



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	48MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3.8K x 8
Voltage - Supply (Vcc/Vdd)	2.15V ~ 3.6V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f24j11-i-so

4.6.7 TYPICAL DEEP SLEEP SEQUENCE

This section gives the typical sequence for using the Deep Sleep mode. Optional steps are indicated, and additional information is given in notes at the end of the procedure.

1. Enable DSWDT (optional).⁽¹⁾
2. Configure DSWDT clock source (optional).⁽²⁾
3. Enable DSBOR (optional).⁽¹⁾
4. Enable RTCC (optional).⁽³⁾
5. Configure the RTCC peripheral (optional).⁽³⁾
6. Configure the ULPWU peripheral (optional).⁽⁴⁾
7. Enable the INT0 Interrupt (optional).⁽⁴⁾
8. Context save SRAM data by writing to the DSGPR0 and DSGPR1 registers (optional).
9. Set the REGSLP bit (WDTCON<7>) and clear the IDLEN bit (OSCCON<7>).
10. If using an RTCC alarm for wake-up, wait until the RTCSYNC (RTCCFG<4>) bit is clear.
11. Enter Deep Sleep mode by setting the DSEN bit (DSCONH<7>) and issuing a *SLEEP* instruction. These two instructions must be executed back to back.
12. Once a wake-up event occurs, the device will perform a POR reset sequence. Code execution resumes at the device's Reset vector.
13. Determine if the device exited Deep Sleep by reading the Deep Sleep bit, DS (WDTCON<3>). This bit will be set if there was an exit from Deep Sleep mode.
14. Clear the Deep Sleep bit, DS (WDTCON<3>).
15. Determine the wake-up source by reading the DSWAKEH and DSWAKEL registers.
16. Determine if a DSBOR event occurred during Deep Sleep mode by reading the DSBOR bit (DSCONL<1>).
17. Read the DSGPR0 and DSGPR1 context save registers (optional).
18. Clear the RELEASE bit (DSCONL<0>).

Note 1: DSWDT and DSBOR are enabled through the devices' Configuration bits. For more information, see **Section 26.1 "Configuration Bits"**.

2: The DSWDT and RTCC clock sources are selected through the devices' Configuration bits. For more information, see **Section 26.1 "Configuration Bits"**.

3: For more information, see **Section 17.0 "Real-Time Clock and Calendar (RTCC)"**.

4: For more information on configuring this peripheral, see **Section 4.7 "Ultra Low-Power Wake-up"**.

4.6.8 DEEP SLEEP FAULT DETECTION

If during Deep Sleep the device is subjected to unusual operating conditions, such as an Electrostatic Discharge (ESD) event, it is possible that the internal circuit states used by the Deep Sleep module could become corrupted. If this were to happen, the device may exhibit unexpected behavior, such as a failure to wake back up.

In order to prevent this type of scenario from occurring, the Deep Sleep module includes automatic self-monitoring capability. During Deep Sleep, critical internal nodes are continuously monitored in order to detect possible Fault conditions (which would not ordinarily occur). If a Fault condition is detected, the circuitry will set the DSFLT status bit (DSWAKEL<7>) and automatically wake the microcontroller from Deep Sleep, causing a POR Reset.

During Deep Sleep, the Fault detection circuitry is always enabled and does not require any specific configuration prior to entering Deep Sleep.

5.0 RESET

The PIC18F46J11 family of devices differentiates among various kinds of Reset:

- Power-on Reset (POR)
- MCLR Reset during normal operation
- MCLR Reset during power-managed modes
- Watchdog Timer (WDT) Reset (during execution)
- Configuration Mismatch (CM)
- Brown-out Reset (BOR)
- RESET Instruction
- Stack Full Reset
- Stack Underflow Reset
- Deep Sleep Reset

This section discusses Resets generated by MCLR, POR and BOR, and covers the operation of the various start-up timers.

For information on WDT Resets, see **Section 26.2 “Watchdog Timer (WDT)”**. For Stack Reset events, see **Section 6.1.4.4 “Stack Full and Underflow Resets”** and for Deep Sleep mode, see **Section 4.6 “Deep Sleep Mode”**.

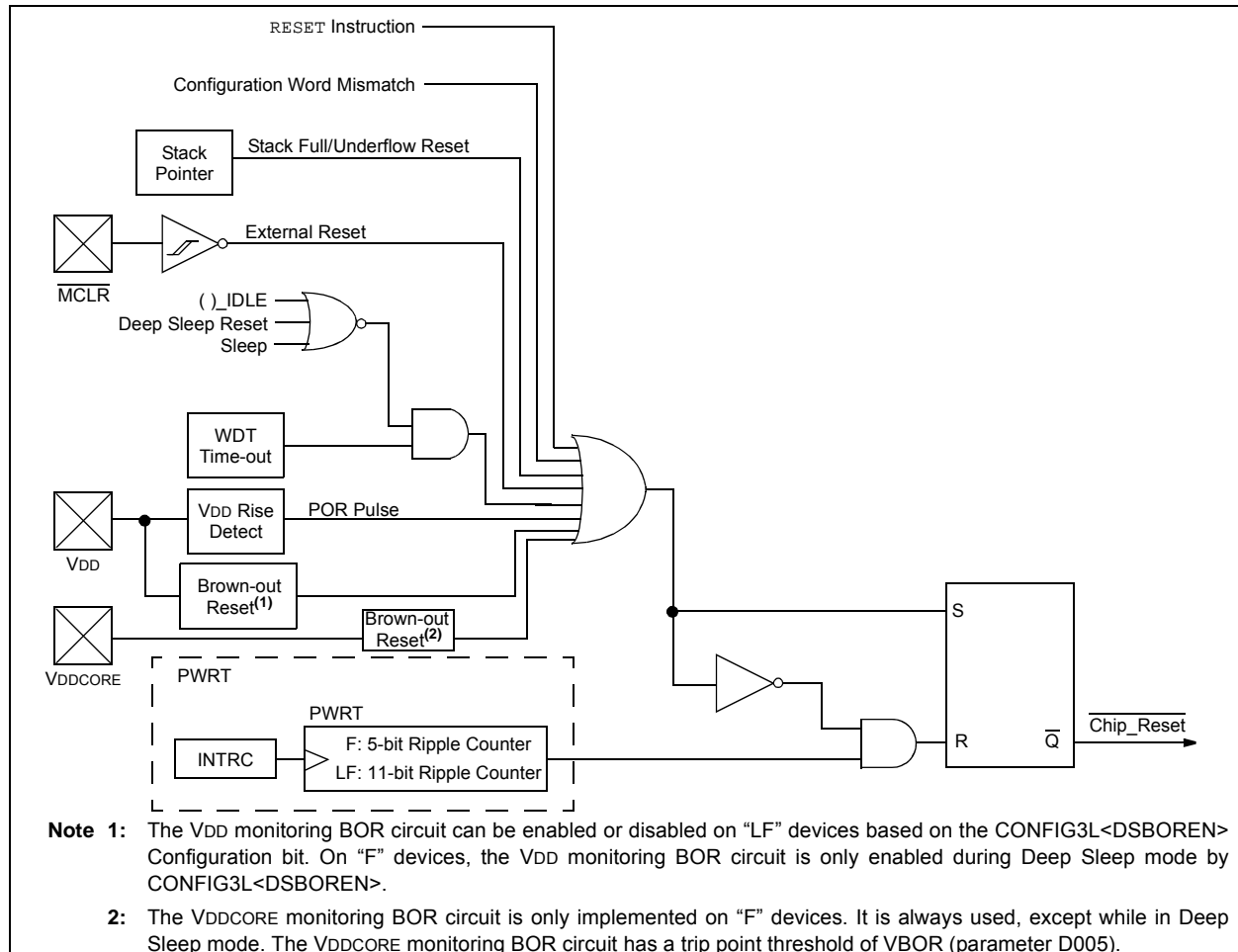
Figure 5-1 provides a simplified block diagram of the on-chip Reset circuit.

