Application Report

Migrating from the MSP430F2xx and MSP430G2xx families to the MSP430FR4xx and MSP430FR2xx family

Lixin Chen, Darren Lu, Xiaodong Li *MSP430 Applications*

ABSTRACT

This application report helps to ease the migration from MSP430F2xx flash-based MCUs to the MSP430FR4xx and MSP430FR2xx FRAM-based MCUs. It discusses programming, system, hardware, core architecture, and peripheral considerations. The intent is to highlight key differences between the two families. For more information on the use of the MSP430FR4xx and MSP430FR2xx devices, see the MSP430FR4xx and MSP430FR2xx family user's guide. Although the MSP430F2xx and MSP430G2xx families are used as a base for comparison, similar considerations apply when migrating from MSP430F1xx and MSP430F4xx families. For the migration guide to MSP430FR57xx, see Migrating from the MSP430F2xx family to the MSP430FR57xx family.

Table of Contents

List of Figures

List of Tables

Trademarks

MSP430™, CapTIvate™, E2E™, MSP430Ware™, and Code Composer Studio™ are trademarks of Texas Instruments.
IAR Embedded Workbench® is a registered trademark of IAR Systems.
All trademarks are the property of their respective owners.

1 Introduction

The MSP430F2xx and MSP430G2xx devices and the MSP430FR4xx and MSP430FR2xx devices are part of TI's MSP430™ ultra-low-power sensing & measurement MCUs portfolio.

- Most of the MSP430FR4xx and MSP430FR2xx devices are part of the MSP430 value line & general purpose microcontrollers portfolio.
- The MSP430FR25xx and MSP430FR26xx devices are part of the MSP430 capacitive touch sensing microcontrollers portfolio. Migrating to these devices is similar to other family members except for the use of CapTIvate™ technology. For information about this technology and its ecosystem, visit the capacitive touch sensing portfolio.
- MSP430F2xx and MSP430G2xx devices are part of the MSP430 MCUs for metrology, monitoring and system control portfolio.

The purpose of this application report is to highlight the key differences between the MSP430F2xx and MSP430G2xx families and the MSP430FR4xx and MSP430FR2xx family to ensure a smoother migration. It is divided into:

- Changes when handling nonvolatile memory
- System-level considerations such as power management and hardware
- Peripheral modifications

With respect to the instruction set, the MSP430FR4xx family is completely backward code compatible with all other MSP430™ families. Any code migration is therefore affected only by register or peripheral feature changes and slight variations in instruction cycle times, while the instruction set remains the same. For any specific information, see the device-specific data sheet and errata.

Note

For the purpose of this application report, the term *F2xx* indicates the MSP430F2xx and MSP430G2xx families, and the term *FR4xx* indicates the MSP430FR4xx and MSP430FR2xx family.

2 Comparison of MSP430FR4xx and MSP430FR2xx Devices

Table 2-1 summarizes the primary differences among the MCUs in the MSP430FR4xx and MSP430FR2xx family.

Table 2-1. Comparison of Features

Feature or Module	FR413x, FR203x	FR2433, FR263x, FR253x	FR231x	FR21xx, FR2000	FR235x, FR215x	FR267x, FR247x
CPU	16-MHz MSP430	16-MHz MSP430	16-MHz MSP430	16-MHz MSP430	24-MHz MSP430	16-MHz MSP430
Program FRAM	15.5KB or 8KB	15.5KB or 8KB	3.75KB or 2KB	3.75KB, 2KB, 1KB, or 0.5KB	32KB or 16KB	64KB or 32KB
Information FRAM	512 bytes	512 bytes	N/A	N/A	512 bytes	512 bytes
SRAM	2KB or 1KB	4KB, 2KB, or 1KB	1KB	1KB or 0.5KB	4KB or 2KB	8KB or 6KB
Maximum GPIOs	60	19	16	12	44	43
Interrupt pins	16 (P1 and P2)	16 (P1 and P2)	12 (8 pins of P1 and 4 pins of P2)	8 (4 pins each of P1 and P2)	32 (P1, P2, P3 and P4)	All GPIOs
USCI	1 eUSCI_A, 1 eUSCI_B	2 eUSCI_A, 1 eUSCI_B	1 eUSCI_A, 1 eUSCI_B	1 eUSCI_A	2 eUSCI_A, 2 eUSCI_B	2 eUSCI_A, 2 eUSCI_B
ADC	ADC10 (10 channel)	ADC10 (8 channels)	ADC10 (8 channels)	ADC10 (8 channels)	ADC12 (12 channels)	ADC12 (12 channels)
Comparator	N/A(1)	N/A	1	1	1 LP eCOMP, 1 HS eCOMP	1 LP eCOMP
Analog features	N/A	N/A	1 SAC-L1 (OA), 1 TIA	N/A	4 SAC-L3	N/A
Timer	2 Timer_A with 3CC(2), RTC counter, WDT	2 Timer_A with 3CC, 2 Timer_A with 2CC, RTC counter, WDT	2 Timer_B with 3CC, RTC counter, WDT	1 Timer_B with 3CC, RTC counter, WDT	3 Timer_B with 3CC, 1 Timer_B with 7CC, RTC counter, WDT	4 Timer_B with 3CC, 1 Timer_B with 7CC, RTC counter, WDT
Additional features	Temperature sensor, brownout reset, capacitive touch I/O, LCD in FR4133	Temperature sensor, brownout reset, MPY32, CapTIvate™ technology in FR263x and FR253x	Temperature sensor, brownout reset, capacitive touch I/O	Temperature sensor, brownout reset, capacitive touch I/O	Shared voltage reference for ADC, DAC, and eCOMP, low-power REFO selectable, temperature sensor, brownout reset, capacitive touch I/O	Shared voltage reference for ADC, DAC, and eCOMP, low-power REFO selectable, temperature sensor, brownout reset, MPY32, CapTIvate technology in FR267x
BSL	UART	I^2C, UART	I^2C, UART	UART	I^2C, UART	I^2C, UART
V_{CC}	1.8 V to 3.6 V	1.8 V to 3.6 V	1.8 V to 3.6 V	1.8 V to 3.6 V	1.8 V to 3.6 V	1.8 V to 3.6 V
Active power	126 µA/MHz	126 µA/MHz	126 µA/MHz	126 µA/MHz	142 µA/MHz	135 µA/MHz
Operating temperature	–40°C to 85°C	–40°C to 85°C	–40°C to 85°C	–40°C to 85°C	–40°C to 105°C	–40°C to 105°C
Package	LQFP64, TSSOP56, TSSOP48	VQFN24	TSSOP20, TSSOP16, QFN16	TSSOP16, QFN24	LQFP48, QFN40, TSSOP38	LQFP48, VQFN40, VQFN32

(1) N/A = not available
(2) CC = capture/compare registers

Table 2-2 summarizes the memory maps of the typical MCUs in the MSP430FR4xx and MSP430FR2xx family.

