Selecting the Right MCU Can Squeeze Nanoamps out of Your Next Internet of Things Application

SILICON LABS

www.silabs.com

Introduction

Industry leaders predict that the number of connected devices for the Internet of Things (IoT) will surpass 15 billion nodes by 2015 and top 50 billion by 2020. Most of these connected devices will not be traditional PCs or even smartphones but smaller, less expensive, simpler ultra-low-power embedded systems ranging from smart thermostats to security sensors and IP cameras to personal fitness trackers and smart watches.

Energy efficiency is critical to IoT end-node applications because many connected devices must often operate on batteries for months or years without maintenance or battery replacement. Some systems will even be designed to operate on harvested energy sources. In either case, developers are challenged with meeting aggressive battery-life specifications, which drive component-related decisions.

When selecting your next microcontroller (MCU) for you IoT application, verify that your MCU has the following hardware and software features to help squeeze the most nanoamps out of your system, extending the battery life of your IoT-enabled device:

- Ultra-low power modes with very fast wake time
- Autonomous peripherals that operate in low-energy modes
- Power estimation and energy profiling software tools

Ultra-Low Power Modes with Quick Wake Time

Ultra-low power applications generally spend the most time in the lowest power mode, wake up to perform a task or measurement, and then quickly re-enter the lowest power mode. *(See Figure 1, "Energy = Power x Time.)* Therefore, it is extremely important to select an MCU with ultra-low power consumption and the fastest wake-up time.

A relatively slow wake-up time results in wasted energy because the MCU cannot perform any additional tasks during this time. As a result, MCUs with a slow wake-up time cannot achieve the optimal low-power consumption required by IoT connected devices.

Developers also must have a basic understanding of what is available in the lowest power mode. For example, are RAM and CPU retention supported? Is a real-time clock (RTC) enabled? And are power-on reset and brown-out detection safety features active? A good benchmark for this low-energy scenario with a Silicon Labs EFM32 Gecko MCU is approximately 900 nA power consumption in Energy Mode 2 (EM2) with a 2 µs wake-up time.

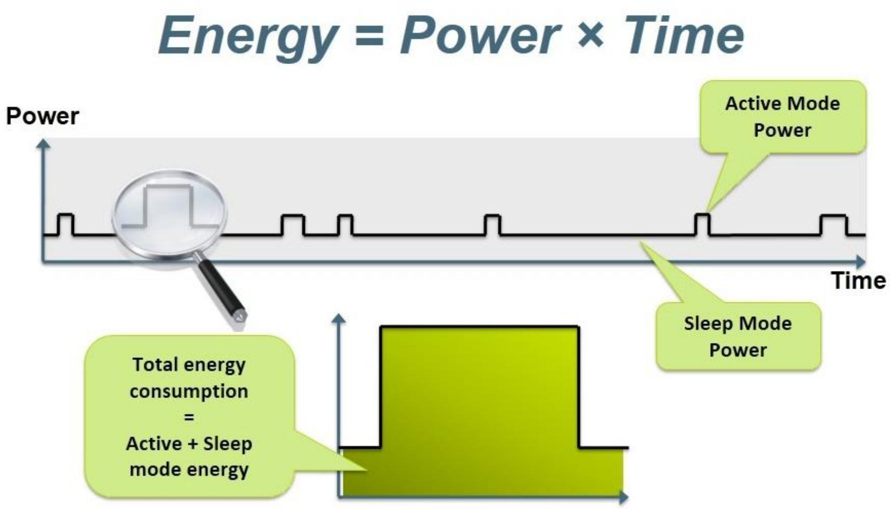

Figure 1. Minimizing energy consumption over time enables optimal energy efficiency.

EFM32 with 3 V power supply Application from memory	EM0 Run Mode	EM1 Sleep Mode	EM2 Deep Sleep	EM3 Stop Mode	EM4 Shutoff Mode
Current consumption	110 µA/MHz	45 µA/MHz	0.9 µA	0.6 µA	20 nA
Wake-up time	-	0	2 µs	2 µs	160 µs
CPU (Cortex-M3/M0)	On	-	-	-	-
High frequency peripherals	Available	Available	-	-	-
Low frequency peripherals	Available	Available	Available	-	-
Asynchronous peripherals	Available	Available	Available	Available	-
Full CPU and SRAM retention	On	On	On	On	-
Power-on Reset/Brown-out Detector	On	On	On	On	On
Wake-up events	Any	Any	32 kHz peripherals	Async IRQ, I^{2}C slave, Analog comparators Voltage comparators	Reset, GPIO rising/ falling edge

Table 1. Silicon Labs Energy Modes

Autonomous Peripherals That Operate in Low-Power Modes

An MCU that can remain in the lowest power mode as along as possible enables developers to achieve the utmost energy efficiency for IoT systems. In most cases, MCUs must wake up quickly to handle some event or take a measurement. By leveraging autonomous MCU peripherals (i.e., peripherals that do not require CPU intervention) in low-power modes, an application can reduce power consumption while still performing very advanced tasks.

To extend this idea, some autonomous peripherals can directly connect to other peripherals without involving the CPU. With this system architecture, a peripheral can produce signals that other peripherals can consume and instantly react to while the CPU remains in an ultra-low-power mode. For example, a timer can initiate an analog-to-digital (ADC) conversion, or a GPIO pin can initiate a UART transmission without

CPU intervention while in a low-power mode. This type of system architecture gives developers a wide variety of options to implement the most energy-friendly systems for IoT applications.

Power Estimation and Profiling Software Tools

Energy-aware software development tools are critical for ultra-low-power applications. Such tools can ultimately help developers reduce not only system power but also development time and costs by estimating battery life before the developer writes any code and determining which lines of code correspond to the highest system-level power consumption. During the information gathering stage of an IoT development effort, battery life estimators help developers quickly identify whether a given MCU solution can meet a battery-life specification. Within a few minutes of using the tool, developers will have a clear idea of the power-consumption capabilities of their application and whether the design meets their power budget.

During the development phase, sophisticated energy profiling tools can help developers quickly verify power consumption claims in data sheets as well as squeeze nanoamps out of their IoT application. By providing a real-time power consumption graph linked to lines of code, a profiling tool makes it very easy to determine when the most power is consumed by the MCU in addition to the code that triggered the higher power consumption. The developer simply clicks on the graphical output, and the profiling tool will show the corresponding line of code being executed. Energy profiling tools, such as Silicon Labs' energyAware profiler as shown in Figure 2, can be very helpful to developers trying to chase down any extra power consumption or meet a battery-life specification as project deadlines approach.

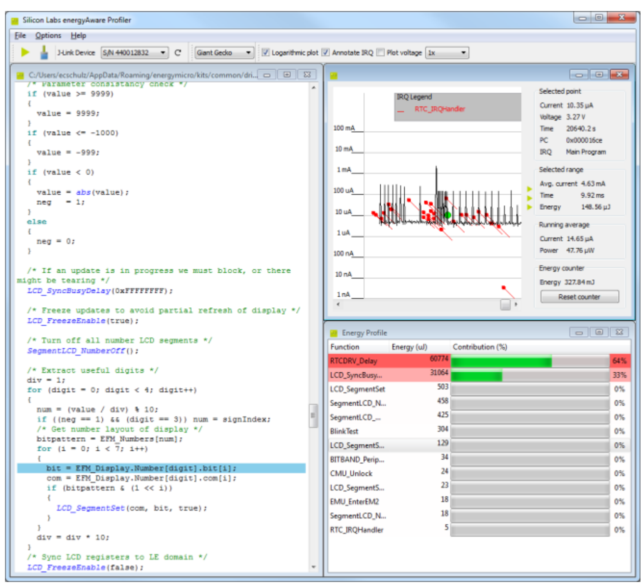

Figure 2. The energyAware Profiler associates system performance with the corresponding code

Enabling IoT Applications

MCUs based on an energy-friendly architecture and supported by sophisticated software development tools are essential to optimizing IoT connected device applications for ultra-low energy consumption. Because energy efficiency is paramount to IoT applications, developers should select MCUs with ultra-low power sleep modes and very fast wake-up times. Selecting an MCU vendor that provides energy-aware software development tools will also help developers create the most energy-friendly applications and extend the battery life of their IoT products by squeezing nanoamps out of their designs.

# # #