Section 10. Watchdog Timer and Power Saving Modes

HIGHLIGHTS

This section of the manual contains the following topics:

10.1 Introduction

This section addresses the Watchdog Timer (WDT) and Power Saving modes of the dsPIC30F device family. The dsPIC DSC devices have two reduced Power modes that can be entered through execution of the PWRSAV instruction:

- Sleep Mode: The CPU, system clock source, and any peripherals that operate on the system clock source are disabled. This is the lowest Power mode for the device.
- Idle Mode: The CPU is disabled, but the system clock source continues to operate. Peripherals continue to operate, but can optionally be disabled.

The WDT, when enabled, operates from the internal LPRC clock source and can be used to detect system software malfunctions by resetting the device if the WDT has not been cleared in software. Various WDT time-out periods can be selected using the WDT postscaler. The WDT can also be used to wake the device from Sleep or Idle mode.

10.2 Power Saving Modes

The dsPIC30F device family has two special Power Saving modes, Sleep mode and Idle mode, that can be entered through the execution of a special PWRSAV instruction.

The assembly syntax of the PWRSAV instruction is as follows:

```
PWRSAV  #SLEEP_MODE    ; Put the device into SLEEP mode
PWRSAV  #IDLE_MODE     ; Put the device into IDLE mode
```

Note: SLEEP_MODE and IDLE_MODE are constants defined in the assembler include file for the selected device.

The Power Saving modes can be exited as a result of an enabled interrupt, WDT time-out, or a device Reset. When the device exits one of these two Operating modes, it is said to 'wake-up'. The characteristics of the Power Saving modes are described in subsequent sections.

10.3 Sleep Mode

The characteristics of Sleep mode are as follows:

- The system clock source is shutdown. If an on-chip oscillator is used, it is turned off.
- The device current consumption will be at a minimum *provided that no I/O pin is sourcing current*.
- The Fail-Safe Clock Monitor (FSCM) does not operate during Sleep mode since the system clock source is disabled.
- The LPRC clock will continue to run in Sleep mode if the WDT is enabled.
- The Low Voltage Detect circuit, if enabled, remains operative during Sleep mode.
- The BOR circuit, if enabled, remains operative during Sleep mode.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some peripherals may continue to operate in Sleep mode. These peripherals include I/O pins that detect a change in the input signal, or peripherals that use an external clock input. Any peripheral that is operating on the system clock source will be disabled in Sleep mode.

The processor will exit, or 'wake-up', from Sleep on one of the following events:

- On any interrupt source that is individually enabled
- On any form of device Reset
- On a WDT time-out

10.3.1 Clock Selection on Wake-up from Sleep

The processor will restart the same clock source that was active when Sleep mode was entered.

10.3.2 Delay on Wake-up from Sleep

The power-up and oscillator start-up delays associated with waking up from Sleep mode are shown in Table 10-1. In all cases, the POR delay time (TPOR = 10 µs nominal) is applied to allow internal device circuits to stabilize before the internal system Reset signal, $\overline{\text{SYSRST}}$, is released.

Table 10-1: Delay Times for Exit from Sleep Mode

Clock Source	$\overline{\text{SYSRST}}$ Delay	Oscillator Delay	FSCM Delay	Notes
EC, EXTRC	TPOR	—	—	**1**
EC + PLL	TPOR	TLOCK	TFSCM	**1, 3, 4**
XT + PLL	TPOR	TOST + TLOCK	TFSCM	**1, 2, 3, 4**
XT, HS, XTL	TPOR	TOST	TFSCM	**1, 2, 4**
LP (OFF during Sleep)	TPOR	TOST	TFSCM	**1, 2, 4**
LP (ON during Sleep)	TPOR	—	—	**1**
FRC, LPRC	TPOR	—	—	**1**

Note 1: TPOR = Power-on Reset delay (10 µs nominal).

2: TOST = Oscillator Start-up Timer. A 10-bit counter counts 1024 oscillator periods before releasing the oscillator clock to the system.

3: TLOCK = PLL lock time (20 µs nominal).

4: TFSCM = Fail-Safe Clock Monitor delay (100 µs nominal).

Note: Please refer to the "Electrical Specifications" section of the dsPIC30F device data sheet for TPOR, TFSCM and TLOCK specification values.

10.3.3 Wake-up from Sleep Mode with Crystal Oscillator or PLL

If the system clock source is derived from a crystal oscillator and/or the PLL, then the Oscillator Start-up Timer (OST) and/or PLL lock times must be applied before the system clock source is made available to the device. As an exception to this rule, no oscillator delays are necessary if the system clock source is the LP oscillator and it was running while in Sleep mode. Note that in spite of various delays applied, the crystal oscillator (and PLL) may not be up and running at the end of the POR delay.

10.3.4 FSCM Delay and Sleep Mode

If the following conditions are true, a nominal 100 µs delay (TFSCM) will be applied after the POR delay expires when waking from Sleep mode:

- The oscillator was shutdown while in Sleep mode.
- The system clock is derived from a crystal oscillator source and/or the PLL.

The FSCM delay provides time for the OST to expire and the PLL to stabilize before device execution resumes in most cases. If the FSCM is enabled, it will begin to monitor the system clock source after the FSCM delay expires.

10.3.5 Slow Oscillator Start-up

The OST and PLL lock times may not have expired when the power-up delays have expired.

If the FSCM is enabled, then the device will detect this condition as a clock failure and a clock fail trap will occur. The device will switch to the FRC oscillator and the user can re-enable the crystal oscillator source in the clock failure Trap Service Routine.

If FSCM is NOT enabled, then the device will simply not start executing code until the clock is stable. From the user's perspective, the device will appear to be in Sleep until the oscillator clock has started.

10.3.6 Wake-up from Sleep on Interrupt

User interrupt sources that are assigned to CPU priority level 0 cannot wake the CPU from Sleep mode, because the interrupt source is effectively disabled. To use an interrupt as a wake-up source, the CPU priority level for the interrupt must be assigned to CPU priority level 1 or greater.

Any source of interrupt that is individually enabled, using its corresponding IE control bit in the IECx registers, can wake-up the processor from Sleep mode. When the device wakes from Sleep mode, one of two actions may occur:

- If the assigned priority for the interrupt is *less than or equal to* the current CPU priority, the device will wake-up and continue code execution from the instruction following the `PWRSAV` instruction that initiated Sleep mode.
- If the assigned priority level for the interrupt source is *greater* than the current CPU priority, the device will wake-up and the CPU exception process will begin. Code execution will continue from the first instruction of the ISR.

The Sleep status bit (RCON<3>) is set upon wake-up.

10.3.7 Wake-up from Sleep on Reset

All sources of device Reset will wake the processor from Sleep mode. Any source of Reset (other than a POR) that wakes the processor will set the Sleep status bit (RCON<3>) to indicate that the device was previously in Sleep mode.

On a Power-on Reset, the Sleep bit is cleared.

10.3.8 Wake-up from Sleep on Watchdog Time-out

If the Watchdog Timer (WDT) is enabled and expires while the device is in Sleep mode, the processor will wake-up. The Sleep and WDTO status bits (RCON<3>, RCON<4>) are both set to indicate that the device resumed operation due to the WDT expiration. Note that this event does not reset the device. Operation continues from the instruction following the `PWRSAV` instruction that initiated Sleep mode.

10.4 Idle Mode

User interrupt sources that are assigned to CPU priority level 0 cannot wake the CPU from Idle mode, because the interrupt source is effectively disabled. To use an interrupt as a wake-up source, the CPU priority level for the interrupt must be assigned to CPU priority level 1 or greater.

When the device enters Idle mode, the following events occur:

- The CPU will stop executing instructions.
- The WDT is automatically cleared.
- The system clock source will remain active and peripheral modules, by default, will continue to operate normally from the system clock source. Peripherals can optionally be shutdown in Idle mode using their ‘stop-in-idle’ control bit. (See peripheral descriptions for further details.)
- If the WDT or FSCM is enabled, the LPRC will also remain active.

The processor will wake from Idle mode on the following events:

- On any interrupt that is individually enabled.
- On any source of device Reset.
- On a WDT time-out.

Upon wake-up from Idle, the clock is re-applied to the CPU and instruction execution begins immediately starting with the instruction following the `PWRSAV` instruction, or the first instruction in the ISR.

10.4.1 Wake-up from Idle on Interrupt

Any source of interrupt that is individually enabled using the corresponding IE control bit in the IECx register and exceeds the current CPU priority level, will be able to wake-up the processor from Idle mode. When the device wakes from Idle mode, one of two options may occur:

- If the assigned priority for the interrupt is *less than or equal to* the current CPU priority, the device will wake-up and continue code execution from the instruction following the `PWRSAV` instruction that initiated Idle mode.
- If the assigned priority level for the interrupt source is *greater* than the current CPU priority, the device will wake-up and the CPU exception process will begin. Code execution will continue from the first instruction of the ISR.

The Idle status bit (RCON<2>) is set upon wake-up.

10.4.2 Wake-up from Idle on Reset

Any Reset, other than a POR, will wake the CPU from Idle mode. On any device Reset, except a POR, the Idle status bit is set (RCON<2>) to indicate that the device was previously in Idle mode. In a Power-on Reset, the Idle bit is cleared.

10.4.3 Wake-up from Idle on WDT Time-out

If the WDT is enabled, then the processor will wake from Idle mode on a WDT time-out and continue code execution with the instruction following the `PWRSAV` instruction that initiated Idle mode. Note that the WDT time-out does not reset the device in this case. The WDTO and Idle status bits (RCON<4>, RCON<2>) will both be set.

10.4.4 Time Delays on Wake from Idle Mode

Unlike a wake-up from Sleep mode, there are no time delays associated with wake-up from Idle mode. The system clock is running during Idle mode, therefore, no start-up times are required at wake-up.

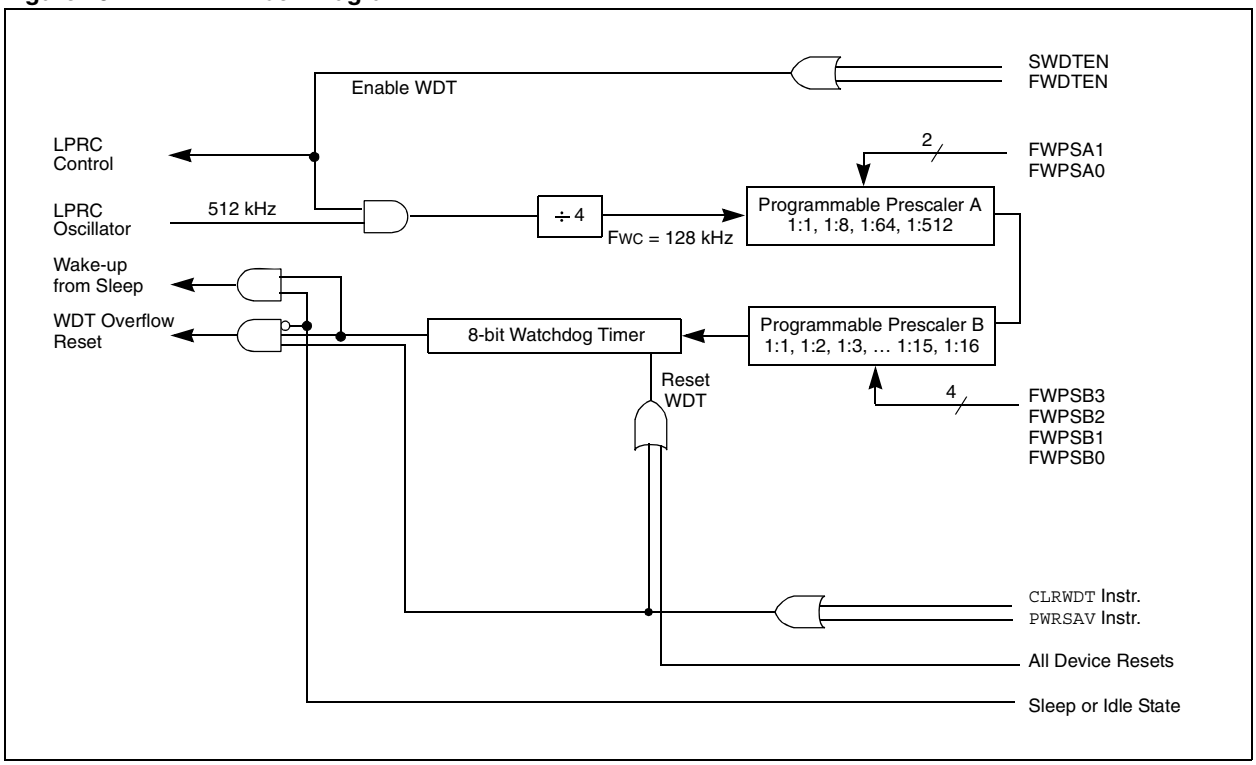
10.5 Interrupts Coincident with Power Save Instructions

Any interrupt that coincides with the execution of a `PWRSAV` instruction will be held off until entry into Sleep or Idle mode has completed. The device will then wake-up from Sleep or Idle mode.

10.6 Watchdog Timer

The primary function of the Watchdog Timer (WDT) is to reset the processor in the event of a software malfunction. The WDT is a free running timer, which runs on the internal LPRC oscillator requiring no external components. Therefore, the WDT timer will continue to operate even if the system clock source (e.g., the crystal oscillator) fails. A block diagram of the WDT is shown in Figure 10-1.

Figure 10-1: WDT Block Diagram



10.6.1 Enabling and Disabling the WDT

The WDT is enabled or disabled by the FWDTEN device configuration bit in the FWDT Device Configuration register. The FWDT Configuration register values are written during device programming. When the FWDTEN configuration bit is set, the WDT is enabled. This is the default value for an erased device. Refer to **Section 24. "Device Configuration"** for further details on the FWDT Device Configuration register.

10.6.1.1 Software Controlled WDT

If the FWDTEN device configuration bit is set, then the WDT is always enabled. However, the WDT can be optionally controlled in the user software when the FWDTEN configuration bit has been programmed to '0'.

The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user to enable the WDT for critical code segments and disable the WDT during non-critical segments for maximum power savings.

10.6.2 WDT Operation

If enabled, the WDT will increment until it overflows or “times out”. A WDT time-out will force a device Reset, except during Sleep or Idle modes. To prevent a WDT Time-out Reset, the user must periodically clear the Watchdog Timer using the CLRWDT instruction. The CLRWDT instruction also clears the WDT prescalers.

If the WDT times out during Sleep or Idle modes, the device will wake-up and continue code execution from where the PWRSAV instruction was executed.

In either case, the WDTO bit (RCON<4>) will be set to indicate that the device Reset or wake-up event was due to a WDT time-out. If the WDT wakes the CPU from Sleep or Idle mode, the Sleep status bit (RCON<3>), or Idle status bit (RCON<2>) will also be set to indicate that the device was previously in a Power Saving mode.

10.6.3 WDT Timer Period Selection

The WDT clock source is the internal LPRC oscillator, which has a nominal frequency of 512 kHz. The LPRC clock is further divided by 4 to provide a 128 kHz clock to the WDT. The counter for the WDT is 8-bits wide, so the nominal time-out period for the WDT (TWDT) is 2 milliseconds.

10.6.3.1 WDT Prescalers

The WDT has two clock prescalers, Prescaler A and Prescaler B, to allow a wide variety of time-out periods. Prescaler A can be configured for 1:1, 1:8, 1:64 or 1:512 divide ratios. Prescaler B can be configured for any divide ratio from 1:1 through 1:16. Time-out periods that range between 2 ms and 16 seconds (nominal) can be achieved using the prescalers.

The prescaler settings are selected using the FWPSA<1:0> (Prescaler A) and FWPSB<3:0> (Prescaler B) configuration bits in the FWDT Device Configuration register. The FWPSA<1:0> and FWPSB<3:0> values are written during device programming. For more information on the WDT prescaler configuration bits, please refer to **Section 24. “Device Configuration”**.

The time-out period of the WDT is calculated as follows:

Equation 10-1: WDT Time-out Period

$$\text{WDT Period} = 2\text{ ms} \bullet \text{Prescale A} \bullet \text{Prescale B}$$

Note:	The WDT time-out period is directly related to the frequency of the LPRC oscillator. The frequency of the LPRC oscillator will vary as a function of device operating voltage and temperature. Please refer to the specific dsPIC30F device data sheet for LPRC clock frequency specifications.

Table 10-2 shows time-out periods for various prescaler selections:

Table 10-2: WDT Time-out Period vs. Prescale A and Prescale B Settings

Prescaler B Value	Prescaler A Value			
	1	**8**	**64**	**512**
1	2	16	128	1024
2	4	32	256	2048
3	6	48	384	3072
4	8	64	512	4096
5	10	80	640	5120
6	12	96	768	6144
7	14	112	896	7168
8	16	128	1024	8192
9	18	144	1152	9216
10	20	160	1280	10240
11	22	176	1408	11264
12	24	192	1536	12288
13	26	208	1664	13312
14	28	224	1792	14336
15	30	240	1920	15360
16	32	256	2048	16384

Note: All time values are in milliseconds.

10.6.4 Resetting the Watchdog Timer

The WDT and all its prescalers are reset:

- On ANY device Reset
- When a `PWRSAV` instruction is executed (i.e., Sleep or Idle mode is entered)
- By a `CLRWDT` instruction during normal execution

10.6.5 Operation of WDT in Sleep and Idle Modes

If the WDT is enabled, it will continue to run during Sleep or Idle modes. When the WDT time-out occurs, the device will wake the device and code execution will continue from where the `PWRSAV` instruction was executed.

The WDT is useful for low power system designs, because it can be used to periodically wake the device from Sleep mode to check system status and provide action if necessary. Note that the SWDTEN bit is very useful in this respect. If the WDT is disabled during normal operation (FWDTEN = `0`), then the SWDTEN bit (RCON<5>) can be used to turn on the WDT just before entering Sleep mode.

10.7 Peripheral Module Disable (PMD) Registers

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled via the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral will also be disabled so writes to those registers will have no effect and read values will be invalid.

A peripheral module will only be enabled if both the associated bit in the the PMD register is cleared and the peripheral is supported by the specific dsPIC variant. If the peripheral is present in the device, it is enabled in the PMD register by default.

Note:	If a PMD bit is set, the corresponding module is disabled after a delay of 1 instruction cycle. Similarly, if a PMD bit is cleared, the corresponding module is enabled after a delay of 1 instruction cycle (assuming the module control registers are already configured to enable module operation).

Please check individual device data sheet for specific operational details of the PMD register.

10.8 Design Tips

Question 1: ***The device resets even though I have inserted a `CLRWDT` instruction in my main software loop.***

Answer: Make sure that the software loop that contains the `CLRWDT` instruction meets the minimum specification of the WDT (not the typical value). Also, make sure that interrupt processing time has been accounted for.

Question 2: ***What should my software do before entering Sleep or Idle mode?***

Answer: Make sure that the sources intended to wake the device have their IE bits set. In addition, make sure that the particular source of interrupt has the ability to wake the device. Some sources do not function when the device is in Sleep mode.

If the device is to be placed in Idle mode, make sure that the 'stop-in-idle' control bit for each device peripheral is properly set. These control bits determine whether the peripheral will continue operation in Idle mode. See the individual peripheral sections of this manual for further details.

Question 3: ***How do I tell which peripheral woke the device from Sleep or Idle mode?***

Answer: You can poll the IF bits for each enabled interrupt source to determine the source of wake-up.

10.9 Related Application Notes

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the dsPIC30F Product Family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Watchdog Timer and Power Saving Modes are:

Title	Application Note #
Low Power Design using PICmicro® Microcontrollers	AN606

Note: Please visit the Microchip web site (www.microchip.com) for additional Application Notes and code examples for the dsPIC30F Family of devices.

10.10 Revision History

Revision A

This is the initial released revision of this document.

Revision B

There were no technical content or editorial revisions to this section of the manual, however, this section was updated to reflect Revision B throughout the manual.

Revision C

This revision incorporates all known errata at the time of this document update.

Revision D

Section 10.7, Peripheral Module Disable (PMD) Registers, has been added.

NOTES: