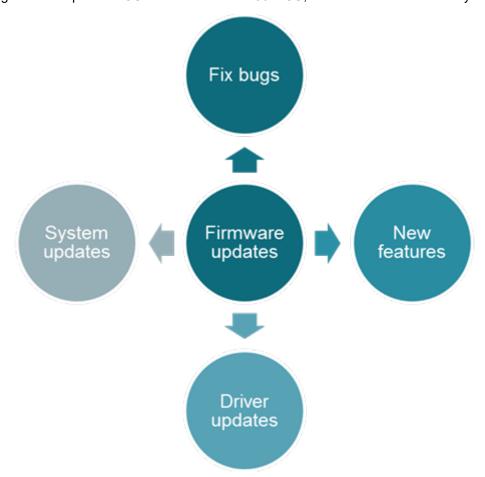*Technical Article*
Learn How to Apply Wireless Firmware Upgrades for MSP430 Microcontrollers

TEXAS INSTRUMENTS

Kris

As more and more devices become connected within the Internet of Things (IoT), and customer demand means that firmware and software updates become important product requirements, building the architecture for these updates is a key component in the up-front product design. Within the context of low-power devices, a *Bluetooth*® or USB connection to an MSP430™ microcontroller (MCU) is an obvious choice for providing over-the-air (OTA) updates, however for more power hungry applications which often include an accompanying processor alongside a low-power MCU such as the MSP430 MCU, a different mechanism may be preferred.

This post focuses on a short technical paper written on interfacing an MSP430 MCU with another off-the-shelf System on Chip (SoC) to provide MSP430 firmware updates through a SPI channel that connects the two processors. In the case of the SoC, it is Wi-Fi® connectivity enabled so that the user application software can directly access the device when it connects to a local area network (LAN) or through a Wi-Fi Direct transmission. The method of updating the MSP430 firmware is for the user to launch a file transfer directly to the SoC in which it can update its own firmware; the MSP430 firmware revision is then read through the SPI bus to determine if it also needs to be updated. The MSP430 MCU does provide its own unique solution called the Bootloader or BSL (http://www.ti.com/tool/mspbsl) to provide firmware updates, however certain design constraints or other

---

requirements may limit the use of the BSL and another mechanism may be required, such as the use of a SPI or UART bus.

One of the key aspects of MSP430 firmware updates is ensuring that all instruction execution happens within RAM while the Flash is getting overwritten. The white paper below helps to shed some light on the design considerations required to make this happen, as well as to provide some context for parsing and sending firmware files to the MSP430 MCU.

The intended applications for this type of design are numerous, however in the specific case of the referenced paper, the device designed was a wireless, battery-powered medical device making use of both a TI MSP430F5438AMCU and WiLink™ 8 Wi-Fi and Bluetooth combo connectivity module, as well as another higher power SoC used to perform intensive real-time calculations. The MSP430 MCU keeps the device in a low power standby mode and is then woken up through Bluetooth to boot-up the SoC for full operation over a Wi-Fi channel.

To learn more about wireless firmware upgrades for MSP430 MCUs, download my white paper:

- Wireless firmware upgrades for the MSP430 MCUs using a SPI connected SoC

IMPORTANT NOTICE AND DISCLAIMER