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	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	44
Program Memory Size	32KB (16K x 16)
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 43x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP Exposed Pad
Supplier Device Package	48-TQFP-EP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf55k42-i-pt

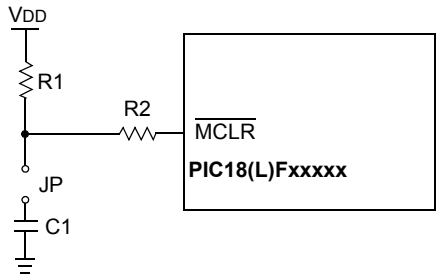
2.3 Master Clear (MCLR) Pin

The MCLR pin provides two specific device functions: Device Reset, and Device Programming and Debugging. If programming and debugging are not required in the end application, a direct connection to VDD may be all that is required. The addition of other components, to help increase the application's resistance to spurious Resets from voltage sags, may be beneficial. A typical configuration is shown in Figure 2-1. Other circuit designs may be implemented, depending on the application requirements.

During programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the MCLR pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R1 and C1 will need to be adjusted based on the application and PCB requirements. For example, it is recommended that the capacitor, C1, be isolated from the MCLR pin during programming and debugging operations by using a jumper (Figure 2-2). The jumper is replaced for normal run-time operations.

Any components associated with the MCLR pin should be placed within 0.25 inch (6 mm) of the pin.

FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



Note 1: R1 ≤ 10 kΩ is recommended. A suggested starting value is 10 kΩ. Ensure that the MCLR pin VIH and VIL specifications are met.

2: R2 ≤ 470Ω will limit any current flowing into MCLR from the external capacitor, C1, in the event of MCLR pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the MCLR pin VIH and VIL specifications are met.

2.4 ICSP™ Pins

The ICSPCLK and ICSPDAT pins are used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of ohms, not to exceed 100Ω.

Pull-up resistors, series diodes and capacitors on the ICSPCLK and ICSPDAT pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits, and pin input voltage high (VIH) and input low (VIL) requirements.

For device emulation, ensure that the "Communication Channel Select" (i.e., ICSPCLK/ICSPDAT pins), programmed into the device, matches the physical connections for the ICSP to the Microchip debugger/emulator tool.

For more information on available Microchip development tools connection requirements, refer to **Section 43.0 "Development Support"**.

3.1.1 PRIORITY LOCK

The System arbiter grants memory access to the peripheral selections (DMAx, Scanner) when the PRLOCKED bit (PRLOCK Register) is set.

Priority selections are locked by setting the PRLOCKED bit of the PRLOCK register. Setting and clearing this bit requires a special sequence as an extra precaution against inadvertent changes. Examples of setting and clearing the PRLOCKED bit are shown in Example 3-1 and Example 3-2.

EXAMPLE 3-1: PRIORITY LOCK SEQUENCE

```
; Disable interrupts
BCF INTCON0,GIE

; Bank to PRLOCK register
BANKSEL PRLOCK
MOVLW 55h

; Required sequence, next 4
instructions
MOVWF PRLOCK
MOVLW AAh
MOVWF PRLOCK
; Set PRLOCKED bit to grant memory
access to peripherals
BSF PRLOCK,0

; Enable Interrupts
BSF INTCON0,GIE
```

EXAMPLE 3-2: PRIORITY UNLOCK SEQUENCE

```
; Disable interrupts
BCF INTCON0,GIE

; Bank to PRLOCK register
BANKSEL PRLOCK
MOVLW 55h

; Required sequence, next 4
instructions
MOVWF PRLOCK
MOVLW AAh
MOVWF PRLOCK
; Clear PRLOCKED bit to allow changing
priority settings
BCF PRLOCK,0

; Enable Interrupts
BSF INTCON0,GIE
```

3.2 Memory Access Scheme

The user can assign priorities to both system level and peripheral selections based on which the system arbiter grants memory access. Let us consider the following priority scenarios between ISR, MAIN, and Peripherals.

Note:	It is always required that the ISR priority be higher than Main priority.

3.2.1 ISR PRIORITY > MAIN PRIORITY > PERIPHERAL PRIORITY

When the Peripheral Priority (DMAx, Scanner) is lower than ISR and MAIN Priority, and the peripheral requires:

1. Access to the Program Flash Memory, then the peripheral waits for an instruction cycle in which the CPU does not need to access the PFM (such as a branch instruction) and uses that cycle to do its own Program Flash Memory access, unless a PFM Read/Write operation is in progress.
2. Access to the SFR/GPR, then the peripheral waits for an instruction cycle in which the CPU does not need to access the SFR/GPR (such as `MOVLW`, `CALL`, `NOP`) and uses that cycle to do its own SFR/GPR access.
3. Access to the Data EEPROM, then the peripheral has access to Data EEPROM unless a Data EEPROM Read/Write operation is being performed.

This results in the lowest throughput for the peripheral to access the memory, and does so without any impact on execution times.

3.2.2 PERIPHERAL PRIORITY > ISR PRIORITY > MAIN PRIORITY

When the Peripheral Priority (DMAx, Scanner) is higher than ISR and MAIN Priority, the CPU operation is stalled when the peripheral requests memory.

The CPU is held in its current state until the peripheral completes its operation. Since the peripheral requests access to the bus, the peripheral cannot be disabled until it completes its operation.

This results in the highest throughput for the peripheral to access the memory, but has the cost of stalling other execution while it occurs.