5.1 RCON Register

Device Reset events are tracked through the RCON register (Register 5-1). The lower five bits of the register indicate that a specific Reset event has occurred. In most cases, these bits can only be set by the event and must be cleared by the application after the event. The state of these flag bits, taken together, can be read to indicate the type of Reset that just occurred. This is described in more detail in **Section 5.7 “Reset State of Registers”**.

The ECON register also has a control bit for setting interrupt priority (IPEN). Interrupt priority is discussed in **Section 9.0 “Interrupts”**.

FIGURE 5-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



6.4 Data Addressing Modes

Note: The execution of some instructions in the core PIC18 instruction set is changed when the PIC18 extended instruction set is enabled. See **Section 6.6 “Data Memory and the Extended Instruction Set”** for more information.

While the program memory can be addressed in only one way, through the PC, information in the data memory space can be addressed in several ways. For most instructions, the addressing mode is fixed. Other instructions may use up to three modes, depending on which operands are used and whether or not the extended instruction set is enabled.

The addressing modes are:

- Inherent
- Literal
- Direct
- Indirect

An additional addressing mode, Indexed Literal Offset, is available when the extended instruction set is enabled (XINST Configuration bit = 1). Its operation is discussed in more detail in **Section 6.6.1 “Indexed Addressing with Literal Offset”**.

6.4.1 INHERENT AND LITERAL ADDRESSING

Many PIC18 control instructions do not need any argument at all; they either perform an operation that globally affects the device, or they operate implicitly on one register. This addressing mode is known as Inherent Addressing. Examples include *SLEEP*, *RESET* and *DAW*.

Other instructions work in a similar way, but require an additional explicit argument in the opcode. This is known as Literal Addressing mode, because they require some literal value as an argument. Examples include *ADDLW* and *MOVLW*, which respectively, add or move a literal value to the W register. Other examples include *CALL* and *GOTO*, which include a 20-bit program memory address.

6.4.2 DIRECT ADDRESSING

Direct Addressing specifies all or part of the source and/or destination address of the operation within the opcode itself. The options are specified by the arguments accompanying the instruction.

In the core PIC18 instruction set, bit-oriented and byte-oriented instructions use some version of Direct Addressing by default. All of these instructions include some 8-bit Literal Address as their LSB. This address specifies either a register address in one of the banks of data RAM (**Section 6.3.3 “General Purpose**

Register File”), or a location in the Access Bank (**Section 6.3.2 “Access Bank”**) as the data source for the instruction.

The Access RAM bit, ‘a’, determines how the address is interpreted. When ‘a’ is ‘1’, the contents of the BSR (**Section 6.3.1 “Bank Select Register”**) are used with the address to determine the complete 12-bit address of the register. When ‘a’ is ‘0’, the address is interpreted as being a register in the Access Bank. Addressing that uses the Access RAM is sometimes also known as Direct Forced Addressing mode.

A few instructions, such as *MOVFF*, include the entire 12-bit address (either source or destination) in their opcodes. In these cases, the BSR is ignored entirely.

The destination of the operation’s results is determined by the destination bit, ‘d’. When ‘d’ is ‘1’, the results are stored back in the source register, overwriting its original contents. When ‘d’ is ‘0’, the results are stored in the W register. Instructions without the ‘d’ argument have a destination that is implicit in the instruction; their destination is either the target register being operated on or the W register.

6.4.3 INDIRECT ADDRESSING

Indirect Addressing allows the user to access a location in data memory without giving a fixed address in the instruction. This is done by using File Select Registers (FSRs) as pointers to the locations to be read or written to. Since the FSRs are themselves located in RAM as SFRs, they can also be directly manipulated under program control. This makes FSRs very useful in implementing data structures such as tables and arrays in data memory.

The registers for Indirect Addressing are also implemented with Indirect File Operands (INDFs) that permit automatic manipulation of the pointer value with auto-incrementing, auto-decrementing or offsetting with another value. This allows for efficient code using loops, such as the example of clearing an entire RAM bank in Example 6-5. It also enables users to perform Indexed Addressing and other Stack Pointer operations for program memory in data memory.

EXAMPLE 6-5: HOW TO CLEAR RAM (BANK 1) USING INDIRECT ADDRESSING

	LFSR	FSR0, 0x100 ;	
NEXT	CLRF	POSTINC0	; Clear INDF
			; register then
			; inc pointer
	BTFSS	FSR0H, 1	; All done with
			; Bank1?
	BRA	NEXT	; NO, clear next
CONTINUE			; YES, continue

PIC18F46J11 FAMILY

6.6.1 INDEXED ADDRESSING WITH LITERAL OFFSET

Enabling the PIC18 extended instruction set changes the behavior of Indirect Addressing using the FSR2 register pair and its associated file operands. Under proper conditions, instructions that use the Access Bank, that is, most bit and byte-oriented instructions, can invoke a form of Indexed Addressing using an offset specified in the instruction. This special addressing mode is known as Indexed Addressing with Literal Offset, or Indexed Literal Offset mode.

When using the extended instruction set, this addressing mode requires the following:

- The use of the Access Bank is forced ('a' = 0); and
- The file address argument is less than or equal to 5Fh.

Under these conditions, the file address of the instruction is not interpreted as the lower byte of an address (used with the BSR in Direct Addressing) or as an 8-bit address in the Access Bank. Instead, the value is interpreted as an offset value to an Address Pointer specified by FSR2. The offset and the contents of FSR2 are added to obtain the target address of the operation.

6.6.2 INSTRUCTIONS AFFECTED BY INDEXED LITERAL OFFSET MODE

Any of the core PIC18 instructions that can use Direct Addressing are potentially affected by the Indexed Literal Offset Addressing mode. This includes all byte and bit-oriented instructions, or almost one-half of the standard PIC18 instruction set. Instructions that only use Inherent or Literal Addressing modes are unaffected.

Additionally, byte and bit-oriented instructions are not affected if they do not use the Access Bank (Access RAM bit is '1') or include a file address of 60h or above. Instructions meeting these criteria will continue to execute as before. A comparison of the different possible addressing modes when the extended instruction set is enabled is provided in Figure 6-9.

Those who desire to use byte or bit-oriented instructions in the Indexed Literal Offset mode should note the changes to assembler syntax for this mode. This is described in more detail in **Section 27.2.1 "Extended Instruction Syntax"**.

PIC18F46J11 FAMILY

NOTES:

PIC18F46J11 FAMILY

NOTES:

PIC18F46J11 FAMILY

13.7 Resetting Timer1 Using the ECCP Special Event Trigger

If ECCP1 or ECCP2 is configured to use Timer1 and to generate a Special Event Trigger in Compare mode (CCPxM<3:0> = 1011), this signal will reset Timer3. The trigger from ECCP2 will also start an A/D conversion if the A/D module is enabled (see **Section 18.3.4 “Special Event Trigger”** for more information).

The module must be configured as either a timer or a synchronous counter to take advantage of this feature. When used this way, the CCPxH:CCPxL register pair effectively becomes a Period register for Timer1.

If Timer1 is running in Asynchronous Counter mode, this Reset operation may not work.

In the event that a write to Timer1 coincides with a Special Event Trigger, the write operation will take precedence.

Note: The Special Event Trigger from the ECCPx module will not set the TMR1IF interrupt flag bit (PIR1<0>).

13.8 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using the Timer1 gate circuitry. This is also referred to as Timer1 gate count enable.

The Timer1 gate can also be driven by multiple selectable sources.

13.8.1 TIMER1 GATE COUNT ENABLE

The Timer1 Gate Enable mode is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 13-4 for timing details.

TABLE 13-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1G	Timer1 Operation
↑	0	0	Counts
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Counts

PIC18F46J11 FAMILY

15.1 Timer3 Gate Control Register

The Timer3 Gate Control register (T3GCON), provided in Register 14-2, is used to control the Timer3 gate.

REGISTER 15-2: T3GCON: TIMER3 GATE CONTROL REGISTER (ACCESS F97h)⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-x	R/W-0	R/W-0
TMR3GE	T3GPOL	T3GTM	T3GSPM	T3GGO/T3DONE	T3GVAL	T3GSS1	T3GSS0
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 **TMR3GE:** Timer3 Gate Enable bit
If TMR3ON = 0:
This bit is ignored.
If TMR3ON = 1:
1 = Timer3 counting is controlled by the Timer3 gate function
0 = Timer3 counts regardless of Timer3 gate function
- bit 6 **T3GPOL:** Timer3 Gate Polarity bit
1 = Timer3 gate is active-high (Timer3 counts when gate is high)
0 = Timer3 gate is active-low (Timer3 counts when gate is low)
- bit 5 **T3GTM:** Timer3 Gate Toggle Mode bit
1 = Timer3 Gate Toggle mode is enabled
0 = Timer3 Gate Toggle mode is disabled and toggle flip-flop is cleared
Timer3 gate flip-flop toggles on every rising edge.
- bit 4 **T3GSPM:** Timer3 Gate Single Pulse Mode bit
1 = Timer3 Gate Single Pulse mode is enabled and is controlling Timer3 gate
0 = Timer3 Gate Single Pulse mode is disabled
- bit 3 **T3GGO/T3DONE:** Timer3 Gate Single Pulse Acquisition Status bit
1 = Timer3 gate single pulse acquisition is ready, waiting for an edge
0 = Timer3 gate single pulse acquisition has completed or has not been started
This bit is automatically cleared when T3GSPM is cleared.
- bit 2 **T3GVAL:** Timer3 Gate Current State bit
Indicates the current state of the Timer3 gate that could be provided to TMR3H:TMR3L. Unaffected by Timer3 Gate Enable bit (TMR3GE).
- bit 1-0 **T3GSS<1:0>:** Timer3 Gate Source Select bits
10 = TMR2 to match PR2 output
01 = Timer0 overflow output
00 = Timer3 gate pin (T3G)

Note 1: Programming the T3GCON prior to T3CON is recommended.

15.3 Timer3 16-Bit Read/Write Mode

Timer3 can be configured for 16-bit reads and writes (see **Section 15.3 “Timer3 16-Bit Read/Write Mode”**). When the RD16 control bit (T3CON<1>) is set, the address for TMR3H is mapped to a buffer register for the high byte of Timer3. A read from TMR3L will load the contents of the high byte of Timer3 into the Timer3 High Byte Buffer register. This provides the user with the ability to accurately read all 16 bits of Timer3 without having to determine whether a read of the high byte, followed by a read of the low byte, has become invalid due to a rollover between reads.

A write to the high byte of Timer3 must also take place through the TMR3H Buffer register. The Timer3 high byte is updated with the contents of TMR3H when a write occurs to TMR3L. This allows a user to write all 16 bits to both the high and low bytes of Timer3 at once.

The high byte of Timer3 is not directly readable or writable in this mode. All reads and writes must take place through the Timer3 High Byte Buffer register.

Writes to TMR3H do not clear the Timer3 prescaler. The prescaler is only cleared on writes to TMR3L.

15.4 Using the Timer1 Oscillator as the Timer3 Clock Source

The Timer1 internal oscillator may be used as the clock source for Timer3. The Timer1 oscillator is enabled by setting the T1OSCEN (T1CON<3>) bit. To use it as the Timer3 clock source, the TMR3CS bit must also be set. As previously noted, this also configures Timer3 to increment on every rising edge of the oscillator source.

The Timer1 oscillator is described in **Section 13.0 “Timer1 Module”**.

15.5 Timer3 Gate

Timer3 can be configured to count freely, or the count can be enabled and disabled using Timer3 gate circuitry. This is also referred to as Timer3 gate count enable.

Timer3 gate can also be driven by multiple selectable sources.

15.5.1 TIMER3 GATE COUNT ENABLE

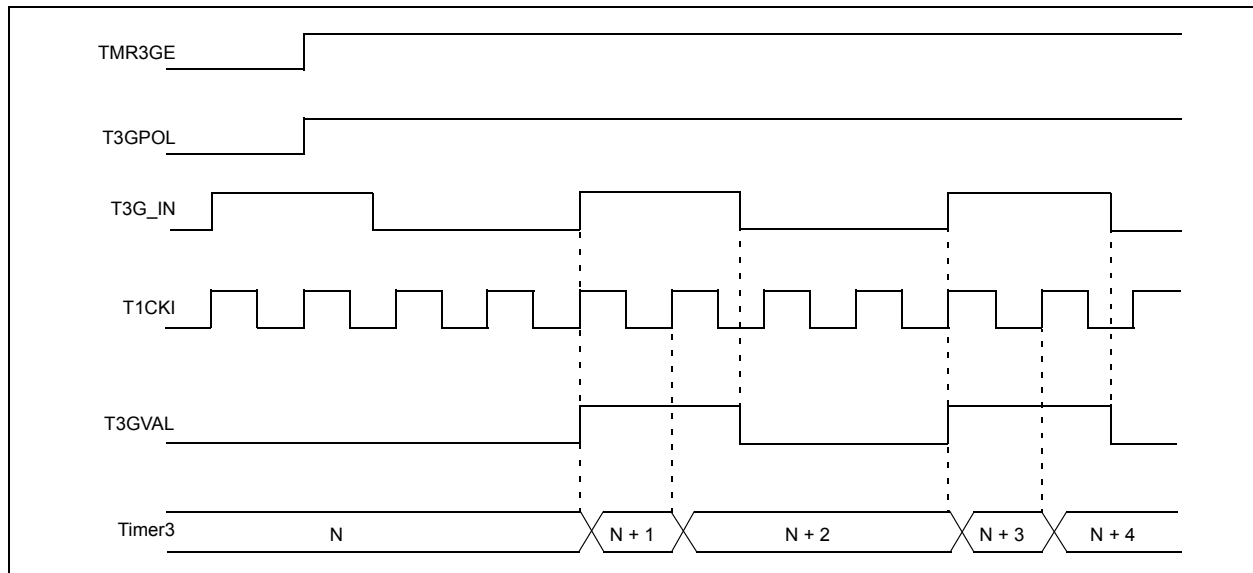
The Timer3 Gate Enable mode is enabled by setting the TMR3GE bit of the T3GCON register. The polarity of the Timer3 Gate Enable mode is configured using the T3GPOL bit of the T3GCON register.

When Timer3 Gate Enable mode is enabled, Timer3 will increment on the rising edge of the Timer3 clock source. When Timer3 Gate Enable mode is disabled, no incrementing will occur and Timer3 will hold the current count. See Figure 15-2 for timing details.

TABLE 15-1: TIMER3 GATE ENABLE SELECTIONS

T3CLK	T3GPOL	T3G	Timer3 Operation
↑	0	0	Counts
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Counts

FIGURE 15-2: TIMER3 GATE COUNT ENABLE MODE



PIC18F46J11 FAMILY

REGISTER 17-16: ALRMWD: ALARM WEEKDAY VALUE REGISTER (ACCESS F8Fh, PTR 01b)⁽¹⁾

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY2	WDAY1	WDAY0
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 7-3 **Unimplemented:** Read as '0'

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit bits
Contains a value from 0 to 6.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 17-17: ALRMHR: ALARM HOURS VALUE REGISTER (ACCESS F8Eh, PTR 01b)⁽¹⁾

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
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

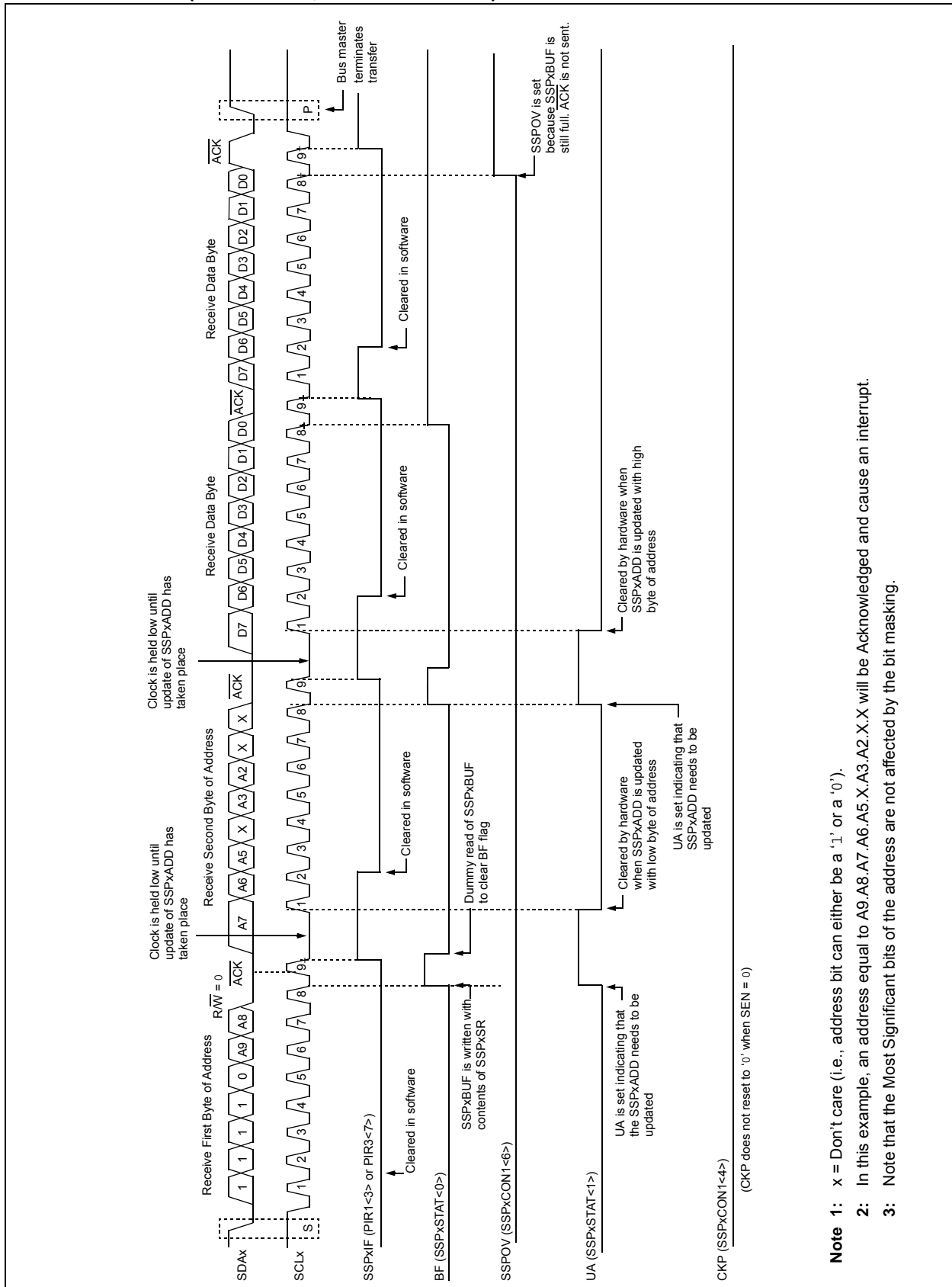
bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **HRTEN<1:0>:** Binary Coded Decimal Value of Hour's Tens Digit bits
Contains a value from 0 to 2.

bit 3-0 **HRONE3:HRONE0:** Binary Coded Decimal Value of Hour's Ones Digit bits
Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

FIGURE 19-11: I²C™ SLAVE MODE TIMING WITH SEN = 0 AND ADMSK<5:1> = 01001 (RECEPTION, 10-BIT ADDRESS)



21.7 A/D Converter Calibration

The A/D Converter in the PIC18F46J11 family of devices includes a self-calibration feature, which compensates for any offset generated within the module. The calibration process is automated and is initiated by setting the ADCAL bit (ADCON1<6>). The next time the GO/DONE bit is set, the module will perform a “dummy” conversion (that is, with reading none of the input channels) and store the resulting value internally to compensate for the offset. Thus, subsequent offsets will be compensated.

Example 21-1 provides an example of a calibration routine.

The calibration process assumes that the device is in a relatively steady-state operating condition. If A/D calibration is used, it should be performed after each device Reset or if there are other major changes in operating conditions.

21.8 Operation in Power-Managed Modes

The selection of the automatic acquisition time and A/D conversion clock is determined in part by the clock source and frequency while in a power-managed mode.

