Revolutionizing context save and restore with MSP FRAM microcontrollers

Texas Instruments

William Cooper
MSP Product Marketing Engineer
Texas Instruments

William Goh
MSP Applications Engineer
Texas Instruments

Traditionally, MSP low-power microcontrollers (MCUs) have helped you do more than competition with a power budget that may be limited by a battery source or even inclusion within a current loop.

This was achieved by focusing on a number of characteristics that included industry leading standby-mode current consumption, minimal active-mode current consumption and integrated power-optimized peripherals. Now, with the integration of non-volatile FRAM offering nearly unlimited write endurance and speeds nearly 100x faster than Flash, there is an opportunity to think beyond conserving energy.

Doing something only possible on our ultra-low-power MSP430FRx FRAM microcontrollers, we created the Compute Through Power Loss (CTPL) software utility. This library is designed to save the CPU and peripheral states into FRAM memory, enabling two new possibilities:

1. Simplifies use of the MSP MCU's lowest power modes (LPMs)
2. Allows application to shut down gracefully in the event of a power loss and wake without resetting the system

The software library provides examples to detect if the system is entering LPMx.5 or losing power. The CTPL embedded software is designed to simplify the user experience using these low-power modes. LPMx.5 modes consume significantly lower current than other modes, but require developers to check if it the system is waking up from LPMx.5 or a cold start. Then, the developer would need to re-initialize the application and peripherals as LPMx.5 does not maintain any volatile data. CTPL APIs, abstract this startup algorithm and with the extra benefit of returning the application where it last left off by restoring the CPU state and the state of key peripherals (i.e. clocks, GPIO, etc). In software, the CTPL library essentially emulates the behavior of the system when entering LPM3. All of this is available without additional hardware. Furthermore, a system could leverage a buffered capacitor such that once line power is down, it can leverage the capacitor, shutting down all but essential functions and backing up critical data in FRAM with the provided CTPL APIs.

Figure 1 shows a flow-chart of the CTPL library functions behind the scenes in a full reference design. In the TI Design, Intelligent System State Restoration after Power Failure with Compute Through Power Loss Utility, an example is provided with options to call the API for entering LPMx.5 or entering shutdown mode.

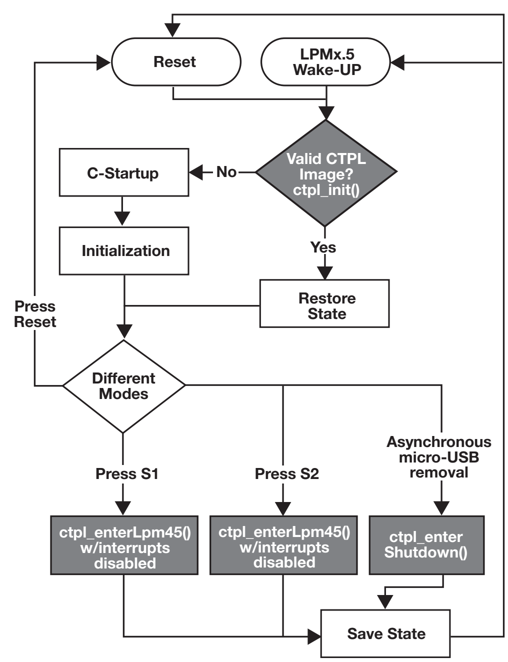

Figure 1: Flow of an Algorithm Leveraging Compute Through Power Loss

CTPL-enabled applications require the ability to constantly save and restore the MCU state into memory; hence, a high-endurance non-volatile memory technology is required. In this case, FRAM memory technology is a requirement with write endurance of greater than 10 trillion. This is not feasible with Flash or EEPROM memory technologies where typical write endurance is limited from 10,000 to 100,000 cycles. In addition to the limited write cycles, Flash memory technology first requires a pre-erase which typically has a high current consumption due to the required charge-pump. FRAM does not require a pre-erase to write nor does it have a charge-pump. These advantages and others described in **Figure 2** are truly opening up new possibilities across applications ranging from home to industrial automation.

Taking a deeper look from an application perspective, energy harvesting is becoming an interesting option as a way to expand battery life and eliminate potential maintenance costs. In an energy harvesting system, energy can be scarce and the system would need to complete its tasks as soon as possible before the energy is consumed. In this situation, it is crucial that your application is optimized for execution-time. When power is available, an application must execute a startup routine and re-initialize various peripherals before entering main(). This can be time and energy intensive. The CTPL library enables automatically restoring your application in an efficient manner by restoring your CPU states, peripheral states, and stack FRAM to its appropriate location. When the application variables are placed in FRAM, the system no longer needs to be re-initialized. With this state storage in fast, non-volatile memory, CTPL reduces your C-code based application start-up time which in turn saves energy. This energy savings can vary from application to application depending on how many variables the compiler needs to initialize. Another time saving benefit of the CTPL software is the return of the application program counter to where the application last executed. One possible use-case of energy harvesting is a solar powered irrigation system. Energy is available during the day to store charge and runs the system as needed. The CTPL utility can enable use of the lowest power state during inactivity and could periodically log temperature, moisture, and valve position data. MSP430FR5x/6x MCUs also feature peripherals such as a 12-bit ADC with internal window comparator that can store data in memory and alert the application when an analog sensor exceeds a threshold without CPU intervention. The ADC can also be used to detect when the system is about to lose power.

All-in-one: FRAM MCU delivers max benefits				
Specifications	**FRAM**	**SRAM**	**EEPROM**	**Flash**
Non-volatile Retains data w/o power	Yes	No	Yes	Yes
Write Speed (13 KB)	10ms	<10ms	2 secs	1 sec
Average active Power [µA/MHz] 16 bit word access by the CPU	100	<60	50,000+	230
Write endurance	10^{15}	Unlimited	100,000	10,000
Soft Errors	Below Mesuarable Limits	Yes	Yes	Yes
Bit-wise programmable	Yes	Yes	No	No
Unified Memory Flexible code and data partitioning	Yes	No	No	No

*Based on devices from Texas Instruments

Figure 2: The FRAM Advantage Quantified

Another application that could leverage compute through power loss, is one with battery power that only needs to be active when a key is pressed or if tampering is detected. Most home electronic door locks are powered by batteries which mean ultra-low-power is crucial. During inactive state, the electronic door lock can leverage CTPL. Many MSP FRAM MCUs also have options of a built-in hardware encryption (256-bit AES module) and a hardware Intellectual Protection Encapsulation (IPE) unit as a secondary level of security to protect secret keys. The IPE module is capable of securing a segment of memory from any unwanted reads or writes during application execution.

In an industrial space, water meters have a permanent battery for the entire life of the product. And depending on country-specific regulations, water meters can last for a minimum of 8-years on a single battery. The battery is often the most expensive item on the bill of materials. Country regulation can change over time towards longer minimum operation in which a larger battery may be needed. Or, the application could leverage lower power standby currents from the MCU. CTPL can be leveraged to prolong the battery life during storage, shipping, and operation of the water meter. Several of the MSP430FRx microcontrollers even have a built-in Extended Scan Interface (ESI) module which is an analog front-end designed specifically to measure rotation as commonly required for measuring flow. In addition, with the FRAM technology, the application gets data-logging capabilities without the cost of an external EEPROM.

To evaluate this technology and discover how it can add value in your designs, check out the MSP-EXP430FR5969 Launchpad Development Kit and see it in action with the reference design that leverages the power of FRAM and this new Compute Through Power Loss software utility.