Table 2-2. Comparison of Memory Maps

	Access	FR4133	FR2633	FR2311	FR2111	FR2355	FR2676
Memory (FRAM)	R/W Optional write protect	15KB	15KB	3.75KB	3.75KB	32KB	64KB
Main: interrupt vectors and signatures		FFFFh to FF80h	FFFFh to FF80h	FFFFh to FF80h	FFFFh to FF80h	FFFFh to FF80h	FFFFh to FF80h
Main: code memory		FFFFh to C400h	FFFFh to C400h	FFFFh to F100h	FFFFh to F100h	FFFFh to 8000h	17FFFh to 8000h
Information Memory (FRAM)	R/W Optional write protect	512B 19FFh to 1800h	512B 19FFh to 1800h	N/A	N/A	512B 19FFh to 1800h	512B 19FFh to 1800h
RAM	R/W	2KB 27FFh to 2000h	4KB 2FFFh to 2000h	1KB 23FFh to 2000h	1KB 23FFh to 2000h	4KB 2FFFh to 2000h	8KB 3FFFh to 2000h
ROM BSL	R	1KB 13FFh to 1000h	2KB 17FFh to 1000h 1KB FFFFFh to FFC00h	2KB 17FFh to 1000h 1KB FFFFFh to FFC00h	1KB 13FFh to 1000h	2KB 17FFh to 1000h	2KB 17FFh to 1000h 1KB FFFFFh to FFC00h
Peripherals	R/W	4KB 0FFFh to 0020h	4KB 0FFFh to 0020h	4KB 0FFFh to 0020h	4KB 0FFFh to 0020h	4KB 0FFFh to 0020h	4KB 0FFFh to 0020h
ROM Library	R	N/A	CapTIvate libraries and driver libraries, 12KB 6FFFh to 4000h	N/A	N/A	CapTIvate libraries, FFT and driver libraries, 20KB FFBFFh to FAC00h	CapTIvate libraries, FFT and driver libraries, 16KB C3FFFh to C0000h

The registers of the SYS module differ by device. For details, see the MSP430FR4xx and MSP430FR2xx family user's guide.

3 In-System Programming of Nonvolatile Memory

3.1 Ferroelectric RAM (FRAM) Overview

Unlike the F2xx family, which has flash memory integrated, the MSP430FRxx devices use FRAM nonvolatile memory. Using FRAM is very similar to using static RAM (SRAM). The introduction of FRAM as an embedded memory in a general-purpose ultra-low-power MCU was on TI's 16-bit MSP430 product line.

Some of the key attributes of FRAM are:

- FRAM is nonvolatile, that is, it retains its contents on loss of power.
- The embedded FRAM on MSP430 devices can be accessed (read or write) at up to a maximum speed of 8 MHz. Above 8 MHz, wait states are used when accessing FRAM.
- Writing to FRAM and reading from FRAM require no setup or preparation such as erase before write or unlocking of control registers (unless the write protection bit is used to protect the FRAM against write access).
- FRAM is not segmented and each bit is individually erasable, writable, and addressable.
- FRAM does not require an erase before a write.
- FRAM write accesses are low power because writing to FRAM does not require a charge pump.
- FRAM writes can be performed across the full voltage range of the device.
- FRAM write speeds can reach up to 8 MBps with a typical write speed of approximately 2 MBps. The high speed of writes is inherent to the technology and aided by the elimination of the erase bottleneck that is prevalent in other nonvolatile memory technologies [9]. In comparison, typical MSP430 flash write speed including the erase time is approximately 14 kBps [9].
- FRAM has far greater write endurance compared to flash: practically unlimited 10^{15} write cycles of FRAM compared to 10^{5} write cycles of flash.

3.2 FRAM Cell

A single FRAM cell can be considered a dipole capacitor that consists of a film of ferroelectric material (ferroelectric crystal) between two electrode plates. Storing a 1 or 0 (writing to FRAM) simply requires polarizing the crystal in a specific direction using an electric field. This makes FRAM very fast, easy to write to, and capable of meeting high-endurance requirements.

Reading from FRAM requires applying an electric field across the capacitor similar to a write. Depending on the state of the crystal, it may be repolarized, thereby emitting a large induced charge. This charge is then compared to a known reference to estimate the state of the crystal. The stored data bit 1 or 0 is inferred from the induced charge. In the process of reading the data, the crystal that is polarized in the direction of the applied field loses its current state. Hence, every read must be accompanied by a write-back to restore the state of the memory location. With TI's MSP430 FRAM MCUs, this is inherent to the FRAM implementation and is completely transparent to the application. The write-back mechanism is also protected from power loss and completes safely under all power-fail events.

The FR4xx power management system achieves this by isolating the FRAM power rails from the device supply rails in the event of a power loss. The FRAM power circuitry also uses built-in low-dropout regulator (LDO) and a capacitor that stores sufficient charge to complete the current write-back in the event of a power failure.

See Section 7.5 for more information about FRAM and the FRAM controller.

3.3 Protecting FRAM Using the Memory Write Protection Bit

Because FRAM is very easy to reprogram, it also makes it easy for erroneous code execution to unintentionally overwrite application code, just as it would if executing from RAM.

To safeguard against erroneous overwriting of FRAM, memory write protection is provided. FR4xx provides two separate write protection bits:

- SYSCFG0.PFWP – User program FRAM protection
- SYSCFG0.DFWP – User data FRAM (information memory 1800h to 19FFh) protection

When writing the SYSCFG0 register, write the protection password (SYSCFG0.FRWPW = 0xA5) in a word with the other bits.

Table 3-1 summarizes the FRAM protection features in the typical devices of MSP430FR4xx and MSP430FR2xx family.

Table 3-1. FRAM Protection Features

Feature	FR2033	FR4133	FR2433	FR2311	FR2111	FR2355	FR2476
Protection password (SYSCFG0.FRWPPW)	N	N	Y	Y	Y	Y	Y
The program FRAM write protection (SYSCFG0.PFWP)	Y	Y	Y	Y	Y	Y	Y
The data FRAM protection (SYSCFG0.DFWP)	Y	Y	Y	N	N	Y	Y
Program FRAM write protection offset address (SYSCFG0.FRWPOA)	N	N	N	N	N	Y	Y

When a write protection bit is set, any write to the protected FRAM is blocked but does not generate an interrupt or reset.

Note

FRAM write protection should be enabled at all times except when a write operation is required. TI recommends completing the write operation within a short time with interrupts disabled to reduce the risk of unintended write operations. After the write operation completes, TI highly recommends enabling the FRAM write protection again as soon as possible to reduce the risk of unintended writes.

When writing data to FRAM, TI recommends using the __persistent attribute to define variables in FRAM memory. Code examples on how to write the FRAM are provided in the "Tools & software" tab of each FR4xx product folder (for example, MSP430FR4133 or MSP430FR2433). For more information about using FRAM technology in MSP430 MCUs from the software development perspective, see MSP430 FRAM technology – how to and best practices.

3.4 FRAM Memory Wait States

The maximum FRAM memory access speed is 8 MHz. If the MCLK is operating faster than 8 MHz and FRAM access is required, wait states are necessary to set to ensure reliable FRAM access. When using MCLK ≥ 8 MHz, configure the FRAM wait states in software before configuring the MCLK frequency.

1. Configure the appropriate wait states.

```
FRCTL0 = FRCTLPW | NWAITS_x
```

2. Configure MCLK ≥ 8 MHz.

For more information, see the *Wait State Control* section of the *FRAM Controller (FRCTRL)* chapter in MSP430FR4xx and MSP430FR2xx family user's guide.

3.5 Bootloader (BSL)

The BSL is software that is used to reprogram the MCU, for example, during field firmware updates. On the F2xx family of devices, the BSL uses a Timer_A-based UART and is located in ROM. The BSL is not erasable or customizable by the user.

The FR4xx family follows a similar approach in which the BSL software is located in ROM. It cannot be erased or reprogrammed. For the BSL address range, see the device-specific data sheet. In terms of the communication interface, the FR4xx is also based on the UART protocol similar to the F2xx devices. However, the FR4xx uses the hardware eUSCI_A module to implement UART communication instead of using Timer_A. Hence, the module pins UCA0TXD and UCA0RXD are used for BSL communication. And the $\overline{\text{RST}}$/NMI/SBWTDIO pin and TEST/SBWTCK pin are used for BSL entry sequence.

The BSL on the MSP430FR2x devices (except for the MSP430FR203x) supports both UART and I^2C communication. This BSL scans the UART and I^2C peripherals to detect which interface is used by the host programmer. When a transmission is detected on one of the peripherals, that interface is selected and the other

is disabled by the BSL. On the F2xx family of devices, only the UART BSL is available. For details, see the MSP430 FRAM device bootloader (BSL) user's guide.

The BSL in the FR4xx family can be disabled by programming a specific signature to the BSL signature location. For details, see Table 3-2. Note that the location and length of the BSL signature is different from the F2xx family.

This process is documented in the SYS chapter in the MSP430FR4xx and MSP430FR2xx family user's guide.

3.6 JTAG and Security

On F2xx devices, the JTAG port is secured by blowing a physical fuse on one of the JTAG lines by subjecting it to a high voltage through a special procedure. This action is irreversible, and further access to the device is only possible through the BSL.

On the FR4xx devices, there is no physical fuse. The JTAG/SBW is locked by programming a specific signature into the device's FRAM memory at JTAG/SBW signature address of FF80h to FF83h.

When JTAG/SBW is locked by programming the JTAG/SBW signature, access to the device is only possible through the BSL (using the BSL password). However, when the BSL is not disabled and the BSL password is supplied, it is possible to clear the JTAG/SBW signature and make JTAG communication available again. Hence, on the FR4xx devices, locking the JTAG/SBW is reversible if the BSL password is known and BSL is not disabled.

Table 3-2 describes the FR4xx device password, BSL signature, and JTAG/SBW signature.

Table 3-2. FR4xx Device Password, BSL Signature, and JTAG/SBW Signature

Name	Address	Value	Device Security	BSL and SBW Behavior After Reset
Device password	FFE0h to FFFFh	Depending on vector table configuration		The value is used to protect BSL.
BSL signature	FF84h to FF87h	5555_5555h	Secured, password not required	BSL is bypassed. User code starts immediately.
		Any other values	Secured, password required through the BSL	BSL is invoked before user code starts if BSL is triggered.
JTAG/SBW signature	FF80h to FF83h	FFFF_FFFFh	Not secured	JTAG/SBW is not locked.
		0000_0000h		
		Any other values	Secured	JTAG/SBW is locked.

3.7 Production Programming

The MSP-GANG430 in-system gang programmer does not support FR4xx MCUs. MSP-GANG430 has been superseded by the MSP-GANG Production Programmer.

4 Hardware Migration Considerations

- For JTAG and SBW connections on the FR4xx devices, see the MSP430 hardware tools user's guide. Note the parallel capacitor on the $\overline{\text{RST}}$/NMI/SBWTDIO pin must be less than 1.1 nF when using SBW for debug or firmware download.
- The FR4xx devices provide an internal pullup resistor on the reset line, which eliminates the need for an external reset resistor. The internal pullup resistor is enabled by default. For details, see the "Reset Pin ($\overline{\text{RST}}$/NMI) Configuration" section in the MSP430FR4xx and MSP430FR2xx family user's guide.
- The FR4xx devices do not provide internal load capacitors on the LFXT oscillator as in the F2xx family. Hence, it is required to have external load capacitors if the LFXT oscillator is used. For layout, the external crystal should be as close as possible to the XIN and XOUT pins on the FR4xx MCUs. Place the load capacitors as close as possible to the crystal pins. Match the capacitor values to the crystal specifications and PCB layout. For more guidance about crystal selection, layout concerns, and crystal oscillator testing, see MSP430 32-kHz crystal oscillators.
- Compared to the F2xx family, the FR4xx clock system is quite different. There is an FLL and an internal trimmed REFO in FR4xx devices that can generate a REFO clock at 32.768 kHz with an accuracy of ±3.5%. For detailed information, see Section 7.2.2 and the "Clock System" chapter of the MSP430FR4xx and MSP430FR2xx family user's guide.
- In the FR4xx family, only the MSP430FR231x MCUs support a high-frequency clock source on the XT1 oscillator. The FLL reference divider FLLREFDIV is available only when XT1 HF mode is supported in the device.
- Instead of having analog voltage supply pins (AVCC and AVSS) and digital voltage supply pins (DVCC and DVSS) as on F2xx devices, there is only one pair of power supply pins (DVCC and DVSS) on FR4xx devices.