If the A/D is expected to operate while the device is in a power-managed mode, the ACQT<2:0> and ADCS<2:0> bits in ADCON1 should be updated in accordance with the power-managed mode clock that will be used. After the power-managed mode is entered (either of the power-managed Run modes), an A/D acquisition or conversion may be started. Once an acquisition or conversion is started, the device should continue to be clocked by the same power-managed mode clock source until the conversion has been completed. If desired, the device may be placed into the corresponding power-managed Idle mode during the conversion.

If the power-managed mode clock frequency is less than 1 MHz, the A/D RC clock source should be selected.

Operation in the Sleep mode requires the A/D RC clock to be selected. If bits, ACQT<2:0>, are set to ‘000’ and a conversion is started, the conversion will be delayed one instruction cycle to allow execution of the SLEEP instruction and entry to Sleep mode. The IDLEN and SCS bits in the OSCCON register must have already been cleared prior to starting the conversion.

EXAMPLE 21-1: SAMPLE A/D CALIBRATION ROUTINE

```

BCF    ANCON0,PCFG0    ;Make Channel 0 analog
BSF    ADCON0,ADON     ;Enable A/D module
BSF    ADCON1,ADCAL    ;Enable Calibration
BSF    ADCON0,GO       ;Start a dummy A/D conversion
CALIBRATION
;
BTFSC  ADCON0,GO       ;Wait for the dummy conversion to finish
BRA    CALIBRATION     ;
BCF    ADCON1,ADCAL    ;Calibration done, turn off calibration enable
;Proceed with the actual A/D conversion
    
```

24.0 HIGH/LOW VOLTAGE DETECT (HLVD)

PIC18F46J11 family devices (including PIC18LF46J11 family devices) have a High/Low Voltage Detect (HLVD) module for monitoring the absolute voltage on VDD or the HLVDIN pin. This is a programmable circuit that allows the user to specify both a device voltage trip point and the direction of change from that point.

If the module detects an excursion past the trip point in that direction, an interrupt flag is set. If the interrupt is enabled, the program execution will branch to the interrupt vector address and the software can then respond to the interrupt.

The High/Low-Voltage Detect Control register (Register 24-1) completely controls the operation of the HLVD module. This allows the circuitry to be “turned off” by the user under software control, which minimizes the current consumption for the device.

Figure 24-1 provides a block diagram for the HLVD module.

REGISTER 24-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER (ACCESS F85h)

R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VDIRMAG	BGVST	IRVST	HLVDEN	HLVDL3 ⁽¹⁾	HLVDL2 ⁽¹⁾	HLVDL1 ⁽¹⁾	HLVDL0 ⁽¹⁾
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 **VDIRMAG:** Voltage Direction Magnitude Select bit

1 = Event occurs when voltage equals or exceeds trip point (HLVDL<3:0>)

0 = Event occurs when voltage equals or falls below trip point (HLVDL<3:0>)

bit 6 **BGVST:** Band Gap Reference Voltages Stable Status Flag bit

1 = Indicates internal band gap voltage references is stable

0 = Indicates internal band gap voltage reference is not stable

bit 5 **IRVST:** Internal Reference Voltage Stable Flag bit

1 = Indicates that the voltage detect logic will generate the interrupt flag at the specified voltage range

0 = Indicates that the voltage detect logic will not generate the interrupt flag at the specified voltage range and the HLVD interrupt should not be enabled

bit 4 **HLVDEN:** High/Low-Voltage Detect Power Enable bit

1 = HLVD enabled

0 = HLVD disabled

bit 3-0 **HLVDL<3:0>:** Voltage Detection Limit bits⁽¹⁾

1111 = External analog input is used (input comes from the HLVDIN pin)

1110 = Maximum setting

•

•

•

1000 = Minimum setting

0xxx = Reserved

Note 1: See Table 29-8 in **Section 29.0 “Electrical Characteristics”** for specifications.

The module is enabled by setting the HLVDEN bit. Each time the module is enabled, the circuitry requires some time to stabilize. The IRVST bit is a read-only bit that indicates when the circuit is stable. The module can generate an interrupt only after the circuit is stable and IRVST is set.

The VDIRMAG bit determines the overall operation of the module. When VDIRMAG is cleared, the module monitors for drops in VDD below a predetermined set point. When the bit is set, the module monitors for rises in VDD above the set point.

PIC18F46J11 FAMILY

24.1 Operation

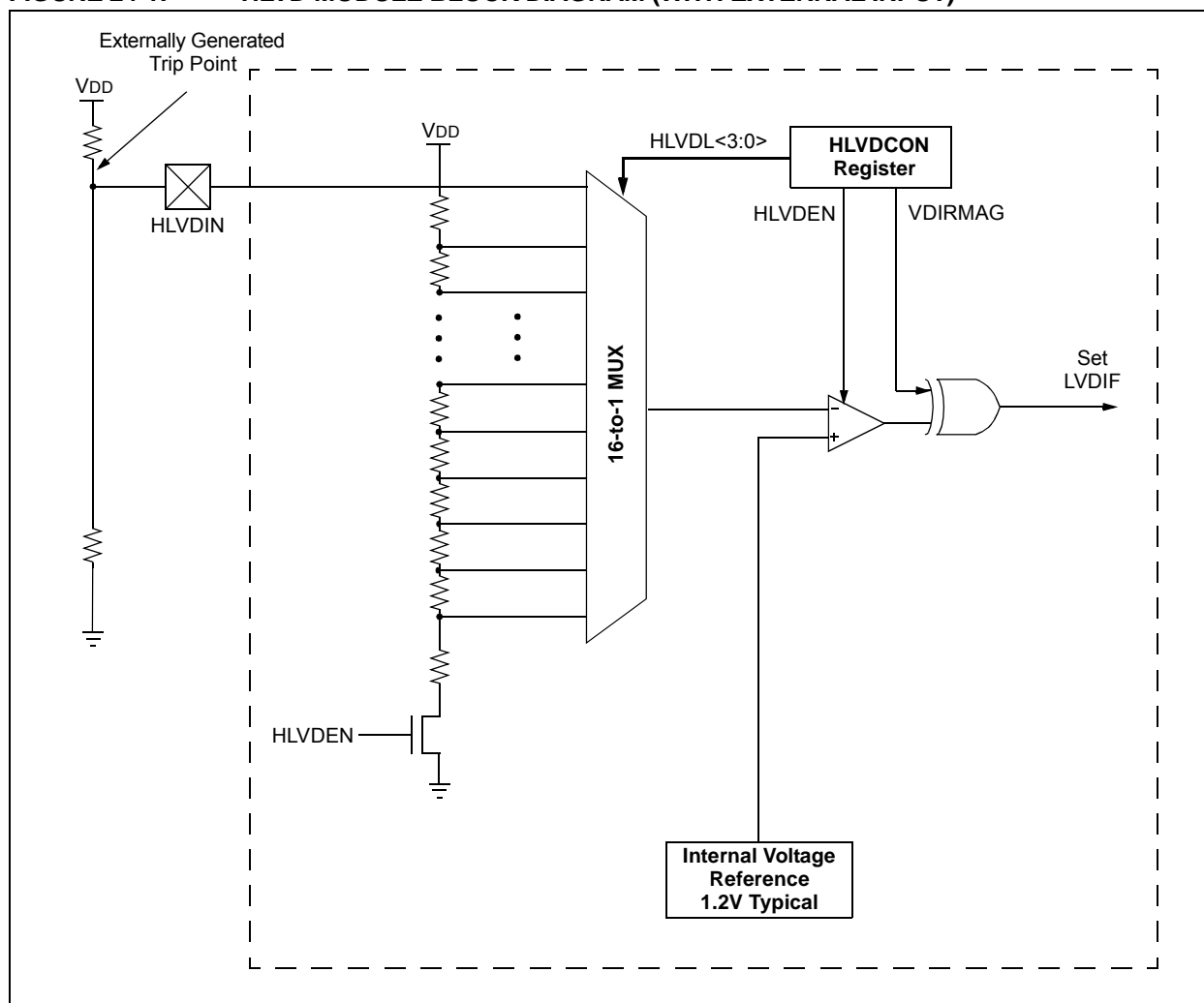
When the HLVD module is enabled, a comparator uses an internally generated reference voltage as the set point. The set point is compared with the trip point, where each node in the resistor divider represents a trip point voltage. The “trip point” voltage is the voltage level at which the device detects a high or low-voltage event, depending on the configuration of the module.