TABLE 4-3: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES BANK 64

40FFh	—	40DFh	—	40BFh	—	409Fh	—	407Fh	—	405Fh	—	403Fh	—	401Fh	—
40FEh	—	40DEh	—	40BEh	—	409Eh	—	407Eh	—	405Eh	—	403Eh	—	401Eh	—
40FDh	—	40DDh	T6PR_M2	40BDh	ADRESH_M2	409Dh	—	407Dh	—	405Dh	—	403Dh	—	401Dh	—
40FCh	—	40DCh	PWM5DCH_M2	40BCh	ADRESL_M2	409Ch	—	407Ch	—	405Ch	—	403Ch	—	401Ch	—
40FBh	TMR5H_M1	40DBh	PWM5DCL_M2	40BBh	ADPCH_M2	409Bh	—	407Bh	—	405Bh	—	403Bh	—	401Bh	—
40FAh	TMR5L_M1	40DAh	T6PR_M1	40BAh	ADCLK_M1	409Ah	—	407Ah	—	405Ah	—	403Ah	—	401Ah	—
40F9h	TMR3H_M1	40D9h	CCPR1H_M2	40B9h	ADACT_M1	4099h	—	4079h	—	4059h	—	4039h	—	4019h	—
40F8h	TMR3L_M1	40D8h	CCPR1L_M2	40B8h	ADREF_M1	4098h	—	4078h	—	4058h	—	4038h	—	4018h	—
40F7h	TMR1H_M1	40D7h	T4PR_M4	40B7h	ADCON3_M1	4097h	—	4077h	—	4057h	—	4037h	—	4017h	—
40F6h	TMR1L_M1	40D6h	PWM8DCH_M1	40B6h	ADCON2_M1	4096h	ADRESH_M1	4076h	—	4056h	—	4036h	—	4016h	—
40F5h	—	40D5h	PWM8DCL_M1	40B5h	ADCON1_M1	4095h	ADRESL_M1	4075h	—	4055h	—	4035h	—	4015h	—
40F4h	—	40D4h	T4PR_M3	40B4h	ADCON0_M1	4094h	ADPCH_M1	4074h	—	4054h	—	4034h	—	4014h	—
40F3h	—	40D3h	PWM7DCH_M1	40B3h	ADCAP_M2	4093h	ADCAP_M1	4073h	—	4053h	—	4033h	—	4013h	—
40F2h	—	40D2h	PWM7DCL_M1	40B2h	ADACQH_M2	4092h	ADACQH_M1	4072h	—	4052h	—	4032h	—	4012h	—
40F1h	—	40D1h	T4PR_M2	40B1h	ADACQL_M2	4091h	ADACQL_M1	4071h	—	4051h	—	4031h	—	4011h	—
40F0h	—	40D0h	CCPR4H_M1	40B0h	ADPREVH_M2	4090h	ADPREVH_M1	4070h	—	4050h	—	4030h	—	4010h	—
40EFh	PWM8DCH_M2	40CFh	CCPR4L_M1	40AFh	ADPREVL_M2	408Fh	ADPREVL_M1	406Fh	—	404Fh	—	402Fh	—	400Fh	—
40EEh	PWM8DCL_M2	40CEh	T4PR_M1	40AEh	ADRPT_M2	408Eh	ADRPT_M1	406Eh	—	404Eh	—	402Eh	—	400Eh	—
40EDh	PWM7DCH_M2	40CDh	CCPR3H_M1	40ADh	ADCNT_M2	408Dh	ADCNT_M1	406Dh	—	404Dh	—	402Dh	—	400Dh	—
40ECh	PWM7DCL_M2	40CCh	CCPR3L_M1	40ACh	ADACCU_M2	408Ch	ADACCU_M1	406Ch	—	404Ch	—	402Ch	—	400Ch	—
40EBh	PWM6DCH_M2	40CBh	T2PR_M3	40ABh	ADACCH_M2	408Bh	ADACCH_M1	406Bh	—	404Bh	—	402Bh	—	400Bh	—
40EAh	PWM6DCL_M2	40CAh	PWM6DCH_M1	40AAh	ADACCL_M2	408Ah	ADACCL_M1	406Ah	—	404Ah	—	402Ah	—	400Ah	—
40E9h	PWM5DCH_M3	40C9h	PWM6DCL_M1	40A9h	ADFLTRH_M2	4089h	ADFLTRH_M1	4069h	—	4049h	—	4029h	—	4009h	—
40E8h	PWM5DCL_M3	40C8h	T2PR_M2	40A8h	ADFLTRL_M2	4088h	ADFLTRL_M1	4068h	—	4048h	—	4028h	—	4008h	—
40E7h	CCPR4H_M2	40C7h	PWM5DCH_M1	40A7h	ADSTPTH_M2	4087h	ADSTPTH_M1	4067h	—	4047h	—	4027h	—	4007h	—
40E6h	CCPR4L_M2	40C6h	PWM5DCL_M1	40A6h	ADSTPTL_M2	4086h	ADSTPTL_M1	4066h	—	4046h	—	4026h	—	4006h	—
40E5h	CCPR3H_M2	40C5h	T2PR_M2	40A5h	ADERRH_M2	4085h	ADERRH_M1	4065h	—	4045h	—	4025h	—	4005h	—
40E4h	CCPR3L_M2	40C4h	CCPR2H_M1	40A4h	ADERRL_M2	4084h	ADERRL_M1	4064h	—	4044h	—	4024h	—	4004h	—
40E3h	CCPR2H_M2	40C3h	CCPR2L_M1	40A3h	ADUTHH_M2	4083h	ADUTHH_M1	4063h	IOCEF_M1	4043h	—	4023h	—	4003h	—
40E2h	CCPR2L_M2	40C2h	T2PR_M1	40A2h	ADUTHL_M2	4082h	ADUTHL_M1	4062h	IOCCF_M1	4042h	—	4022h	—	4002h	—
40E1h	CCPR1H_M3	40C1h	CCPR1H_M1	40A1h	ADLTHH_M2	4081h	ADLTHH_M1	4061h	IOCBF_M1	4041h	—	4021h	—	4001h	—
40E0h	CCPR1L_M3	40C0h	CCPR1L_M1	40A0h	ADLTHL_M2	4080h	ADLTHL_M1	4060h	IOCAF_M1	4040h	—	4020h	—	4000h	—

Note 1: Addresses in Bank 64 are accessible ONLY through DMA Source and Destination Address Registers. CPU does not have access to registers in Bank 64.

TABLE 4-6: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES BANK 61

3DFFh	—	3DDFh	U2FIFO	3DBFh	—	3D9Fh	—	3D7Fh	—	3D5Fh	I2C2CON2	3D3Fh	—	3D1Fh	—
3DFEh	—	3DDEh	U2BRGH	3DBEh	—	3D9Eh	—	3D7Eh	—	3D5Eh	I2C2CON1	3D3Eh	—	3D1Eh	—
3DFDh	—	3DDDh	U2BRGL	3DBDh	—	3D9Dh	—	3D7Dh	—	3D5Dh	I2C2CON0	3D3Dh	—	3D1Dh	—
3DFCh	—	3DDCh	U2CON2	3DBCh	—	3D9Ch	—	3D7Ch	I2C1BTO	3D5Ch	I2C2ADR3	3D3Ch	—	3D1Ch	SPI1CLK
3DFBh	—	3DDBh	U2CON1	3DBBh	—	3D9Bh	—	3D7Bh	I2C1CLK	3D5Bh	I2C2ADR2	3D3Bh	—	3D1Bh	SPI1INTE
3DFAh	U1ERRIE	3DDAh	U2CON0	3DBAh	—	3D9Ah	—	3D7Ah	I2C1PIE	3D5Ah	I2C2ADR1	3D3Ah	—	3D1Ah	SPI1INTF
3DF9h	U1ERRIR	3DD9h	—	3DB9h	—	3D99h	—	3D79h	I2C1PIR	3D59h	I2C2ADR0	3D39h	—	3D19h	SPI1BAUD
3DF8h	U1UIR	3DD8h	U2P3L	3DB8h	—	3D98h	—	3D78h	I2C1STAT1	3D58h	I2C2ADB1	3D38h	—	3D18h	SPI1TWIDTH
3DF7h	U1FIFO	3DD7h	—	3DB7h	—	3D97h	—	3D77h	I2C1STAT0	3D57h	I2C2ADB0	3D37h	—	3D17h	SPI1STATUS
3DF6h	U1BRGH	3DD6h	U2P2L	3DB6h	—	3D96h	—	3D76h	I2C1ERR	3D56h	I2C2CNT	3D36h	—	3D16h	SPI1CON2
3DF5h	U1BRGL	3DD5h	—	3DB5h	—	3D95h	—	3D75h	I2C1CON2	3D55h	I2C2TXB	3D35h	—	3D15h	SPI1CON1
3DF4h	U1CON2	3DD4h	U2P1L	3DB4h	—	3D94h	—	3D74h	I2C1CON1	3D54h	I2C2RXB	3D34h	—	3D14h	SPI1CON0
3DF3h	U1CON1	3DD3h	—	3DB3h	—	3D93h	—	3D73h	I2C1CON0	3D53h	—	3D33h	—	3D13h	SPI1TCNTH
3DF2h	U1CON0	3DD2h	U2TXB	3DB2h	—	3D92h	—	3D72h	I2C1ADR3	3D52h	—	3D32h	—	3D12h	SPI1TCNTL
3DF1h	U1P3H	3DD1h	—	3DB1h	—	3D91h	—	3D71h	I2C1ADR2	3D51h	—	3D31h	—	3D11h	SPI1TXB
3DF0h	U1P3L	3DD0h	U2RXB	3DB0h	—	3D90h	—	3D70h	I2C1ADR1	3D50h	—	3D30h	—	3D10h	SPI1RXB
3DEFh	U1P2H	3DCFh	—	3DAFh	—	3D8Fh	—	3D6Fh	I2C1ADR0	3D4Fh	—	3D2Fh	—	3D0Fh	—
3DEEh	U1P2L	3DCEh	—	3DAEh	—	3D8Eh	—	3D6Eh	I2C1ADB1	3D4Eh	—	3D2Eh	—	3D0Eh	—
3DEDh	U1P1H	3DCDh	—	3DADh	—	3D8Dh	—	3D6Dh	I2C1ADB0	3D4Dh	—	3D2Dh	—	3D0Dh	—
3DECh	U1P1L	3DCCh	—	3DACh	—	3D8Ch	—	3D6Ch	I2C1CNT	3D4Ch	—	3D2Ch	—	3D0Ch	—
3DEBh	U1TXCHK	3DCBh	—	3DABh	—	3D8Bh	—	3D6Bh	I2C1TXB	3D4Bh	—	3D2Bh	—	3D0Bh	—
3DEAh	U1TXB	3DCAh	—	3DAAh	—	3D8Ah	—	3D6Ah	I2C1RXB	3D4Ah	—	3D2Ah	—	3D0Ah	—
3DE9h	U1RXCHK	3DC9h	—	3DA9h	—	3D89h	—	3D69h	—	3D49h	—	3D29h	—	3D09h	—
3DE8h	U1RXB	3DC8h	—	3DA8h	—	3D88h	—	3D68h	—	3D48h	—	3D28h	—	3D08h	—
3DE7h	—	3DC7h	—	3DA7h	—	3D87h	—	3D67h	—	3D47h	—	3D27h	—	3D07h	—
3DE6h	—	3DC6h	—	3DA6h	—	3D86h	—	3D66h	I2C2BTO	3D46h	—	3D26h	—	3D06h	—
3DE5h	—	3DC5h	—	3DA5h	—	3D85h	—	3D65h	I2C2CLK	3D45h	—	3D25h	—	3D05h	—
3DE4h	—	3DC4h	—	3DA4h	—	3D84h	—	3D64h	I2C2PIE	3D44h	—	3D24h	—	3D04h	—
3DE3h	—	3DC3h	—	3DA3h	—	3D83h	—	3D63h	I2C2PIR	3D43h	—	3D23h	—	3D03h	—
3DE2h	U2ERRIE	3DC2h	—	3DA2h	—	3D82h	—	3D62h	I2C2STAT1	3D42h	—	3D22h	—	3D02h	—
3DE1h	U2ERRIR	3DC1h	—	3DA1h	—	3D81h	—	3D61h	I2C2STAT0	3D41h	—	3D21h	—	3D01h	—
3DE0h	U2UIR	3DC0h	—	3DA0h	—	3D80h	—	3D60h	I2C2ERR	3D40h	—	3D20h	—	3D00h	—

Legend: Unimplemented data memory locations and registers, read as '0'.

Note 1: Unimplemented in LF devices.

2: Unimplemented in PIC18(L)F26/27K42.

3: Unimplemented in PIC18(L)F26/27/45/46/47K42.

REGISTER 9-20: PIE6: PERIPHERAL INTERRUPT ENABLE REGISTER 6

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR3GIE	TMR3IE	U2IE	U2EIE	U2TXIE	U2RXIE	I2C2EIE	I2C2IE
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 **TMR3GIE:** TMR3 Gate Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 6 **TMR3IE:** TMR3 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **U2IE:** UART2 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **U2EIE:** UART2 Framing Error Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **U2TXIE:** UART2 Transmit Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **U2RXIE:** UART2 Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **I2C2EIE:** I^2C2 Error Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **I2C2IE:** I^2C2 Interrupt Enable bit
1 = Enabled
0 = Disabled

REGISTER 9-24: PIE10: PERIPHERAL INTERRUPT ENABLE REGISTER 10

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
—	—	—	—	—	—	CLC4IE	CCP4IE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **CLC4IE:** CLC4 Interrupt Enable bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

bit 0 **CCP4IE:** CCP4 Interrupt Enable bit
- 1 = Interrupt has occurred (must be cleared by software)
- 0 = Interrupt event has not occurred

REGISTER 14-18: SCANTRIG: SCAN TRIGGER SELECTION REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	TSEL<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **TSEL<3:0>**: Scanner Data Trigger Input Selection bits

1111 = Reserved
- •
- •
- •

1010 = Reserved
1001 = SMT1_output
1000 = TMR6_postscaled
0111 = TMR5_output
0110 = TMR4_postscaled
0101 = TMR3_output
0100 = TMR2_postscaled
0011 = TMR1_output
0010 = TMR0_output
0001 = CLKREF_output
0000 = LFINTOSC

17.0 PERIPHERAL PIN SELECT (PPS) MODULE

The Peripheral Pin Select (PPS) module connects peripheral inputs and outputs to the device I/O pins. Only digital signals are included in the selections. All analog inputs and outputs remain fixed to their assigned pins. Input and output selections are independent as shown in the simplified block diagram Figure 17-1.

The peripheral input is selected with the peripheral xxxPPS register (Register 17-1), and the peripheral output is selected with the PORT RxyPPS register (Register 17-2). For example, to select PORTC<7> as the UART1 RX input, set U1RXPPS to 0b1 0111, and to select PORTC<6> as the UART1 TX output set RC6PPS to 0b01 0011.

17.1 PPS Inputs

Each peripheral has a PPS register with which the inputs to the peripheral are selected. Inputs include the device pins.

Multiple peripherals can operate from the same source simultaneously. Port reads always return the pin level regardless of peripheral PPS selection. If a pin also has analog functions associated, the ANSEL bit for that pin must be cleared to enable the digital input buffer.

Although every peripheral has its own PPS input selection register, the selections are identical for every peripheral as shown in Register 17-1.

Note:	The notation “xxx” in the register name is a place holder for the peripheral identifier. For example, INT0PPS.

17.2 PPS Outputs

Each I/O pin has a PPS register with which the pin output source is selected. With few exceptions, the port TRIS control associated with that pin retains control over the pin output driver. Peripherals that control the pin output driver as part of the peripheral operation will override the TRIS control as needed. These peripherals include:

- UART

Although every pin has its own PPS peripheral selection register, the selections are identical for every pin as shown in Register 17-2.

Note:	The notation “Rxy” is a place holder for the pin identifier. For example, RA0PPS.

REGISTER 22-5: TxCON: TIMERx CONTROL REGISTER

R/W/HC-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ON	CKPS<2:0>			OUTPS<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HC = Bit is cleared by hardware

bit 7 **ON:** Timerx On bit(1)
- 1 = Timerx is On
- 0 = Timerx is Off: all counters and state machines are reset

bit 6-4 **CKPS<2:0>:** Timerx-type Clock Prescale Select bits
- 111 = 1:128 Prescaler
- 110 = 1:64 Prescaler
- 101 = 1:32 Prescaler
- 100 = 1:16 Prescaler
- 011 = 1:8 Prescaler
- 010 = 1:4 Prescaler
- 001 = 1:2 Prescaler
- 000 = 1:1 Prescaler

bit 3-0 **OUTPS<3:0>:** Timerx Output Postscaler Select bits
- 1111 = 1:16 Postscaler
- 1110 = 1:15 Postscaler
- 1101 = 1:14 Postscaler
- 1100 = 1:13 Postscaler
- 1011 = 1:12 Postscaler
- 1010 = 1:11 Postscaler
- 1001 = 1:10 Postscaler
- 1000 = 1:9 Postscaler
- 0111 = 1:8 Postscaler
- 0110 = 1:7 Postscaler
- 0101 = 1:6 Postscaler
- 0100 = 1:5 Postscaler
- 0011 = 1:4 Postscaler
- 0010 = 1:3 Postscaler
- 0001 = 1:2 Postscaler
- 0000 = 1:1 Postscaler

Note 1: In certain modes, the ON bit will be auto-cleared by hardware. See **Section 22.1.2 "One-Shot Mode"**.

FIGURE 25-15: TIME OF FLIGHT MODE SINGLE ACQUISITION TIMING DIAGRAM

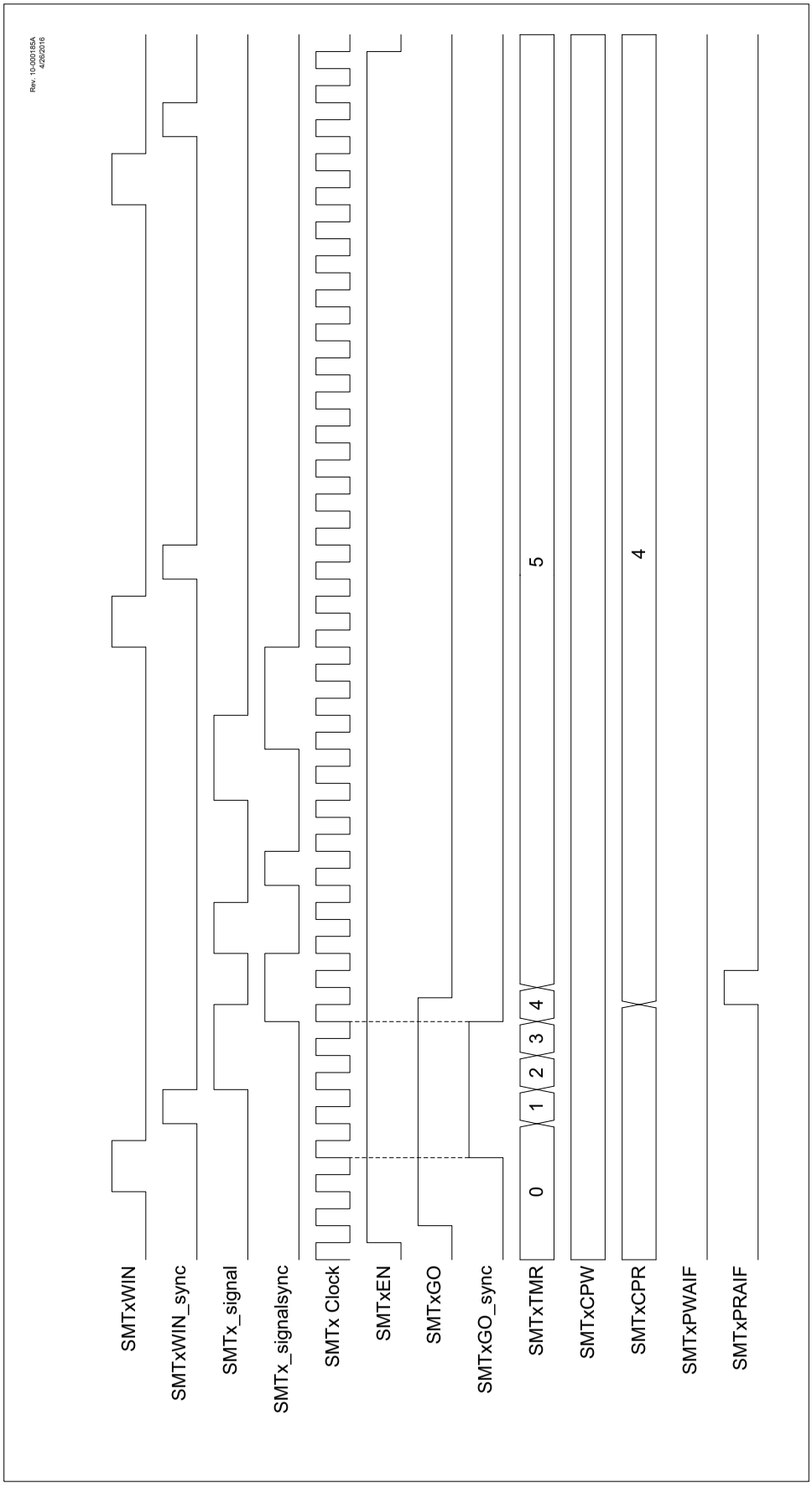


FIGURE 26-14: CWG SHUTDOWN BLOCK DIAGRAM

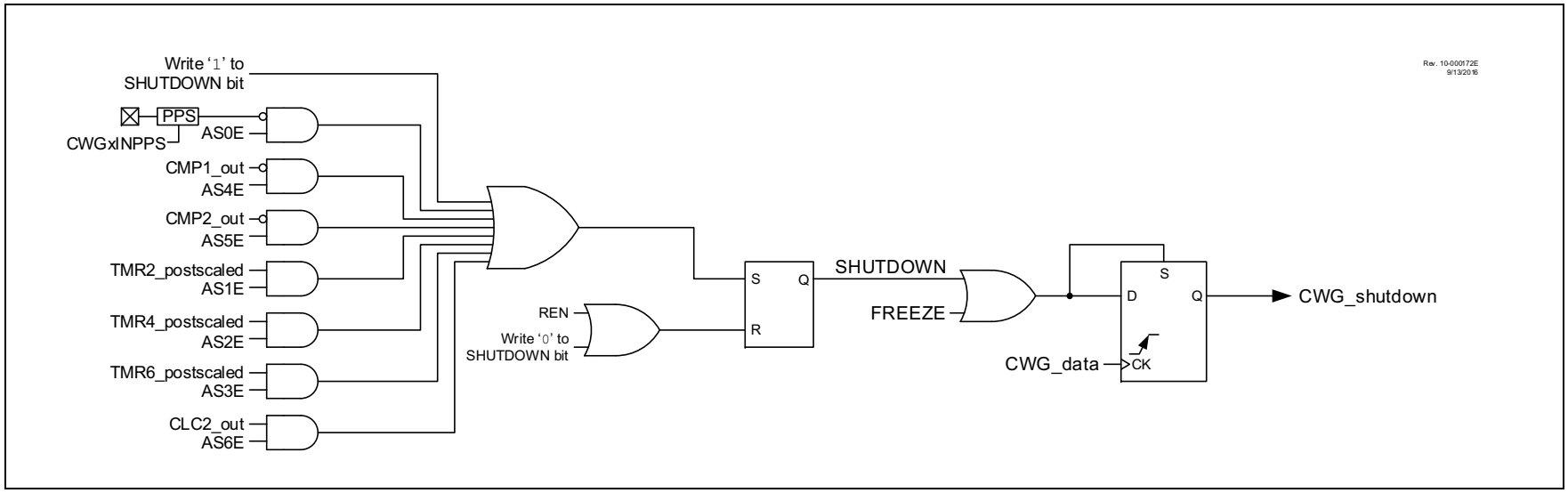
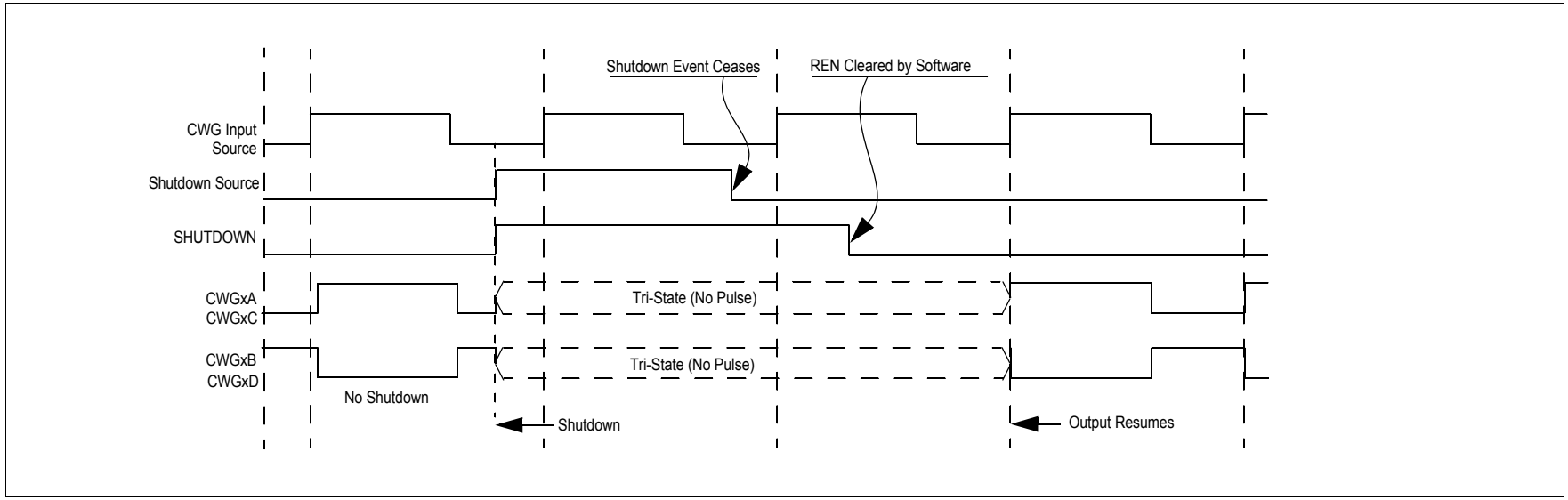