5 Device Calibration Information

Some F2xx devices provide a TLV structure that supplies calibration values for the DCO frequency, ADC reference, and internal temperature sensor. The TLV structure is stored in information memory segment A (Info A) and can be erased by the user. A mass erase of the device that occurs if an incorrect BSL password is supplied also erases the factory-calibrated constants.

To prevent this, the TLV information on FR4xx devices is stored in protected FRAM area where it cannot be erased by unintended write operation. For details on the location and access of the TLV, see the device-specific data sheet.

Note

Another information memory area at 1800h to 19FFh is available for application use on the FR4xx devices, except on the MSP430FR231x and MSP430FR211x devices, with write protection bit of SYSCFG0.DFWP (see Section 3.3).

The TLV structure contains calibration values that can be used to improve the measurement accuracy of various functions. The calibration values available on a given device are shown in the TLV structure of the device-specific data sheet. In the FR4xx data sheets, ADC offset and gain calibration data, and temperature sensor calibration data are provided.

For more information about how to use the TLV in the FR4xx devices, see the device descriptor section in the MSP430FR4xx and MSP430FR2xx family user's guide.

6 Important Device Specifications

Table 6-1 shows important differences in device-level electrical specifications [3][4].

Table 6-1. Device Specifications

Parameter	FR4xx	F2xx
Supply voltage range	1.8 V to 3.6 V(1) (2)	1.8 or 2.2 V to 3.6 V
Maximum system frequency, f_{SYSTEM} (3)	16 MHz at V_{CC} = 1.8 V	4 MHz at V_{CC} = 1.8 V 8 to 12 MHz at V_{CC} = 2.7 V 8 to 16 MHz at V_{CC} = 3.3 V
Minimum supply voltage for nonvolatile memory programming	1.8 V	2.2 V
Minimum analog supply voltage for ADC operation	2.0 V	2.2 V

(1) The minimum operating voltage depends on the SVSH voltage levels.
(2) Supply voltage changes faster than 0.2 V/µs can trigger a BOR reset even within the recommended supply voltage range.
(3) See the device-specific data sheet for the specified operating conditions.

A significant improvement that migrating to an FR4xx device brings to your system is in terms of the power consumption. The FR4xx demonstrates significant improvement in active and standby power consumption, during both typical conditions and across the voltage and temperature range of the device. The same is also true for peripherals in the FR4xx family, such as the ADC, which show significant power improvements when compared to their F2xx counterparts. For details on the power consumption for each peripheral and of the device in active and standby modes, see the device-specific data sheet.

7 Core Architecture Considerations

7.1 Power Management Module (PMM)

7.1.1 Core LDO and LPM3.5 LDO

The F2xx family devices use a single voltage rail to power the chip, that is, a single power rail supplies both the analog peripherals and the digital core on the chip. The FR4xx family also uses a single voltage supply. However unlike F2xx, there are no AVCC and AVSS pins, only DVCC and DVSS. The external voltage supply on the DVCC pin is fed to an internal low-dropout voltage regulator (LDO) (see Figure 7-1).

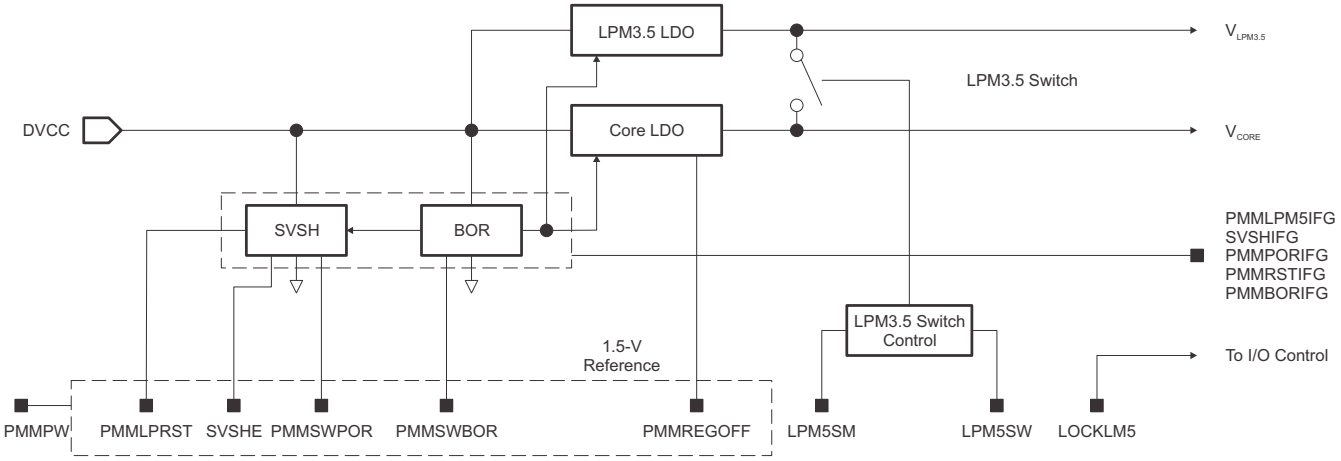

Figure 7-1. PMM Block Diagram

The PMM manages all functions related to the core voltage and its supervision. Its primary functions are, first, to generate a supply voltage for the core logic and, second, to provide several mechanisms for the supervision of both the voltage supplied to the device (DVCC) and the voltage generated for the core (VCORE).

Using the PMM is especially advantageous as it allows the core to operate at a lower voltage, which brings significant power savings. It also ensures that the core receives a stable and regulated voltage over a wide supply range.

This allows the FR4xx devices to operate across the entire voltage range of the device at the maximum device frequency of 16 MHz. In comparison, F2xx devices have a relationship between system frequency and supply voltage that must be followed to ensure proper operation of the device (see Table 6-1).

There is a second LDO integrated in FR4xx, the LPM3.5 LDO. This LDO supplies the current for the LPM3.5 power domain logic, which contains the RTC and LCD modules (only the MSP430FR4xx MCUs have the LCD module). In LPMx.5 low-power modes, the core LDO is turned off. When entering LPM3.5, the LPM3.5 switch is turned off to save power consumption. When exiting LPM3.5 power mode, the LPM3.5 switch is turned on so that the core LDO can supply power to the LPM3.5 domain logic to support high-frequency operation (see Figure 7-1).

7.1.2 SVS

Because supply voltage supervision (SVS) is an important aspect of providing a stable supply or a notification in case of power failure, the FR4xx provides a high-side supply voltage supervision (SVSH) block. The SVSH supervises the external chip supply (DVCC), and the PMM internally supervises the low-side supply to the core.

In FR4xx devices, the SVS threshold tracks directly with the device minimum supply of 1.8 V, and there is no need to program SVS high-side levels as in the F2xx family (in which the SVS feature is available on selected devices only). Also, the SVSH block in the FR4xx is highly simplified. It is on by default at power-up and stays on. It can trigger a BOR reset when the supply falls below the SVS level. It can be turned off in LPM3, LPM4, and LPMx.5 modes by setting SVSHE = 0, if required.

7.1.3 VREF

Unlike in F2xx, the FR4xx includes a VREF generation block and a high-accuracy bandgap in the PMM module designed for low-power applications. Two voltage references are generated for internal use (1.5-V V_{REF}) and external use (1.2-V V_{REF}) (see Figure 7-2).

The 1.5-V V_{REF} is connected to the ADC module and can be used as a reference voltage for the ADC. It is also internally connected to ADC channel 13. This makes a possible to monitor the DVCC voltage by using the ADC sampling 1.5-V V_{REF} (set DVCC as the ADC reference) without the support of any external components. For detailed information, see the sections "Power Management Module (PMM)" and "On-Chip Reference Voltages" in the device-specific data sheet.

The 1.2-V V_{REF} can be buffered and output to a pin when the ADC channel on that pin is selected as the function. To determine which pin can output the 1.2-V refererence, see the device-specific data sheet. The 1.2-V V_{REF} has 1-mA drive capability (see Figure 7-2). For more detailed information, see the PMM and ADC chapters in the MSP430FR4xx and MSP430FR2xx family user's guide.

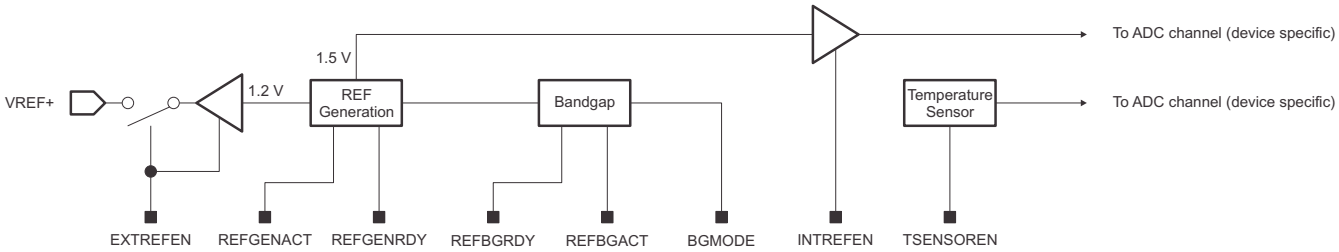

Figure 7-2. VREF Generation Block Diagram

7.1.4 Debug in Low-Power Mode

One of the main differences between the two families that can be observed while debugging can be attributed to the PMM module. In the FR4xx family, the VCORE regulator operates in two modes to conserve power: high-performance mode (used in active and LPM0 modes) and low-power mode (used in LPM3 and LPM4 modes). When the FR4xx device is plugged into the debugger, it automatically forces the LDO to the high-performance mode regardless of the operating mode (active or LPM) that is set by the application code. In an application, this affects current consumption and wake-up times, which may cause the device to behave differently between stand-alone and debugger modes. When debugging with the lower LPMs (LPM3 and LPM4), ensure that the debugger is disconnected to observe device performance accurately.

For LPMx.5, the debugger mode is not supported because the core LDO is turned off. See Section 7.3.1 for more information about LPMx.5 low-power modes.

7.2 Clock System

7.2.1 DCO Frequencies

The F2xx basic clock system (BCS) uses an internal digitally controlled oscillator (DCO) to provide precalibrated frequencies. Unlike F2xx, the FR4xx clock system (CS) uses internal digitally controlled oscillator (DCO) plus a frequency-locked loop (FLL) to provide frequencies.

A significant difference in the FR4xx DCO+FLL is that it can be configured only to the factory-provided frequencies and does not provide the in-between frequency steps that are available on the F2xx DCO.

If an in-between frequency is necessary when using an FR4xx MCU, there is a complex method to implement it using the DCOFTRIMEN and DCOFTRIM bits in user code. For detailed information, post on the MSP430 E2E™ forum with the tags "MSP430FR4xx" and "SLAA649".

The FR4xx MCUs provide the same clock source options and system clocks as the F2xx MCUs. However, the clock distribution system is simplified. For example, the SMCLK is derived from MCLK. There is a SMCLK divider (DIVS) between MCLK and SMCLK. However, the MCLK divider DIVM also affects SMCLK. For more detailed information, see the clock system chapter in the MSP430FR4xx and MSP430FR2xx family user's guide. Each FR4xx device may have a differences in its clock distribution system (see the clock distribution table in the device-specific data sheet for details).

7.2.2 FLL, REFO, and DCO Tap

Another significant difference in the FR4xx CS module is that it has the frequency-locked loop (FLL) and internal trimmed low-frequency reference oscillator (REFO), which are not integrated in the F2xx BCS module.

The FLL stabilizes the DCO frequency to a programmable multiple of the selected FLL reference frequency of FLLREFCLK/n. The FLL reference frequency can be XT1CLK (external crystal plus internal XT1 oscillator), or the internal 32-kHz reference oscillator REFOCLK. The value of n is defined by the FLLREFDIV bits (n = 1, 2, 4, 8, 12, or 16). The default is n = 1. On the devices that support only low frequency on XT1, FLLREFDIV is always read and written as 0 (n = 1).

For applications in which accurate frequency is needed, the FLL should be checked to determine if it is locked or not. The FLL lock status can be detected by reading the FLLUNLOCK bits. When changing clock frequency or changing FLL reference clock, the FLL locks again if it is not disabled.

There are two types of DCO trim values. If the DCO range is selected as the maximum valid value, the DCO factory trim (default) value is applied. If the DCO range is any value other that the maximum valid value, the DCO software trim process is needed. Otherwise, the FLLUNLOCK bit might always be 1. In the DCO software trim process, DCOFTRIMEN and DCOFTRIM are adjusted by software to achieve a suitable DCO trim value, FLLUNLOCK is set to 0, and the FLL is locked. For a detailed description of how to perform software trimming of the DCO, see the DCO section of the clock system chapter in the MSP430FR4xx and MSP430FR2xx family user's guide.

For a detailed guide on how to check the FLL lock status, see the FLL unlock detection section in the MSP430FR4xx and MSP430FR2xx family user's guide. Code examples that show how to set clock frequency and check the FLL lock status are available in the device-specific product folders on www.ti.com.

Nine of the integrator bits (CSCTL0 bits 8 to 0) set the DCO frequency tap. The nine DCOx bits divide the DCO range selected by the DCORSEL bits into 512 frequency steps, separated by approximately 0.1% (F2xx only has three DCOx bits to divide the DCO range selected by the RSELx bits into 8 frequency steps, separated by approximately 10%). One benefit from the nine DCOx bits is that the jitter performance for the DCOCLK is much better. See the device-specific data sheet for detailed specifications.

The modulator mixes two adjacent DCO frequencies to produce fractional taps. When FLL operation is enabled, the modulator settings and DCOx are controlled by the FLL hardware. When FLL operation is not desired, the modulator settings and DCOx control can be configured with software.

The DCO modulator is disabled when DISMOD is set. When the DCO modulator is disabled, the DCOCLK is adjusted to the DCO tap selected by the DCOx bits.

7.2.3 FRAM Access at 16 MHz, ADC Clock, and Clocks-on-Demand

While the FR4xx can source MCLK at 16 MHz, FRAM access is limited to 8 MHz by the FRAM controller, and wait states are required when MCLK is greater than 8 MHz. For configuring wait states, see Section 3.4. Code execution from RAM and accesses to peripherals can be carried out at 16 MHz.

The ADC module's internal oscillator on the F2xx family has been renamed to MODOSC in the FR4xx family (similar to the F5xx family).

The FR4xx CS supports the 'clocks-on-demand' feature. In the F2xx family, the availability of a system clock is affected by entry into a low-power mode. For example, SMCLK is turned off in LPM3 and, hence, any peripheral such as a timer that uses SMCLK is inactive in LPM3. The FR4xx, however, allows the LPM settings to be overridden by a clock request. As long as there is an active request for a clock from a peripheral, the clock remains on, regardless of the LPM setting. This is most easily seen when there is increased power consumption when porting code between families. It is left to the user to disable any modules that request the clock source and prevent the device from entering the required LPM. As an option, this feature can be disabled using the Clock System Control 8 (CSCTL8) register bits MODOSCREQEN, SMCLKREQEN, MCLKREQEN, and ACLKREQEN.

Table 7-1 lists important differences between the clock systems.

Table 7-1. Comparison of FR4xx and F2xx Clock Systems

Parameter	FR4xx	F2xx
Maximum system frequency, f_{SYSTEM}	16 MHz or 24 MHz	16 MHz
XT1 oscillator	Supports LF or LF and HF modes(1)	Supports LF and HF modes
XT2 oscillator	Not available	Supports up to 16 MHz
DCO range	Factory-provided frequencies only	0.06 to 26 MHz
FLL	Available	Not available
REFO	Available, low-power mode	Not available
LFMODCLK (MODOSC/128)	Not available	Not available
VLO control	Available with VLOAUTOOFF	Available with OSCOFF in LPM4
Production calibrated frequencies	None	1 MHz, 8 MHz, 12 MHz, and 16 MHz
Clock sources for MCLK	DCOCLKDIV, XT1CLK, REFOCLK, VLOCLK	DCOCLK, VLOCLK, LFXT1CLK, or XT2CLK(2)
Clock sources for SMCLK	MCLK	DCOCLK, (LFXT1CLK and VLOCLK), or XT2CLK(2)
Clock sources for ACLK	XT1CLK, REFOCLK, VLO(3)	LFXT1CLK, VLOCLK(2)
External crystal fail-safe operations	XT1, LF: defaults to REFOCLK, XT1, HF: defaults to DCOCLKDIV	For any crystal failure: OFIFG is set, MCLK sourced by crystal defaults to DCO. Other clock sources do not have a fail-safe option.
OFIE reset in fail-safe operation	Not reset automatically	Reset automatically
Registers	CSCTL0 through CSCTL8	DCOCTL, BCSCTL1 through BCSCTL3
DCO bits	9	3
Internal load capacitors for XT1 oscillator	Not available	Available

(1) Some FR4xx MCUs support XT1 HF mode. See the device-specific data sheet for details.
(2) See the device-specific data sheet for the clock sources.
(3) Unlike in F2xx devices, VLOCLK cannot be selected as clock source of ACLK in most FR4xx devices. In the FR4xx family, the enhanced clock system devices including FR235x, FR215x, FR267x, and FR247x support VLO as the ACLK source. The VLO calibration method differs between F2xx and FR4xx devices (for details, see VLO calibration on the MSP430FR4xx and MSP430FR2xx family).

For more information about the clock system in the FR4xx devices, see the MSP430FR4xx and MSP430FR2xx family user's guide.

7.3 Operating Modes, Wake-up Times, and Reset

Table 7-2 compares the operating modes that are available and the wake-up times from LPMs.

Table 7-2. Comparison of Operating Modes and Wake-up Times

Parameter or Feature	FR4xx(1)	F2xx(1)
LPM0, LPM1, LPM2, LPM3, LPM4	Available except LPM1 and LPM2	Available
LPM3.5, LPM4.5	Available	Not available
Wake-up time from LPM0	0.2 + 2.5 / f_{DCO} ≈ 2.7 µs (f_{DCO} = 1 MHz) (max)	2 µs
Wake-up time from LPM1 or LPM2	Not applicable	2 µs
Wake-up time from LPM3 or LPM4	10 µs	2 µs
Wake-up time from LPM3.5	350 µs	Not applicable
Wake-up time from LPM4.5	350 µs (SVSHE = 1) 1 ms (SVSHE = 0)	Not applicable
Wake-up time from BOR event	1 ms	2 ms (max)

(1) The values in this table are approximations. To determine the values for a specific device, see the data sheet.

