To Extend Battery Life, Use Two Processors (Not One)

John Dixon, J. B. Fowler, Zack Albus and Adrian Valenzuela, Texas Instruments

When system designers must extend battery life, many believe one chip uses less power than two. The reasons seem straightforward: Chip-to-chip communications consume more power than on-chip communications, and two chips will inevitably have more transistors and thus more leakage than a single chip with equivalent functions. But power-saving design techniques often turn conventional wisdom on its head.

Digital signal processor (DSP) designers have integrated more functions, such as accelerators, communications modules and network peripherals, onto DSP chips to make them more useful to engineers. But when one of these more capable chips powers up to perform simple housekeeping or supervisory tasks, it can dissipate more power than the tasks require. In many cases, the designer cannot turn on just the portion of the DSP chip they want to use.

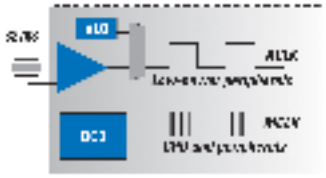
In some applications, a microcontroller (MCU) can perform equivalent system-supervision tasks and consume less power than a DSP. So, a dual-processor architecture -- DSP and MCU -- may make sense. Thus, by using a low-power DSP as a main processor and a lower-power MCU as a system supervisor, designers can extend battery life beyond what they could achieve with just one DSP. To help save power, engineers should consider the following points when they select a DSP:

- Look for large on-chip memories. Every time a DSP accesses off-chip memory, it uses extra power. External DRAM requires constant power, which can drain batteries.
- Select a DSP that allows activation and deactivation of peripherals. Several types of DSPs can automatically power down inactive on-chip peripherals. This capability offers several levels of control and power savings.
- Choose a DSP that offers multiple standby states at different power levels. More power options mean more energy savings.
- Select a DSP that offers development software that optimizes power use and minimizes power consumption. The tool should let developers easily alter the chip's voltage and frequency on the fly, manage power states and help them measure and analyze power-consumption information.

MCUs Draw Less Current

On the MCU side of an application, a low-power semiconductor process can reduce transistor-leakage currents and help chip designers optimize low-power operation. Unfortunately, low-power processes can limit MCU performance. A Texas Instruments MSP430 MCU, for example, can consume as little as 500 nA in standby

mode; but it has a maximum clock frequency of only 16 MHz. By comparison, a TMS320C5506 DSP operates at a maximum clock frequency of 108 MHz but uses 10 µA in standby mode. That represents a 20-fold increase above the current drawn by the MSP430.



Historically, software controlled internal MCU peripherals. This approach meant the CPU always remained active. But newer interrupt-driven peripherals that require little software overhead let the MCU stay in a standby mode most of the time. Internal ADC hardware, for example, can automatically scan input channels, trigger conversions and execute DMA transfers to handle repetitive data-sampling tasks. As a result, the ADC runs almost autonomously; the CPU spends little time servicing it; and the MCU conserves power.

Multiple Clocks Reduce Power Demands

An MCU's clock-system design also can help reduce power consumption. The diagram in Figure 1 shows two clocks that operate from a single crystal. The MCU typically uses one 32-kHz crystal, but it does not necessarily generate both of the internal clock signals, MCLK and ACLK. Typically, the crystal produces only the ACLK signal. The MCU's low-power peripherals use the 32-kHz auxiliary clock (ACLK) that also drives the MCU's real-time clock. A high-speed digitally controlled oscillator (DCO) produces a master-clock (MCLK) signal for the CPU and high-speed peripherals.

The DCO can generate a clock signal in several ways, each of which delivers different performance and power-consumption results. From low to higher power consumption, the clock modes range from VLO, 32-kHz crystal to DCO. To lower power consumption, designers use the lowest power clock (VLO or 32-kHz crystal) in idle modes and enable the higher frequency DCO when an application requires active processing from the CPU. The DCO can become active and fully stable in less than 1 µs. This "instant-on" capability saves time and power. Note that using the lower frequency low-power clocks during active processing would consume more power than switching to a faster clock. The low-frequency clocks would cause the CPU to spend more time on a given task in a higher power active mode.

In addition to the power saved by using a low-speed clock for some peripherals, MSP430 MCUs provide a very-low power oscillator (VLO) to produce the ACLK signal. In its standby power operating mode (LPM3), with ACLK running and all interrupts enabled, a MSP430 MCU typically consumes less than 1 µA. So, low-power MCUs use less energy than DSPs to maintain a real-time clock or manage battery charging procedures. Also, moving these types of tasks to an MCU frees a DSP to perform the signal-processing tasks at which it excels.

Power Savings Add Up

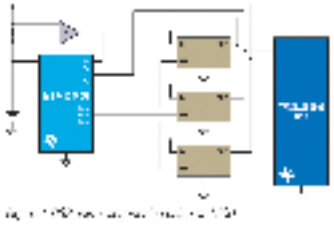
Engineers can see striking results delivered by a two-processor design. Consider a hypothetical system that relies on a high-end DSP to handle supervisory tasks. That processor would quickly deplete a typical 2,500 mAh NiMH AA battery. At an average current draw of 10 mA, two batteries in series would die in about 10.5 days. A dual-processor application would decrease the current to 1 mA, and the batteries would last about 120 days.

Some of the system or supervisory functions that an MCU can handle to decrease power consumption in a dual-processor system include:

- Real-time clock maintenance
- Power-supply sequencing
- Power-supply monitoring and reset
- Keypad or human-interface management
- Battery management
- Display control

Manage the DSP Power

Many DSPs require several power "rails" that must apply power in a set sequence for proper operation of the DSP and surrounding devices. Typically, these rails apply core (CPU) power as well as power for DDR memories and I/O devices. Although application-specific devices can apply voltages to a DSP chip in a set sequence, they cannot perform other functions. A small low-power MCU, though, can sequence and monitor supply voltages and perform power-control tasks (Figure 2). In this example, software turns on the three power-regulator circuits in the proper order. The MCU uses its internal ADC to verify when each supply rail has reached the proper voltage. When the overall circuit does not need the DSP chip, the MCU can shut down the regulators to turn off the DSP.



In fact, the MCU can control both the DSP's voltage and frequency by communicating directly with the voltage-controlled oscillator or with a PLL to control the DSP's clock frequency. Thus, when the DSP has finished a compute-intensive task, the MCU can adjust the clock and put the DSP into a standby mode to save power.

Bi-directional monitoring lets the MCU "probe" the DSP to see how busy it is. In this mode the MCU operates as an intelligent controller. On the other hand, the DSP can

both read from and write to the MCU. So the DSP can inform the MCU that it needs to slow down or speed up the DSP's clock, depending on the application's requirements.

Designers can reap additional benefits when they have an MCU take on other tasks that a DSP would normally perform in a one-processor system. An MCU can use less power than a DSP to handle keypad operations, for example. The MCU can send an interrupt to the DSP only after it detects a pressed and released key. This approach helps avoid excess current drain caused by a stuck key; a common occurrence in some handheld equipment. To further off-load a DSP chip, a supervisory MCU might also provide:

- A driver for a segmented LCD
- Standard SPI, UART and I^2C ports
- An interface for RF-communications peripherals
- Battery-management circuitry
- General-purpose I/O ports

For each type of peripheral listed above, and others mentioned previously, the MCU could "auto start" from a low-power mode. Thus, the MCU would not have to continuously poll peripherals to determine which ones need service -- and draw full power to do so. Peripherals would power up as needed.

Every milliamp is precious in low-power portable applications. Ultimately, designers must determine whether to use one or two processors in an application based on calculations, measurements and tradeoffs between functions and operations on a DSP or MCU.

Adrian Valenzuela *is a product marketing engineer for the Texas Instruments MSP430 16-bit microcontroller family. He has worked with the MSP430 since 2002. Valenzuela holds a BSEE and MSEE from Rice University.* ***Zack Albus*** *is the applications manager for TI's MSP430 16-bit microcontrollers. He supports customers with applications such as electricity meters, handheld instruments and controls. Zack holds BSEE and MSEE degrees from Texas Tech University.* ***John Dixon*** *is the low-power catalog DSP marketing manager for TMS320C5000 platforms. Dixon has three years of DSP marketing experience with TI's Semiconductor Group. John received a degree in electronics from Trinity College Dublin, Ireland and an MBA from the Michael Smirfit School of Business, Ireland.* ***John "J.B." Fowler*** *has worked as an applications engineer for Texas Instruments since 2000. He focuses on TI's TMS320C6000 DSP and DaVinci technology with an emphasis in video-security applications. J.B. graduated from the Colorado School of Mines with a BSEE and an MS in system engineering. For more information, contact Texas Instruments, 12500 TI Boulevard, P.O. Box 660199, Dallas, TX 75266-0199; (972) 995-2011; www.ti.com* [1]

Source URL (retrieved on *07/28/2014 - 7:55pm*):
http://www.ecnmag.com/articles/2007/07/extend-battery-life-use-two-processors-

not-one?qt-recent_content=0

Links:
[1] http://www.ti.com/