FIGURE 26-15: SHUTDOWN FUNCTIONALITY, AUTO-RESTART DISABLED (REN = 0, LSAC = 01, LSBD = 01)



32.7 SPI Operation in Sleep Mode

SPI master mode will operate in Sleep, provided the clock source selected by SPIxCLK is active in Sleep mode. FIFOs will operate as they would when the part is awake. When TXR = 1, the TXFIFO will need to contain data in order for transfers to take place in Sleep. All interrupts will still set the interrupt flags in Sleep but only enabled interrupts will wake the device from Sleep.

SPI Slave mode will operate in Sleep, because the clock is provided by an external master device. FIFOs will still operate and interrupts will set interrupt flags, and enabled interrupts will wake the device from Sleep.

32.8 SPI Interrupts

There are three top level SPI interrupts in the PIRx register:

- SPI Transmit
- SPI Receive
- SPI Module status

The status interrupts are enabled at the module level in the SPIxINTE register. Only enabled status interrupts will cause the single top level SPIxIF flag to be set.

32.8.1 SPI RECEIVER DATA INTERRUPT

The SPI Receiver Data Interrupt is set when RXFIFO contains data, and is cleared when the RXFIFO is empty. The interrupt flag SPI1RXIF is located in PIRx and the interrupt enable SPI1RXIE is located in PIEx. This interrupt flag is read-only.

32.8.2 SPI TRANSMITTER DATA INTERRUPT

The SPI Transmitter Data Interrupt is set when TXFIFO is not full, and is cleared when the TXFIFO is full. The interrupt flag SPI1TXIF is located in PIRx and the interrupt enable SPI1TXIE is located in PIEx. The interrupt flag is read-only.

32.8.3 SPI MODULE STATUS INTERRUPTS

The SPIxIF flag in the respective PIR register is set when any of the individual status flags in SPIxINTF and their respective SPIxINTE bits are set. In order for the setting of any specific interrupt flag to interrupt normal program flow both the SPIxIE bit as well as the specific bit in SPIxINTE associated with that interrupt must be set.

The Status Interrupts are:

- Shift Register Empty Interrupt
- Transfer Counter is Zero Interrupt
- Start of Slave Select Interrupt
- End of Slave Select Interrupt
- Receiver Overflow Interrupt
- Transmitter Underflow Interrupt

REGISTER 32-6: SPIxBAUD: SPI BAUD RATE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'

bit 7-0 BAUD<7:0>: Baud Clock Prescaler Select bits

SCK high or low time: TSC=SPI Clock Period*(BAUD+1)

SCK toggle frequency: FSCK=FBAUD= SPI Clock Frequency/(2*(BAUD+1))

Note: This register should not be written while the SPI is enabled (EN bit of SPIxCON0 = 1)

REGISTER 32-7: SPIxCON0: SPI CONFIGURATION REGISTER 0

R/W-0/0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
EN	—	—	—	—	LSBF	MST	BMODE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'

bit 7 **EN**: SPI Module Enable Control bit

1 =SPI is enabled

0 = SPI is disabled,

bit 6-3 **Unimplemented**: Read as '0'

bit 2 **LSBF**: LSb-First Data Exchange bit

1 = Data is exchanged LSb first

0 = Data is exchanged MSb first (traditional SPI operation)

bit 1 **MST**: SPI Operating Mode Master Select bit

1 = SPI module operates as the bus master

0 = SPI module operates as a bus slave

bit 0 **BMODE**: Bit-Length Mode Select bit

1 = SPIxTWIDTH setting applies to every byte: total bits sent is SPIxTWIDTH*SPIxTCNT, end-of-packet occurs when SPIxTCNT = 0

0 = SPIxTWIDTH setting applies only to the last byte exchanged; total bits sent is SPIxTWIDTH + (SPIxTCNT*8)

Note: This register should only be written when the EN bit is cleared, or to clear the EN bit.

33.4.3 SLAVE OPERATION IN 7-BIT ADDRESSING MODE

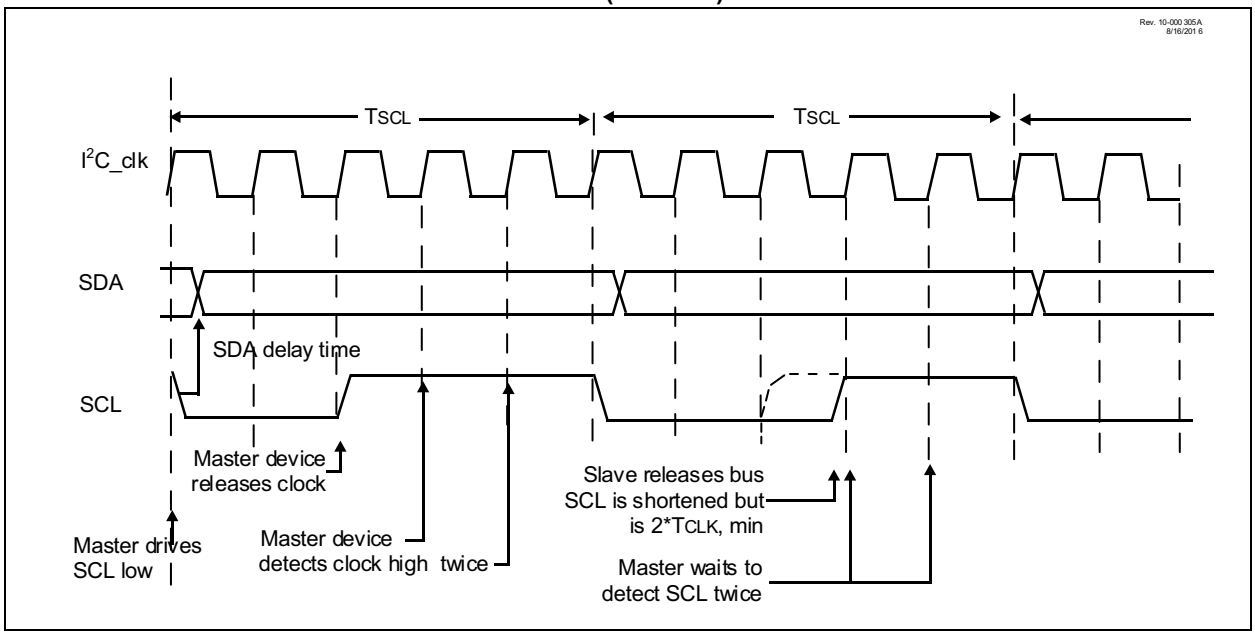
The 8th bit in an address byte transmitted by the master is used to determine if the Master wants to read from or write to the Slave device. If set, it denotes that the Master wants to read from the slave and if cleared it means the master wants to write to the slave device. If there is an address match, the R/W bit is copied to the R/W bit of the I2CxSTAT0 register.

33.4.3.1 Slave Reception (7-bit Addressing Mode)

This section describes the sequence of events for the I^2C module configured as an I^2C slave in 7-bit Addressing mode and is receiving data. Figure 33-6, Figure 33-7, and Figure 33-8 are used as a visual reference for this description.

1. Master asserts Start condition (can also be a restart) on the bus. Start condition Interrupt Flag (SCIF) in I2CxPIR register is set.
2. If Start condition interrupt is enabled (SCIE bit is set), generic interrupt I2CxIF is set.
3. Master transmits eight bits – 7-bit address and R/W = 0.
4. Received address is compared with the values in I2CxADR0/I2CxADR1/I2CxADR2/I2CxADR3 registers. Refer to section **Section 33.4.1 "Slave Addressing Modes"** for slave addressing modes.
5. If address matches; SMA in I2CxSTAT0 register is set, R/W is copied to R/W bit, D/A bit is cleared. If the address does not match; module becomes idle.
6. The matched address data is loaded into I2CxADB0 and ADRIF in I2CxPIR register is set.
7. If Address hold interrupt is enabled (ADRIE = 1), CSTR is set. I2CxIF is set. Slave software can read address from I2CxADB0 and set/clear ACKDT before releasing SCL.
8. If there are any previous error conditions, e.g., Receive buffer overflow or transmit buffer underflow errors, Slave will force a NACK and the module becomes idle.
9. ACKDT value is copied out to SDA for ACK pulse to be read by the Master on the 9th SCL pulse.
10. If the Acknowledge interrupt and hold is enabled (ACKTIE = 1), CSTR is set, I2CxIF is set, then Slave software can read address from I2CxADB0 register and change the value of ACKDT before releasing SCL by clearing CSTR.
11. Master sends first seven SCL pulses of the data byte or a Stop condition (in the case of NACK).
12. If Stop condition; PCIF in I2CxPIR register is set, module becomes idle.
13. If the receive buffer is full from the previous transaction i.e. RXBF = 1 (I2CxRXIF = 1), CSTR is set. Slave software must read data out of I2CxRXB to resume communication.
14. Master sends 8th SCL pulse of the data byte. D/A bit is set, WRIF is set.
15. I2CxRXB is loaded with new data, RXBF bit is set, I2CxRXIF is set.
16. If Data write interrupt and hold is enabled (WRIE = 1), CSTR is set, I2CxIF is set. Slave software can read data from I2CxRXB and set/clear ACKDT before releasing SCL by clearing CSTR.
17. If I2CxCNT = 0, the ACKCNT value is output to the SDA; else, if I2CxCNT!= 0, the ACKDT value is used and the value of I2CxCNT is decremented.
18. The ACK value is copied out to SDA to be read by the Master on the 9th SCL pulse.
19. If I2CxCNT = 0, CNTIF is set.
20. If a NACK was sent, NACKIF is set, module becomes idle.
21. If ACKTIE = 1, CSTR is set, I2CxIF is set. Slave software can read data from I2CxRXB clearing RXBF, before releasing SCL by clearing CSTR.
22. Go to step 11.

FIGURE 33-13: CLOCK SYNTHESIS TIMING (FME = 0)

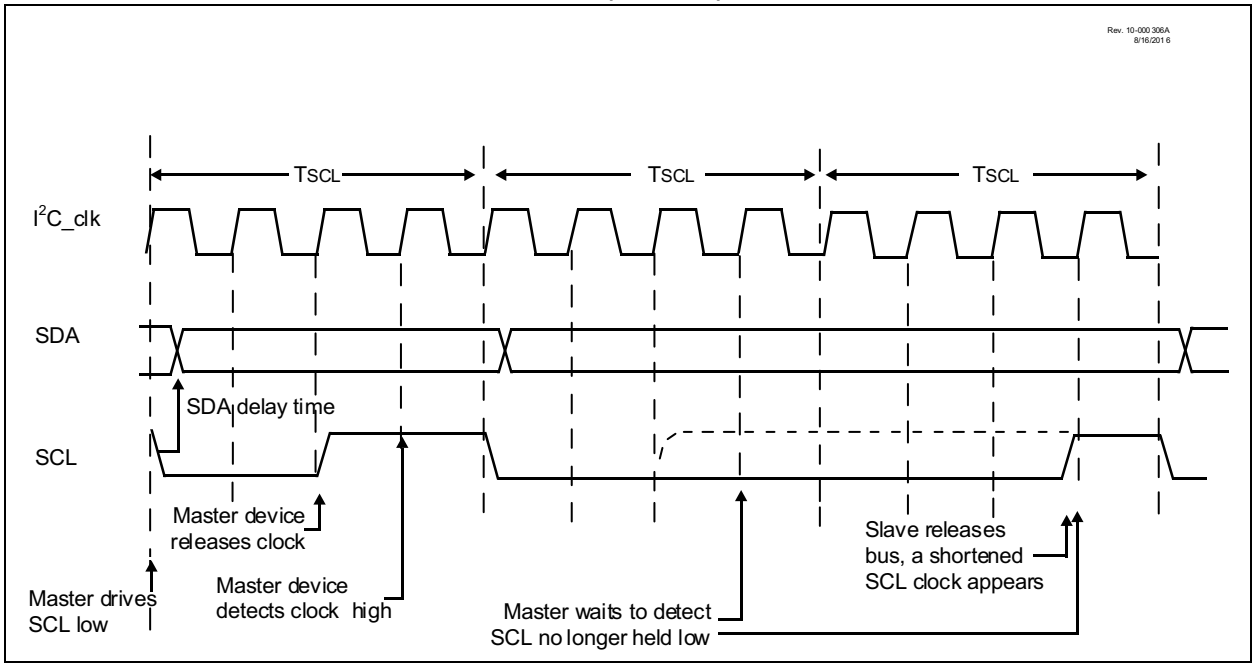


33.5.4.2 Clock Timing with FME = 1

One TSCL, consists of four clocks of the I2C clock input. The first clock is used to drive SCL low, the third releases SCL high, and the fourth is used to detect if the clock is, in fact, high or being stretched by a slave.

