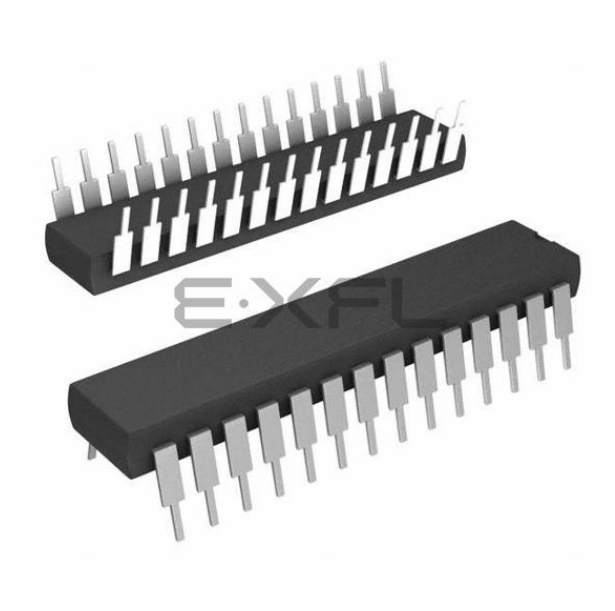


Welcome to E-XFL.COM

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	25
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 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf25k42-i-sp

TABLE 4-11: REGISTER FILE SUMMARY FOR PIC18(L)F24/25K42 DEVICES (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
3DE0h	U2UIR	WUIF	ABDIF	—	—	—	ABDIE	—	—	00---0--
3DDFh	U2FIFO	TXWRE	STPMD	TXBE	TXBF	RXIDL	XON	RXBE	RXBF	00101110
3DDEh	U2BRGH	BRGH								00000000
3DDDh	U2BRGL	BRGL								00000000
3DDCh	U2CON2	RUNOVF	RXPOL	STP		—	TXPOL	FLO		0000-000
3DDBh	U2CON1	ON	—	—	WUE	RXBIMD	—	BRKOVR	SENDB	0--00-00
3DDAh	U2CON0	BRGS	ABDEN	TXEN	RXEN	MODE				0000-000
3DD9h	—	Unimplemented								—
3DD8h	U2P3L	P3L								xxxxxxxx
3DD7h	—	Unimplemented								—
3DD6h	U2P2L	P2L								00000000
3DD5h	—	Unimplemented								—
3DD4h	U2P1L	P1L								00000000
3DD3h	—	Unimplemented								—
3DD2h	U2TXB	TXB								00000000
3DD1h	—	Unimplemented								—
3DD0h	U2RXB	RXB								00000000
3DCFh - 3D7Dh	—	Unimplemented								—
3D7Ch	I2C1BTO	BTO								-----000
3D7Bh	I2C1CLK	CLK								----0000
3D7Ah	I2C1PIE	CNTIE	ACKTIE	—	WRIE	ADRIE	PCIE	RSCIE	SCIE	00000000
3D79h	I2C1PIR	CNTIF	ACKTIF	—	WRIF	ADRIF	PCIF	RSCIF	SCIF	00000000
3D78h	I2C1STAT1	TXWE	—	TXBE	—	RXRE	CLRBF	—	RXBF	00100000
3D77h	I2C1STAT0	BFRE	SMA	MMA	R	D	—	—	—	00000000
3D76h	I2C1ERR	—	BTOIF	BCLIF	NACKIF	—	BTOIE	BCLIE	NACKIE	00000000
3D75h	I2C1CON2	ACNT	GCEN	FME	ABD	SDAHT		BFRET		00000000
3D74h	I2C1CON1	ACKCNT	ACKDT	ACKSTAT	ACKT	—	RXO	TXU	CSD	00000000
3D73h	I2C1CON0	EN	RSEN	S	CSTR	MDR	MODE			00000000
3D72h	I2C1ADR3	ADR							—	11111110
3D71h	I2C1ADR2	ADR								11111111
3D70h	I2C1ADR1	ADR							—	11111110
3D6Fh	I2C1ADR0	ADR								11111111
3D6Eh	I2C1ADB1	ADB								xxxxxxxx
3D6Dh	I2C1ADB0	ADB								xxxxxxxx
3D6Ch	I2C1CNT	CNT								xxxxxxxx
3D6Bh	I2C1TXB	TXB								xxxxxxxx
3D6Ah	I2C1RXB	RXB								xxxxxxxx
3D69h - 3D67h	—	Unimplemented								—
3D66h	I2C2BTO	BTO								-----000
3D65h	I2C2CLK	CLK								----0000
3D64h	I2C2PIE	CNTIE	ACKTIE	—	WRIE	ADRIE	PCIE	RSCIE	SCIE	00000000
3D63h	I2C2PIR	CNTIF	ACKTIF	—	WRIF	ADRIF	PCIF	RSCIF	SCIF	00000000
3D62h	I2C2STAT1	TXWE	—	—	—	RXRE	CLRBF	—	RXBF	00100000
3D61h	I2C2STAT0	BFRE	—	MMA	—	D	—	—	—	00000000
3D60h	I2C2ERR	—	BTOIF	BCLIF	NACKIF	—	BTOIE	BCLIE	NACKIE	00000000
3D5Fh	I2C2CON2	ACNT	GCEN	FME	ABD	SDAHT		BFRET		00000000
3D5Eh	I2C2CON1	ACKCNT	ACKDT	ACKSTAT	ACKT	—	RXO	TXU	CSD	00000000

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition
Note 1: Not present in LF devices.

8.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

8.2 Brown-out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- BOR is always on
- BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 8-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV<1:0> bits in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Table 46-11 for more information.

8.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

8.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

8.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

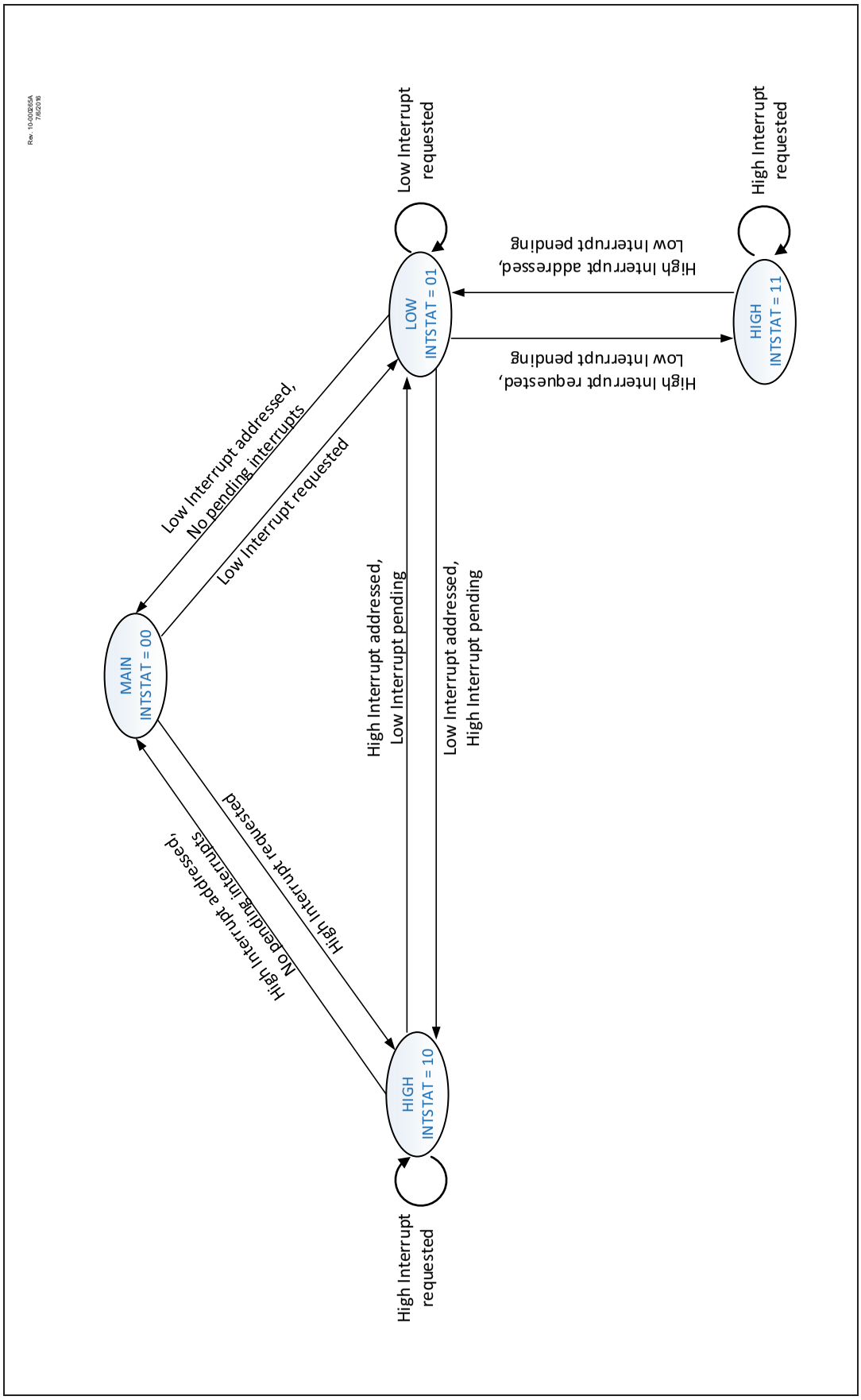
BOR protection is unchanged by Sleep.

8.2.4 BOR AND BULK ERASE

BOR is forced ON during PFM Bulk Erase operations to make sure that a safe erase voltage is maintained for a successful erase cycle.

During Bulk Erase, the BOR is enabled at 2.45V for F and LF devices, even if it is configured to some other value. If VDD falls, the erase cycle will be aborted, but the device will not be reset.

FIGURE 11-1: VECTORED INTERRUPTS STATE TRANSITION DIAGRAM



REGISTER 13-3: WDTPSL: WWDT PRESCALE SELECT LOW BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<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 **PSCNT<7:0>:** Prescale Select Low Byte bits(1)

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should not be read during normal operation.

REGISTER 13-4: WDTPSH: WWDT PRESCALE SELECT HIGH BYTE REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
PSCNT<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 **PSCNT<15:8>:** Prescale Select High Byte bits(1)

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should not be read during normal operation.

REGISTER 16-9: CRCXORH: CRC XOR HIGH BYTE REGISTER

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x
X<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 **X<15:8>**: XOR of Polynomial Term X^n Enable bits

REGISTER 16-10: CRCXORL: CRC XOR LOW BYTE REGISTER

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	U-1
X<7:1>							—
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit
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 7-1 **X<7:1>**: XOR of Polynomial Term X^n Enable bits

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

Depending on the priority of the DMA with respect to CPU execution (Refer to **Section 3.2 "Memory Access Scheme"** for more information), the DMA Controller can move data through two methods:

- Stalling the CPU execution until it has completed its transfers (DMA has higher priority over the CPU in this mode of operation)
- Utilizing unused CPU cycles for DMA transfers (CPU has higher priority over the DMA in this mode of operation). Unused CPU cycles are referred to as bubbles which are instruction cycles available for use by the DMA to perform read and write operations. In this way, the effective bandwidth for handling data is increased; at the same time, DMA operations can proceed without causing a processor stall.

17.4 DMA Interface

The DMA module transfers data from the source to the destination one byte at a time, this smallest data movement is called a DMA data transaction. A DMA Message refers to one or more DMA data transactions.

Each DMA data transaction consists of two separate actions:

- Reading the Source Address Memory and storing the value in the DMA Buffer register
- Writing the contents of the DMA Buffer register to the Destination Address Memory

Note:	DMA data movement is a two-cycle operation.

The XIP bit (DMAxCON0 register) is a status bit to indicate whether or not the data in the DMAxBUF register has been written to the destination address. If the bit is set then data is waiting to be written to the destination. If clear it means that either data has been written to the destination or that no source read has occurred.

The DMA has read access to PFM, Data EEPROM, and SFR/GPR space, and write access to SFR/GPR space. Based on these memory access capabilities, the DMA can support the following memory transactions:

TABLE 17-1: DMA MEMORY ACCESS

Read Source	Write Destination
Program Flash Memory	GPR
Program Flash Memory	SFR
Data EE	GPR
Data EE	SFR
GPR	GPR
SFR	GPR
GPR	SFR
SFR	SFR

Even though the DMA module has access to all memory and peripherals that are also available to the CPU, it is recommended that the DMA does not access any register that is part of the System arbitration. The DMA, as a system arbitration client should not be read or written by itself or by another DMA instantiation.

The following sections discuss the various control interfaces required for DMA data transfers.

17.4.1 DMA ADDRESSING

The start addresses for the source read and destination write operations are set using the DMAxSSA <21:0> and DMAxDSA <15:0> registers, respectively.

When the DMA Message transfers are in progress, the DMAxSPTR <21:0> and DMAxDPTR <15:0> registers contain the current address pointers for each source read and destination write operation, these registers are modified after each transaction based on the Address mode selection bits.

The SMODE and DMODE bits in the DMAxCON1 control register determine the address modes of operation by controlling how the DMAxSPTR <21:0> and DMAxDPTR <15:0> bits are updated after every DMA data transaction combination (Figure 17-2).

Each address can be separately configured to:

- Remain unchanged
- Increment by 1
- Decrement by 1

18.4 PORTE Functionality

18.4.1 MASTER CLEAR INPUT $\overline{\text{MCLR}}$

For PIC18(L)F2xK42 devices, PORTE is only available when Master Clear functionality is disabled (MCLRE = 0). In this case, PORTE is a single bit, input-only port comprised of RE3 only. The pin operates as previously described. RE3 in PORTE register is a read-only bit and will read '1' when MCLRE = 1 (i.e., Master Clear enabled).

18.4.2 RE3 WEAK PULL-UP

The port RE3 pin has an individually controlled weak internal pull-up. When set, the WPUE3 bit enables the RE3 pin pull-up. When the RE3 port pin is configured as $\overline{\text{MCLR}}$, (CONFIG2L, MCLRE = 1 and CONFIG4H, LVP = 0), or configured for Low-Voltage Programming, (MCLRE = x and LVP = 1), the pull-up is always enabled and the WPUE3 bit has no effect.

18.4.3 INTERRUPT-ON-CHANGE

The interrupt-on-change feature is available only on the RE3 pin of PORTE for all devices. If MCLRE = 1 or LVP = 1, RE3 port functionality is disabled and interrupt-on-change on RE3 is not available. For further details refer to **Section 9.11 "Interrupt-on-Change"**.

25.0 CAPTURE/COMPARE/PWM MODULE

The Capture/Compare/PWM module is a peripheral that allows the user to time and control different events, and to generate Pulse-Width Modulation (PWM) signals. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate Pulse-Width Modulated signals of varying frequency and duty cycle.

This family of devices contains four standard Capture/Compare/PWM modules (CCP1, CCP2, CCP3 and CCP4). Each individual CCP module can select the timer source that controls the module. Each module has an independent timer selection which can be accessed using the CxTSEL bits in the CCPTMRS register (Register 25-2). The default timer selection is TMR1 when using Capture/Compare mode and TMR2 when using PWM mode in the CCPx module.

Please note that the Capture/Compare mode operation is described with respect to TMR1 and the PWM mode operation is described with respect to TMR2 in the following sections.

The Capture and Compare functions are identical for all CCP modules.

Note 1: In devices with more than one CCP module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the CCP1CON and CCP2CON control the same operational aspects of two completely different CCP modules.

2: Throughout this section, generic references to a CCP module in any of its operating modes may be interpreted as being equally applicable to CCPx module. Register names, module signals, I/O pins, and bit names may use the generic designator 'x' to indicate the use of a numeral to distinguish a particular module, when required.

25.1 CCP Module Configuration

Each Capture/Compare/PWM module is associated with a control register (CCPxCON), a capture input selection register (CCPxCAP) and a data register (CCPRx). The data register, in turn, is comprised of two 8-bit registers: CCPRxL (low byte) and CCPRxH (high byte).

25.1.1 CCP MODULES AND TIMER RESOURCES

The CCP modules utilize Timers 1 through 6 that vary with the selected mode. Various timers are available to the CCP modules in Capture, Compare or PWM modes, as shown in Table 25-1.

TABLE 25-1: CCP MODE – TIMER RESOURCE

CCP Mode	Timer Resource
Capture	Timer1, Timer3 or Timer5
Compare	
PWM	Timer2, Timer4 or Timer6

The assignment of a particular timer to a module is determined by the timer to CCP enable bits in the CCPTMRS register (see Register 25-2) All of the modules may be active at once and may share the same timer resource if they are configured to operate in the same mode (Capture/Compare or PWM) at the same time.

25.1.2 OPEN-DRAIN OUTPUT OPTION

When operating in Output mode (the Compare or PWM modes), the drivers for the CCPx pins can be optionally configured as open-drain outputs. This feature allows the voltage level on the pin to be pulled to a higher level through an external pull-up resistor and allows the output to communicate with external circuits without the need for additional level shifters.

FIGURE 27-8: HIGH AND LOW MEASURE MODE REPEAT ACQUISITION TIMING DIAGRAM

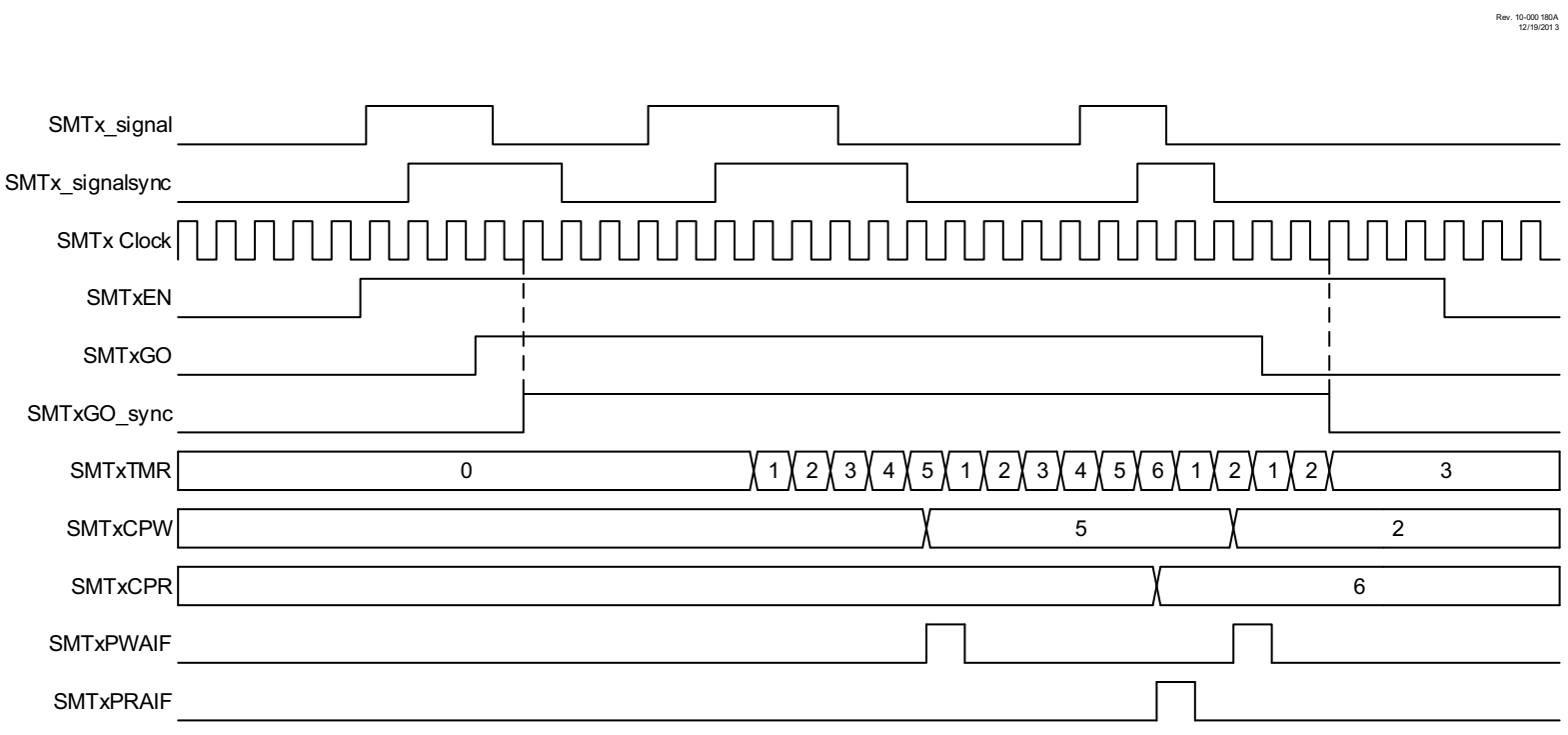


FIGURE 27-11: WINDOWED MEASURE MODE SINGLE ACQUISITION TIMING DIAGRAM

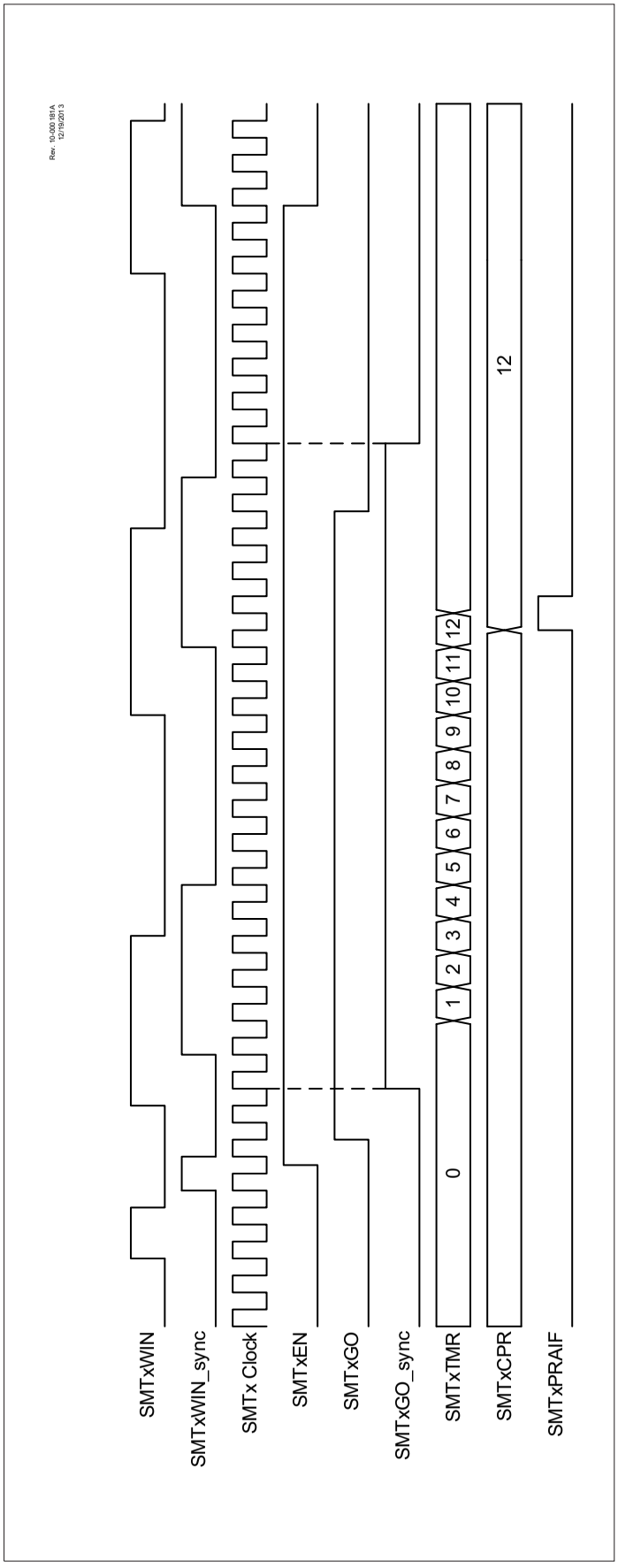
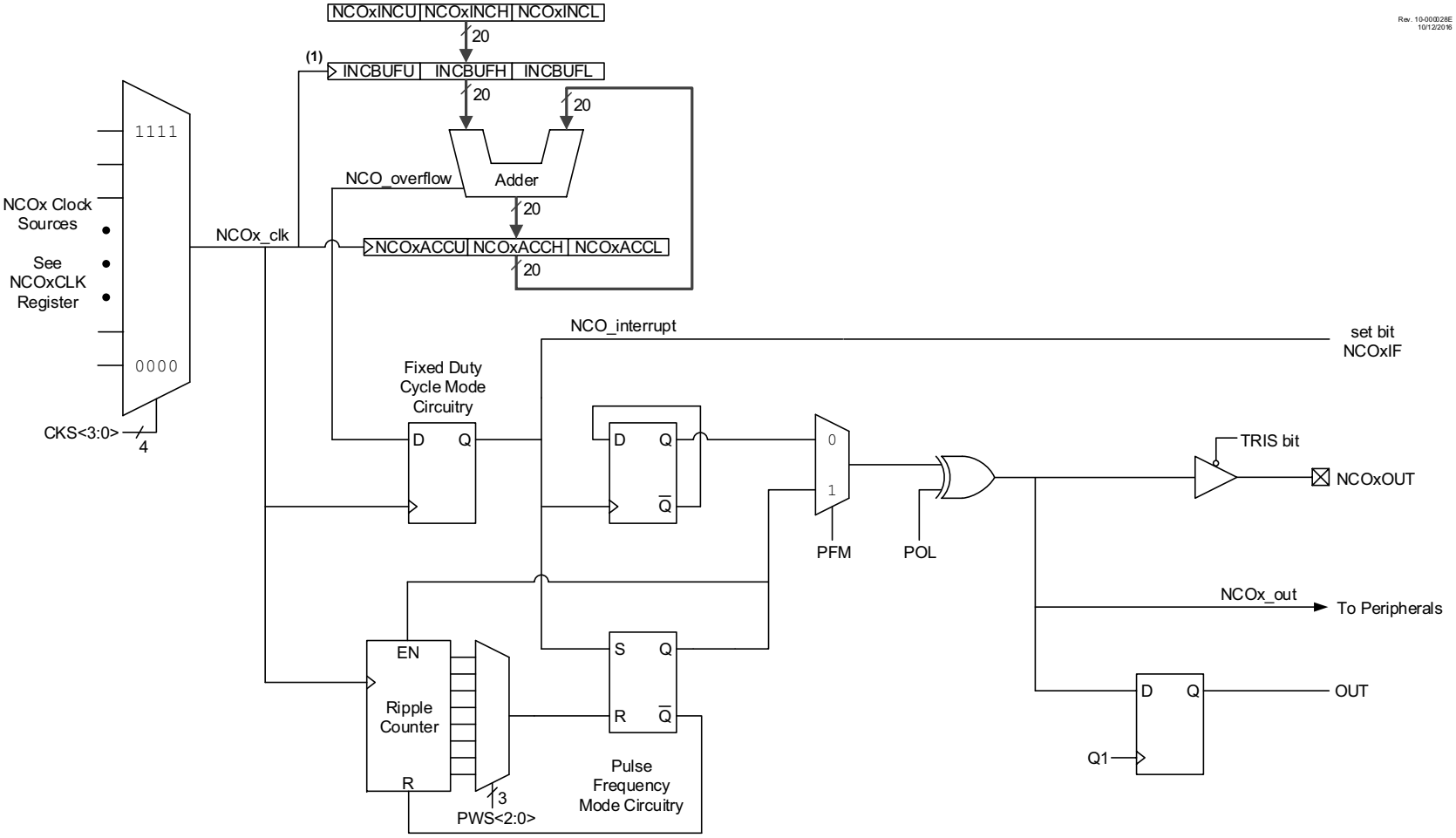


FIGURE 30-1: DIRECT DIGITAL SYNTHESIS MODULE SIMPLIFIED BLOCK DIAGRAM



Note 1: The increment registers are double-buffered to allow for value changes to be made without first disabling the NCO module. The full increment value is loaded into the buffer registers on the second rising edge of the NCOx_clk signal that occurs immediately after a write to NCOxINCL register. The buffers are not user-accessible and are shown here for reference.

33.9 Stop Bits

The number of Stop bits is user selectable with the STP bits in the UxCON2 register.The STP bits affect all modes of operation.

Stop bits selections include:

- 1 transmit with receive verify on first
- 1.5 transmit with receive verify on first
- 2 transmit with receive verify on both
- 2 transmit with receive verify on first only

In all modes, except DALI, the transmitter is idle for the number of Stop bit periods between each consecutively transmitted word. In DALI, the Stop bits are generated after the last bit in the transmitted data stream.

The input is checked for the idle level in the middle of the first Stop bit, when receive verify on first is selected, as well as in the middle of the second Stop bit, when verify on both is selected. If any Stop bit verification indicates a non-idle level, the framing error FERIF bit is set for the received word.

33.9.1 DELAYED UXRXIF

When operating in Half-Duplex mode, where the microcontroller needs to reverse the transceiver direction after a reception, it may be more convenient to hold off the UxRXIF interrupt until the end of the Stop bits to avoid line contention. The user selects when the UxRXIF interrupt occurs with the STPMD bit in the UxFIFO register. When STPMD is '1', the UxRXIF occurs at the end of the last Stop bit. When STPMD is '0', UxRXIF occurs when the received byte is stored in the receive FIFO. When STP<1:0> = 10, the store operation is performed in the middle of the second Stop bit, otherwise, it is performed in the middle of the first Stop bit. The FERIF and PERIF interrupts are not delayed with STPMD. Only UxRXIF is delayed when STPMD is set and should be the only indicator for reversing transceiver direction.

33.10 Operation after FIFO overflow

The Receive Shift Register (RSR) can be configured to stop or continue running during a receive FIFO overflow condition. Stopped operation is the Legacy mode.

When the RSR continues to run during an overflow condition, the first word received after clearing the overflow will always be valid.

When the RSR is stopped during an overflow condition, synchronization with the Start bits is lost. Therefore, the first word received after the overflow is cleared may start in the middle of a word.

Operation during overflow is selected with the RUNOVF bit in the UxCON2 register. Setting the RUNOVF bit selects the run during overflow method.

33.11 Receive and Transmit Buffers

The UART uses small buffer areas to transmit and receive data. These are sometimes referred to as FIFOs.

The receiver has a Receive Shift Register (RSR) and two buffer registers. The buffer at the top of the FIFO (earliest byte to enter the FIFO) is by retrieved by reading the UxRXB register.

The transmitter has one Transmit Shift Register (TSR) and one buffer register. Writes to UxTXB go to the transmit buffer then immediately to the TSR, if it is empty. When the TSR is not empty, writes to UxTXB are held then transferred to the TSR when it becomes available.

33.11.1 FIFO STATUS

The UxFIFO register contains several status bits for determining the state of the receive and transmit buffers.

The RXBE bit indicates that the receive FIFO is empty. This bit is essentially the inverse of UxRXIF. The RXBF bit indicates that the receive FIFO is full.

The transmitter has only one buffer register so the status bits are essentially a copy and inverse of the UxTXIF bit. The TXBE bit indicates that the buffer is empty (same as UxTXIF) and the TXBF bit indicates that the buffer is full (UxTXIF inverse). A third transmitter status bit, TXWRE (transmit write error), is set whenever a UxTXB write is performed when the TXBF bit is set. This indicates that the write was unsuccessful.

33.11.2 FIFO RESET

All modes support resetting the receive and transmit buffers.

The receive buffer is flushed and all unread data discarded when the RXBE bit in the UxFIFO register is written to '1'. The MOVWF instruction with the TXBE bit cleared should be used to avoid inadvertently clearing a byte pending in the TSR when UxTXB is empty.

Data written to UxTXB when TXEN is low will be held in the Transmit Shift Register (TSR) then sent when TXEN is set. The transmit buffer and inactive TSR are flushed by setting the TXBE bit in the UxFIFO register. Setting TXBE while a character is actively transmitting from the TSR will complete the transmission without being flushed.

Clearing the ON bit will discard all received data and transmit data pending in the TSR and UxTXB.

35.1 I²C Features

- Inter-Integrated Circuit (I²C) interface supports the following modes in hardware:
  - Master mode
  - Slave mode with byte NACKing
  - Multi-Master mode
- Dedicated Address, Receive and Transmit buffers
- Up to four Slave addresses matching
- General Call address matching
- 7-bit and 10-bit addressing with masking
- Start, Restart, Stop, Address, Write, and ACK Interrupts
- Clock Stretching hardware for:
  - RX Buffer Full
  - TX Buffer Empty
  - After Address, Write, and ACK
- Bus Collision Detection with arbitration
- Bus Timeout Detection
- SDA hold time selection
- I²C, SMBus 2.0, and SMBus 3.0 input level selections

35.2 I²C Module Overview

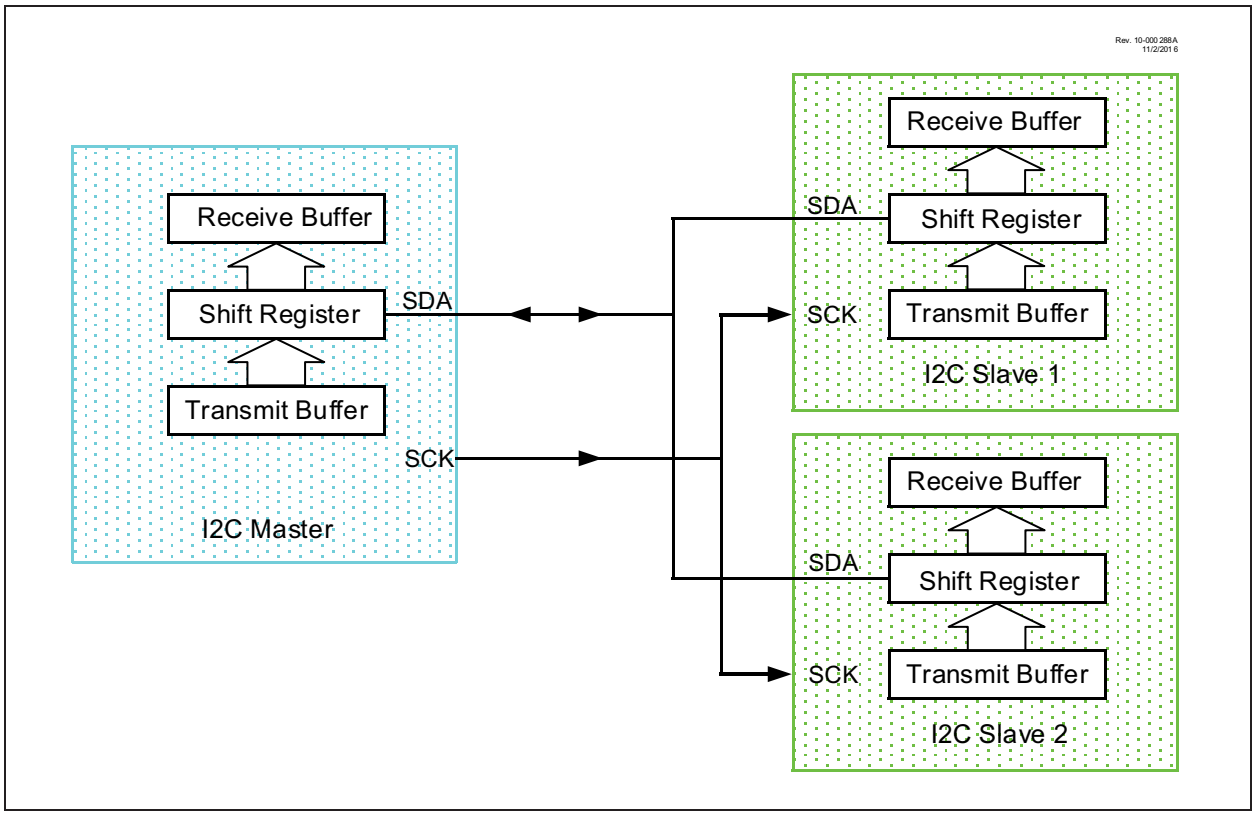
The I²C module provides a synchronous interface between the microcontroller and other I²C-compatible devices using the two-wire I²C serial bus. Devices communicate in a master/slave environment. The I²C bus specifies two signal connections:

- Serial Clock (SCL)
- Serial Data (SDA)

Both the SCL and SDA connections are bidirectional open-drain lines, each requiring pull-up resistors to the supply voltage. Pulling the line to ground is considered a logical zero and letting the line float is considered a logical one. Every transaction on the I²C bus has to be initiated by the Master.

Figure 35-25 shows a typical connection between a master and more than one slave.

FIGURE 35-2: I²C MASTER/SLAVE CONNECTIONS



35.5.11 MASTER TRANSMISSION 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 transmitting data. Figure 35-21 is used as a visual reference for this description

1. If ABD = 0; i.e. Address buffers are enabled

Master software loads number of bytes to be transmitted in one sequence in I2CxCNT, high address byte of slave address in I2CxADB1 with R/W = 0, low address byte in I2CxADB0 and the first byte of data in I2CxTXB. Master software has to set the Start (S) bit to initiate communication.

If ABD = 1; i.e. Address buffers are disabled

Master software loads the number of bytes to be transmitted in one sequence in I2CxCNT and the high address byte of the slave address with R/W = 0 into the I2CxTXB register. Writing to the I2CxTXB will assert the start condition on the bus and sets the S bit. Software writes to the S bit are ignored in this case.

2. Master hardware waits for BFRE bit to be set; then shifts out the start and high address and waits for acknowledge.
3. If NACK, master hardware sends Stop.
4. If ABD = 0; i.e. Address buffer are enabled

If ACK, master hardware sends the low address byte from I2CxADB0.

If ABD = 1; i.e., Address buffer are disabled

If ACK, master hardware sets TXIF and MDR bits and the software has to write the low address byte into I2CxTXB. Writing to I2CxTXB sends the low address on the bus.

5. If TXBE = 1 and I2CxCNT! = 0, I2CxTXIF and MDR bits are set. Clock is stretched on 8th falling SCL edge till master software writes next data byte to I2CxTXB.
6. Master hardware sends ninth SCL pulse for ACK from slave and loads the shift register from I2CxTXB. I2CxCNT is decremented.
7. If slave sends a NACK, master hardware sends Stop and ends transmission.
8. If slave sends an ACK, master hardware outputs data in the shift register on SDA. I2CxCNT value is checked on the 8th falling SCL edge. If I2CxCNT = 0; master hardware sends 9th SCL pulse for ACK and CNTIF is set.
9. If I2CxCNT != 0; go to step 5.

REGISTER 38-15: ADCNT: ADC REPEAT COUNTER REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
CNT<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 **CNT<7:0>**: ADC Repeat Count bits
Determines the number of times that the ADC is triggered before the threshold is checked when the computation is Low-pass Filter, Burst Average, or Average modes. See Table 38-2 for more details.

REGISTER 38-16: ADFLTRH: ADC FILTER HIGH BYTE REGISTER

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FLTR<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 **FLTR<15:8>**: ADC Filter Output Most Significant bits
In Accumulate, Average, and Burst Average mode, this is equal to ACC right shifted by the ADCRS bits of ADCON2. In LPF mode, this is the output of the Low-pass Filter.

REGISTER 38-17: ADFLTRL: ADC FILTER LOW BYTE REGISTER

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FLTR<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 **FLTR<7:0>**: ADC Filter Output Least Significant bits
In Accumulate, Average, and Burst Average mode, this is equal to ACC right shifted by the ADCRS bits of ADCON2. In LPF mode, this is the output of the Low-pass Filter.

BRA **Unconditional Branch**

Syntax: BRA n

Operands: -1024 ≤ n ≤ 1023

Operation: (PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1101	0nnn	nnnn	nnnn

Description: Add the 2's complement number '2n' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is a 2-cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

Example: `HERE BRA Jump`

Before Instruction
PC = address `(HERE)`
After Instruction
PC = address `(Jump)`

BSF **Bit Set f**

Syntax: BSF f, b {,a}

Operands: 0 ≤ f ≤ 255
0 ≤ b ≤ 7
a ∈ [0,1]

Operation: 1 → f<b>

Status Affected: None

Encoding:

1000	bbba	ffff	ffff

Description: Bit 'b' in register 'f' is set.
If 'a' is '0', the Access Bank is selected.
If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 43.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: `BSF FLAG_REG, 7, 1`

Before Instruction
FLAG_REG = 0Ah
After Instruction
FLAG_REG = 8Ah

MOVFFL **Move f to f (Long Range)**

Syntax: MOVFFL f_s,f_d

Operands: $0 \le f_s \le 16383$
$0 \le f_d \le 16383$

Operation: $(f_s) \rightarrow f_d$

Status Affected: None

Encoding:

1st word	0000	0000	0110	$f_sf_sf_sf_s$
2nd word	1111	$f_sf_sf_sf_s$	$f_sf_sf_sf_s$	$f_sf_sf_df_d$
3rd word	1111	$f_df_df_df_d$	$f_df_df_df_d$	$f_df_df_df_d$

Description: The contents of source register 'f_s' are moved to destination register 'f_d'. Location of source 'f_s' can be anywhere in the 16 Kbyte data space (0000h to 3FFFh). Either source or destination can be W (a useful special situation).
MOVFFL is particularly useful for transferring a data memory location to a peripheral register (such as the transmit buffer or an I/O port).
The MOVFFL instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

Words: 3

Cycles: 3

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
Decode	Read register 'f_s' (src)	Process data	No operation
Decode	No operation No dummy read	No operation	Write register 'f_d' (dest)

Example: MOVFFL 2000h, 200Ah

Before Instruction
Contents of 2000h = 33h
Contents of 200Ah = 11h
After Instruction
Contents of 2000h = 33h
Contents of 200Ah = 33h

MOVLB **Move literal to BSR**

Syntax: MOVLW k

Operands: $0 \le k \le 63$

Operation: $k \rightarrow$ BSR

Status Affected: None

Encoding:

0000	0001	00kk	kkkk

Description: The 6-bit literal 'k' is loaded into the Bank Select Register (BSR<5:0>). The value of BSR<7:6> always remains '0'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write literal 'k' to BSR

Example: MOVLB 5

Before Instruction
BSR Register = 02h
After Instruction
BSR Register = 05h

TABLE 44-1: REGISTER FILE SUMMARY FOR PIC18(L)F24/25K42 DEVICES (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3ECAh	HLVDCON1	—	—	—	—	SEL				661
3EC9h	HLVDCON0	EN	—	OUT	RDY	—	—	INTH	INTL	662
3EC8h - 3EC4h	—	Unimplemented								
3EC3h	ZCDCON	SEN	—	OUT	POL	—	—	INTP	INTN	465
3EC2h	—	Unimplemented								
3EC1h	FVRCON	EN	RDY	TSEN	TSRNG	CDAFVR		ADFVR		600
3EC0h	CMOUT	—	—	—	—	—	—	C2OUT	C1OUT	654
3EBFh	CM1PCH	—	—	—	—	—	PCH			654
3EBEh	CM1NCH	—	—	—	—	—	NCH			653
3EBDh	CM1CON1	—	—	—	—	—	—	INTP	INTN	653
3EBCh	CM1CON0	EN	OUT	—	POL	—	—	HYS	SYNC	652
3EBBh	CM2PCH	—	—	—	—	—	PCH			654
3EBAh	CM2NCH	—	—	—	—	—	NCH			653
3EB9h	CM2CON1	—	—	—	—	—	—	INTP	INTN	653
3EB8h	CM2CON0	EN	OUT	—	POL	—	—	HYS	SYNC	652
3EB7h - 3E9Fh	—	Unimplemented								
3E9Eh	DAC1CON0	EN	—	OE1	OE2	PSS		—	NSS	644
3E9Dh	—	Unimplemented								
3E9Ch	DAC1CON1	—	—	—	DATA					645
3E9Bh - 3DFBh	—	Unimplemented								
3DFAh	U1ERRIE	TXMTIE	PERIE	ABDOVE	CERIE	FERIE	RXBKIE	RXFOIE	TXCIE	504
3DF9h	U1ERRIR	TXMTIF	PERIF	ABDOVF	CERIF	FERIF	RXBKIF	RXFOIF	TXCIF	503
3DF8h	U1UIR	WUIF	ABDIF	—	—	—	ABDIE	—	—	505
3DF7h	U1FIFO	TXWRE	STPMD	TXBE	TXBF	RXIDL	XON	RXBE	RXBF	506
3DF6h	U1BRGH	BRGH								507
3DF5h	U1BRGL	BRGL								507
3DF4h	U1CON2	RUNOVF	RXPOL	STP		C0EN	TXPOL	FLO		502
3DF3h	U1CON1	ON	—	—	WUE	RXBIMD	—	BRKOVR	SENDB	501
3DF2h	U1CON0	BRGS	ABDEN	TXEN	RXEN	MODE				500
3DF1h	U1P3H	—	—	—	—	—	—	—	P3H	511
3DF0h	U1P3L	P3L								511
3DEFh	U1P2H	—	—	—	—	—	—	—	P2H	510
3DEEh	U1P2L	P2L								510
3DEDh	U1P1H	—	—	—	—	—	—	—	P1H	509
3DECh	U1P1L	P1L								509
3DEBh	U1TXCHK	TXCHK								512
3DEAh	U1TXB	TXB								508
3DE9h	U1RXCHK	RXCHK								512
3DE8h	U1RXB	RXB								508
3DE7h - 3DE3h	—	Unimplemented								
3DE2h	U2ERRIE	TXMTIE	PERIE	ABDOVE	CERIE	FERIE	RXBKIE	RXFOIE	TXCIE	504
3DE1h	U2ERRIR	TXMTIF	PERIF	ABDOVF	CERIF	FERIF	RXBKIF	RXFOIF	TXCIF	503
3DE0h	U2UIR	WUIF	ABDIF	—	—	—	ABDIE	—	—	505
3DDFh	U2FIFO	TXWRE	STPMD	TXBE	TXBF	RXIDL	XON	RXBE	RXBF	506
3DDEh	U2BRGH	BRGH								507
3DDDh	U2BRGL	BRGL								507

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition
Note 1: Not present in LF devices.

TABLE 46-7: EXTERNAL CLOCK/OSCILLATOR TIMING REQUIREMENTS (CONTINUED)

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS21	F_{CY}	Instruction Frequency	—	FOSC/4	—	MHz	
OS22	T_{CY}	Instruction Period	62.5	$1/F_{CY}$	—	ns	

* 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 12.0 "Power-Saving Operation Modes"**.

3: The system clock frequency (FOSC) must meet the voltage requirements defined in the **Section 46.2 "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.

FIGURE 46-7: CLKOUT AND I/O TIMING

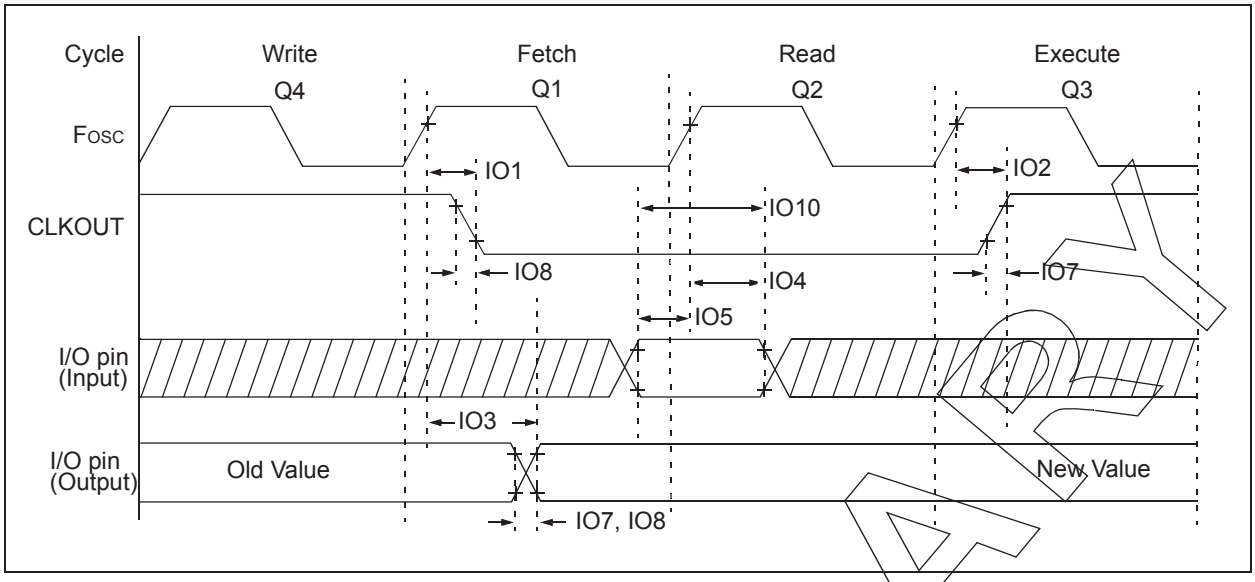


TABLE 46-10: I/O AND CLKOUT TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
IO1*	$T_{CLKOUTH}$	CLKOUT rising edge delay (rising edge Fosc (Q1 cycle) to falling edge CLKOUT	—	—	70	ns	
IO2*	$T_{CLKOUTL}$	CLKOUT falling edge delay (rising edge Fosc (Q3 cycle) to rising edge CLKOUT	—	—	72	ns	
IO3*	T_{IO_VALID}	Port output valid time (rising edge Fosc (Q1 cycle) to port valid)	—	50	70	ns	
IO4*	T_{IO_SETUP}	Port input setup time (Setup time before rising edge Fosc – Q2 cycle)	20	—	—	ns	
IO5*	T_{IO_HOLD}	Port input hold time (Hold time after rising edge Fosc – Q2 cycle)	50	—	—	ns	
IO6*	T_{IOR_SLREN}	Port I/O rise time, slew rate enabled	—	25	—	ns	V_{DD} = 3.0V
IO7*	T_{IOR_SLRDIS}	Port I/O rise time, slew rate disabled	—	5	—	ns	V_{DD} = 3.0V
IO8*	T_{IOF_SLREN}	Port I/O fall time, slew rate enabled	—	25	—	ns	V_{DD} = 3.0V
IO9*	T_{IOF_SLRDIS}	Port I/O fall time, slew rate disabled	—	5	—	ns	V_{DD} = 3.0V
IO10*	T_{INT}	INT pin high or low time to trigger an interrupt	25	—	—	ns	
IO11*	T_{IOC}	Interrupt-on-Change minimum high or low time to trigger interrupt	25	—	—	ns	

*These parameters are characterized but not tested.