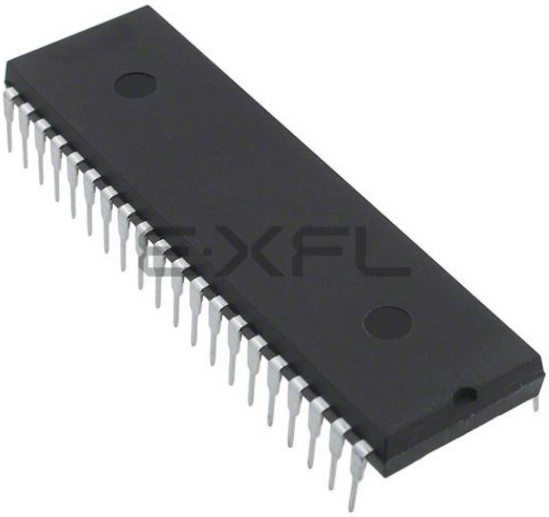
Microchip Technology - PIC18F46K42-E/P Datasheet



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	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K 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 ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f46k42-e-p

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	I²C	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	$\overline{\text{MCLR}}$ VPP
VDD	20	17	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VSS	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 I²C 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 I²C specific or SMBUS input buffer thresholds.

REGISTER 5-2: CONFIGURATION WORD 1H (30 0001h)

U-1	U-1	R/W-1	U-1	R/W-1	U-1	R/W-1	R/W-1
—	—	FCMEN	—	CSWEN	—	PR1WAY	$\overline{\text{CLKOUTEN}}$
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 **Unimplemented:** Read as ‘1’

bit 5 **FCMEN:** Fail-Safe Clock Monitor Enable bit
1 = FSCM timer is enabled
0 = FSCM timer is disabled

bit 4 **Unimplemented:** Read as ‘1’

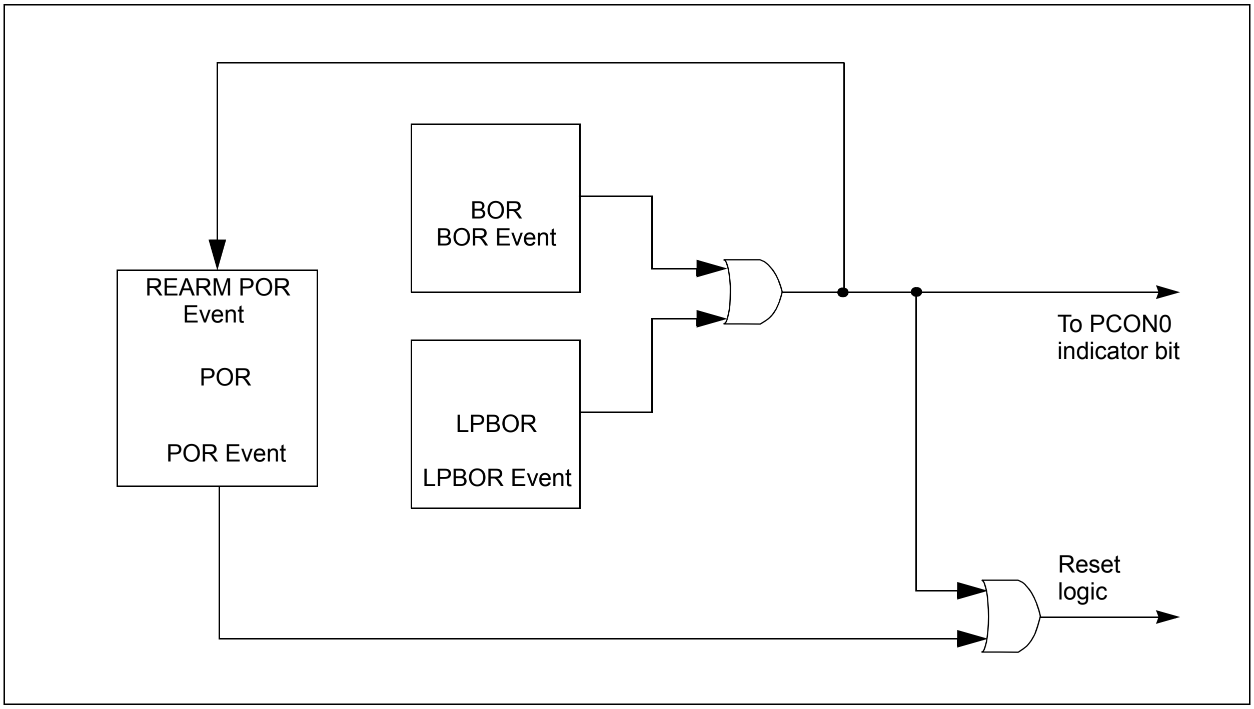
bit 3 **CSWEN:** Clock Switch Enable bit
1 = Writing to NOSC and NDIV is allowed
0 = The NOSC and NDIV bits cannot be changed by user software

bit 2 **Unimplemented:** Read as ‘1’

bit 1 **PR1WAY:** PRLOCKED One-Way Set Enable bit
1 = PRLOCKED bit can be cleared and set only once; Priority registers remain locked after one clear/set cycle
0 = PRLOCKED bit can be set and cleared multiple times (subject to the unlock sequence)

bit 0 **$\overline{\text{CLKOUTEN}}$:** Clock Out Enable bit
If FEXTOSC<2:0> = EC (high, mid or low) or Not Enabled:
1 = CLKOUT function is disabled; I/O or oscillator function on OSC2
0 = CLKOUT function is enabled; FOSC/4 clock appears at OSC2
Otherwise:
This bit is ignored.

FIGURE 6-2: LPBOR, BOR, POR RELATIONSHIP



REGISTER 9-29: IPR4: PERIPHERAL INTERRUPT PRIORITY REGISTER 4

R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
CLC1IP	CWG1IP	NCO1IP	—	CCP1IP	TMR2IP	TMR1GIP	TMR1IP
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 **CLC1IP:** CLC1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **CWG1IP:** CWG1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **NCO1IP:** NCO1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **CCP1IP:** CCP1 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 2 **TMR2IP:** TMR2 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **TMR1GIP:** TMR1 Gate Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **TMR1IP:** TMR1 Interrupt Priority bit
1 = High priority
0 = Low priority

10.2.1 WAKE-UP FROM SLEEP

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

1. External Reset input on $\overline{\text{MCLR}}$ pin, if enabled
2. BOR Reset, if enabled
3. Low-Power Brown-Out Reset (LPBOR), if enabled
4. POR Reset
5. Windowed Watchdog Timer, if enabled
6. All interrupt sources except clock switch interrupt can wake up the part.

The first five events will cause a device Reset. The last one event is considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 6.13 "Power Control (PCON0/PCON1) Register"**.

When the `SLEEP` instruction is being executed, the next instruction (PC + 2) is prefetched. For the device to wake-up through an interrupt event, the corresponding Interrupt Enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the `SLEEP` instruction. If the GIE bit is enabled, the device executes the instruction after the `SLEEP` instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

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

Upon a wake from a Sleep event, the core will wait for a combination of three conditions before beginning execution. The conditions are:

- PFM Ready
- COSC-Selected Oscillator Ready
- BOR Ready (unless BOR is disabled)

10.2.2 WAKE-UP USING INTERRUPTS

When any interrupt source, with the exception of the clock switch interrupt, has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a `SLEEP` instruction
  - `SLEEP` instruction will execute as a `NOP`
  - WDT and WDT prescaler will not be cleared
  - $\overline{\text{TO}}$ bit of the STATUS register will not be set
  - $\overline{\text{PD}}$ bit of the STATUS register will not be cleared
- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction
  - `SLEEP` instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - $\overline{\text{TO}}$ bit of the STATUS register will be set
  - $\overline{\text{PD}}$ bit of the STATUS register will be cleared

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the $\overline{\text{PD}}$ bit. If the $\overline{\text{PD}}$ bit is set, the `SLEEP` instruction was executed as a `NOP`.

17.8 Register Definitions: PPS Input Selection

REGISTER 17-1: xxxPPS: PERIPHERAL xxx INPUT SELECTION

U-0	U-0	R/W-m/u(1,3)	R/W-m/u(1)	R/W-m/u(1)	R/W-m/u(1)	R/W-m/u(1)	R/W-m/u(1)
—	—	xxxPPS<5:0>					
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	-n/n = Value at POR and BOR/Value at all other Resets
u = Bit is unchanged	x = Bit is unknown	q = value depends on peripheral
‘1’ = Bit is set	U = Unimplemented bit, read as ‘0’	m = value depends on default location for that input
‘0’ = Bit is cleared		

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-3 **xxxPPS<5:3>:** Peripheral xxx Input PORTx Pin Selection bits

See Table 17-1 for the list of available ports and default pin locations.
101 = PORTF(2)
100 = PORTE(3)
011 = PORTD(3)
010 = PORTC
001 = PORTB
000 = PORTA

bit 2-0 **xxxPPS<2:0>:** Peripheral xxx Input PORTx Pin Selection bits

111 = Peripheral input is from PORTx Pin 7 (Rx7)
110 = Peripheral input is from PORTx Pin 6 (Rx6)
101 = Peripheral input is from PORTx Pin 5 (Rx5)
100 = Peripheral input is from PORTx Pin 4 (Rx4)
011 = Peripheral input is from PORTx Pin 3 (Rx3)
010 = Peripheral input is from PORTx Pin 2 (Rx2)
001 = Peripheral input is from PORTx Pin 1 (Rx1)
000 = Peripheral input is from PORTx Pin 0 (Rx0)

Note 1: The Reset value ‘m’ of this register is determined by device default locations for that input.
2: Reserved on PIC18LF26/27/45/46/57K42 parts.
3: Reserved on PIC18LF26/27K42 parts.

REGISTER 21-4: TxGATE: TIMERx GATE ISM REGISTER

U-0	U-0	U-0	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	GSS<4:0>				
bit 7							bit 0

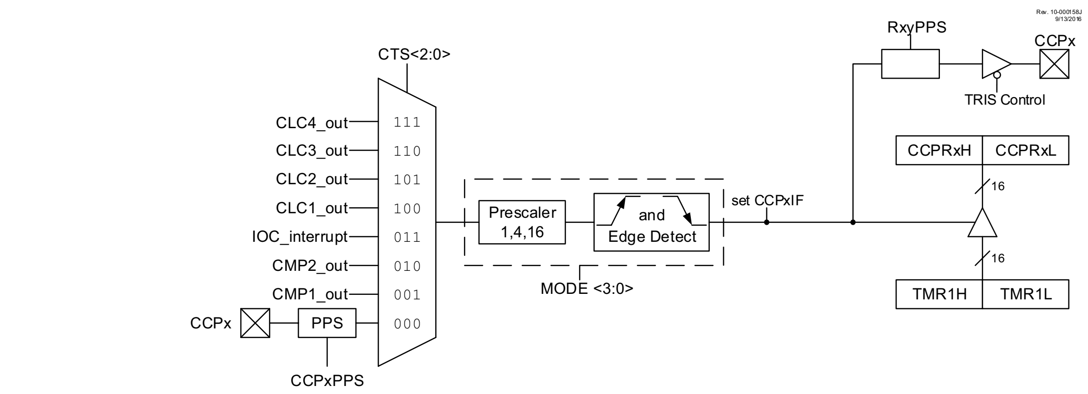
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	u = unchanged

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **GSS<4:0>:** Timerx Gate Source Selection bits

GSS	Timer1	Timer3	Timer5
	Gate Source	Gate Source	Gate Source
11111-11011	Reserved	Reserved	Reserved
11010	CLC4_out	CLC4_out	CLC4_out
11001	CLC3_out	CLC3_out	CLC3_out
11000	CLC2_out	CLC2_out	CLC2_out
10111	CLC1_out	CLC1_out	CLC1_out
10110	ZCDOUT	ZCDOUT	ZCDOUT
10101	CMP2OUT	CMP2OUT	CMP2OUT
10100	CMP1OUT	CMP1OUT	CMP1OUT
10011	NCO1OUT	NCO1OUT	NCO1OUT
10010-10001	Reserved	Reserved	Reserved
10000	PWM8OUT	PWM8OUT	PWM8OUT
01111	PWM7OUT	PWM7OUT	PWM7OUT
01110	PWM6OUT	PWM6OUT	PWM6OUT
01101	PWM5OUT	PWM5OUT	PWM5OUT
01100	CCP4OUT	CCP4OUT	CCP4OUT
01011	CCP3OUT	CCP3OUT	CCP3OUT
01010	CCP2OUT	CCP2OUT	CCP2OUT
01001	CCP1OUT	CCP1OUT	CCP1OUT
01000	SMT1_match	SMT1_match	SMT1_match
00111	TMR6OUT (postscaled)	TMR6OUT (postscaled)	TMR6OUT (postscaled)
00110	TMR5 overflow	TMR5 overflow	Reserved
00101	TMR4OUT (postscaled)	TMR4OUT (postscaled)	TMR4OUT (postscaled)
00100	TMR3 overflow	Reserved	TMR3 overflow
00011	TMR2OUT (postscaled)	TMR2OUT (postscaled)	TMR2OUT (postscaled)
00010	Reserved	TMR1 overflow	TMR1 overflow
00001	TMR0 overflow	TMR0 overflow	TMR0 overflow
00000	Pin selected by T1GPPS	Pin selected by T3GPPS	Pin selected by T5GPPS

FIGURE 23-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



25.1 SMT Operation

The core of the module is the 24-bit counter, SMT1TMR combined with a complex data acquisition front-end. Depending on the mode of operation selected, the SMT can perform a variety of measurements summarized in Table 25-1.

25.1.1 CLOCK SOURCES

Clock sources available to the SMT include:

- FOSC
- FOSC/4
- HFINTOSC 16 MHz
- LFINTOSC
- MFINTOSC 31.25 kHz

The SMT clock source is selected by configuring the CSEL<2:0> bits in the SMT1CLK register. The clock source can also be prescaled using the PS<1:0> bits of the SMT1CON0 register. The prescaled clock source is used to clock both the counter and any synchronization logic used by the module.

25.1.2 PERIOD MATCH INTERRUPT

Similar to other timers, the SMT triggers an interrupt when SMT1TMR rolls over to '0'. This happens when SMT1TMR = SMT1PR, regardless of mode. Hence, in any mode that relies on an external signal or a window to reset the timer, proper operation requires that SMT1PR be set to a period larger than that of the expected signal or window.

25.2 Basic Timer Function Registers

The SMT1TMR time base and the SMT1CPW/SMT1PR/SMT1CPR buffer registers serve several functions and can be manually updated using software.

25.2.1 TIME BASE

The SMT1TMR is the 24-bit counter that is the center of the SMT. It is used as the basic counter/timer for measurement in each of the modes of the SMT. It can be reset to a value of 24'h00_0000 by setting the RST bit of the SMT1STAT register. It can be written to and read from software, but it is not guarded for atomic access, therefore reads and writes to the SMT1TMR should only be made when the GO = 0, or the software should have other measures to ensure integrity of SMT1TMR reads/writes.

25.2.2 PULSE-WIDTH LATCH REGISTERS

The SMT1CPW registers are the 24-bit SMT pulse-width latch. They are used to latch in the value of the SMT1TMR when triggered by various signals, which are determined by the mode the SMT is currently in. The SMT1CPW registers can also be updated with the current value of the SMT1TMR value by setting the CPWUP bit of the SMT1STAT register.

25.2.3 PERIOD LATCH REGISTERS

The SMT1CPR registers are the 24-bit SMT period latch. They are used to latch in other values of the SMT1TMR when triggered by various other signals, which are determined by the mode the SMT is currently in.

The SMT1CPR registers can also be updated with the current value of the SMT1TMR value by setting the CPRUP bit in the SMT1STAT register.

25.3 Halt Operation

The counter can be prevented from rolling-over using the STP bit in the SMT1CON0 register. When halting is enabled, the period match interrupt persists until the SMT1TMR is reset (either by a manual Reset, **Section 25.2.1 "Time Base"**) or by clearing the GO bit of the SMT1CON1 register and writing the SMT1TMR values in software.

25.4 Polarity Control

The three input signals for the SMT have polarity control to determine whether or not they are active-high/positive edge or active-low/negative edge signals.

The following bits apply to Polarity Control:

- WSEL bit (Window Polarity)
- SSEL bit (Signal Polarity)
- CSEL bit (Clock Polarity)

These bits are located in the SMT1CON0 register.

25.5 Status Information

The SMT provides input status information for the user without requiring the need to deal with the polarity of the incoming signals.

25.5.1 WINDOW STATUS

Window status is determined by the WS bit of the SMT1STAT register. This bit is only used in Windowed Measure, Gated Counter and Gated Window Measure modes, and is only valid when TS = 1, and will be delayed in time by synchronizer delays in non-Counter modes.

25.5.2 SIGNAL STATUS

Signal status is determined by the AS bit of the SMT1STAT register. This bit is used in all modes except Window Measure, Time of Flight and Capture modes, and is only valid when TS = 1, and will be delayed in time by synchronizer delays in non-Counter modes.

25.5.3 GO STATUS

Timer run status is determined by the TS bit of the SMT1STAT register, and will be delayed in time by synchronizer delays in non-Counter modes.

30.0 DATA SIGNAL MODULATOR (DSM) MODULE

The Data Signal Modulator (DSM) is a peripheral which allows the user to mix a data stream, also known as a modulator signal, with a carrier signal to produce a modulated output.

Both the carrier and the modulator signals are supplied to the DSM module either internally, from the output of a peripheral, or externally through an input pin.

The modulated output signal is generated by performing a logical “AND” operation of both the carrier and modulator signals and then provided to the MDOUT pin.

The carrier signal is comprised of two distinct and separate signals. A carrier high (CARH) signal and a carrier low (CARL) signal. During the time in which the modulator (MOD) signal is in a logic high state, the DSM mixes the carrier high signal with the modulator signal. When the modulator signal is in a logic low state, the DSM mixes the carrier low signal with the modulator signal.

Using this method, the DSM can generate the following types of Key Modulation schemes:

- Frequency-Shift Keying (FSK)
- Phase-Shift Keying (PSK)
- On-Off Keying (OOK)

Additionally, the following features are provided within the DSM module:

- Carrier Synchronization
- Carrier Source Polarity Select
- Programmable Modulator Data
- Modulated Output Polarity Select
- Peripheral Module Disable, which provides the ability to place the DSM module in the lowest power consumption mode

Figure 30-1 shows a Simplified Block Diagram of the Data Signal Modulator peripheral.

REGISTER 30-3: MD1CARH: MODULATION HIGH CARRIER CONTROL REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	CH<4:0>[(1)]				
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 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **CH<4:0>:** Modulator Carrier High Selection bits[(1)]
See Table 30-1 for signal list

Note 1: Unused selections provide an input value.

REGISTER 30-4: MD1CARL: MODULATION LOW CARRIER CONTROL REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	CL<4:0>[(1)]				
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 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **CL<4:0>:** Modulator Carrier Low Input Selection bits[(1)]
See Table 30-1 for signal list

Note 1: Unused selections provide a zero as the input value.

TABLE 30-1: MD1CARH/MD1CARL SELECTION MUX CONNECTIONS

MD1CARH			MD1CARL		
CH<4:0>		Connection	CL<4:0>		Connection
11111-10011	31-19	Reserved	11111-10011	31-19	Reserved
10010	18	CLC4OUT	10010	18	CLC4OUT
10001	17	CLC3OUT	10001	17	CLC3OUT
10000	16	CLC2OUT	10000	16	CLC2OUT
01111	15	CLC1OUT	01111	15	CLC1OUT
01110	14	NCO1OUT	01110	14	NCO1OUT
01101-01100	13-12	Reserved	01101-01100	13-12	Reserved
01011	11	PWM8 OUT	01011	11	PWM8 OUT
01010	10	PWM7 OUT	01010	10	PWM7 OUT
01001	9	PWM6 OUT	01001	9	PWM6 OUT
01000	8	PWM5 OUT	01000	8	PWM5 OUT
00111	7	CCP4 OUT	00111	7	CCP4 OUT
00110	6	CCP3 OUT	00110	6	CCP3 OUT
00101	5	CCP2 OUT	00101	5	CCP2 OUT
00100	4	CCP1 OUT	00100	4	CCP1 OUT
00011	3	CLKREF output	00011	3	CLKREF output
00010	2	HFINTOSC	00010	2	HFINTOSC
00001	1	FOSC (system clock)	00001	1	FOSC (system clock)
00000	0	Pin selected by MD1CARHPPS	00000	0	Pin selected by MD1CARLPPS

REGISTER 30-5: MD1SRC: MODULATION SOURCE CONTROL REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	MS<4:0>				
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 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **MS<4:0>:** Modulator Source Selection bits[(1)]
See Table 30-2 for signal list

Note 1: Unused selections provide a zero as the input value.

31.4 DMX Mode (UART1 only)

DMX is a protocol used in stage and show equipment. This includes lighting, fog machines, motors, etc. The protocol consists of a controller that sends out commands, and receiver such as theater lights that receive these commands. DMX protocol is usually unidirectional, but can be a bidirectional protocol in either Half or Full-duplex modes. An example of Half-duplex mode is the RDM (Remote Device Management) protocol that sits on DMX512A. The controller transmits commands and the receiver receives them. Also there are no error conditions or re-transmit mechanisms.

DMX, or DMX512A as it is known, consists of a “Universe” of 512 channels. This means that one controller can output up to 512 bytes on a single DMX link. Each equipment on the line is programmed to listen to a consecutive sequence of one or more of these bytes.

For example, a fog machine connected to one of the universes may be programmed to receive one byte, starting at byte number 10, and a lighting unit may be programmed to receive four bytes starting at byte number 22.

31.4.1 DMX CONTROLLER

DMX Controller mode is configured with the following settings:

- MODE<3:0> = 1010
- TXEN = 1
- RXEN = 0
- TXPOL = 0
- UxP1 = One less than the number of bytes to transmit (excluding the Start code)
- UxBRGH:L = Value to achieve 250K baud rate
- STP<1:0> = 10 for 2 Stop bits
- RxyPPS = TX pin output code
- ON = 1

Each DMX transmission begins with a Break followed by a byte called the ‘Start Code’. The width of the BREAK is fixed at 25 bit times. The Break is followed by a “Mark After Break” (MAB) Idle period. After this Idle period, the 1st through ‘n’th byte is transmitted, where ‘n-1’ is the value in UxP1. See Figure 31-6.

Software sends the Start Code and the ‘n’ data bytes by writing the UxTXB register with each byte to be sent in the desired order. A UxTXIF value of ‘1’ indicates when the UxTXB is ready to accept the next byte.

The internal byte counter is not accessible to software. Software needs to keep track of the number of bytes written to UxTXB to ensure that no more and no less than ‘n’ bytes are sent because the DMX state machine will automatically insert a Break and reset its internal counter after ‘n’ bytes are written. One way to ensure synchronization between hardware and software is to toggle TXEN after the last byte of the universe is completely free of the transmit shift register as indicated by the TXMTIF bit.

31.4.2 DMX RECEIVER

DMX Receiver mode is configured with the following settings:

- MODE<3:0> = 1010
- TXEN = 0
- RXEN = 1
- RXPOL = 0
- UxP2 = number of first byte to receive
- UxP3 = number of last byte to receive
- UxBRGH:L = Value to achieve 250K baud rate
- STP<1:0> = 10 for 2 Stop bits
- ON = 1
- UxRXPPS = code for desired input pin
- Input pin ANSEL bit = 0

When configured as DMX Receiver, the UART listens for a Break character that is at least 23 bit periods wide. If the Break is shorter than 23 bit times, the Break is ignored and the DMX state machine remains in Idle mode. Upon receiving the Break, the DMX counters will be reset to align with the incoming data stream. Immediately after the Break, the UART will see the “Mark after Break” (MAB). This space is ignored by the UART. The Start Code follows the MAB and will always be stored in the receive FIFO.

After the Start Code, the 1st through 512th byte will be received, but not all of them are stored in the receive FIFO. The UART ignores all received bytes until the ones of interest are received. This is done using the UxP2 and UxP3 registers. The UxP2 register holds the value of the byte number to start the receive process. The byte counter starts at 0 for the first byte after the Start Code. For example, to receive four bytes starting at the 10th byte after the Start Code, write 009h (9 decimal) to UxP2H:L and 00Ch (12 decimal) to UxP3H:L. The receive FIFO is only 2 bytes deep, therefore the bytes must be retrieved by reading UxRXB as they come in to avoid a receive FIFO overrun condition.

Typically two Stop bits are inserted between bytes. If either Stop bit is detected as a ‘0’ then the framing error for that byte will be set.

Since the DMX sequence always starts with a Break, the software can verify that it is in sync with the sequence by monitoring the RXBKIF flag to ensure that the next byte received after the RXBKIF is processed as the Start Code and subsequent bytes are processed as the expected data.

32.0 SERIAL PERIPHERAL INTERFACE (SPI) MODULE

32.1 SPI Module Overview

The SPI (Serial Peripheral Interface) module is a synchronous serial data communication bus that operates in Full-Duplex mode. Devices communicate in a master/slave environment where the master device initiates the communication. A slave device is controlled through a Chip Select known as Slave Select. Example slave devices include serial EEPROMs, shift registers, display drivers, A/D converters, or another PIC® device.

The SPI bus specifies four signal connections:

- Serial Clock (SCK)
- Serial Data Out (SDO)
- Serial Data IN (SDI)
- Slave Select ($\overline{\text{SS}}$)

The SPI interface supports the following modes and features:

- Master mode
- Slave mode
- Clock Polarity and Edge Select
- SDI, SDO, and SS Polarity Control
- Separate Transmit and Receive Enables
- Slave Select Synchronization
- Daisy-chain connection of slave devices
- Separate Transmit and Receive Buffers with 2-byte FIFO and DMA capabilities

Figure 32-1 shows the block diagram of the SPI module.

32.3.3 TRANSMIT AND RECEIVE FIFOS

The transmission and reception of data from the SPI module is handled by two FIFOs, one for reception and one for transmission (addressed by the SFRs SPIxRXB and SPIxTXB, respectively.). The TXFIFO is written by software and is read by the SPI module to shift the data onto the SDO pin. The RXFIFO is written by the SPI module as it shifts in the data from the SDI pin and is read by software. Setting the CLRBF bit of SPIxSTATUS resets the occupancy for both FIFOs, emptying both buffers. The FIFOs are also reset by disabling the SPI module.

Note: TXFIFO occupancy and RXFIFO occupancy simply refer to the number of bytes that are currently being stored in each FIFO. These values are used in this chapter to illustrate the function of these FIFOs and are not directly accessible through software.

The SPIxRXB register addresses the receive FIFO and is read-only. Reading from this register will read from the first FIFO location that was written to by hardware and decrease the RXFIFO occupancy. If the FIFO is empty, reading from this register will instead return a value of zero and set the RXRE (Receive Buffer Read Error) bit of the SPIxSTATUS register. The RXRE bit must then be cleared in software in order to properly reflect the status of the read error. When RXFIFO is full, the RXBF bit of the SPIxSTATUS register will be set. When the device receives data on the SDI pin, the receive FIFO may be written to by hardware and the occupancy increased, depending on the mode and receiver settings, as summarized in Table 32-1.

The SPIxTXB register addresses the transmit FIFO and is write-only. Writing to the register will write to the first empty FIFO location and increase the occupancy. If the FIFO is full, writing to this register will not affect the data and will set the TXWE bit of the SPIxSTATUS register. When the TXFIFO is empty, the TXBE bit of SPIxSTATUS will be set. When a data transfer occurs, data may be read from the first FIFO location written to and the occupancy decreases, depending on mode and transmitter settings, as summarized in Table 32-1 and **Section 32.6.1 “Slave Mode Transmit options”**.

32.3.4 LSB VS. MSB-FIRST OPERATION

Typically, SPI communication is output Most-Significant bit first, but some devices/buses may not conform to this standard. In this case, the LSBF bit may be used to alter the order in which bits are shifted out during the data exchange. In both Master and Slave mode, the LSBF bit of SPIxCON0 controls if data is shifted MSb or LSb first. Clearing the bit (default) configures the data to transfer MSb first, which is traditional SPI operation, while setting the bit configures the data to transfer LSb first.

32.3.5 INPUT AND OUTPUT POLARITY BITS

SPIxCON1 has three bits that control the polarity of the SPI inputs and outputs. The SDIP bit controls the polarity of the SDI input, the SDOP bit controls the polarity of the SDO output, and the SSP bit controls the polarity of both the slave $\overline{\text{SS}}$ input and the master SS output. For all three bits, when the bit is clear, the input or output is active-high, and when the bit is set, the input or output is active-low. When the EN bit of SPIxCON0 is cleared, SS(out) and SCK(out) both revert to the inactive state dictated by their polarity bits. The SDO output state when the EN bit of SPIxCON0 is cleared is determined by several factors.

- When the associated TRIS bit for the SDO pin is cleared, and the SPI goes idle after a transmission, the SDO output will remain at the last bit level. The SDO pin will revert to the Idle state if EN is cleared.
- When the associated TRIS bit for the SDO pin is set, behavior varies in Slave and Master mode.
  - In Slave mode, the SDO pin tri-states when:
  - Slave Select is inactive,
  - the EN bit of SPIxCON0 is cleared, or when
  - the TXR bit of SPIxCON2 is cleared.
  - In Master mode, the SDO pin tri-states when TXR = `0`. When TXR = `1` and the SPI goes idle after a transmission, the SDO output will remain at the last bit level. The SDO pin will revert to the Idle state if EN is cleared.

32.8.3.4 Receiver Overflow and Transmitter Underflow Interrupts

The receiver overflow interrupt triggers if data is received when the RXFIFO is already full and RXR = `1`. In this case, the data will be discarded and the RXOIF bit will be set. The receiver overflow interrupt flag is the RXOIF bit of SPIxINTF. The receiver overflow interrupt enable bit is the RXOIE bit of SPIxINTE.

The Transmitter Underflow interrupt flag triggers if a data transfer begins when the TXFIFO is empty and TXR = `1`. In this case, the most recently received data will be transmitted and the TXUIF bit will be set. The transmitter underflow interrupt flag is the TXUIF bit of SPIxINTF. The transmitter underflow interrupt enable bit is the TXUIE bit of SPIxINTE.

Both of these interrupts will only occur in Slave mode, as Master mode will not allow the RXFIFO to overflow or the TXFIFO to underflow.

33.4.3.3 Slave operation in 10-bit Addressing Mode

In 10-bit Addressing mode, the first received byte is compared to the binary value of '11110A9A80'. A9 and A8 are the two MSb of the 10-bit address. The first byte is compared with the value in I2CxADR1 and I2CxADR3 registers. After the high byte is acknowledged, the low address byte is clocked in and all eight bits are compared to the low address value in the I2CxADR0 and I2CxADR2 registers. A high and low address match as a write request is required at the start of all 10-bit addressing communication. To initiate a read, the Master needs to issue a Restart once the slave is addressed and clock in the high address with the R/W bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. The SMA (slave active) bit is set only when both the high and low address bytes match.

Note: All seven bits of the received high address are compared to the values in the I2CxADR1 and I2CxADR3 registers. The five-bit '11110' high address format is not enforced by module hardware. It is up to the user to configure these bits correctly.

33.4.3.4 Slave Reception (10-bit Addressing Mode)

This section describes the sequence of events for the I^2C module configured as an I^2C slave in 10-bit Addressing mode and is receiving data. Figure 33-11 is 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. If Start condition interrupt is enabled (SCIE bit is set), generic interrupt I2CxIF is set.
2. Master transmits high address byte with R/W = 0.
3. The received high address is compared with the values in I2CxADR1 and I2CxADR3 registers.
4. If high address matches; R/W is copied to R/W bit, D/A bit is cleared, high address data is copied to I2CxADB1. If the address does not match; module becomes idle.
5. If Address hold interrupt is enabled (ADRIE = 1), CSTR is set. I2CxIF is set.
6. Slave software can read high address from I2CxADB1 and set/clear ACKDT before releasing SCL.
7. ACKDT value is copied out to SDA for ACK pulse. SCL line is released by clearing CSTR.
8. Master sends ninth SCL pulse for ACK
9. Slave can force a NACK at this point due to previous error not being cleared. E.g. Receive buffer overflow or transmit buffer underflow errors. In these cases the Slave hardware forces a NACK and the module becomes idle.
10. Master transmits low address data byte
11. If the low address matches; SMA is set, ADRIF is set, R/W is copied to R/W bit, D/A bit is cleared, low address data is copied to I2CxADB0, and ACTDT is copied to SDA. If the address does not match; module becomes idle.
12. If address hold interrupt is enabled, the CSTR bit is set as mentioned in step 6. Slave software can read low address byte from I2CxADB0 register and change ACKDT value before releasing SCL.
13. Master sends ninth SCL pulse for ACK.
14. If the Acknowledge interrupt and hold is enabled (ACKTIE = 1), CSTR is set, I2CxIF is set.
15. Slave software can read address from I2CxADB0 and I2CxADB1 registers and change the value of ACKDT before releasing SCL by clearing CSTR.
16. Master sends first seven SCL pulses of the data byte or a Stop condition (in the case of NACK).
17. If Stop condition; PCIF in I2CxPIR register is set, module becomes idle.
18. 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.
19. Master sends eighth SCL pulse of the data byte. D/A bit is set, WRIF is set. I2CxRXB is loaded with new data, RXBF bit is set.
20. 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.
21. If I2CxCNT = 0, the ACKCNT value is output to the SDA; else, the ACKDT value is used and the value of I2CxCNT is decremented.
22. Master sends SCL pulse for ACK.
23. If I2CxCNT = 0, CNTIF is set.
24. If the response was a NACK; NACKIF is set, module becomes idle.
25. If ACKTIE = 1, CSTR is set, I2CxIF is set. Slave software can read data from I2CxRXB clearing RXBF; before releasing SCL by clearing CSTR
26. Go to step 16.

33.7 Register Definitions: I^2C Control

This section defines all the registers associated with the control and status of the I^2C bus.

REGISTER 33-1: I2CxCON0: I^2C CONTROL REGISTER 0

R/W-0	R/W-0	R/W/HC/HS-0	R/C/HS/HC-0	R-0	R/W-0	R/W-0	R/W-0
EN[1,2]	RSEN	S	CSTR[3]	MDR	MODE <2:0>		
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	HS = Hardware set HC = Hardware clear

bit 7 **EN:** I^2C Module Enable bit
1 = Enables the I^2C module[1,2]
0 = Disables the I^2C module.

bit 6 **RSEN:** Restart Enable bit (Only mode<2:0> = 1xx)
1 = When (I2CCNT = 0 or ACKSTAT = 1), on 9th falling SCL sets MDR.
0 = When (I2CCNT = 0 or ACKSTAT = 1), on 9th falling SCL; master shifts out a Stop condition

bit 5 **S:** Master Start/Restart bit (Only Mode<2:0> = 1xx)
<u>When MMA = 0</u>
1 = Set by user set of START bit or write to I2CTXB, waits for BFRE = 1 to begin with a Start
0 = Cleared by hardware after sending Start
<u>When (MMA = 1 & MDR = 1 & pause_for_Restart)</u>
1 = Set by user set of START bit or write to I2CTXB, resumes communication with a Restart
0 = Cleared by hardware after sending Restart
<u>Else</u> - Writes to I2CTXB or set has no effect on Start bit

bit 4 **CSTR:** Slave Clock Stretching bit [3]
1 = Clock is held low (clock stretching)
0 = Enable clocking, SCL control is released

<u>SMA = 1 and RXBF = 1</u>[6]
- Set by hardware on 7th falling SCL edge
- User must read byte I2CRXB to release SCL
<u>SMA = 1 and TXBE = 1 and I2CCNT!= 0</u>
- Set by hardware on 8th falling SCL edge
- User must write byte to I2CTXB to release SCL
<u>when ADRIE is set</u> [4]
- Set by hardware on 8th falling SCL edge of matching received address
- User must clear CSTR to release SCL
<u>SMA = 1 & WRIE = 1</u>
- Set by hardware on 8th falling SCL edge of received data byte
- User must clear CSTR to release SCL
<u>SMA = 1 & ACKTIE = 1</u>
- Set by hardware on 9th falling SCL edge
- User must clear CSTR to release SCL

TABLE 44-13: RESET, WDT, OSCILLATOR START-UP TIMER, POWER-UP TIMER, BROWN-OUT RESET AND LOW-POWER BROWN-OUT RESET SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
RST01*	T_{MCLR}	$\overline{MCLR}$ Pulse Width Low to ensure Reset	2	—	—	µs	
RST02*	T_{IOZ}	I/O high-impedance from Reset detection	—	—	2	µs	
RST03	T_{WDT}	Watchdog Timer Time-out Period	—	16	—	ms	1:512 Prescaler
RST04*	T_{PWRT}	Power-up Timer Period	—	1 16 64	—	ms ms ms	PWRTS = 00 PWRTS = 01 PWRTS = 10
RST05	T_{OST}	Oscillator Start-up Timer Period[1,2]	—	1024	—	T_{OSC}	
RST06	V_{BOR}	Brown-out Reset Voltage[4]	2.7 2.55 2.3 2.3 1.8	2.85 2.7 2.45 2.45 1.9	3.0 2.85 2.6 2.6 2.05	V V V V V	BORV = 00 BORV = 01 BORV = 10 BORV = 11 (PIC18Fxxx) BORV = 11 (PIC18LFxxx)
RST07	V_{BORHYS}	Brown-out Reset Hysteresis	—	40	—	mV	
RST08	T_{BORDC}	Brown-out Reset Response Time	—	3	—	µs	
RST09	V_{LPBOR}	Low-Power Brown-out Reset Voltage	1.8	1.9	2.5	V	PIC18LFXXX only

* 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: By design, the Oscillator Start-up Timer (OST) counts the first 1024 cycles, independent of frequency.

2: To ensure these voltage tolerances, V_{DD} and V_{SS} must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

TABLE 44-14: HIGH/LOW-VOLTAGE DETECT CHARACTERISTICS

Standard Operating Conditions (unless otherwise stated)

Param. No.	Symbol	Characteristic	Min.	Typ†	Max.	Units	Conditions
HLVD01	V_{DET}	Voltage Detection	—	1.85	—	V	HLVDSEL<3:0>=0000
			—	2.06	—	V	HLVDSEL<3:0>=0001
			—	2.26	—	V	HLVDSEL<3:0>=0010
			—	2.47	—	V	HLVDSEL<3:0>=0011
			—	2.57	—	V	HLVDSEL<3:0>=0100
			—	2.78	—	V	HLVDSEL<3:0>=0101
			—	2.88	—	V	HLVDSEL<3:0>=0110
			—	3.09	—	V	HLVDSEL<3:0>=0111
			—	3.40	—	V	HLVDSEL<3:0>=1000
			—	3.60	—	V	HLVDSEL<3:0>=1001
			—	3.71	—	V	HLVDSEL<3:0>=1010
			—	3.91	—	V	HLVDSEL<3:0>=1011
			—	4.12	—	V	HLVDSEL<3:0>=1100
			—	4.32	—	V	HLVDSEL<3:0>=1101
			—	4.63	—	V	HLVDSEL<3:0>=1110

TABLE 44-15: ANALOG-TO-DIGITAL CONVERTER (ADC) ACCURACY SPECIFICATIONS(1,2):

Operating Conditions (unless otherwise stated)
VDD = 3.0V, TA = 25°C, TAD = 1μs

Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
AD01	NR	Resolution	—	—	12	bit	
AD02	EIL	Integral Error	—	±0.1	±2.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD03	EDL	Differential Error	—	±0.1	±1.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD04	EOFF	Offset Error	—	0.5	6.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD05	EGN	Gain Error	—	±0.2	±6.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD06	VADREF	ADC Reference Voltage (ADREF+ - ADREF-)	1.8	—	VDD	V	
AD07	VAIN	Full-Scale Range	ADREF-	—	ADREF+	V	
AD08	ZAIN	Recommended Impedance of Analog Voltage Source	—	1	—	kΩ	
AD09	RVREF	ADC Voltage Reference Ladder Impedance	—	50	—	kΩ	**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: Total Absolute Error is the sum of the offset, gain and integral non-linearity (INL) errors.

2: The ADC conversion result never decreases with an increase in the input and has no missing codes.

3: This is the impedance seen by the VREF pads when the external reference pads are selected.