Pack a Punch in Battery Pack Systems with Low-cost MCUs

Evan Lew

Battery packs have enabled a new generation of personal electronics. Products with demanding power requirements like power tools, e-bikes, and electric vehicles have been redefined by advancements in high performance battery technology. Today, batteries must be safer, more efficient, and smarter than ever before. The increased demand for intelligent battery pack systems highlights the importance of picking the right MCU for the job. In this article we will discuss MSP430™ hardware and software features that can help solve some of the unique challenges in battery pack systems.

Battery Packs Background Information

High-performance battery cells such as lithium-ion (Li-ion) batteries require protection integrated circuits (ICs) such as the BQ77915 to ensure that the cells are operating within their temperature and current limits. Some systems require more information about the real-time status of the battery system. Battery protection ICs such as the BQ76952 offer more sophisticated features to monitor and control the battery system.

Generally, higher-end battery management ICs are paired with a microcontroller (MCU) for configuration and communication, as shown in Figure 1.
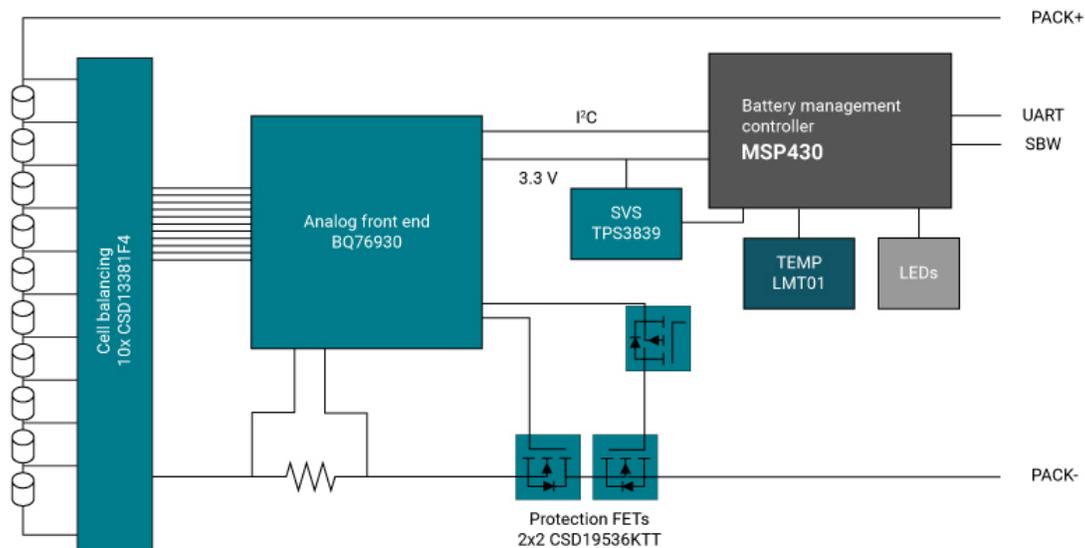
Figure 1. Block Diagram of a Li-ion Battery Pack

When a system requires more advanced features, such as state-of-charge tracking or data logging, an MCU becomes an even more important component in the system.

Reduce Bill of Materials with Low-cost Sensing

Some battery pack designs may use the MCU to detect and respond to different environmental conditions such as temperature. MSP430 features a collection of integrated analog peripherals known as Smart analog combo (SAC) on select devices. SAC can replace operational amplifiers and other analog signal-conditioning elements to reduce the bill of materials (BOM). Using discrete components consumes board space and increases costs. SAC is able to perform many signal-conditioning and sensing tasks with no external circuitry.

Furthermore, SAC is a runtime configurable peripheral that enables dynamic optimization and adjustment. For example, SAC has a programmable gain mode that is configurable across multiple designs without needing to change any external components such as feedback resistors. This mode can add flexibility during prototyping, simplify in-system calibration and reduce BOM costs in production by limiting external components.

In a battery pack system, SAC is configurable as a general-purpose operational amplifier and pairs with the integrated 12-bit analog-to-digital converter to perform current measurements. For example, SAC-enabled devices such as the MSP430FR2355 can perform low- or high-side current measurements. SAC can also take temperature measurements with minimal external system requirements.

Improve Reliability with High-performance Memory

New applications are demanding high performance from battery packs, making data-based algorithms such as state-of-health tracking and state-of-charge tracking more important. Algorithms that require frequent logging, low latency and robust data integrity may be challenging to support with low-cost MCUs in unstable power environments. However, MSP430 devices with ferroelectric random-access memory (FRAM) are well equipped for this type of challenge. FRAM is a high-performance memory technology that provides fast, low-latency, low-power nonvolatile memory. Why does this matter in a battery pack system? Because batteries may operate near their limits, which could trip overcurrent or thermal protection circuitry without any advance notice to the MCU. FRAM's low-latency greatly minimizes the chance of data corruption during an unexpected power down. The MSP430 MCU also offers a save-restore flow that allows software to seamlessly pick up where it left off after a power interruption.

Extend Battery Life with Low-power Features

Lower standby power consumption means longer run times, and more power available for demanding scenarios. MSP430 devices offer multiple sleep configurations to meet virtually any power requirement. For example, the MSP430FR2355 has an operating current of 142 µA/MHz during normal operation, a standby current of 620 nA and a shutdown current of 42 nA.

SAC can also help reduce system power usage. Since SAC is an onboard peripheral, it can power up dynamically when needed and subsequently power down to save power. Traditional signal-conditioning techniques use discrete operational amplifiers that draw power continuously, whether or not they are actively in use.

Software analysis tools such as EnergyTrace can help pinpoint and visualize energy-intensive software routines and help optimize code for low-power operation. Additionally, techniques in the Ultra-Low Power (ULP) Advisor tool offer easy ways to make code as power-efficient as possible. The ULP advisor is enabled by default in Code Composer Studio™ software and can be used with IAR Embedded Workbench as well.

Save Development Time with a Ready-to-go GUI

Getting initial prototypes up and running can be a challenging task. MSP430T offers a graphical user interface (GUI) to speed up development and debug for the BQ76952 as shown in Figure 2. Pairing an MSP430FR2355 LaunchPad™ development kit with the BQ76952 evaluation module (EVM) gives you a detailed view of important battery cell data without the headache of debugging interfaces and initialization routines.
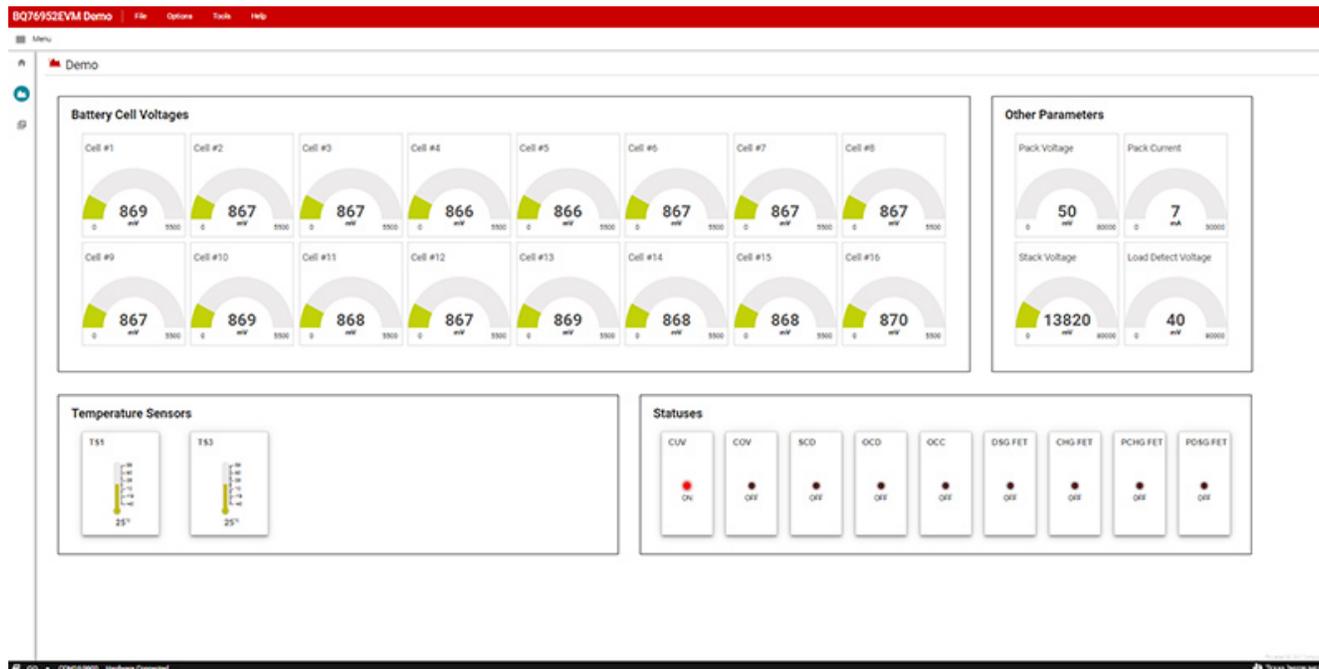

Figure 2. MSP+BQ Dashboard Interface

Conclusion

Battery pack systems of today present many design challenges and in the future will demand even higher energy density, top-notch safety, and more on-board intelligence. Choosing the right MCU for a design can help streamline product development enabling faster releases to market. MSP430's diverse portfolio of MCUs makes it easy to select a device with the right balance of performance, power, and cost. Check out the resources below to explore devices and technical resources.

Additional Resources

- Get started by exploring MSP430 devices.
- Download the application report, "EEPROM Emulation Using Low-Memory MSP430 FRAM MCUs."
- Learn more about compute through power loss and other FRAM features on the MSP FRAM utilities.
- To learn about the use of MSP430 low-power techniques in a thermostat, see the TI E2E article, "Step 6 to Build a Smart Thermostat Using an MCU: Energy Optimization."
- Check out the MSP+BQ graphical user interface (GUI).

IMPORTANT NOTICE AND DISCLAIMER