When the supply voltage is equal to the trip point, the voltage tapped off of the resistor array is equal to the internal reference voltage generated by the voltage reference module. The comparator then generates an interrupt signal by setting the LVDIF bit.

The trip point voltage is software programmable to any one of 8 values. The trip point is selected by programming the HLVDL<3:0> bits (HLVDCON<3:0>).

Additionally, the HLVD module allows the user to supply the trip voltage to the module from an external source. This mode is enabled when bits, HLVDL<3:0>, are set to '1111'. In this state, the comparator input is multiplexed from the external input pin, HLVDIN. This gives users flexibility because it allows them to configure the HLVD interrupt to occur at any voltage in the valid operating range.

FIGURE 24-1: HLVD MODULE BLOCK DIAGRAM (WITH EXTERNAL INPUT)



PIC18F46J11 FAMILY

CALLW Subroutine Call using WREG

Syntax:	CALLW				
Operands:	None				
Operation:	(PC + 2) → TOS, (W) → PCL, (PCLATH) → PCH, (PCLATU) → PCU				
Status Affected:	None				
Encoding:	<table><tr><td>0000</td><td>0000</td><td>0001</td><td>0100</td></tr></table>	0000	0000	0001	0100
0000	0000	0001	0100		
Description	<p>First, the return address (PC + 2) is pushed onto the return stack. Next, the contents of W are written to PCL; the existing value is discarded. Then, the contents of PCLATH and PCLATU are latched into PCH and PCU, respectively. The second cycle is executed as a NOP instruction while the new next instruction is fetched.</p> <p>Unlike CALL, there is no option to update W, STATUS or BSR.</p>				
Words:	1				
Cycles:	2				

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read WREG	Push PC to stack	No operation
No operation	No operation	No operation	No operation

Example: HERE CALLW

Before Instruction

PC = address (HERE)
PCLATH = 10h
PCLATU = 00h
W = 06h

After Instruction

PC = 001006h
TOS = address (HERE + 2)
PCLATH = 10h
PCLATU = 00h
W = 06h

MOVSF Move Indexed to f

Syntax:	MOVSF [z _s], f _d			
Operands:	0 ≤ z _s ≤ 127 0 ≤ f _d ≤ 4095			
Operation:	((FSR2) + z _s) → f _d			
Status Affected:	None			
Encoding:				
1st word (source)	1110	1011	0zzz	zzzz _s
2nd word (destin.)	1111	ffff	ffff	ffff _d

Description: The contents of the source register are moved to destination register 'f_d'. The actual address of the source register is determined by adding the 7-bit literal offset 'z_s', in the first word, to the value of FSR2. The address of the destination register is specified by the 12-bit literal 'f_d' in the second word. Both addresses can be anywhere in the 4096-byte data space (000h to FFFh).

The MOVSF instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

If the resultant source address points to an Indirect Addressing register, the value returned will be 00h.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	No operation No dummy read	No operation	Write register 'f' (dest)

Example: MOVSF [0x05], REG2

Before Instruction

FSR2 = 80h
Contents
of 85h = 33h
REG2 = 11h

After Instruction

FSR2 = 80h
Contents
of 85h = 33h
REG2 = 33h

PIC18F46J11 FAMILY

FIGURE 29-13: EXAMPLE SPI MASTER MODE TIMING (CKE = 0)

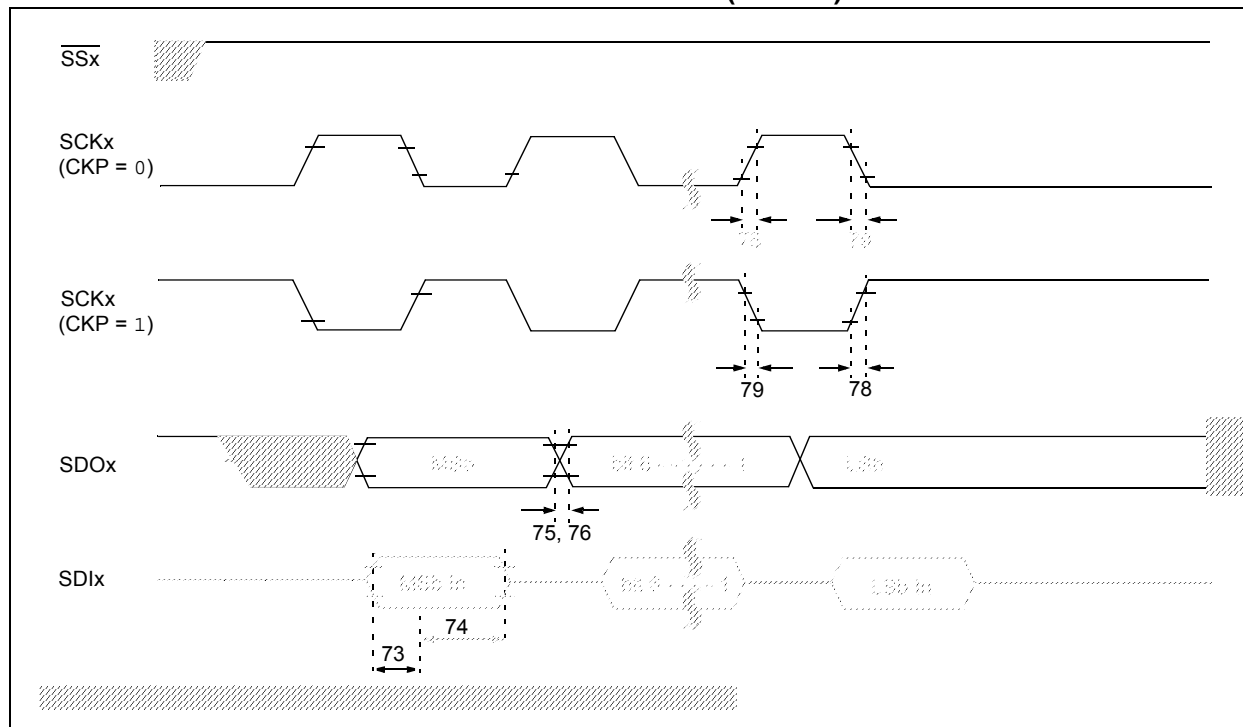


TABLE 29-20: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)

Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
73	TdIV2sCH, TdIV2sCL	Setup Time of SDIx Data Input to SCKx Edge	35 100	— —	ns ns	VDD = 3.3V, VDDCORE = 2.5V VDD = 2.15V, VDDCORE = 2.15V
74	Tsch2DiL, TscL2DiL	Hold Time of SDIx Data Input to SCKx Edge	30 83	— —	ns ns	VDD = 3.3V, VDDCORE = 2.5V VDD = 2.15V
75	TDoR	SDOx Data Output Rise Time	—	25	ns	PORTB or PORTC
76	TDoF	SDOx Data Output Fall Time	—	25	ns	PORTB or PORTC
78	TscR	SCKx Output Rise Time (Master mode)	—	25	ns	PORTB or PORTC
79	TscF	SCKx Output Fall Time (Master mode)	—	25	ns	PORTB or PORTC

INDEX

A

A/D	351
A/D Converter Interrupt, Configuring	355
Acquisition Requirements	356
ADCAL Bit	359
ADRESH Register	354
Analog Port Pins, Configuring	357
Associated Registers	360
Automatic Acquisition Time	357
Calibration	359
Configuring the Module	355
Conversion Clock (TAD)	357
Conversion Requirements	506
Conversion Status (GO/DONE Bit)	354
Conversions	358
Converter Characteristics	505
Operation in Power-Managed Modes	359
Special Event Trigger (ECCPx)	358
Use of the ECCP2 Trigger	358
Absolute Maximum Ratings	467
AC (Timing) Characteristics	485
Load Conditions for Device Timing Specifications	486
Parameter Symbolology	485
Temperature and Voltage Specifications	486
Timing Conditions	486
ACKSTAT	316
ACKSTAT Status Flag	316
ADCAL Bit	359
ADCON0 Register	
GO/DONE Bit	354
ADDFSR	456
ADDLW	419
ADDULNK	456
ADDWF	419
ADDWFC	420
ADRESL Register	354
Analog-to-Digital Converter. <i>See</i> A/D.	
ANDLW	420
ANDWF	421
Assembler	
MPASM Assembler	464
Auto-Wake-up on Sync Break Character	340

B

Bank Select Register	84
Baud Rate Generator	312
BC	421
BCF	422
BF	316
BF Status Flag	316
Block Diagrams	
+5V System Hardware Interface	132
8-Bit Multiplexed Address and Data Application	193
A/D	354
Analog Input Model	355
Baud Rate Generator	313
Capture Mode Operation	250
Comparator Analog Input Model	364
Comparator Configurations	366
Comparator Output	361
Comparator Voltage Reference	369
Comparator Voltage Reference Output Buffer	
Example	371

CTMU	379
CTMU Current Source Calibration Circuit	382
CTMU Typical Connections and Internal	
Configuration for Pulse Delay Generation	390
CTMU Typical Connections and Internal	
Configuration for Time Measurement	389
Demultiplexed Addressing Mode	186
Device Clock	38
EUSART Transmit	337
EUSARTx Receive	339
Fail-Safe Clock Monitor	410
Fully Multiplexed Addressing Mode	186
Generic I/O Port Operation	131
High/Low-Voltage Detect with External Input	374
Interrupt Logic	116
LCD Control	194
Legacy Parallel Slave Port	180
MSSPx (I ² C Master Mode)	311
MSSPx (I ² C Mode)	291
MSSPx (SPI Mode)	272
Multiplexed Addressing Application	193
On-Chip Reset Circuit	63
Parallel EEPROM (Up to 15-Bit Address, 16-Bit	
Data)	194
Parallel EEPROM (Up to 15-Bit Address, 8-Bit	
Data)	194
Parallel Master/Slave Connection Addressed Buffer	183
Parallel Master/Slave Connection Buffered	182
Partially Multiplexed Addressing Application	193
Partially Multiplexed Addressing Mode	186
PIC18F2XJ11 (28-Pin)	14
PIC18F4XJ11 (44-Pin)	15
PMP Module	171
PWM (Enhanced)	255
PWM Operation (Simplified)	252
Reads From Flash Program Memory	107
RTCC	227
Simplified Steering	268
Single Comparator	364
Table Read Operation	103
Table Write Operation	104
Table Writes to Flash Program Memory	109
Timer0 in 16-Bit Mode	198
Timer0 in 8-Bit Mode	198
Timer1	205
Timer2	214
Timer3	218
Timer4	226
Using the Open-Drain Output	132
Watchdog Timer	405
BN	422
BNC	423
BNN	423
BN OV	424
BNZ	424
BOR. <i>See</i> Brown-out Reset.	
BOV	427
BRA	425
Break Character (12-Bit) Transmit and Receive	342
BRG. <i>See</i> Baud Rate Generator.	
Brown-out Reset (BOR)	65
and On-Chip Voltage Regulator	408
Detecting	65

PIC18F46J11 FAMILY

Power-up Timer (PWRT)	46, 66
Time-out Sequence	66
Prescaler, Timer0	199
Prescaler, Timer2 (Timer4)	253
PRI_IDLE Mode	52
PRI_RUN Mode	48
Product Identification System	535
Program Counter	79
PCL, PCH and PCU Registers	79
PCLATH and PCLATU Registers	79
Program Memory	
ALU Status	96
Extended Instruction Set	99
Flash Configuration Words	78
Hard Memory Vectors	78
Instructions	83
Two-Word	83
Interrupt Vector	78
Look-up Tables	81
Memory Maps	77
Hard Vectors and Configuration Words	78
Reset Vector	78
Program Verification and Code Protection	411
Programming, Device Instructions	413
Pulse Steering	266
PUSH	442
PUSH and POP Instructions	80
PUSHL	458
PWM (CCP Module)	
Associated Registers	254
Duty Cycle	252
Example Frequencies/Resolutions	253
Operation Setup	253
Period	252
PR2/PR4 Registers	252
TMR2 (TMR4) to PR2 (PR4) Match	252
PWM (ECCP Module)	
Effects of a Reset	269
Operation in Power-Managed Modes	269
Operation with Fail-Safe Clock Monitor	269
Pulse Steering	266
Steering Synchronization	268
TMR4 to PR4 Match	225
PWM Mode. See Enhanced Capture/Compare/PWM	255
Q	
Q Clock	253
R	
RAM. See Data Memory.	
RBF Bit	138
RC_IDLE Mode	54
RC_RUN Mode	50
RCALL	443
RCON Register	
Bit Status During Initialization	68
Reader Response	534
Real-Time Clock and Calendar (RTCC)	227
Operation	239
Registers	228
Reference Clock Output	45
Register File	86
Register File Summary	90–95
Registers	
ADCON0 (A/D Control 0)	351
ADCON1 (A/D Control 1)	352

