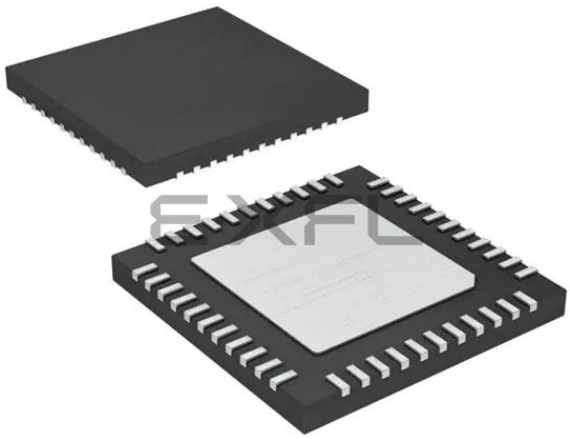


Welcome to E-XFL.COM

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, DMA, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	128KB (64K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 35x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f47k42-i-ml

Analog Peripherals

- Analog-to-Digital Converter with Computation (ADC^2):
  - 12-bit with up to 35 external channels
  - Automated post-processing
  - Automated math functions on input signals: averaging, filter calculations, oversampling and threshold comparison
  - Operates in Sleep
  - Integrated charge pump for improved low-voltage operation
- Hardware Capacitive Voltage Divider (CVD):
  - Automates touch sampling and reduces software size and CPU usage when touch or proximity sensing is required
  - Adjustable sample and hold capacitor array
  - Two guard ring output drives
- Temperature Sensor
  - Internal connection to ADC
  - Can be calibrated for improved accuracy
- Two Comparators:
  - Low-Power/High-Speed mode
  - Fixed Voltage Reference at noninverting input(s)
  - Comparator outputs externally accessible
- 5-bit Digital-to-Analog Converter (DAC):
  - 5-bit resolution, rail-to-rail
  - Positive Reference Selection
  - Unbuffered I/O pin output
  - Internal connections to ADCs and comparators
- Voltage Reference
  - Fixed Voltage Reference with 1.024V, 2.048V and 4.096V output levels

Flexible Oscillator Structure

- High-Precision Internal Oscillator
  - Selectable frequency range up to 64 MHz
  - ±1% at calibration (nominal)
- Low-Power Internal 32 kHz Oscillator (LFINTOSC)
- External 32 kHz Crystal Oscillator (SOSC)
- External Oscillator Block with:
  - x4 PLL with external sources
  - Three crystal/resonator modes up to 20 MHz
  - Three external clock modes up to 20 MHz
- Fail-Safe Clock Monitor
- Oscillator Start-up Timer (OST)
  - Ensures stability of crystal oscillator sources

TABLE 1: 28-PIN ALLOCATION TABLE (PIC18(L)F2XK42) (CONTINUED)

I/O	28-Pin SPDIP/SOIC/SSOP	28-Pin (U)QFN	ADC	Voltage Reference	DAC	Comparators	Zero Cross Detect	I^2C	SPI	UART	DSM	Timers/SMT	CCP and PWM	CWG	CLC	NCO	Clock Reference (CLKR)	Interrupt-on-Change	Basic
RC0	11	8	ANC0	—	—	—	—	—	—	—	—	T1CKI(1) T3CKI(1) T3G(1) SMTWIN1(1)	—	—	—	—	—	IOCC0	SOSCO
RC1	12	9	ANC1	—	—	—	—	—	—	—	—	SMTSIG1(1)	CCP2(1)	—	—	—	—	IOCC1	SOSCI
RC2	13	10	ANC2	—	—	—	—	—	—	—	—	T5CKI(1)	CCP1(1)	—	—	—	—	IOCC2	—
RC3	14	11	ANC3	—	—	—	—	SCL1(3,4)	SCK1(1)	—	—	T2IN(1)	—	—	—	—	—	IOCC3	—
RC4	15	12	ANC4	—	—	—	—	SDA1(3,4)	SDI1(1)	—	—	—	—	—	—	—	—	IOCC4	—
RC5	16	13	ANC5	—	—	—	—	—	—	—	—	T4IN(1)	—	—	—	—	—	IOCC5	—
RC6	17	14	ANC6	—	—	—	—	—	—	CTS1(1)	—	—	—	—	—	—	—	IOCC6	—
RC7	18	15	ANC7	—	—	—	—	—	—	RX1(1)	—	—	—	—	—	—	—	IOCC7	—
RE3	1	26	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IOCE3	$\overline{MCLR}$ V_{PP}
V_{DD}	20	17	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
V_{SS}	8, 19	5, 16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
OUT(2)	—	—	ADGRDA ADGRDB	—	—	C1OUT C2OUT	—	SDA1 SCL1 SDA2 SCL2	SS1 SCK1 SDO1	DTR1 RTS1 TX1 DTR2 RTS2 TX2	DSM	TMR0	CCP1 CCP2 CCP3 CCP4 PWM5OUT PWM6OUT PWM7OUT PWM8OUT	CWG1A CWG1B CWG1C CWG1D CWG2A CWG2B CWG2C CWG2D CWG3A CWG3B CWG3C CWG3D	CLC1OUT CLC2OUT CLC3OUT CLC4OUT	NCO	CLKR	—	—

Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins.

2: All output signals shown in this row are PPS remappable.

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 can be configured for I^2C and SMB™ 3.0/2.0 logic levels; The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) 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.

TABLE 3: 48-PIN ALLOCATION TABLE FOR PIC18(L)F5XK42 (CONTINUED)

I/O	48-Pin TQFP	48-Pin UQFN	ADC	Voltage Reference	DAC	Comparators	Zero Cross Detect	I^2C	SPI	UART	DSM	Timers/SMT	CCP and PWM	CWG	CLC	NCO	Clock Reference (CLKR)	Interrupt-on-Change	Basic
VDD	7, 30	7, 30	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VSS	6, 31	6, 31	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
OUT(2)	—	—	ADGRDA ADGRDB	—	—	C1OUT C2OUT	—	SDA1 SCL1 SDA2 SCL2	SS1 SCK1 SDO1	DTR1 RTS1 TX1 DTR2 RTS2 TX2	DSM	TMR0	CCP1 CCP2 CCP3 CCP4 PWM5OUT PWM6OUT PWM7OUT PWM8OUT	CWG1A CWG1B CWG1C CWG1D CWG2A CWG2B CWG2C CWG2D CWG3A CWG3B CWG3C CWG3D	CLC1OUT CLC2OUT CLC3OUT CLC4OUT	NCO	CLKR	—	—

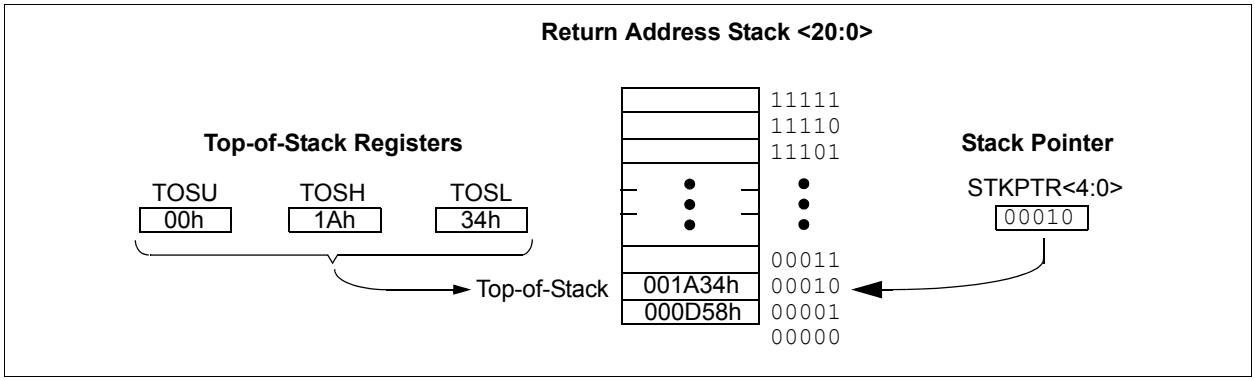
Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins.

2: All output signals shown in this row are PPS remappable.

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 can be configured for I^2C and SMB™ 3.0/2.0 logic levels; The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4/RD0/RD1 pins. PPS assignments to the other pins (e.g., RA5) 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.

FIGURE 4-1: RETURN ADDRESS STACK AND ASSOCIATED REGISTERS



4.2.5.2 Return Stack Pointer (STKPTR)

The STKPTR register (Register 4-4) contains the Stack Pointer value. The STKOVF (Stack Overflow) Status bit and the STKUNF (Stack Underflow) Status bit can be accessed using the PCON0 register. The value of the Stack Pointer can be 0 through 31. On Reset, the Stack Pointer value will be zero. The user may read and write the Stack Pointer value. This feature can be used by a Real-Time Operating System (RTOS) for stack maintenance. After the PC is pushed onto the stack 32 times (without popping any values off the stack), the STKOVF bit is set. The STKOVF bit is cleared by software or by a POR. The action that takes place when the stack becomes full depends on the state of the STVREN (Stack Overflow Reset Enable) Configuration bit. (Refer to **Section 5.1 "Configuration Words"** for a description of the device Configuration bits.)

If STVREN is set (default), a Reset will be generated and a Stack Overflow will be indicated by the STKOVF bit when the 32nd push is initiated. This includes `CALL` and `CALLW` instructions, as well as stacking the return address during an interrupt response. The STKOVF bit will remain set and the Stack Pointer will be set to zero.

If STVREN is cleared, the STKOVF bit will be set on the 32nd push and the Stack Pointer will remain at 31 but no Reset will occur. Any additional pushes will overwrite the 31st push but the STKPTR will remain at 31.

Setting STKOVF = `1` in software will change the bit, but will not generate a Reset.

The STKUNF bit is set when a stack pop returns a value of zero. The STKUNF bit is cleared by software or by POR. The action that takes place when the stack becomes full depends on the state of the STVREN (Stack Overflow Reset Enable) Configuration bit. (Refer to **Section 5.1 "Configuration Words"** for a description of the device Configuration bits).

If STVREN is set (default) and the stack has been popped enough times to unload the stack, the next pop will return a value of zero to the PC, it will set the STKUNF bit and a Reset will be generated. This condition can be generated by the `RETURN`, `RETLW` and `RETFIE` instructions.

When STVREN = `0`, STKUNF will be set but no Reset will occur.

Note:	Returning a value of zero to the PC on an underflow has the effect of vectoring the program to the Reset vector, where the stack conditions can be verified and appropriate actions can be taken. This is not the same as a Reset, as the contents of the SFRs are not affected.

4.2.5.3 `PUSH` and `POP` Instructions

Since the Top-of-Stack is readable and writable, the ability to push values onto the stack and pull values off the stack without disturbing normal program execution is a desirable feature. The PIC18 instruction set includes two instructions, `PUSH` and `POP`, that permit the TOS to be manipulated under software control. TOSU, TOSH and TOSL can be modified to place data or a return address on the stack.

The `PUSH` instruction places the current PC value onto the stack. This increments the Stack Pointer and loads the current PC value onto the stack.

The `POP` instruction discards the current TOS by decrementing the Stack Pointer. The previous value pushed onto the stack then becomes the TOS value.

REGISTER 5-3: CONFIGURATION WORD 2L (30 0002h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
BOREN<1:0>		$\overline{\text{LPBOREN}}$	IVT1WAY	MVECEN	PWRTS<1:0>		MCLRE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Value for blank device	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **BOREN<1:0>:** Brown-out Reset Enable bits

When enabled, Brown-out Reset Voltage (V_{BOR}) is set by the BORV bit.
11 = Brown-out Reset is enabled, SBOREN bit is ignored
10 = Brown-out Reset is enabled while running, disabled in Sleep; SBOREN is ignored
01 = Brown-out Reset is enabled according to SBOREN
00 = Brown-out Reset is disabled

bit 5 **$\overline{\text{LPBOREN}}$:** Low-Power BOR Enable bit

1 = Low-Power BOR is disabled
0 = Low-Power BOR is enabled

bit 4 **IVT1WAY:** IVTLOCK bit One-Way Set Enable bit

1 = IVTLOCKED bit can be cleared and set only once; IVT registers remain locked after one clear/set cycle
0 = IVTLOCK ED bit can be set and cleared multiple times (subject to the unlock sequence)

bit 3 **MVECEN:** Multi-vector Enable bit

1 = Multi-vector enabled; Vector table used for interrupts
0 = Legacy interrupt behavior

bit 2-1 **PWRTS<1:0>:** Power-up Timer Selection bits

11 = PWRT is disabled
10 = PWRT set at 64 ms (2048 LFINTOSC Cycles)
01 = PWRT set at 16 ms (512 LFINTOSC Cycles)
00 = PWRT set at 1 ms (32 LFINTOSC Cycles)

bit 0 **MCLRE:** Master Clear ($\overline{\text{MCLR}}$) Enable bit

If LVP = 1:
RE3 pin function is $\overline{\text{MCLR}}$
If LVP = 0:
1 = $\overline{\text{MCLR}}$ pin is $\overline{\text{MCLR}}$
0 = $\overline{\text{MCLR}}$ pin function is a port defined function

REGISTER 5-9: CONFIGURATION WORD 5L (30 0008h)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	R/W-1
—	—	—	—	—	—	—	$\overline{CP}$
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Value for blank device	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-1 **Unimplemented:** Read as '1'

bit 0 **$\overline{CP}$:** User Program Flash Memory and Data EEPROM Code Protection bit
1 = User Program Flash Memory and Data EEPROM code protection is disabled
0 = User Program Flash Memory and Data EEPROM code protection is enabled

REGISTER 5-10: CONFIGURATION WORD 5H (30 0009h)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '1'	
-n = Value for blank device	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **Unimplemented:** Read as '1'

TABLE 5-2: SUMMARY OF CONFIGURATION WORDS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default/ Unprogrammed Value
30 0000h	CONFIG1L	—	RSTOSC<2:0>			—	FEXTOSC<2:0>			1111 1111
30 0001h	CONFIG1H	—	—	FCMEN	—	CSWEN	—	PR1WAY	$\overline{CLKOUTEN}$	1111 1111
30 0002h	CONFIG2L	BOREN<1:0>		$\overline{LPBOREN}$	IVT1WAY	MVECEN	PWRTS<1:0>		MCLRE	1111 1111
30 0003h	CONFIG2H	$\overline{XINST}$	—	$\overline{DEBUG}$	STVREN	PPS1WAY	$\overline{ZCD}$	BORV<1:0>		1111 1111
30 0004h	CONFIG3L	—	WDTE<1:0>		WDTCPS<4:0>					1111 1111
30 0005h	CONFIG3H	—	—	WDTCCS<2:0>			WDTCWS<2:0>			1111 1111
30 0006h	CONFIG4L	$\overline{WRTAPP}$	—	—	$\overline{SAFEN}$	$\overline{BBEN}$	BBSIZE<2:0>			1111 1111
30 0007h	CONFIG4H	—	—	LVP	—	$\overline{WRTSAF}$	$\overline{WRTD}$	$\overline{WRTC}$	$\overline{WRTB}$	1111 1111
30 0008h	CONFIG5L	—	—	—	—	—	—	—	$\overline{CP}$	1111 1111
30 0009h	CONFIG5H	—	—	—	—	—	—	—	—	1111 1111

9.5 Context Saving

The Interrupt controller supports a two-level deep context saving (Main routine context and Low ISR context). Refer to state machine shown in Figure 9-6 for details.

The Program Counter (PC) is saved on the dedicated device PC stack. CPU registers saved include STATUS, WREG, BSR, FSR0/1/2, PRODL/H and PCLATH/U.

After WREG has been saved to the context registers, the resolved vector number of the interrupt source to be serviced is copied into WREG. Context save and restore operation is completed by the interrupt controller based on current state of the interrupts and the order in which they were sent to the CPU.

Context save/restore works the same way in both states of MVECEN. When IPEN = 0, there is only one level interrupt active. Hence, only the main context is saved when an interrupt is received.

9.5.1 ACCESSING SHADOW REGISTERS

The Interrupt controller automatically saves the context information in the shadow registers available in Bank 56. Both the saved context values (i.e., main routine and low ISR) can be accessed using the same set of shadow registers. By clearing the SHADLO bit in the SHADCON register (Register 9-43), the CPU register values saved for main routine context can accessed, and by setting the SHADLO bit of the CPU register, values saved for low ISR context can accessed. Low ISR context is automatically restored to the CPU registers upon exiting the high ISR. Similarly, the main context is automatically restored to the CPU registers upon exiting the low ISR.

The Shadow registers in Bank 56 are readable and writable, so if the user desires to modify the context, then the corresponding shadow register should be modified and the value will be restored when exiting the ISR. Depending on the user's application, other registers may also need to be saved.

TABLE 18-1: IOC REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOCAP	IOCAP7	IOCAP6	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
IOCAN	IOCAN7	IOCAN6	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
IOCAF	IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0
IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0
IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0
IOCCP	IOCCP7	IOCCP6	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0
IOCCN	IOCCN7	IOCCN6	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0
IOCCF	IOCCF7	IOCCF6	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0
IOCEP	—	—	—	—	IOCEP3(1)	—	—	—
IOCEN	—	—	—	—	IOCEN3(1)	—	—	—
IOCEF	—	—	—	—	IOCEF3(1)	—	—	—

Note 1: If MCLRE = 1 or LVP = 1, RE3 port functionality is disabled and IOC on RE3 is not available.

TABLE 18-2: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPT-ON-CHANGE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
IOCxF	IOCxF7	IOCxF6	IOCxF5	IOCxF4	IOCxF3	IOCxF2	IOCxF1	IOCxF0	287
IOCxN	IOCxN7	IOCxN6	IOCxN5	IOCxN4	IOCxN3	IOCxN2	IOCxN1	IOCxN0	287
IOCxP	IOCxP7	IOCxP6	IOCxP5	IOCxP4	IOCxP3	IOCxP2	IOCxP1	IOCxP0	287

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by interrupt-on-change.

19.0 PERIPHERAL MODULE DISABLE (PMD)

Sleep, Idle and Doze modes allow users to substantially reduce power consumption by slowing or stopping the CPU clock. Even so, peripheral modules still remain clocked, and thus, consume some amount of power. There may be cases where the application needs what these modes do not provide: the ability to allocate limited power resources to the CPU while eliminating power consumption from the peripherals.

The PIC18F26/27/45/46/47/55/56/57K42 microcontrollers address this requirement by allowing peripheral modules to be selectively enabled or disabled, placing them into the lowest possible power mode.

All modules are ON by default following any Reset.

19.1 Disabling a Module

Disabling a module has the following effects:

- All clock and control inputs to the module are suspended; there are no logic transitions, and the module will not function.
- The module is held in Reset.
- Any SFR becomes “unimplemented”
  - Writing is disabled
  - Reading returns 00h
- I/O functionality is prioritized as per **Section 16.1, I/O Priorities**
- All associated Input Selection registers are also disabled

19.2 Enabling a Module

When the PMD register bit is cleared, the module is re-enabled and will be in its Reset state (Power-on Reset). SFR data will reflect the POR Reset values.

Depending on the module, it may take up to one full instruction cycle for the module to become active. There should be no interaction with the module (e.g., writing to registers) for at least one instruction after it has been re-enabled.

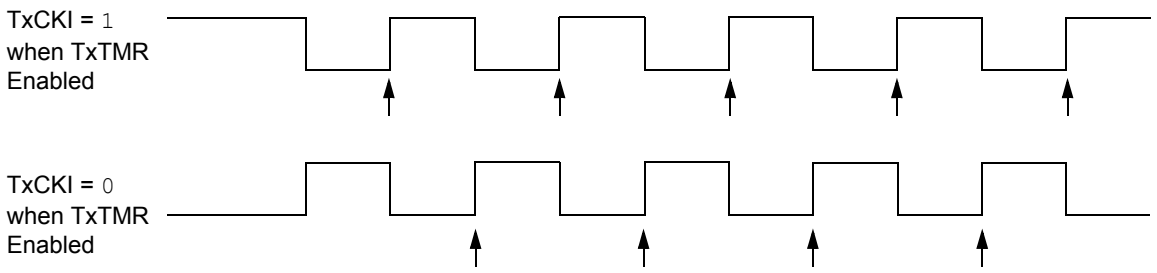
19.3 Effects of a Reset

Following any Reset, each control bit is set to ‘0’, enabling all modules.

19.4 System Clock Disable

Setting SYSCMD (PMD0, Register 19-1) disables the system clock (FOSC) 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.

FIGURE 21-3: TIMER1/3/5 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

FIGURE 21-4: TIMER1/3/5 GATE ENABLE MODE

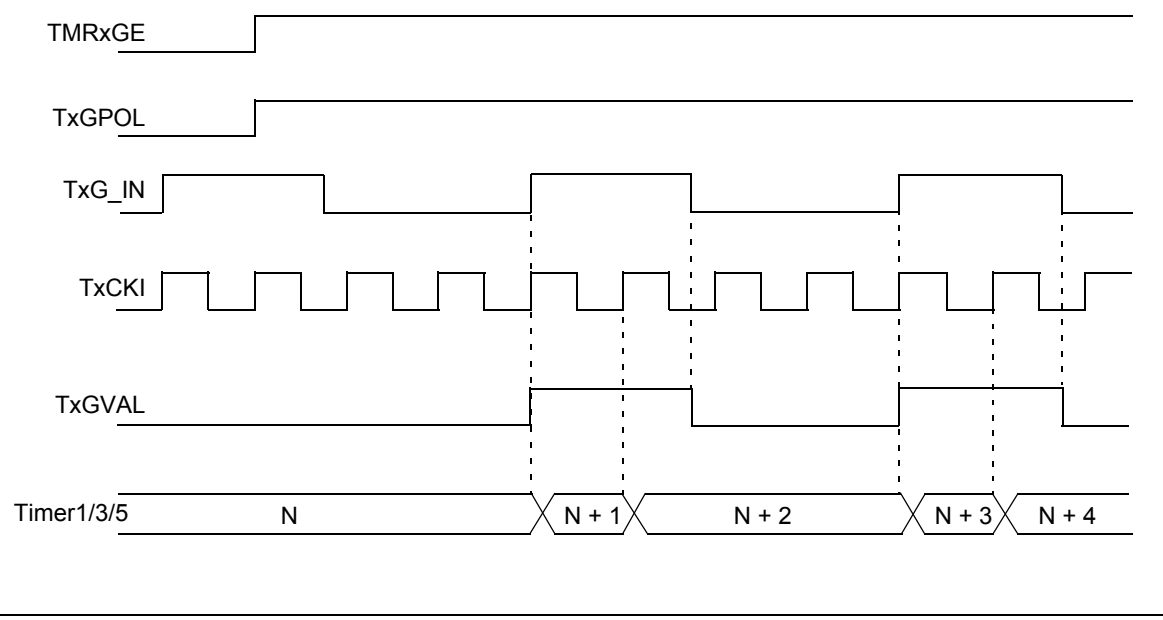


TABLE 22-1: TIMER2 OPERATING MODES

Mode	MODE<4:0>		Output Operation	Operation	Timer Control		
	<4:3>	<2:0>			Start	Reset	Stop
Free Running Period	00	000	Period Pulse	Software gate (Figure 22-6)	ON = 1	—	ON = 0
		001		Hardware gate, active-high (Figure 22-7)	ON = 1 & TMRx_ers = 1	—	ON = 0 or TMRx_ers = 0
		010		Hardware gate, active-low	ON = 1 & TMRx_ers = 0	—	ON = 0 or TMRx_ers = 1
		011	Period Pulse with Hardware Reset	Rising or Falling Edge Reset	ON = 1	TMRx_ers ↕	ON = 0
		100		Rising Edge Reset (Figure 22-8)		TMRx_ers ↑	
		101		Falling Edge Reset		TMRx_ers ↓	
		110		Low Level Reset		TMRx_ers = 0	ON = 0 or TMRx_ers = 0
		111		High Level Reset (Figure 22-9)		TMRx_ers = 1	ON = 0 or TMRx_ers = 1
One-shot	01	000	One-Shot	Software Start (Figure 22-10)	ON = 1	—	ON = 0 or Next clock after TMRx = PRx **(Note 2)**
		001	Edge Triggered Start (Note 1)	Rising Edge Start (Figure 22-9)	ON = 1 & TMRx_ers ↑	—	
		010		Falling Edge Start	ON = 1 & TMRx_ers ↓	—	
		011		Any eEdge Start	ON = 1 & TMRx_ers ↕	—	
		100	Edge Triggered Start and Hardware Reset (Note 1)	Rising Edge Start & Rising Edge Reset (Figure 22-12)	ON = 1 & TMRx_ers ↑	TMRx_ers ↑	
		101		Falling Edge Start & Falling Edge Reset	ON = 1 & TMRx_ers ↓	TMRx_ers ↓	
		110		Rising Edge Start & Low Level Reset (Figure 22-13)	ON = 1 & TMRx_ers ↑	TMRx_ers = 0	
		111		Falling Edge Start & High Level Reset	ON = 1 & TMRx_ers ↓	TMRx_ers = 1	
Monostable	10	000	Reserved				
		001	Edge Triggered Start (Note 1)	Rising Edge Start (Figure 22-12)	ON = 1 & TMRx_ers ↑	—	ON=0 or Next clock after TxTMR = TxPR **(Note 3)**
		010		Falling Edge Start	ON = 1 & TMRx_ers ↓	—	
		011		Any Edge Start	ON = 1 & TMRx_ers ↕	—	
Reserved		100	Reserved				
Reserved		101	Reserved				
One-shot		110	Level Triggered Start and Hardware Reset	High Level Start & Low Level Reset (Figure 22-13)	ON = 1 & TMRx_ers = 1	TMRx_ers = 0	ON = 0 or Held in Reset **(Note 2)**
		111		Low Level Start & High Level Reset	ON = 1 & TMRx_ers = 0	TMRx_ers = 1	
Reserved	11	xxx	Reserved				

Note 1: If ON = 0 then an edge is required to restart the timer after ON = 1.
2: When TxTMR = TxPR then the next clock clears ON and stops TxTMR at 00h.
3: When TxTMR = TxPR then the next clock stops TxTMR at 00h but does not clear ON.

23.5 Register Definitions: CCP Control

Long bit name prefixes for the CCP peripherals are shown below. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

Peripheral	Bit Name Prefix
CCP1	CCP1
CCP2	CCP2
CCP3	CCP3
CCP4	CCP4

REGISTER 23-1: CCPxCON: CCPx CONTROL REGISTER

R/W-0/0	U-0	R-x	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
EN	—	OUT	FMT	MODE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **EN:** CCP Module Enable bit
1 = CCP is enabled
0 = CCP is disabled

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

bit 5 **OUT:** CCPx Output Data bit (read-only)

bit 4 **FMT:** CCPW (pulse-width) Alignment bit
MODE = Capture mode:
Unused
MODE = Compare mode:
Unused
MODE = PWM mode:
1 = Left-aligned format
0 = Right-aligned format

bit 3-0 **MODE<3:0>:** CCPx Mode Select bits

MODE	Operating Mode	Operation	Set CCPxIF
11xx	PWM	PWM operation	Yes
1011	Compare	Pulse output; clear TMR1(2)	Yes
1010		Pulse output	Yes
1001		Clear output(1)	Yes
1000		Set output(1)	Yes
0111	Capture	Every 16th rising edge of CCPx input	Yes
0110		Every 4th rising edge of CCPx input	Yes
0101		Every rising edge of CCPx input	Yes
0100		Every falling edge of CCPx input	Yes
0011		Every edge of CCPx input	Yes
0010	Compare	Toggle output	Yes
0001		Toggle output; clear TMR1(2)	Yes
0000	Disabled		—

Note 1: The set and clear operations of the Compare mode are reset by setting MODE = 4'b0000 or EN = 0.
2: When MODE = 0001 or 1011, then the timer associated with the CCP module is cleared. TMR1 is the default selection for the CCP module, so it is used for indication purpose only.

25.6.8 CAPTURE MODE

This mode captures the Timer value based on a rising or falling edge on the SMTWINx input and triggers an interrupt. This mimics the capture feature of a CCP module. The timer begins incrementing upon the GO bit being set, and updates the value of the SMT1CPR register on each rising edge of SMTWINx, and updates the value of the CPW register on each falling edge of the SMTWINx. The timer is not reset by any hardware conditions in this mode and must be reset by software, if desired. See Figure 25-16 and Figure 25-17.

REGISTER 28-3: NCO1ACCL: NCO1 ACCUMULATOR REGISTER – LOW BYTE

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
ACC<7:0>							
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 7-0 **ACC<7:0>:** NCO1 Accumulator, Low Byte

REGISTER 28-4: NCO1ACCH: NCO1 ACCUMULATOR REGISTER – HIGH BYTE

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
ACC<15:8>							
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 7-0 **ACC<15:8>:** NCO1 Accumulator, High Byte

When TXMTIF goes true, indicating the transmit shift register has completed sending the last byte in the frame, the TX output is held in Idle state for the number of half-bit periods selected by the STP bits in the UxCON2 register.

After the last Stop bit, the TX output is held in Idle state for an additional wait time determined by the half-bit period count in the UxP1 register. For example, a 2450 µs delay (~6 half-bit times) requires a value of 6 in UxP1L.

Any writes to the UxTXB register that occur after TXMTIF goes true, but before the UxP1 wait time expires, are held and then transmitted immediately following the wait time. If a backward frame is received during the wait time, any bytes that may have been written to UxTXB will be transmitted after completion of the backward frame reception plus the UxP1 wait time.

The wait timer is reset by the backward frame and starts over immediately following the reception of the Stop bits of the backward frame. Data pending in the transmit shift register will be sent when the wait time elapses.

To replace or delete any pending forward frame data, the TXBE bit needs to be set to flush the shift register and transmit buffer. A new control byte can then be written to the UxTXB register. The control byte will be held in the buffer and sent at the beginning of the next forward frame following the UxP1 wait time.

In Control Device mode, PERIF is set when a forward frame is received. This helps the software to determine whether the received byte is part of a forward frame from a Control Device (either from the Control Device under consideration or from another Control Device on the bus) or a backward frame from a Control Gear.

31.6.2 CONTROL GEAR

The Control Gear mode is configured with the following settings:

- MODE = `0b1001`
- TXEN = `1`
- RXEN = `1`
- UxP1 = Back Frames are held for transmission this number of half-bit periods after the completion of a Forward Frame.
- UxP2 = Forward/Back Frame threshold delimiter. Idle periods more than this number of half-bit periods are detected as Forward Frames.
- UxBRGH:L = Value to achieve 1200 baud rate
- TXPOL = appropriate polarity for interface circuit
- RXPOL = same as TXPOL
- STP = `0b10` for two Stop bits
- RxyPPS = TX pin output code
- TX pin TRIS control = `0`
- RXPPS = RX pin selection code
- RX pin TRIS control = `1`
- Input pin ANSEL bit = `0`
- ON = `1`

The UART starts listening for a forward frame when the Control Gear mode is entered. Only the frames that follow an Idle period longer than UxP2 half-bit periods are detected as forward frames. Backward frames from other Control Gear are ignored. Only forward frames will be stored in UxRXB. This is necessary because a backward frame can be sent only as a response to a forward frame.

The forward frame is received one byte at a time in the receive FIFO and retrieved by reading the UxRXB register. The end of the forward frame starts a timer to delay the backward frame response by wait time equal to the number of half-bit periods stored in UxP1.

The data received in the forward frame is processed by the application software. If the application decides to send a backward frame in response to the forward frame, the value of the backward frame is written to UxTXB. This value is held for transmission in the transmit shift register until the wait time expires and is then transmitted.

If the backward frame data is written to UxTXB after the wait time has expired, it is held in the UxTXB register until the end of the wait time following the next forward frame. The TXMTIF bit is false when the backward frame data is held in the transmit shift register. Receiving a UxRXIF interrupt before the TXMTIF goes true indicates that the backward frame write was too late and another forward frame was received before sending the backward frame. The pending backward frame has to be flushed by setting the TXBE bit, to prevent it from being sent after the next Forward Frame.

REGISTER 31-12: UxP1H: UART PARAMETER 1 HIGH REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	P1<8>
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 7-6 **Unimplemented:** Read as '0'

bit 0 **P1<8>:** Most Significant Bit of Parameter 1

DMX mode:
Most Significant bit of number of bytes to transmit between Start Code and automatic Break generation

DALI Control Device mode:
Most Significant bit of idle time delay after which a Forward Frame is sent. Measured in half-bit periods

DALI Control Gear mode:
Most Significant bit of delay between the end of a Forward Frame and the start of the Back Frame Measured in half-bit periods

Other modes:
Not used

REGISTER 31-13: UxP1L: UART PARAMETER 1 LOW 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
P1<7:0>							
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 7-0 **P1<7:0>:** Least Significant Bits of Parameter 1

DMX mode:
Least Significant Byte of number of bytes to transmit between Start Code and automatic Break generation

DALI Control Device mode:
Least Significant Byte of idle time delay after which a Forward Frame is sent. Measured in half-bit periods

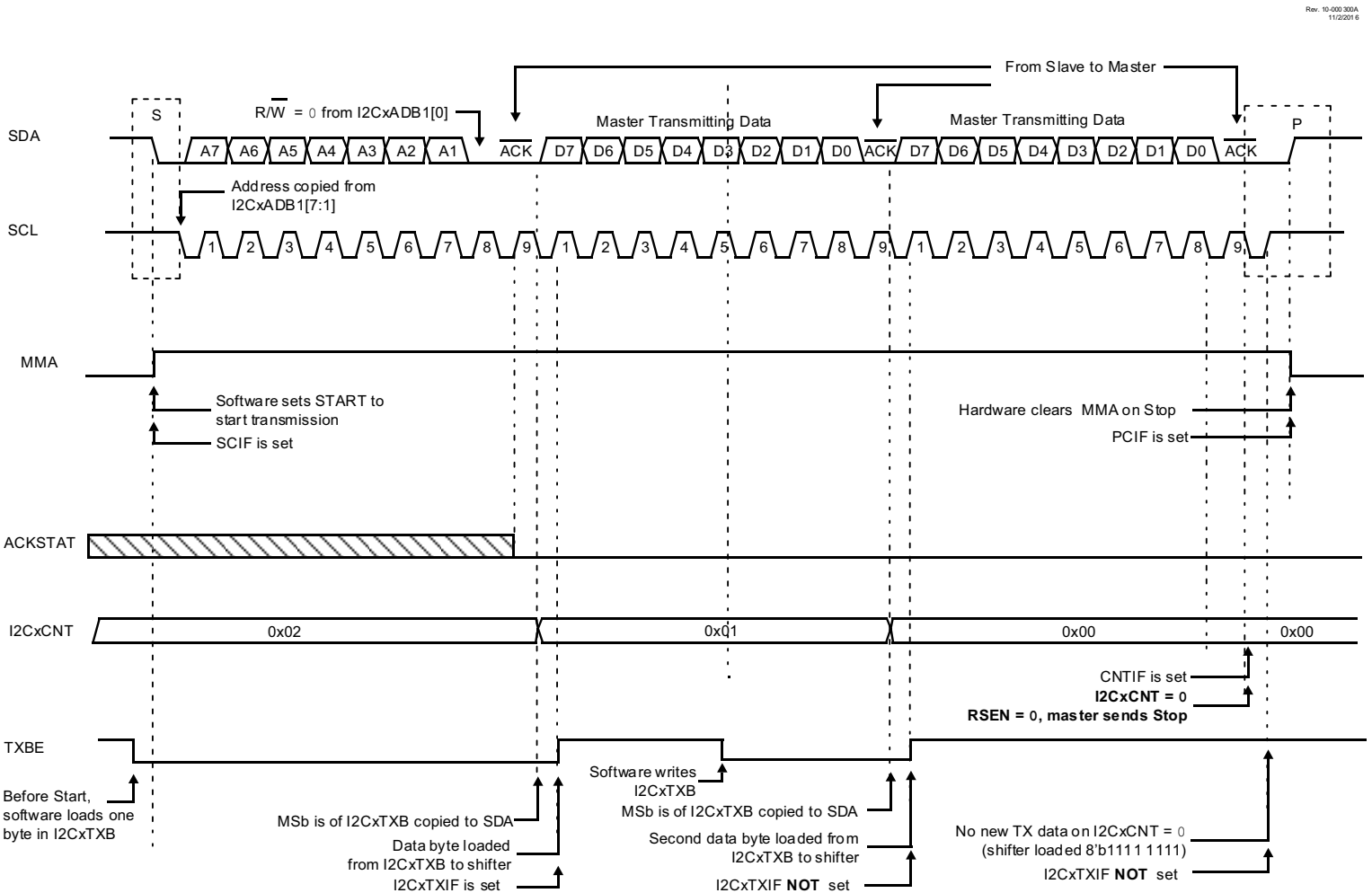
DALI Control Gear mode:
Least Significant Byte of delay between the end of a Forward Frame and the start of the Back Frame Measured in half-bit periods

LIN mode:
PID to transmit (Only Least Significant 6 bits used)

Asynchronous Address mode:
Address to transmit (9th transmit bit automatically set to '1')

Other modes:
Not used

FIGURE 33-19: I²C MASTER, 7-BIT ADDRESS, TRANSMISSION WITH STOP



39.10 Register Definitions: HLVD Control

Long bit name prefixes for the HLVD peripheral is shown in Table 39-1. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

TABLE 39-1:

Peripheral	Bit Name Prefix
HLVD	HLVD

REGISTER 39-1: HLVDCON0: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER 0

R/W-0/0	U-0	R-x	R-x	U-0	U-0	R/W-0/0	R/W-0/0
EN	—	OUT	RDY	—	—	INTH	INTL
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **EN:** High/Low-voltage Detect Power Enable bit
- 1 = Enables HLVD, powers up HLVD circuit and supporting reference circuitry
- 0 = Disables HLVD, powers down HLVD and supporting circuitry

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

bit 5 **OUT:** HLVD Comparator Output bit
- 1 = Voltage ≤ selected detection limit (HLVDL<3:0>)
- 0 = Voltage ≥ selected detection limit (HLVDL<3:0>)

bit 4 **RDY:** Band Gap Reference Voltages Stable Status Flag bit
- 1 = Indicates HLVD Module is ready and output is stable
- 0 = Indicates HLVD Module is not ready

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **INTH:** HLVD Positive going (High Voltage) Interrupt Enable
- 1 = HLVDIF will be set when voltage ≥ selected detection limit (SEL<3:0>)
- 0 = HLVDIF will not be set

bit 0 **INTL:** HLVD Negative going (Low Voltage) Interrupt Enable
- 1 = HLVDIF will be set when voltage ≤ selected detection limit (SEL<3:0>)
- 0 = HLVDIF will not be set

42.0 REGISTER SUMMARY

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
3FFFh	TOSU	—	—	—	Top of Stack Upper byte					37
3FFEh	TOSH	Top of Stack High byte								37
3FFDh	TOSL	Top of Stack Low byte								37
3FFCh	STKPTR	—	—	—	Stack Pointer					39
3FFBh	PCLATU	—	—	—	Holding Register for PC Upper byte					36
3FFAh	PCLATH	Holding Register for PC High byte								36
3FF9h	PCL	PC Low byte								36
3FF8h	TBLPTRU	—	—	Program Memory Table Pointer Upper byte						192
3FF7h	TBLPTRH	Program Memory Table Pointer High byte								192
3FF6h	TBLPTRL	Program Memory Table Pointer Low byte								192
3FF5h	TABLAT	Table Latch								192
3FF4h	PRODH	Product Register High byte								187
3FF3h	PRODL	Product Register Low byte								187
3FF2h	—	Unimplemented								
3FF1h	PCON1	—	—	—	—	—	—	$\overline{\text{MEMV}}$	—	91
3FF0h	PCON0	STKOVF	STKUNF	$\overline{\text{WDTWV}}$	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	90
3FEFh	INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed								60
3FEEh	POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented								61
3FEDh	POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented								61
3FECh	PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented								61
3FEBh	PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented – value of FSR0 offset by W								61
3FEAh	FSR0H	—	—	Indirect Data Memory Address Pointer 0 High						61
3FE9h	FSR0L	Indirect Data Memory Address Pointer 0 Low								61
3FE8h	WREG	Working Register								
3FE7h	INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed								61
3FE6h	POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented								61
3FE5h	POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented								61
3FE4h	PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented								61
3FE3h	PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented – value of FSR1 offset by W								61
3FE2h	FSR1H	—	—	Indirect Data Memory Address Pointer 1 High						61
3FE1h	FSR1L	Indirect Data Memory Address Pointer 1 Low								61
3FE0h	BSR	—	—	Bank Select Register						44
3FDFh	INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed								61
3FDEh	POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented								61
3FDDh	POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented								61
3FDCh	PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented								61
3FDBh	PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented – value of FSR2 offset by W								61
3FDAh	FSR2H	—	—	Indirect Data Memory Address Pointer 2 High						61
3FD9h	FSR2L	Indirect Data Memory Address Pointer 2 Low								61
3FD8h	STATUS	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	N	OV	Z	DC	C	58
3FD7h	IVTBASEU	—	—	—	BASE20	BASE19	BASE18	BASE17	BASE16	166
3FD6h	IVTBASEH	BASE15	BASE14	BASE13	BASE12	BASE11	BASE10	BASE9	BASE8	166
3FD5h	IVTBASEL	BASE7	BASE6	BASE5	BASE4	BASE3	BASE2	BASE1	BASE0	166
3FD4h	IVTLOCK	—	—	—	—	—	—	—	IVTLOCKED	168
3FD3h	INTCON1	STAT		—	—	—	—	—	—	136
3FD2h	INTCON0	GIE	GIEL	IPEN	—	—	INT2EDG	INT1EDG	INT0EDG	135

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 44-11: ADC CONVERSION TIMING (ADC CLOCK FROM ADCRC)