If a slave is clock stretching, the hardware waits; checking SCL on each successive I2C clock, proceeding only after detecting SCL high. Figure 33-14 shows the clock synthesis timing when FME = 1.

FIGURE 33-14: CLOCK SYNTHESIS TIMING (FME = 1)



33.5.12 MASTER RECEPTION IN 10-BIT ADDRESSING MODE

This section describes the sequence of events for the I^2C module configured as an I^2C master in 10-bit Addressing mode and is receiving data. Figure 33-22 is used as a visual reference for this description.

1. Master software loads high address byte in I2CxADB1 and low address byte in I2CxADB0 for write and sets restart enable (RSTEN) bit.
2. Master software sets START bit.
3. Master hardware waits for BFRE bit to be set; then shifts out start, high address and waits for acknowledge.
4. If slave responds with a NACK, master hardware sends Stop and ends communication.
5. If slave responds with ACK, master hardware shifts out the low address.
6. If the transmit buffer empty flag (TXBE) is set and I2CxCNT! = 0, the clock is stretched on 8th falling SCL edge. Allowing master software writes next data to I2CxTXB.
7. Master hardware sends 9th SCL pulse for ACK from slave and loads the shift register from I2CxTXB.
8. If slave responds with a NACK, master hardware sends Stop and ends communication.
9. If slave responds with an ACK and I2CxCNT = 0, master hardware sets MDR bit, go to Step 11.
10. If slave responds with an ACK and I2CxCNT! = 0, master hardware outputs data in shift register on SDA and waits for ACK from slave. Go to step 4.
11. Master software loads I2CxADB0 for read, and I2CCNT with the number of bytes to be received in the current transaction.
12. Master software sets Start bit.
13. Master hardware shifts out Restart and high address with R/W = 1.
14. Master sends out 9th SCL pulse for ACK from Slave.
15. If slave responds with a NACK, master hardware sends Stop or sets MDR (RSEN bit).
16. If slave responds with an ACK, master hardware shifts 7 bits of data into the shift register from the slave.
17. If the receive buffer full flag (RXBF) is set, clock is stretched on seventh falling SCL edge.
18. Master software can clear clock stretching by reading the previous data in the receive buffer.
19. Master hardware shifts 8th bit of data into the shift register from slave and loads it into I2CxRXB.
20. Master software reads data from I2CxRXB register.
21. If I2CxCNT! = 0, master hardware clocks out ACKDT as ACK value to slave.
22. If I2CxCNT = 0, master hardware clocks out ACKCNT as ACK value to slave
23. Go to step 4.

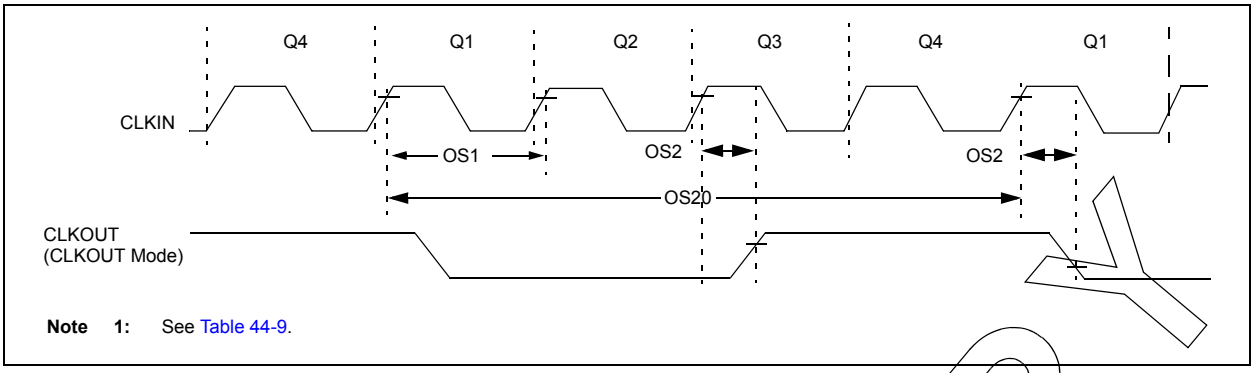
TABLE 42-1: REGISTER FILE SUMMARY FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3DF0h	U1P3L	P3L								509
3DEFh	U1P2H	—	—	—	—	—	—	—	P2H	508
3DEEh	U1P2L	P2L								508
3DEDh	U1P1H	—	—	—	—	—	—	—	P1H	507
3DECh	U1P1L	P1L								507
3DEBh	U1TXCHK	TXCHK								510
3DEAh	U1TXB	TXB								506
3DE9h	U1RXCHK	RXCHK								510
3DE8h	U1RXB	RXB								506
3DE7h - 3DE3h	—	Unimplemented								
3DE2h	U2ERRIE	TXMTIE	PERIE	ABDOVE	CERIE	FERIE	RXBKIE	RXFOIE	TXCIE	502
3DE1h	U2ERRIR	TXMTIF	PERIF	ABDOVF	CERIF	FERIF	RXBKIF	RXFOIF	TXCIF	501
3DE0h	U2UIR	WUIF	ABDIF	—	—	—	ABDIE	—	—	503
3DDFh	U2FIFO	TXWRE	STPMD	TXBE	TXBF	RXIDL	XON	RXBE	RXBF	504
3DDEh	U2BRGH	BRGH								505
3DDDh	U2BRGL	BRGL								505
3DDCh	U2CON2	RUNOVF	RXPOL	STP		—	TXPOL	FLO		500
3DDBh	U2CON1	ON	—	—	WUE	RXBIMD	—	BRKOVR	SENDB	499
3DDAh	U2CON0	BRGS	ABDEN	TXEN	RXEN	MODE				498
3DD9h	—	Unimplemented								
3DD8h	U2P3L	P3L								508
3DD7h	—	Unimplemented								
3DD6h	U2P2L	P2L								508
3DD5h	—	Unimplemented								
3DD4h	U2P1L	P1L								507
3DD3h	—	Unimplemented								
3DD2h	U2TXB	TXB								506
3DD1h	—	Unimplemented								
3DD0h	U2RXB	RXB								506
3DCFh - 3D7Dh	—	Unimplemented								
3D7Ch	I2C1BTO	BTO								582
3D7Bh	I2C1CLK	CLK								581
3D7Ah	I2C1PIE	CNTIE	ACKTIE	—	WRIE	ADRIE	PCIE	RSCIE	SCIE	588
3D79h	I2C1PIR	CNTIF	ACKTIF	—	WRIF	ADRIF	PCIF	RSCIF	SCIF	587
3D78h	I2C1STAT1	TXWE	—	TXBE	—	RXRE	CLRBF	—	RXBF	584
3D77h	I2C1STAT0	BFRE	SMA	MMA	R	D	—	—	—	583
3D76h	I2C1ERR	—	BTOIF	BCLIF	NACKIF	—	BTOIE	BCLIE	NACKIE	585
3D75h	I2C1CON2	ACNT	GCEN	FME	ABD	SDAHT		BFRET		580
3D74h	I2C1CON1	ACKCNT	ACKDT	ACKSTAT	ACKT	—	RXO	TXU	CSD	579
3D73h	I2C1CON0	EN	RSEN	S	CSTR	MDR	MODE			577
3D72h	I2C1ADR3	ADR							—	592
3D71h	I2C1ADR2	ADR								591
3D70h	I2C1ADR1	ADR							—	590
3D6Fh	I2C1ADR0	ADR								589
3D6Eh	I2C1ADB1	ADB								594
3D6Dh	I2C1ADB0	ADB								593

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition

Note 1: Unimplemented in LF devices.
2: Unimplemented in PIC18(L)F26/27K42.
3: Unimplemented on PIC18(L)F26/27/45/46/47K42 devices.
4: Unimplemented in PIC18(L)F45/55K42.

FIGURE 44-5: CLOCK TIMING



Note 1: See Table 44-9.

TABLE 44-9: EXTERNAL CLOCK/OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
ECL Oscillator							
OS1	F_{ECL}	Clock Frequency	—	—	500	kHz	
OS2	T_{ECL_DC}	Clock Duty Cycle	40	—	60	%	
ECM Oscillator							
OS3	F_{ECM}	Clock Frequency	—	—	8	MHz	
OS4	T_{ECM_DC}	Clock Duty Cycle	40	—	60	%	
ECH Oscillator							
OS5	F_{ECH}	Clock Frequency	—	—	64	MHz	
OS6	T_{ECH_DC}	Clock Duty Cycle	40	—	60	%	
LP Oscillator							
OS7	F_{LP}	Clock Frequency	—	—	100	kHz	**Note 4**
XT Oscillator							
OS8	F_{XT}	Clock Frequency	—	—	4	MHz	**Note 4**
HS Oscillator							
OS9	F_{HS}	Clock Frequency	—	—	20	MHz	**Note 4**
Secondary Oscillator							
OS10	F_{SEC}	Clock Frequency	32.4	32.768	33.1	kHz	
System Oscillator							
OS20	F_{OSC}	System Clock Frequency	—	—	64	MHz	**(Note 2, Note 3)**

* These parameters are characterized but not tested.

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

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

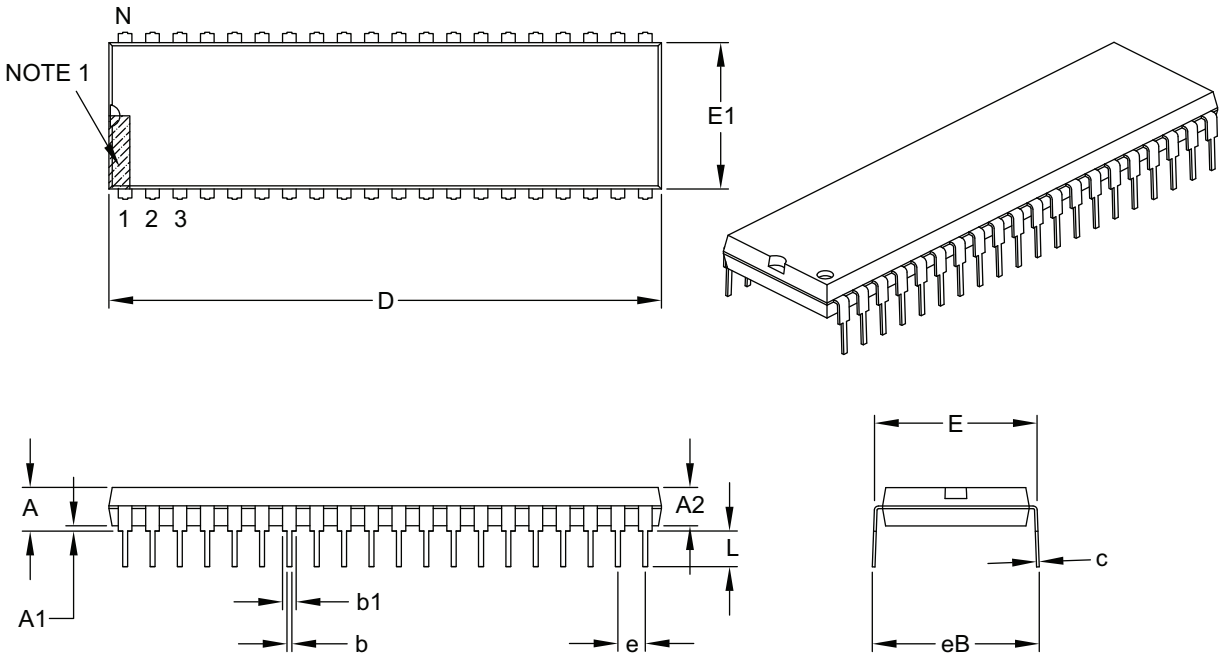
2: The system clock frequency (FOSC) is selected by the "main clock switch controls" as described in **Section 10.0 "Power-Saving Operation Modes"**.

3: The system clock frequency (FOSC) must meet the voltage requirements defined in the **Section 44.2 "Standard Operating Conditions"**.

4: LP, XT and HS oscillator modes require an appropriate crystal or resonator to be connected to the device. For clocking the device with the external square wave, one of the EC mode selections must be used.

40-Lead Plastic Dual In-Line (P) – 600 mil Body [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		40	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.250
Molded Package Thickness	A2	.125	–	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.590	–	.625
Molded Package Width	E1	.485	–	.580
Overall Length	D	1.980	–	2.095
Tip to Seating Plane	L	.115	–	.200
Lead Thickness	c	.008	–	.015
Upper Lead Width	b1	.030	–	.070
Lower Lead Width	b	.014	–	.023
Overall Row Spacing §	eB	–	–	.700

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-016B

PRODUCT IDENTIFICATION SYSTEM

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

PART NO.	[X](2)	-	X	/XX	XXX
Device	Tape and Reel Option		Temperature Range	Package	Pattern

Device:	PIC18F26K42, PIC18LF26K42, PIC18F27K42, PIC18LF27K42 PIC18F45K42, PIC18LF45K42 PIC18F46K42, PIC18LF46K42 PIC18F47K42, PIC18LF47K42 PIC18F55K42, PIC18LF55K42 PIC18F56K42, PIC18LF56K42 PIC18F57K42, PIC18LF57K42
Tape and Reel Option:	Blank = standard packaging (tube or tray) T = Tape and Reel(1), (2)
Temperature Range:	E = -40°C to +125°C (Extended) I = -40°C to +85°C (Industrial)
Package:	ML = 28-lead QFN 6x6mm ML = 44-lead QFN 8x8x0.9mm MX = 28-lead UQFN 6x6x0.5mm MV = 40-lead UQFN 5x5x0.5mm MV = 48-lead UQFN P = 40-lead PDIP PT = 44-lead TQFP (Thin Quad Flatpack) PT = 48-lead TQFP SO = 28-lead SOIC SP = 28-lead Skinny Plastic DIP SS = 28-lead SSOP
Pattern:	QTP, SQTP, Code or Special Requirements (blank otherwise)

Examples:

a) PIC18F26K42-E/P 301 = Extended temp., PDIP package, QTP pattern #301.

b) PIC18F45K42-E/SO = Extended temp., SOIC package.

c) PIC18F46K42T-I/ML = Tape and reel, Industrial temp., QFN package.

Note 1: Tape and Reel option is available for ML, MV, PT, SO and SS packages with industrial Temperature Range only.

2: Tape and Reel identifier only appears in catalog part number description. This identifier is used for ordering purposes and is not printed on the device package.