ALRMCFG (Alarm Configuration)	231
ALRMDAY (Alarm Day Value)	236
ALRMHR (Alarm Hours Value)	237
ALRMMIN (Alarm Minutes Value)	238
ALRMMNTH (Alarm Month Value)	236
ALMRPT (Alarm Calibration)	232
ALRMSEC (Alarm Seconds Value)	238
ALRMWD (Alarm Weekday Value)	237
ANCON0 (A/D Port Configuration 2)	353
ANCON1 (A/D Port Configuration 1)	353
Associated with Comparator	361
Associated with Watchdog Timer	406
BAUDCONx (Baud Rate Control)	330
CCPxCON (Enhanced Capture/Compare/PWM	
x Control)	248
CMSTAT (Comparator Status)	363
CMxCON (Comparator Control x)	362
CONFIG1H (Configuration 1 High)	398
CONFIG1L (Configuration 1 Low)	397
CONFIG2H (Configuration 2 High)	400
CONFIG2L (Configuration 2 Low)	399
CONFIG3H (Configuration 3 High)	402
CONFIG3L (Configuration 3 Low)	401
CONFIG4H (Configuration 4 High)	403
CONFIG4L (Configuration 4 Low)	402
CTMUCONH (CTMU Control High)	391
CTMUCONL (CTMU Control Low)	392
CTMUICON (CTMU Current Control)	393
CVRCON (Comparator Voltage Reference	
Control)	370
DAY (Day Value)	234
DEVID1 (Device ID 1)	403
DEVID2 (Device ID 2)	404
DMACON1 (DMA Control 1)	284
DMACON2 (DMA Control 2)	285
DSCONH (Deep Sleep Control High Byte)	58
DSCONL (Deep Sleep Control Low Byte)	58
DSGPR0 (Deep Sleep Persistent General	
Purpose 0)	59
DSGPR1 (Deep Sleep Persistent General	
Purpose 1)	59
DSWAKEH (Deep Sleep Wake High Byte)	60
DSWAKEL (Deep Sleep Wake Low Byte)	60
ECCPxAS (ECCPx Auto-Shutdown Control)	263
ECCPxDEL (Enhanced PWM Control)	266
EECON1 (EEPROM Control 1)	105
HLVDCON (High/Low-Voltage Detect Control)	373
HOURS (Hours Value)	235
I ² C Mode (MSSP)	291
INTCON (Interrupt Control)	117
INTCON2 (Interrupt Control 2)	118
INTCON3 (Interrupt Control 3)	119
IPR1 (Peripheral Interrupt Priority 1)	126
IPR2 (Peripheral Interrupt Priority 2)	127
IPR3 (Peripheral Interrupt Priority 3)	128
MINUTES (Minutes Value)	235
MONTH (Month Value)	233
ODCON1 (Peripheral Open-Drain Control 1)	133
ODCON2 (Peripheral Open-Drain Control 2)	133
ODCON3 (Peripheral Open-Drain Control 3)	134
OSCCON (Oscillator Control)	44
OSCTUNE (Oscillator Tuning)	42
PADCFG1 (Pad Configuration Control 1)	134
PADCFG1 (Pad Configuration)	230
Parallel Master Port	172

PIC18F46J11 FAMILY

ALRMVAL Register Mapping	242
Calibration	242
Clock Source	240
Digit Carry Rules	240
General Functionality	241
Leap Year	241
Register Mapping	241
RTCVAL Register Mapping	242
Safety Window for Register Reads and Writes	241
Write Lock	241
Peripheral Module Disable (PMD) Register	244
Register Interface	239
Register Maps	245
Reset	244
Device	244
Power-on Reset (POR)	244
Value Registers (RTCVAL)	233
RTCCN Bit Write	239
S	
SCKx	272
SDIx	272
SDOx	272
SEC_IDLE Mode	52
SEC_RUN Mode	48
Serial Clock, SCKx	272
Serial Data In (SDIx)	272
Serial Data Out (SDOx)	272
Serial Peripheral Interface. <i>See</i> SPI Mode.	
SETF	447
Shoot-Through Current	265
Slave Select (SSx)	272
SLEEP	448
Software Simulator (MPLAB SIM)	464
Special Event Trigger. <i>See</i> Compare (ECCP Mode).	
Special Features of the CPU	395
SPI Mode (MSSP)	272
Associated Registers	281
Bus Mode Compatibility	280
Clock Speed, Interactions	280
Effects of a Reset	280
Enabling SPI I/O	276
Master Mode	277
Master/Slave Connection	276
Operation	275
Open-Drain Output Option	275
Operation in Power-Managed Modes	280
Registers	273
Serial Clock	272
Serial Data In	272
Serial Data Out	272
Slave Mode	278
Slave Select	272
Slave Select Synchronization	278
SPI Clock	277
SSPxBUF Register	277
SSPxSR Register	277
Typical Connection	276
SSPOV	316
SSPOV Status Flag	316
SSPxSTAT Register	
R/W Bit	296, 299
SSx	272
Stack Full/Underflow Resets	81
SUBFSR	459
SUBFWB	448

SUBLW	449
SUBULNK	459
SUBWF	449
SUBFWB	450
SWAPF	450

T

Table Pointer Operations with TBLRD, TBLWT (table) ...	106
Table Reads/Table Writes	81
TAD	357
TBLRD	451
TBLWT	452
Timer0	197
Associated Registers	199
Operation	198
Overflow Interrupt	199
Prescaler	199
Switching Assignment	199
Prescaler Assignment (PSA Bit)	199
Prescaler Select (T0PS2:T0PS0 Bits)	199
Reads and Writes in 16-Bit Mode	198
Source Edge Select (T0SE Bit)	198
Source Select (T0CS Bit)	198
Timer1	201
16-Bit Read/Write Mode	206
Associated Registers	212
Clock Source Selection	204
Gate	208
Interrupt	207
Operation	204
Oscillator	201, 206
Layout Considerations	207
Resetting, Using the ECCP Special Event Trigger ..	208
TMR1H Register	201
TMR1L Register	201
Use as a Clock Source	207
Timer2	213
Associated Registers	214
Interrupt	214
Operation	213
Output	214
Timer3	215
16-Bit Read/Write Mode	219
Associated Registers	223
Gate	219
Operation	218
Oscillator	215, 219
Overflow Interrupt	215, 223
Special Event Trigger (ECCP)	223
TMR3H Register	215
TMR3L Register	215
Timer4	225
Associated Registers	226
Interrupt	226
MSSP Clock Shift	226
Operation	225
Output	226
Postscaler. <i>See</i> Postscaler, Timer4.	
PR4 Register	225
Prescaler. <i>See</i> Prescaler, Timer4.	
TMR4 Register	225
TMR4 to PR4 Match Interrupt	225, 226
Timing Diagrams	
A/D Conversion	506
Asynchronous Reception	340
Asynchronous Transmission	338

PIC18F46J11 FAMILY

NOTES: