AN1267

eXtreme Low-Power (XLP) PIC® Microcontrollers: An Introduction to Microchip's Low-Power Devices

Authors: *Brant Ivey*
Rajat Hegde
Akhila Bhat
Microchip Technology Inc.

INTRODUCTION

Power consumption has always been an important consideration for the design of any electrical system. This includes the embedded systems at the heart of countless modern devices and the microcontrollers that make most of these systems work. The expansion of embedded systems into markets, such as portable electronics, metering applications and medical devices, has driven power consumption to become one of the foremost concerns for embedded system designers. It is important that a microcontroller not only consumes as little power as possible, but also provides features that allow for minimal power consumption in the rest of the design as well. To design the best possible system, the application developer must understand all of the power-saving features that a microcontroller might offer – not only to make the best device selection, but also to exploit these features for the most economical power system.

This application note reviews the power-saving technology in current PIC® microcontrollers, particularly with eXtreme Low-Power (XLP) Technology. It also discusses how to select the best low-power device for a design and how to use these features to the best advantage.

UNDERSTANDING POWER CONSUMPTION

Before discussing the details of low-power operation, it may be useful to review the factors that make up power consumption. When we consider power consumption in microcontrollers, we are actually considering two components: dynamic power and static power.

Dynamic power is the current consumed by the switching of digital logic. It is mainly influenced by clock speed, although voltage and temperature also have an impact. For this reason, controlling dynamic power is largely a matter of controlling clock speed.

Static power is the current consumed when the main clock is disabled. It is composed mainly of transistor leakage and the current used by voltage supervisors. For many PIC MCUs, it also includes the clocking of logic necessary to resume operation from the Static mode (e.g., Watchdog Timers).

Static power is affected by the voltage level and temperature, which both have a large impact on the major component of transistor leakage. So, while much of static power consumption is dictated by device design and the manufacturing process, some elements may be influenced by the user.

Since voltage contributes to both static and dynamic power, an application with flexible voltage requirements can benefit from using the lowest supply voltage as the application will allow. For PIC MCUs with a separate core voltage input (V_{DDCORE}), it is important to note that the core voltage has the most impact on both static and dynamic power.

XLP TECHNOLOGY

For PIC microcontrollers, the low-power standard is referred to as eXtreme Low-Power or XLP Technology. XLP technology is continuously evolving to meet the latest low-power design requirements by introducing new power-saving features, including:

- Idle mode
- Sleep mode
- Deep Sleep mode
- Low-Voltage/Retention Sleep mode
- V_{BAT} mode
- On-chip, high-speed oscillator (INTOSC) with PLL and programmable postscaler
- WDT with extended time-out interval
- Ultra Low-Power Wake-up (ULPWU)
- Low-power option for Timer1 and the Secondary Oscillator (32 kHz)
- Low-power, software-controllable BOR

Together, these features offer a significant reduction in power consumption. The target specifications for an XLP microcontroller include typical current consumption less than:

- 100 nA for Power-Down Current (I_{PD})
- 500 nA Watchdog Timer Current (I_{WDT})
- 400 nA Real-Time Clock and Calendar Current (I_{RTCC})

XLP Technology uses a combination of proprietary process geometry design techniques, as well as power management features to reduce power consumption wherever possible. A key part of this strategy is the use of operating modes: a range of software-selectable hardware configurations that allow an application to change its power consumption during run time.

Table 1 summarizes the different operating modes available in XLP Technology. All of these (with the exception of Run mode, which represents baseline full-power operation) are explained in subsequent sections. A brief comparison of power consumption specifications for several Microchip XLP devices is provided in Table 2.

TABLE 1: POWER-SAVING OPERATING MODES FOR XLP TECHNOLOGY DEVICES

Operating Mode	Active Clocks	Active Peripherals	Wake-up Sources	Typical Current	Typical Usage
Deep Sleep(1)	• Timer1/SOSC • LPRC	• RTCC • DSWDT • DSBOR • INT0	• RTCC • DSWDT • DSBOR • INT0 • $\overline{MCLR}$	< 100 nA	Long life, battery-based applications; applications with increased Sleep times(2)
Low-Voltage/ Retention Sleep	• Timer1/SOSC • INTRC/LPRC • A/D RC	• RTCC • WDT • ADC • Comparators • CV_{REF} • INTx • Timer1 • HLVD • BOR	All device wake-up sources (see device data sheet)	200-400 nA	Most low-power applications
V_{BAT}	• SOSC • LPRC	• RTCC • DSGPRx	V_{BAT} mode is exited when V_{DD} is restored	400-500 nA(3)	Long life, battery-based applications
Sleep	Same as Low-Voltage/ Retention Sleep	Same as Low-Voltage/ Retention Sleep	Same as Low-Voltage/ Retention Sleep	4-5 µA	Most low-power applications
Sleep with fast wake-up(4)	Same as Sleep	Same as Sleep	Same as Sleep	4-5 times the Sleep current	Most low-power applications
Idle	• Timer1/SOSC • INTRC/LPRC • A/D RC	All peripherals	All device wake-up sources (see device data sheet)	25% of run current	Any time the device is waiting for an event to occur (e.g., external or peripheral interrupts)
Doze	All clocks	All peripherals	Software or interrupt wake-up	35-75% of run current	Applications with high-speed peripherals, but requiring low CPU use
Run	All clocks	All peripherals	Not applicable	See device data sheet	Normal operation

Note 1: Available on PIC18 and PIC24 devices.

2: Refer to the **"Deciding Between Sleep and Deep Sleep"** section for guidance on when to use Sleep or Deep Sleep mode.

3: RTCC is running.

4: Refer to the **"Sleep with Fast Wake-up"** section for more information on this feature.

TABLE 2: COMPARISON OF ELECTRICAL SPECIFICATIONS FOR SELECT LOW-POWER DEVICES(1)

Parameter	8-Bit PIC® MCUs			16-Bit PIC® MCUs		
	PIC18(L)F67K40	PIC16(L)F18855/75	PIC16(L)F1717/8/9	PIC24F16KA102	PIC24FJ128GA310	PIC24FJ256GB410
Deep Sleep (nA)	No Deep Sleep	No Deep Sleep	No Deep Sleep	20	10	80
Sleep (nA)	50(2)	50(2)	50(2)	25	330(3)	70(3)
WDT (nA)	400(2)	400(2)	500(2)	350(4)	270(4)	100(4)
32 kHz Oscillator with Timer/RTCC (nA)	600(2)	600(2)	500(2)	490	400	170
I/O Port Leakage (nA)	±5	±5	±5	±50	±100 to ±200	±100 to ±200
1 MHz Run Current (µA)	108(2)	90(2)	35(2)	195	150	170
Minimum VDD	1.8	1.8	1.8	1.8	2.0	2.0

Note 1: All numbers are typical values at minimum device VDD as reported in the most recent device data sheet. Values for WDT and/or RTCC may include base Sleep mode current. Sleep data is taken with BOR disabled, if possible. Nearest possible operating conditions are considered for comparison.

2: Data for minimum VDD is not available for these specifications; data for 3V is shown.

3: Low-Voltage/Retention Sleep currents are considered.

4: DSWDT currents are considered.

Deep Sleep Mode

Deep Sleep mode is the lowest static power mode, producing the lowest power consumption possible without removing power to the part completely. Deep Sleep reaches this low-power state by internally removing power from most of the components of the part. The core, on-chip voltage regulator (if present), most peripherals, and (in some cases) RAM, are all powered down in Deep Sleep mode.

Deep Sleep offers exceptionally low current, even on devices using an internal regulator, which normally requires a few microamperes of current. Removing the power from most of the part has the additional benefit of lower current consumption at high temperatures, since there are fewer active circuits that leak current.

Reaching power consumption this low has some trade-offs. Deep Sleep has only a few wake-up sources compared to the variety available in Sleep mode:

- POR Event
- $\overline{\text{MCLR}}$ Event
- RTCC Alarm
- External Interrupt
- Deep Sleep WDT

As a result of removing power from the core, a wake-up from Deep Sleep causes a device Reset rather than resuming from the next instruction, like Sleep mode. The Program Counter (PC) and SFRs are reset, and the device resumes program execution from the Reset vector. Unlike other Resets, all I/O states, as well as the Timer1/SOSC and RTCC, are maintained to allow for uninterrupted operation of the system as a whole. Additionally, Deep Sleep indication bits are set, and some RAM locations are maintained. This is in order to notify the software that the Reset is a Deep Sleep wake-up and allow the firmware state to be properly restored.

After a Deep Sleep wake-up occurs, the application needs to Acknowledge the wake-up, reconfigure peripherals and I/O registers, and then resume operation as normal. A high-level flow of the process is shown in Figure 1. Refer to the device data sheet for specific Deep Sleep entry and exit sequences.

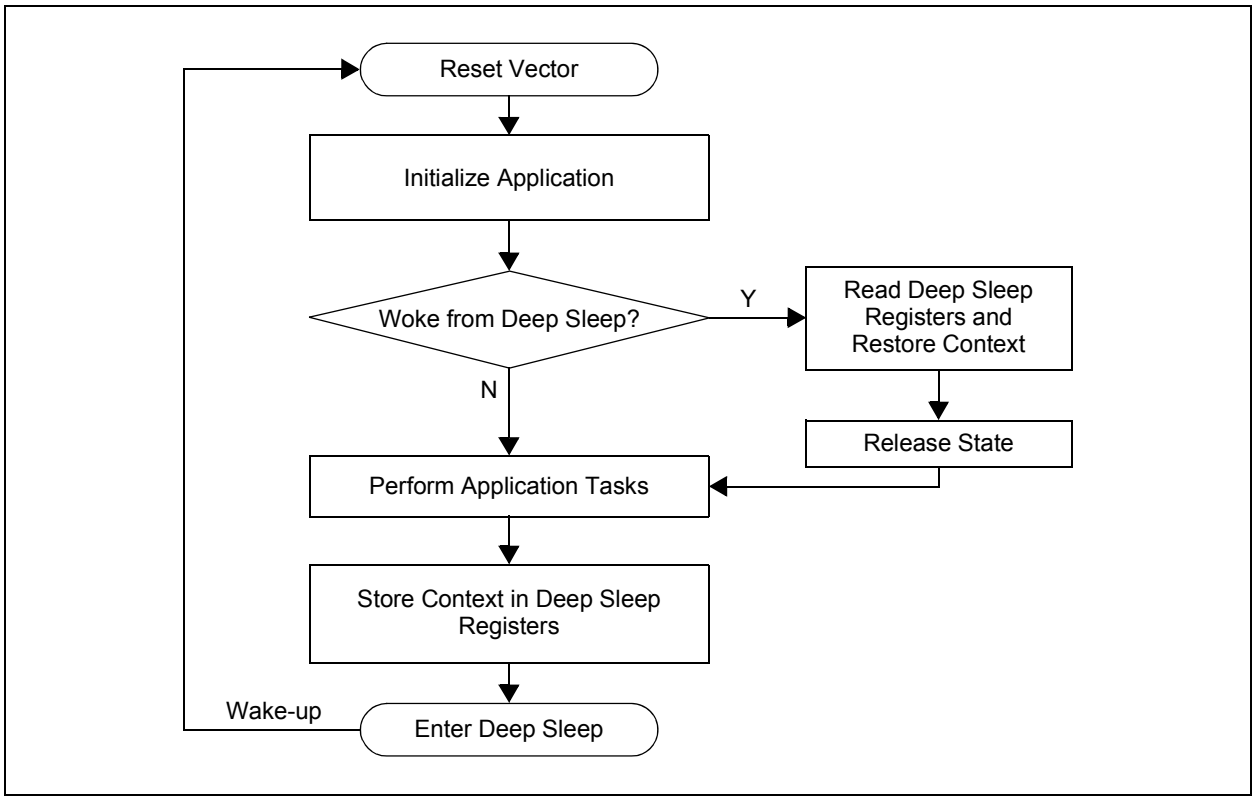
WHEN TO USE DEEP SLEEP MODE

It is important when designing an application to know which low-power mode to use. Deep Sleep mode is intended for use with applications that require very long battery life. The additional requirements for reconfiguring the device after wake-up mean that Sleep mode is better for some applications and Deep Sleep for others.

Ideally, applications that use the Deep Sleep mode have one or more of these characteristics:

- Use long Sleep times (one second or more, typical)
- Do not require any peripherals while asleep
- Require accurate timekeeping with minimal current
- Operate in environments with extreme temperatures

FIGURE 1: PROCEDURE FOR WAKE-UP FROM DEEP SLEEP MODE



SAVING CONTEXT DATA WITH THE DSGPRx REGISTERS

As exiting Deep Sleep mode causes a POR, most Special Function Registers (SFRs) reset to their default POR values. In addition, because VCORE power is not supplied in Deep Sleep mode, information in data RAM will be lost when exiting this mode.

Applications which require critical data to be saved prior to Deep Sleep may use the Deep Sleep General Purpose registers, DSGPR0 and DSGPR1, or data EEPROM (if available). Unlike other SFRs, the contents of these registers are preserved while the device is in Deep Sleep mode. After exiting Deep Sleep, the software can restore the data by reading the registers and clearing the RELEASE bit (DSCON<0>).

Sleep Mode

Sleep mode is the standard low-power mode for virtually all PIC® microcontrollers. In Sleep mode, the main CPU clock and most peripheral clock sources are shut down, bringing the device to a low-power state. The current device state is maintained, including RAM, SFRs and the Program Counter (PC).

Wake-up sources vary between device families. All PIC MCUs can use the WDT, the 32 kHz Timer (Timer1 on most devices) and one or more external interrupt sources. PIC18, PIC24 and PIC32 devices also have a number of peripherals that are capable of waking up the device; these include the ADC, comparators and serial communication modules. Total wake-up times also vary between families; most devices implement options to change wake-up time and allow flexibility in design.

WHEN TO USE SLEEP MODE

Sleep mode is the most commonly used and most flexible of the available modes. Typically, there is a very fast wake-up time that requires little to no overhead to handle entry and exit. As a result, it is the best low-power mode for applications that require short Sleep times, and fast wake-up and processing. Sleep is often used in applications with the following characteristics:

- Short loop times with frequent wake-up (generally less than 1 second)
- Require peripheral wake-up sources
- Perform analog sampling with ADC or comparators while asleep

Low-Voltage/Retention Sleep Mode

Some of the PIC MCUs incorporate a second on-chip voltage regulator, called low-voltage/retention regulator, designed to power select microcontroller modules at 1.2V. This regulator facilitates modules, such as data RAM and WDT, to function at a lower power than in Sleep mode.

Low-Voltage/Retention Sleep mode is similar to Sleep mode with the same set of active peripherals and wake-up triggers. The difference is that the low-voltage/retention regulator allows the core digital logic voltage (V_{CORE}) to drop to 1.2V. This permits an incremental reduction in power consumption over what would be required if V_{CORE} was maintained at a 1.8V (minimum) level.

Low-Voltage Sleep mode requires a longer wake-up time than Sleep mode, due to the additional time required to bring V_{CORE} back to 1.8V (known as T_{VREG}). In addition, the use of the low-voltage/retention regulator limits the amount of current that can be sourced to any active peripherals, such as the RTCC, LCD, etc. Refer to the device data sheet for availability of this mode.

Sleep with Fast Wake-up

Waking up from Sleep typically takes 20 µs. If the application requires a faster wake-up time, it is possible by using the fast wake-up feature. This feature keeps the on-chip voltage regulator enabled during Sleep, thus maintaining the program memory bias voltage to facilitate fast resumption of code execution. In PIC24F devices, the fast wake-up feature can be enabled by setting the VREGS bit in the RCON register. However, in some PIC24F devices, this bit is named PMSLP. In 8-bit XLP devices, the same feature is available as the VREGPM bit in VREGCON register.

Using this feature during Sleep consumes slightly higher current compared to that of normal Sleep mode. For a very low-power requirement, the voltage regulator must be configured to go into Standby mode by clearing the PMSLP/VREGS/VREGPM bit (default configuration). Refer to device-specific data sheet for availability of this mode.

Inrush Current Considerations

When a system, consisting of high value capacitances or decoupling capacitors, is powered on, a large momentary current flows to charge the capacitors. This inrush current may exceed the steady-state current value. Thus, while designing a Low-Dropout (LDO) regulator for low-power applications, inrush current must be taken into consideration.

For example, PIC24FJ256GA410 MCUs power their core digital logic at a constant voltage of 1.8V by incorporating an on-chip regulator. This regulator provides the constant voltage to the core digital logic from a V_{DD} of about 2.1V, all the way up to the device's V_{DDMAX}. A low-ESR (< 5Ω) capacitor, V_{CAP}, is required on the V_{CAP} pin to stabilize the output voltage of the on-chip voltage regulator. The recommended capacitor value of V_{CAP} is 10 µF. This capacitor contributes to the above stated inrush current during power-up and wake-up from Deep Sleep.

Theoretically, inrush current can be calculated using the formula, $I = C \times dV/dT$, where:

- C is the V_{CAP} capacitor value
- dV/dT is the rate of change of the V_{CAP} voltage

The typical inrush current observed, with a recommended V_{CAP} value of 10 µF, is 250 mA to 300 mA for a period of approximately 50 µs. This factor should be considered while designing a power supply or LDO for a circuit.

It is important to note that there will be inrush current even when the device wakes up from Deep Sleep and/or Low-Voltage Sleep. Therefore, if an application chooses to use Deep Sleep mode or Low-Voltage Sleep over Sleep mode, this must be taken into account. Please refer to the following section to decide between Sleep and Deep Sleep modes.

Note:	For the ease of calculation, inrush current is considered as part of I_{POR} in formula [2] calculations.

Deciding Between Sleep and Deep Sleep

A helpful way to determine whether Sleep or Deep Sleep is more effective is to calculate the Breakeven Time (T_{BE}) for a particular application. This time indicates how long a device must remain in Deep Sleep mode to have lower total power consumption than Sleep mode, once the higher power requirements for restart from Deep Sleep are accounted for. T_{BE} can be calculated using the three formulas shown in Example 1.

The first step is to calculate the total charge consumed using Sleep (Q_{SLP}) and Deep Sleep (Q_{DS}). In Sleep, this is simply the Sleep static current (I_{PDSLP}) multiplied by the time the device is in Sleep (T_{PD}) (formula [1]). Charge is used instead of energy because in both cases, the voltage will stay constant, so it can be ignored. Charge also gives an easy comparison to battery capacity specifications when performing power budgeting.

For Deep Sleep, there are three components to the equation: power-up, software initialization and Deep Sleep (formula [2]). The Deep Sleep component, similar to the Sleep energy calculation, is just the Deep Sleep static current (I_{PDDS}) times the Sleep period (T_{PD}).

The POR component includes the POR time (T_{POR}), which starts when the Deep Sleep wake-up interrupt occurs, until the first instruction is executed. Details on POR time can be found in device data sheets. The POR current (I_{POR}) varies based on a number of device settings and application factors, so it is best taken experimentally. Note that on devices with an internal regulator, the POR time and current will include the time and inrush current required for the regulator to charge the capacitor on the VCAP pin if it has discharged while the device is in Deep Sleep.

The initialization component is the initialization time (T_{INIT}) and current (I_{DD}), starting when the device begins code execution and lasting until the main loop is entered. Both of these vary by application and are best assessed with measurement. However, they can be approximated using published dynamic current specifications to determine current and the Stopwatch feature in MPLAB® IDE to measure the initialization execution time.

Breakeven Time is the point where Q_{DS} and Q_{SLP} are equal. Mathematically, this is the same as setting [1] and [2] to be equal to each other. Solving generically for T_{PD} provides formula [3]; at this point, time in Sleep or Deep Sleep is equivalent to T_{BE}. Deep Sleep should be used if the Sleep duration is longer than T_{BE} and Sleep mode should be used if the Sleep time is shorter than T_{BE}. An application with varying Sleep times can use both Sleep and Deep Sleep to get the most efficient current consumption.

Similarly, Breakeven Time can be calculated to decide between Sleep and Low-Voltage Sleep modes.

EXAMPLE 1: CALCULATING BREAKEVEN TIME (DEEP SLEEP vs. SLEEP MODES)

$$Q_{SLP} = T_{PD} \times I_{PDSLP} \quad [1]$$

$$Q_{DS} = (T_{INIT} \times I_{DD}) + (T_{POR} \times I_{POR}) + (T_{PD} \times I_{PDDS}) \quad [2]$$

$$T_{BE} = T_{PD} = \frac{(T_{INIT} \times I_{DD}) + (T_{POR} \times I_{POR})}{I_{PDSLP} - I_{PDDS}} \quad [3]$$

Where:
- Q_{DS} = Total Charge Spent in Deep Sleep
- Q_{SLP} = Total Charge Spent in Sleep
- T_{BE} = Breakeven Time (interval at which $Q_{DS} = Q_{SLP}$)
- T_{INIT} = Initialization Time to Resume Full-Power Operation
- T_{PD} = Sleep or Deep Sleep Period (defined by context)
- T_{POR} = Time Required for Power-on Reset
- I_{POR} = POR Current (inrush current included)
- I_{PDSLP} = Static Current in Sleep mode
- I_{PDDS} = Static Current in Deep Sleep mode

VBAT Mode

VBAT mode is a hardware-based power-saving mode that powers only the most critical systems when a power loss occurs on VDD. This is done by powering these systems from a backup power source, such as a battery connected to the VBAT pin. The Deep Sleep Retention registers (DSGPRx), RTCC and Key RAM block of the cryptographic engine, along with anti-tamper control, can be powered in VBAT mode. These modules can be independently enabled/disabled based on the requirement. Please refer to the device-specific data sheet for availability of this feature.

The device enters VBAT mode whenever power is removed from VDD. An on-chip power switch detects the power loss from the VDD and connects the VBAT pin so it can supply the power.

As Deep Sleep Retention registers, DSGPR0 and DSGPR1, are powered in VBAT mode through the supply from the VBAT pin, they can be used to store critical data. In devices having RTCC with the timestamping feature, the two Timestamping registers can also be used for storing critical data when the timestamping features are not used. Since the power loss on VDD may be unforeseen, it is recommended to load any data to be saved in the DSGPRx registers in advance, while saving the critical data during VBAT mode.

Idle and Doze Modes

Idle and Doze modes are dynamic power reduction modes that are intended to allow more peripheral functionality than static power modes, such as Sleep, while still reducing current consumption below Run mode. These modes allow for significant power reduction at times when peripheral operation is critical, but CPU activity is not.

In Idle mode, the system clock is removed from the CPU, but is still provided to the peripherals. Depending on the device family, some or all of the peripherals may continue to operate in Idle mode. For PIC24, dsPIC® and PIC32 devices, operation in Idle is configurable on a 'per module' basis.

In Doze mode (available on PIC24 and dsPIC33 devices only), the system clock is split into separate CPU and peripheral clocks. The CPU clock is divided by a specific user-defined factor, while the peripheral clock continues to run at the system clock speed.

WHEN TO USE IDLE AND DOZE MODES

Idle and Doze modes are dynamic modes, so while they consume less power than Run mode, they still consume significantly more power than static modes, like Sleep. As a result, they should be used in cases where it is not possible to enter Sleep, such as:

- Making large DMA transfers (on devices with DMA only)
- Sending or receiving serial data
- Performing high-speed ADC sampling
- Waiting for time-out from the synchronous timer
- Waiting for data capture with Input Capture (IC)
- Waiting for event using Output Compare (OC)

Any time a loop, waiting for a peripheral interrupt to occur, would be used, it can be replaced with an entry into Idle or Doze mode. These cases are frequently overlooked, so it is important to review a design for places where the CPU is not being fully utilized to minimize power consumption.

Clock Switching

Also introduced in the XLP Technology, clock switching is an important low-power feature. Clock switching offers enormous flexibility for reducing dynamic current consumption, as clock speed is an important factor in dynamic power. While Idle and Doze modes allow reduction of the speed of the CPU clock, the peripherals are still clocked at full speed and consume full current. Therefore, it is important to be able to reduce the speed of the clocks to the entire device.

The flexible clock switching systems implemented in PIC microcontrollers allow for switching to the most appropriate clock source for a given situation. For example, an application may use a slow clock for code sections that are not time-critical, then switch to a full-speed clock source for processing computation intensive or time-critical code. Such flexibility is necessary when implementing a low-power system in order to ensure the lowest power consumption possible.

WHEN TO USE CLOCK SWITCHING

As with the other dynamic power-saving modes, clock switching is best used in cases where the use of Sleep or Deep Sleep is not possible. Clock switching should be used instead of Idle or Doze mode in situations where clock speed is not critical for both the CPU and the peripherals, as it can provide significantly lower power than Idle and Doze modes.

Special Features of the Modules Active in Low-Power Modes

ADC THRESHOLD DETECT FEATURE

Some PIC microcontrollers include a unique Threshold Detect feature in the ADC module that allows the ADC to make simple decisions based on the conversion results. The Threshold Scan feature can be used to determine if a conversion meets specific user-defined criteria, storing or discarding the converted value as appropriate, and then setting semaphore flags to indicate the event. This allows conversions in low-power modes and wakes up the device only when specific conditions are met.

Threshold Detect is a significant extension of the auto-scan feature. In addition to repeated sampling of a predefined sequence of analog channels, Threshold Detect allows the programmer to define filter criteria and generate an interrupt. In low-power applications, this allows the CPU to remain in power-saving modes for longer until the specified conditions are met.

Independently selectable comparison and buffer storage settings make a wide range of operating combinations possible.

Types of comparisons that can be performed are:

- Result of the current conversion is greater than a reference threshold
- Result of the current conversion is less than a reference threshold
- Result of the current conversion is between two predefined thresholds (“Inside Window”)
- Result of the current conversion is outside of the predefined thresholds (“Outside Window”)

Buffer storage options that determine the disposition of the conversion are:

- Discard the conversion after the comparison has been performed
- Store the conversion after the comparison has been performed
- Store the conversion without comparison (Legacy mode)

Please refer to the device-specific data sheet for details and availability of this feature.

CRYPTOGRAPHIC KEY RAM ANTI-TAMPER AND TIMESTAMPING IN V_{BAT} MODE

The PIC MCUs with Cryptographic Engine are provided with a hardware anti-tampering feature to erase cryptographic key data in the event of application tampering. Hardware anti-tampering monitors the $\overline{TMPR}$ pin. If a low pulse or sustained low-voltage level is detected, the Key RAM gets automatically erased. Key RAM data can also be retained and protected in Deep Sleep and V_{BAT} modes.

When a device is in low-power mode and a tamper event occurs on the $\overline{TMPR}$ pin, the event can be captured using the RTCC timestamp feature and optionally wake-up the device from low-power modes using RTCC timestamp interrupts. In the case of such tamper events, the present time and date values are stored in the TSATIMEL/H and TSADATEL/H registers, the TSAEVT status bit (RTCSTATL<3>) becomes set and an RTCC interrupt occurs. Please refer to the device-specific data sheet for details and availability of this feature.

Note:	To record the timestamp accurately in any event of tampering during V_{BAT} mode, it is recommended to pull-up the $\overline{TMPR}$ pin to V_{BAT}, instead of V_{DD}, to avoid false event recording.

Tips and Tricks for Low-Power Applications

- **Lowering the CPU speed:** For an application that does not require high system speed (or MIPS), it is advised to run at lower MIPS as unnecessarily running in high speed would increase power consumption.
- **Using Doze mode or reducing the clock speed dynamically:** Switch the CPU to Doze mode or lower the system clock speed any time during non-critical operation. In Doze mode, the system clock to the CPU is postscaled so that the CPU runs at a lower speed than the peripherals. Dynamically reducing the clock speed allows significant reduction in power by choosing different clocks for different portions of the code.
- **Using PMD bits to disable peripherals:** Using only the necessary peripherals at a given time and disabling all the unused peripherals saves significant power. By default, all the peripherals will be clocked. To disable the unused peripherals, the PMD registers can be used.
- **CPU offloading:** Offloading the CPU by using necessary core-independent modules, such as the DMA, Configurable Logic Cell (CLC) and Peripheral Trigger Generator (PTG), not only increases efficiency but also reduces power consumption. If available, peripheral FIFO or DMA can also be used. When using FIFO, the CPU can be in Idle or Sleep mode until the FIFO fills, or the DMA completes transfers.
- **Using code optimization:** Using the performance optimization feature in compilers optimizes the code execution to consume less time in active mode, thereby consuming less power. Microchip's XC8, XC16 and XC32 compilers provide different optimization levels suitable for different applications.
- **Using Near Data Space for frequent RAM access:** Storage of the frequently used variables, or any other frequent RAM access, should be done from the Near Data Space. RAM access to the Near Data Space takes only one instruction cycle, thereby improving the execution speed and reducing the overall power requirements.

  Near Data Space can be accessed directly using the file register instructions that encode a 13-bit address into the opcode. The data memory region beyond near memory is addressable only by the Indirect Addressing modes. For example, in the PIC24FJ256GA410 family, the 8-Kbyte area between 0000h and 1FFFh is referred to as the Near Data Space.

  The `near` attribute in the XC16 compiler tells the compiler that the variable is allocated in Near Data Space. Please refer to the respective family C compiler user's guide for more details.

EXAMPLE 2:

```
int num __attribute__ ((near)); //
declare num in near data memory
```

- Using the C storage class `register` for the most frequently used variables improves the execution time in Active mode. Since accessing registers is faster than accessing memory, allocating such variables in the CPU Working registers improves performance.

EXAMPLE 3:

(For XC16 C Compiler only. Please refer to the respective C compiler user's guide for other families)

A global register variable can be defined in a specific W register, as shown below:

```
register int *foo asm ("w8"); //use W8 register
```

A local register variable can be defined with a specific W register, as shown below:

```
register int *foo asm ("w8"); // use W8 register
```

Note: This is the same syntax used for defining global register variables, but for a local variable, it would appear within a function.

Please refer to the respective family C compiler user's guide before using register variables, as using too many registers, in particular, the W0 register, may impair the ability of the XC16 compiler to compile.

- **Interrupts over polling:** It is recommended to avoid polling and use interrupts instead. In polling, the CPU remains active, waiting for an event and consuming power. With interrupts, the device can be put into Sleep or Idle mode, thereby reducing the power consumption significantly.
- **Configuring unused I/Os:** It is recommended to configure any unused I/O as a digital output pin, driving logic '0'.

EXAMPLE 4:

(For PIC24FJ256GA410 family)

```
ANSAbits.ANSA0 = 0;    //Configure RA0 as digital
TRISAbits.TRISA0 = 0; //Configure RA0 as ouput
LATAbits.LATA0 = 0;    //Write '0' to RA0
```

- **Code reorganizing:** In an application using multiple modules, it is better to configure all modules before enabling them. This saves power as it avoids a module running unnecessarily, while the other modules are being configured. This can be useful in applications which have frequent reinitialization or wake-up from Deep Sleep.
- **Disabling device functions:** Some unused device functions can be disabled using the Configuration fuse bits during start-up. For example, JTAG can be disabled, both BOR and WDT can be disabled or configured as software controlled, and CLKO can be configured as an I/O pin.
- **Writing instructions:** Writing the most executed instructions in a loop in assembly language, or using the compiler built-in functions instead of C, can be a good option to save power as it takes less instruction cycles.

  For example, using the REPEAT instruction for looping an instruction takes only (n+1) instructions, where n is the number of loops, which is significantly less instruction cycles when compared to the same using C-based `for` or `while` loops.

EXAMPLE 5:

Replace:

```
for (i=0; i<10 ; i++)
LATCbits.LATC3 = !LATCbits.LATC3;
```

with:

```
REPEAT #9
BTG LATC, #0x3
```

For more low-power tips and tricks, please refer to **Chapter 2 "PIC® Microcontroller Low-Power Tips 'n Tricks"** in the "*Compiled Tips 'N Tricks Guide*" (DS01146).

CONCLUSIONS

With innovations in XLP Technology, Microchip continues to focus on lowering power consumption as a key design goal. The result is devices that not only have impressive features and performance, but power consumption below long-standing industry minimums.

When creating a low-power application, it is important to approach all aspects of the design from a low-power perspective. This application note has taken an initial look at the low-power modes on PIC® microcontrollers with XLP Technology, which are a central source of power savings for many designs.

It is important to be very familiar with how and when these features are used in order to maintain the lowest possible power consumption. Check www.microchip.com/lowpower for future documents covering other important aspects of low-power design.

REFERENCES

E. Schlunder, "*Deep Sleep Mode on Microchip PIC18 and PIC24 Microcontrollers*" (archived WebSeminar), http://webtrain.microchip.com/webseminars.

Microchip Applications Staff, *"Compiled Tips 'N Tricks Guide"* (DS01146), © 2009 Microchip Technology Inc.

"PIC18(L)F67K40 Family Data Sheet" (DS40001841), © 2016 Microchip Technology Inc.

"PIC16(L)F18855/75 Family Data Sheet" (DS40001802), © 2016 Microchip Technology Inc.

"PIC16(L)F1717/8/9 Family Data Sheet" (DS40001740), © 2014-2015 Microchip Technology Inc.

"PIC24F16KA102 Family Data Sheet" (DS39927), © 2008-2011 Microchip Technology Inc.

"PIC24FJ128GA310 Family Data Sheet" (DS30009996), © 2010-2014 Microchip Technology Inc.

"PIC24FJ256GA412/GB412 Family Data Sheet" (DS30010089), © 2015-2016 Microchip Technology Inc.

"MSP430x22x2, MSP430x22x4 Mixed Signal Microcontroller Data Sheet" (SLAS504G), © 2006-2012, Texas Instruments Incorporated.

"MSP430x13x, MSP430x14x, MSP430x14x1 Mixed Signal Microcontroller Data Sheet" (SLAS272F), © 2000−2004, Texas Instruments Incorporated.

"MSP430F552x, MSP430F551x Mixed Signal Microcontrollers Data Sheet" (SLAS590M), © 2009-2015, Texas Instruments Incorporated.

"R8C/LA3A Group, R8C/LA5A Group, R8C/LA6A Group, R8C/LA8A Group RENESAS MCU Data Sheet" (R01DS0011EJ0101), Renesas, Oct 28, 2011.

"RL78/L1C RENESAS MCU Data Sheet" (R01DS0192EJ0210), Renesas, Oct 28, 2011.

"MPLAB® XC16 C Compiler User's Guide" (DS50002071), © 2012-2016 Microchip Technology Inc.

Section 57. "Power-Saving Features with VBAT" in the *"dsPIC33/PIC24F Family Reference Manual"* (DS30622), © 2011 Microchip Technology Inc.

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-1645-6

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-3326-8000
Fax: 86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

France - Saint Cloud
Tel: 33-1-30-60-70-00

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820