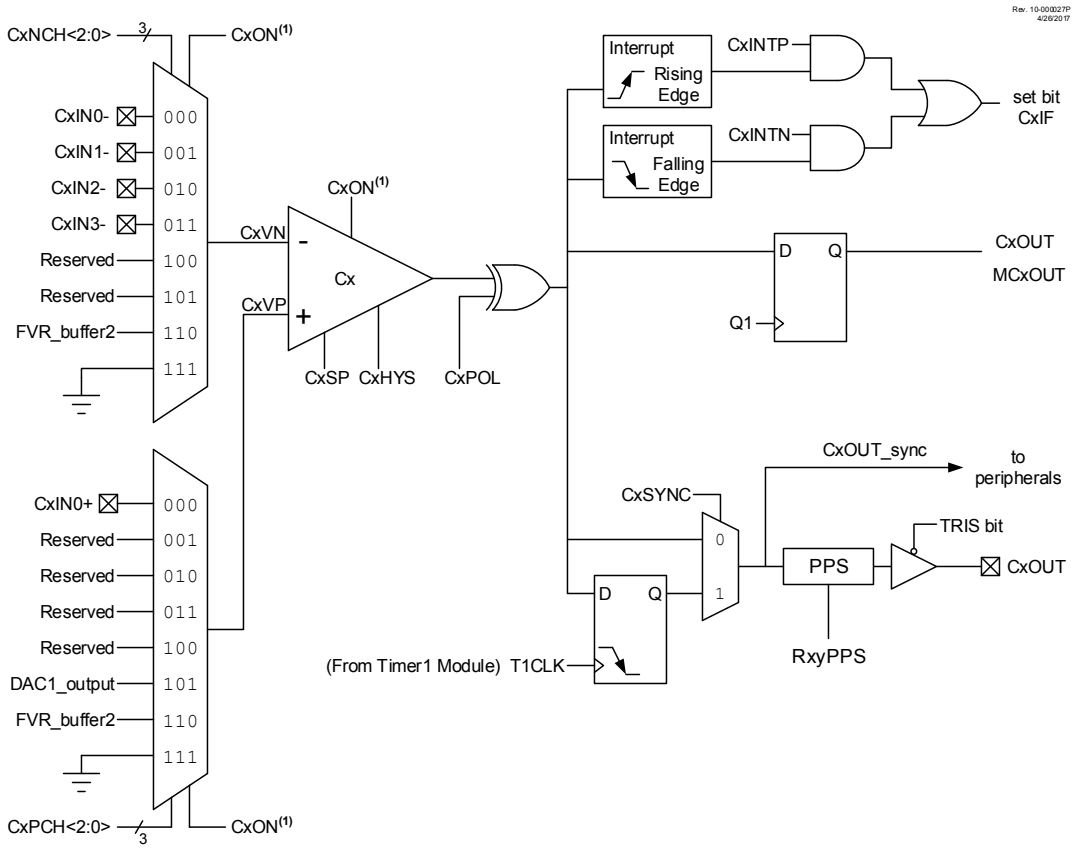
Bits 15:8 – INCH[7:0] NCO Increment – High Byte(1)

Bits 7:0 – INCL[7:0] NCO Increment – Low Byte(1,2)

Note:

1. The logical increment spans NCOxINCU:NCOxINCH:NCOxINCL.
2. NCOxINC is double-buffered as INCBUF; INCBUF is updated on the next falling edge of NCOxCLK after writing to NCOxINCL; NCOxINCU and NCOxINCH should be written prior to writing NCOxINCL.

Figure 23-2. Comparator Module Simplified Block Diagram



Related Links

CMxNCH

CMxPCH

23.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Hysteresis enable
- Timer1 output synchronization

The CMxCON1 register contains Control bits for the following:

- Interrupt on positive/negative edge enables
- Positive input channel selection
- Negative input channel selection

PCH	Positive Input Source
101	CxV_P connects to DAC1 output
100	CxV_P not connected
011	CxV_P not connected
010	CxV_P not connected
001	CxV_P connects to CxIN1+ pin
000	CxV_P connects to CxIN0+ pin

Important: To use CxINy+ pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

See Fixed Voltage Reference (FVR) for more information on the Fixed Voltage Reference module.

See 5-Bit Digital-to-Analog Converter (DAC) module for more information on the DAC input signal.

Any time the comparator is disabled (CxEN = 0), all comparator inputs are disabled.

Related Links

(FVR) Fixed Voltage Reference

(DAC) 5-Bit Digital-to-Analog Converter Module

23.7 Comparator Negative Input Selection

The NCH bits direct an analog input pin and internal reference voltage or analog ground to the inverting input of the comparator:

NCH	Negative Input Sources
111	CxV_N connects to AV_{SS}
110	CxV_N connects to FVR Buffer 2
101	CxV_N not connected
100	CxV_N not connected
011	CxV_N connects to CxIN3- pin
010	CxV_N connects to CxIN2- pin
001	CxV_N connects to CxIN1- pin
000	CxV_N connects to CxIN0- pin

Important: To use CxINy- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

23.10 CWG1 Auto-Shutdown Source

The output of the Comparator module can be used as an auto-shutdown source for the CWG1 module. When the output of the comparator is active and the corresponding WGASxE is enabled, the CWG operation will be suspended immediately.

Related Links

External Input Source

23.11 ADC Auto-Trigger Source

The output of the Comparator module can be used to trigger an ADC conversion. When the ADACT register is set to trigger on a comparator output, an ADC conversion will trigger when the comparator output goes high.

23.12 Even Numbered Timers Reset

The output of the Comparator module can be used to reset the even numbered timers (Timer2, Timer4, etc.). When the TxERS register is appropriately set, the timer will reset when the comparator output goes high.

23.13 Operation in Sleep Mode

The Comparator module can operate during Sleep. The comparator clock source is based on the Timer1 clock source. If the Timer1 clock source is either the system clock (F_{OSC}) or the instruction clock ($F_{OSC}/4$), Timer1 will not operate during Sleep, and synchronized comparator outputs will not operate.

A comparator interrupt will wake the device from Sleep. The CxIE bits of the PIEx register must be set to enable comparator interrupts.

$$T_{\Phi} = \frac{\Phi}{2\pi f} = 125.6\mu s$$

24.5.2 Correction By Offset Current

When the waveform is varying relative to V_{SS}, then the zero cross is detected too early as the waveform falls and too late as the waveform rises. When the waveform is varying relative to V_{DD}, then the zero cross is detected too late as the waveform rises and too early as the waveform falls. The actual offset time can be determined for sinusoidal waveforms with the corresponding equations shown below.

Equation 24-4. ZCD Event Offset

When External Voltage source is relative to V_{SS}

$$T_{offset} = \frac{\sin^{-1}\left(\frac{Z_{CPINV}}{V_{PEAK}}\right)}{2\pi f}$$

When External Voltage source is relative to V_{DD}

$$T_{offset} = \frac{\sin^{-1}\left(\frac{V_{DD} - Z_{CPINV}}{V_{PEAK}}\right)}{2\pi f}$$

This offset time can be compensated for by adding a pull-up or pull-down biasing resistor to the ZCD pin. A pull-up resistor is used when the external voltage source is varying relative to V_{SS}. A pull-down resistor is used when the voltage is varying relative to V_{DD}. The resistor adds a bias to the ZCD pin so that the target external voltage source must go to zero to pull the pin voltage to the Z_{CPINV} switching voltage. The pull-up or pull-down value can be determined with the equations shown below.

Equation 24-5. ZCD Pull-up/Pull-down Resistor

When External Voltage source is relative to V_{SS}

$$R_{pullup} = \frac{R_{SERIES}\left(V_{pullup} - Z_{CPINV}\right)}{Z_{CPINV}}$$

When External Voltage source is relative to V_{DD}

$$R_{pulldown} = \frac{R_{SERIES}(Z_{CPINV})}{(V_{DD} - Z_{CPINV})}$$

24.6 Handling V_{PEAK} Variations

If the peak amplitude of the external voltage is expected to vary, the series resistor must be selected to keep the ZCD current source and sink below the design maximum range of ± 600 µA and above a reasonable minimum range. A general rule of thumb is that the maximum peak voltage can be no more than six times the minimum peak voltage. To ensure that the maximum current does not exceed ± 600 µA and the minimum is at least ± 100 µA, compute the series resistance as shown in Equation 24-6. The compensating pull-up for this series resistance can be determined with the equations shown in Equation 24-5 because the pull-up value is independent from the peak voltage.

30.12.2 PWMxDC

Name: PWMxDC
Address: 0x38C,0x390

PWM Duty Cycle Register

Bit	15	14	13	12	11	10	9	8
	DCH[7:0]							
Access								
Reset	x	x	x	x	x	x	x	x

Bit	7	6	5	4	3	2	1	0
	DCL[1:0]							
Access								
Reset	x	x						

Bits 15:8 – DCH[7:0] PWM Duty Cycle Most Significant bits
These bits are the MSbs of the PWM duty cycle.

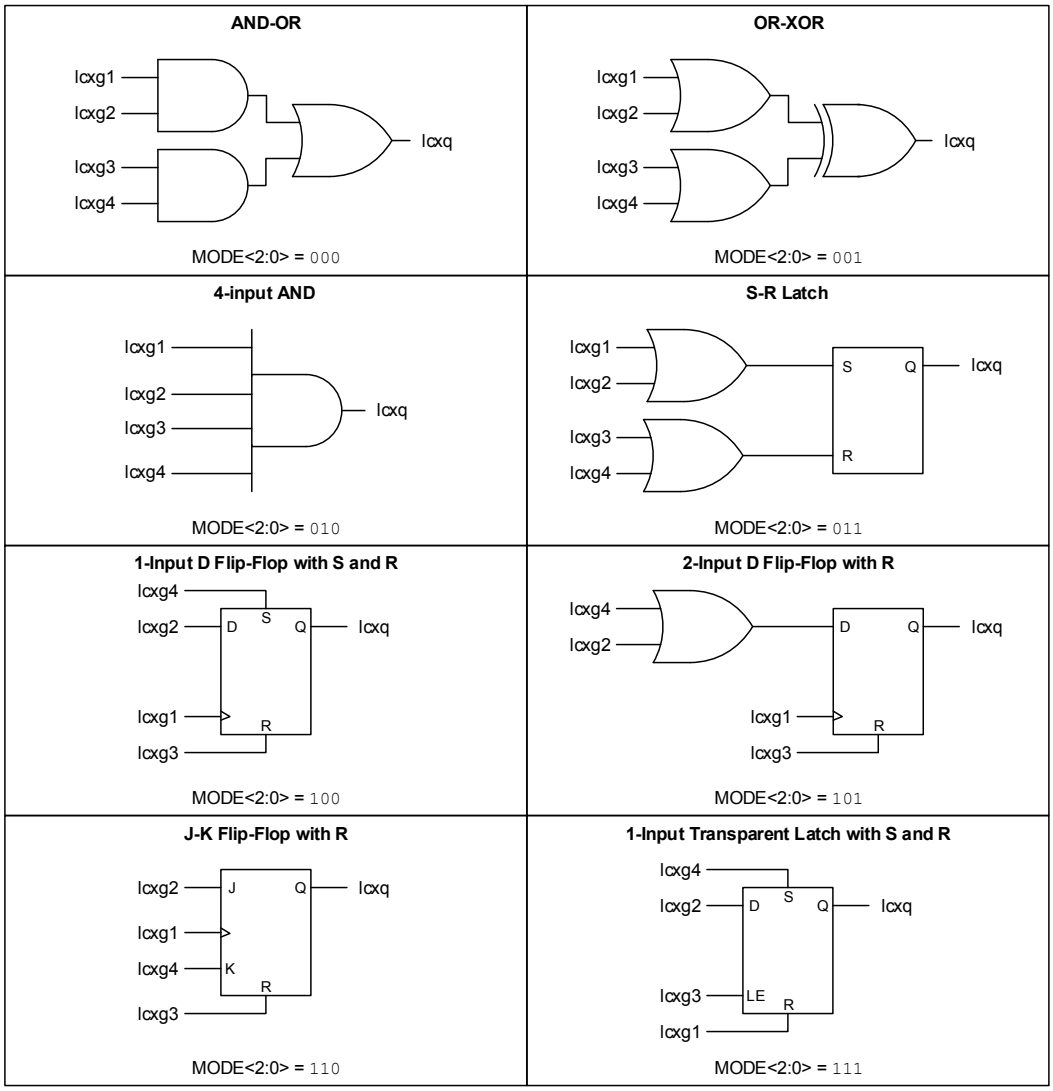
Reset States: POR/BOR = xxxxxxxx
All Other Resets = uuuuuuuu

Bits 7:6 – DCL[1:0] PWM Duty Cycle Least Significant bits
These bits are the LSbs of the PWM duty cycle.

Reset States: POR/BOR = xx
All Other Resets = uu

Figure 33-3. Programmable Logic Functions

Rev. 10-000122B
9/13/2016



33.1.4 Output Polarity

The last stage in the Configurable Logic Cell is the output polarity. Setting the POL bit 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.

33.2 CLC Interrupts

An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

35.5.3.3 7-bit Transmission with Address Hold Enabled

Setting the AHEN bit enables additional clock stretching and interrupt generation after the eighth falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPxIF interrupt is set.

Figure 35-19 displays a standard waveform of a 7-bit address slave transmission with AHEN enabled.

1. Bus starts Idle.
2. Master sends Start condition; the S bit is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Master sends matching address with R/W bit set. After the 8th falling edge of the SCL line the CKP bit is cleared and SSPxIF interrupt is generated.
4. Slave software clears SSPxIF.
5. Slave software reads the ACKTIM, R/W and D/A bits to determine the source of the interrupt.
6. Slave reads the address value from the SSPxBUF register clearing the BF bit.
7. Slave software decides from this information if it wishes to $\overline{ACK}$ or not $\overline{ACK}$ and sets the ACKDT bit accordingly.
8. Slave sets the CKP bit releasing SCL.
9. Master clocks in the $\overline{ACK}$ value from the slave.
10. Slave hardware automatically clears the CKP bit and sets SSPxIF after the $\overline{ACK}$ if the R/W bit is set.
11. Slave software clears SSPxIF.
12. Slave loads value to transmit to the master into SSPxBUF setting the BF bit.

Important: SSPxBUF cannot be loaded until after the $\overline{ACK}$.

13. Slave sets the CKP bit releasing the clock.
14. Master clocks out the data from the slave and sends an $\overline{ACK}$ value on the ninth SCL pulse.
15. Slave hardware copies the $\overline{ACK}$ value into the ACKSTAT bit.
16. Steps 10-15 are repeated for each byte transmitted to the master from the slave.
17. If the master sends a not $\overline{ACK}$ the slave releases the bus allowing the master to send a Stop and end the communication.

Important: Master must send a not $\overline{ACK}$ on the last byte to ensure that the slave releases the SCL line to receive a Stop.

35.6.1 I²C Master Mode Operation

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the R/W bit. In this case, the R/W bit will be logic '0'. Serial data is transmitted eight bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

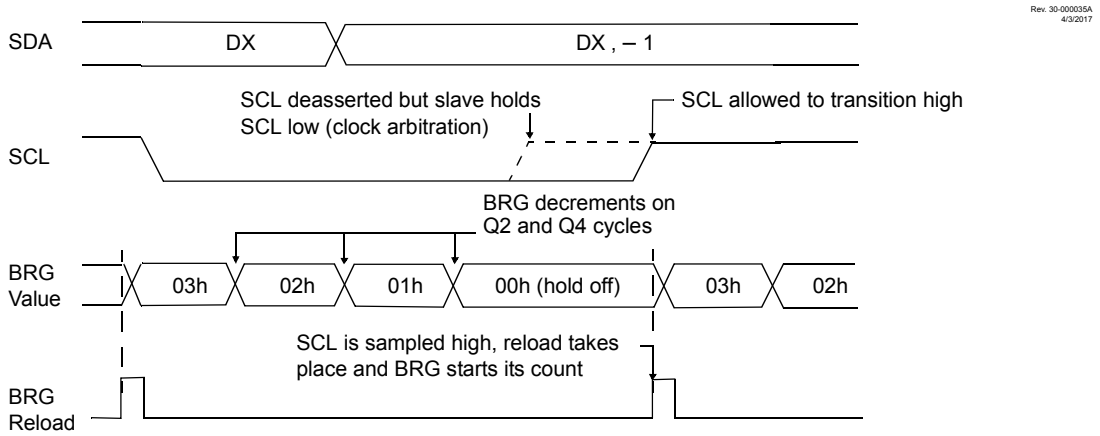
In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case, the R/$\overline{W}$ bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate the receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received eight bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

A Baud Rate Generator is used to set the clock frequency output on SCL. See Baud Rate Generator for more detail.

35.6.2 Clock Arbitration

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, releases the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device as shown in the following figure.

Figure 35-25. Baud Rate Generator Timing with Clock Arbitration



35.6.3 WCOL Status Flag

If the user writes the SSPxBUF when a Start, Restart, Stop, Receive or Transmit sequence is in progress, the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur). Any time the WCOL bit is set it indicates that an action on SSPxBUF was attempted while the module was not idle.

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPxBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

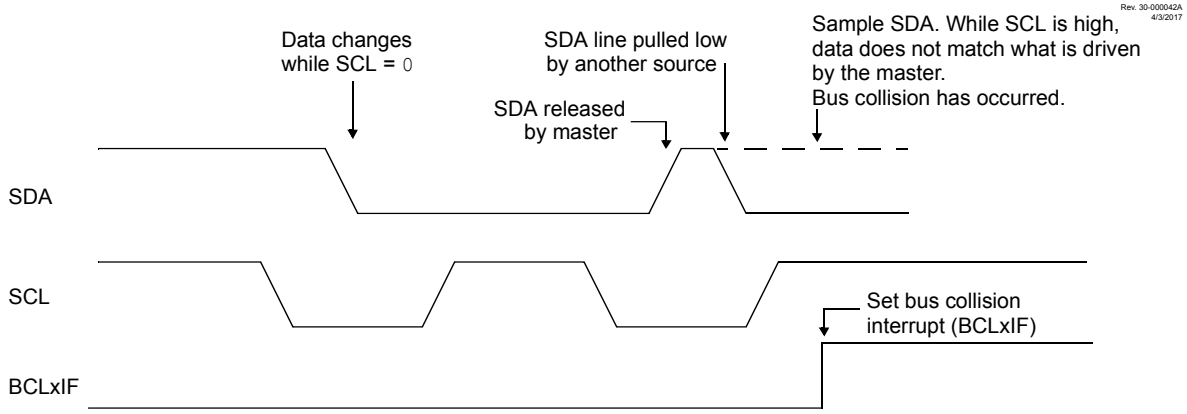
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPxCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPxIF bit will be set.

A write to the SSPxBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set, or the bus is Idle and the S and P bits are cleared.

Figure 35-32. Bus Collision Timing for Transmit and Acknowledge



35.6.13.1 Bus Collision During a Start Condition

During a Start condition, a bus collision occurs if:

1. SDA or SCL are sampled low at the beginning of the Start condition (Figure 35-33).
2. SCL is sampled low before SDA is asserted low (Figure 35-34).

During a Start condition, both the SDA and the SCL pins are monitored.

If the SDA pin is already low, or the SCL pin is already low, then all of the following occur:

- the Start condition is aborted,
- the BCLxIF flag is set and
- the MSSP module is reset to its Idle state (Figure 35-33).

38. Register Summary

Offset	Name	Bit Pos.								
0x00	INDF0	7:0	INDF0[7:0]							
0x01	INDF1	7:0	INDF1[7:0]							
0x02	PCL	7:0	PCL[7:0]							
0x03	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x04	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x06	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x08	BSR	7:0			BSR[5:0]					
0x09	WREG	7:0	WREG[7:0]							
0x0A	PCLATH	7:0		PCLATH[6:0]						
0x0B	INTCON	7:0	GIE	PEIE						INTEDG
0x0C	PORTA	7:0			RA5	RA4	RA3	RA2	RA1	RA0
0x0D	PORTB	7:0	RB7	RB6	RB5	RB4				
0x0E	PORTC	7:0	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
0x0F ... 0x11	Reserved									
0x12	TRISA	7:0			TRISA5	TRISA4		TRISA2	TRISA1	TRISA0
0x13	TRISB	7:0	TRISB7	TRISB6	TRISB5	TRISB4				
0x14	TRISC	7:0	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
0x15 ... 0x17	Reserved									
0x18	LATA	7:0			LATA5	LATA4		LATA2	LATA1	LATA0
0x19	LATB	7:0	LATB7	LATB6	LATB5	LATB4				
0x1A	LATC	7:0	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
0x1B ... 0x7F	Reserved									
0x80	INDF0	7:0	INDF0[7:0]							
0x81	INDF1	7:0	INDF1[7:0]							
0x82	PCL	7:0	PCL[7:0]							
0x83	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x84	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x86	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x88	BSR	7:0			BSR[5:0]					
0x89	WREG	7:0	WREG[7:0]							
0x8A	PCLATH	7:0		PCLATH[6:0]						
0x8B	INTCON	7:0	GIE	PEIE						INTEDG
0x8C	ADLTH	7:0	LTHL[7:0]							