The code flow for entry into and exit out of low-power modes LPM0 to LPM4 remains the same in the FR4xx family as in the F2xx family. There are differences in functionality between the low-power modes on the F2xx compared to the FR4xx devices. These differences are described in the SYS chapter of the MSP430FR4xx and MSP430FR2xx family user's guide.

7.3.1 LPMx.5

Two new low-power modes introduced in the FR4xx family are LPM3.5 and LPM4.5. In both modes, the VCORE LDO is turned off, which powers down the digital core, RAM, and peripherals. To wake up from LPM3.5, RTC interrupts, LCD interrupts, oscillator fault, or port interrupts are required. All other system interrupts are not available. The RTC module and LCD module on the FR4xx device are powered from the LPM3.5 LDO rail and can, therefore, stay functional even when the core LDO is turned off. In LPM4.5, only port interrupts can be used to wake up the device.

It is important to understand that the LPMx.5 modes are inherently different from the typical LPMs (LPM0 through LPM4) in that a wakeup from these modes constitutes a device reset. Because RAM is not retained (except backup memory and LCD memory in LPM3.5), the state of the application (if stored in variables located in RAM) and register initialization are lost.

LPM3.5 is different from LPM4.5 for entering low-power mode and RAM retention.

- The register setting is the same for entering LPM3.5 and LPM4.5. If the RTC or LCD is active, the FR4xx enters LPM3.5. If the RTC and LCD are off, the FR4xx enters LPM4.5. Measure the power supply current to determine the FR4xx power mode.
- In LPM3.5, backup memory (32 bytes) and LCD memory (40 bytes) are retained. If the application must retain some data after wakeup from LPM3.5, these 72 bytes can be used. In addition, FRAM can also be used for storing data because FRAM is nonvolatile (see the FRAM special features in Section 3.1).

These LPMx.5 modes are suited for applications that spend large amounts of time in deep sleep and in which wake-up time is not critical. To decide which power mode is appropriate for the application, the frequency of wakeup needs to be considered because there is an energy penalty associated with the time spent during wakeup.

- For example, an F2xx application that wakes up every 1 ms from LPM3 to sample a signal can be more efficiently ported to LPM3 in FR4xx rather than LPM3.5. This is because LPM3.5 requires approximately 350 µs to wake up, and the application would spend 35% of its duty cycle during wakeup, which significantly affects the power gains that were achieved in moving from LPM3 to LPM3.5.
- Consider a different application that wakes up once per minute to update a time stamp. In this case, LPM3.5 could be a better fit for maximizing power savings because the average power during on time for LPM3.5 is 50% that of LPM3. Hence, selecting LPMs when migrating depends on the application and the on/off duty cycle that is required.

7.3.2 Reset

7.3.2.1 Behavior of POR and BOR

One important difference between the families is the behavior on reset. There are multiple levels of reset (such as PUC, POR, and BOR) across all MSP430 families. In the F2xx family, the program counter (PC) is reinitialized to the reset vector location on executing a PUC. In the case of a power cycle (POR), the PC is reinitialized after t_{DBOR} has elapsed [3]. In the FR4xx family, the behavior on executing a PUC is the same as the F2xx family as regards reinitialization of the PC and specific peripheral registers.

However, a deeper level of reset such as POR or BOR executes a boot code that is present in protected ROM. This boot code sets up the device and loads calibration settings that are essential to establish device functionality. Hence, the time to start from a POR or BOR in the FR4xx family is different from the time in the F2xx family. For details, see the device-specific data sheet.

7.3.2.2 Reset Generation

There are also some differences between FR4xx and F2xx in which events generate BOR, POR, and PUC. For detailed information, see the SYS chapter of the MSP430FR4xx and MSP430FR2xx family user's guide.

The FR4xx can initiate all levels of reset in software (in the F2xx family, only a PUC can be initiated by software). The resets are initiated by setting the PMMSWBOR or PMMSWPOR bit in the PMMCTL0 control register.

7.3.2.3 Determining the Cause of Reset

In F2xx devices, a PUC can be triggered by multiple sources such as WDT timer expiration, WDT key violation, or flash key violation. To determine the cause of reset, it is necessary to investigate multiple registers because each reset source is tracked by different interrupt flags and registers.

In the FR4xx devices, all sources of reset are combined into one System Reset Vector (SYSRSTIV) register, and it is no longer necessary to read multiple registers to determine the cause of reset. The SYSRSTIV register is useful when debugging and lists all sources from all levels of reset (PUC, POR, and BOR). See the device-specific data sheet for a list of SYSRSTIV values for different reset sources.

7.4 Interrupt Vectors

The FR4xx devices use an interrupt vector (IV) structure for any interrupt service routine that is sourced by multiple flags.

For example, in the F2xx family, the USCI TX interrupt sources the RX and TX interrupt flags, and the USCI RX interrupt sources all the status flags. In the case of the FR4xx family, all of these interrupt flags are captured using a single interrupt vector UCBxIV. This allows interrupt servicing to be more efficient and ensures the same predefined latency for all interrupts.

7.5 FRAM and the FRAM Controller

7.5.1 Flash and FRAM Overview Comparison

The F2xx family flash controller is replaced by the FRAM controller in the FR4xx family.

The most significant differences between using FRAM and flash pertain to (1) timing and (2) power requirements (see Table 7-3).

Table 7-3. Comparison of Flash and FRAM on MSP430 MCUs

Parameter	FRAM (FR4133)[1]	Flash (F2274)[1]
Program time for byte or word (maximum)	120 ns	116 µs (approximately)
Erase time for segment (maximum)	Not applicable (pre-erase not required)	18 ms
Supply current during program (maximum)	No extra current during write (included in active power specification)	5 mA
Supply current during erase (maximum)	Not applicable (pre-erase not required)	7 mA
Nonvolatile memory maximum read frequency	8 MHz	16 MHz

(1) The values in this table are approximations; to find the values for a specific device, see the data sheet.

Because every read from an FRAM location is also a write, there is no current penalty due to write or erase. Hence, the power consumption when writing to a block of FRAM is the same as when reading from it. This is different from flash, where the writing process consumes excess power due to the operation of a device-internal charge pump. Similarly, FRAM does not require an erase before write and is not segmented like flash. Hence, there is no added erase current (or erase time) when writing to FRAM.

In terms of the write time, FRAM is written in four-word blocks, and the write time is built into each read cycle. Hence, there is no difference between the read time and write time for an FRAM byte, word, or 4-word block. With regards to the read frequency, FRAM accesses (both read and write) are capped at 8 MHz. However, flash reads can take place at the maximum speed allowed by the device (f_{SYSTEM}), which is 16 MHz in the F2xx devices.

The speed of instruction execution in an FRAM-based system is affected by the architecture. The FR4xx uses a cache-based architecture that employs a combination of register and FRAM accesses when executing from nonvolatile memory. This allows the system throughput to be higher than the maximum allowable read frequency of 8 MHz.

7.5.2 Cache Architecture

The FRAM controller uses a 2-way associative cache that has a 64-bit line size. The cache stores prefetched instructions. The function of the FRAM controller is to prefetch four instruction words depending on the current PC location. The actual execution of these instructions is carried out in the cache. When the end of the cache buffer is reached, the FRAM controller preserves the four current words in the same cache line and fetches the next four words. If a code discontinuity is encountered at the end of a 2-way associative cache line, the cache is refreshed and the following four instruction words are retrieved from FRAM. However, if the application code loops back to a location already present in the cache when the last instruction in the cache is reached, the relevant instruction is executed directly from the cache instead of fetching code from FRAM again.

Only FRAM accesses are subject to the 8-MHz access limitation. When executing from cache, a system clock of up to 16 MHz can be used. Thus the cache is useful in (1) overcoming the 8-MHz limitation and increasing the average system throughput and (2) reducing overall active power by ensuring that most instructions are executed from it. Note that this is an instruction-only cache; all data is fetched directly from FRAM and is not cached.

The cached execution of instructions in the FR4xx family differs from the F2xx family, in which every instruction is directly executed from flash with no prefetches or caching. This direct execution provides a 1:1 relationship between MCLK and instruction execution. For example, at MCLK = 16 MHz, eight two-cycle instructions can be executed in 16 clocks. This relationship is application-dependent for the FR4xx family. The 1:1 relationship is true for MCLK ≤ 8 MHz. For MCLK > 8 MHz, the number of inserted wait states (directly proportional to how many times FRAM is accessed) determines the MCLK:instruction-execution ratio.

To provide another application example, with MCLK = 16 MHz, a JMP $ instruction (single cycle) is executed at the same rate in both devices. This is because the FR4xx fetches this instruction once and stores it in cache where it can be executed at the maximum MCLK speed. However, a loop that has more than eight instruction words would require accessing the FRAM every time a cache refresh is needed. These FRAM accesses take place at MCLK / 2 = 8 MHz, thereby reducing the overall throughput of the system when compared to an F2xx device.

8 Peripheral Considerations

Some of the peripherals in the FR4xx family have new features or existing features that are implemented differently. This section highlights the differences in the peripherals.

8.1 Watchdog Timer

The main difference between the two families lies in the fail-safe operation.

In the F2xx family, WDT is typically timed by ACLK, which is sourced by a crystal or the VLO. If ACLK or SMCLK failure occurs, WDT defaults to MCLK. If MCLK is also sourced by a crystal, and crystal has failed, the DCO is automatically activated.

In the FR4xx family, the WDT_A fail-safe defaults to VLOCLK if SMCLK or ACLK fails as the WDT_A clock source.

8.2 Ports

8.2.1 Digital Input/Output

The main differences in the FR4xx general-purpose I/O (GPIO) pins are:

- P1 and P2 ports support interrupt inputs in the FR4xx devices, the same as in the F2xx devices. In addition, P1 and P2 interrupts can be used to wake up FR4xx devices from LPMx.5 power modes. In MSP430FR231x devices, P2.2, P2.3, P2.4, and P2.5 do not support interrupts. In MSP430FR211x devices, P1.4, P1.5, P1.6, and P1.7 do not support interrupts. In MSPFR235x devices, P1, P2, P3 and P4 all support interrupts. In MSP430FR267x and FR247x devices, all GPIOs including P1, P2, P3, P4, P5, and P6 support interrupts.
- Peripheral function select in the MSP430FR413x and MSP430FR203x devices uses one register for Port x function selection: PxSEL0. Other FR2xx devices use two registers for Port x function selection: PxSEL0 and PxSEL1. F2xx also uses two registers for Port x function selection: PxSEL and PxSEL2. See the device-specific data sheet for details.
- In F2xx, the high-impedance leakage current is ±50 nA. In FR4xx, this specification is ±20 nA.
- Configuration of digital I/Os after BOR reset

  To prevent cross currents during startup of the device, all port pins are high-impedance with Schmitt triggers and module functions are disabled. To enable the I/O functions after a BOR reset, first configure the ports, and then clear the LOCKLPM5 bit. For details, see the section on configuration after reset in the digital I/O chapter of the MSP430FR4xx and MSP430FR2xx family user's guide.
- Configuration for LPMx.5 power modes

  During LPMx.5 the I/O pin states are held and locked based on the settings before entry to LPMx.5, regardless of the default I/O register settings. Only the pin conditions are retained. All port configuration register settings such as PxDIR, PxREN, PxOUT, PxIES, and PxIE contents are lost and must be reconfigured after exit from LPMx.5. After wake from LPMx.5, the LOCKLPM5 bit can be cleared to release I/O pin conditions and I/O interrupt configuration. For details, see the section on configuration for LPMx.5 low-power modes in the digital I/O chapter of the MSP430FR4xx and MSP430FR2xx family user's guide.
- Configuration of unused port pins

  To prevent a floating input and to reduce power consumption, unused I/O pins should be configured as I/O function, output direction, and left unconnected on the PCB. Alternatively, the integrated pullup or pulldown resistor can be enabled by setting the PxREN bit of the unused pin to prevent a floating input.

- Configuration of unbonded pins

  In the MSP430FR413x and MSP430FR203x MCUs, some pins are not bonded out in packages with fewer pins than the 64-pin PM package. Configure these unbonded pins as unused port pins. See Table 8-1 for the unbonded pins for different MSP430FR413x and MSP430FR203x packages.

Table 8-1. Pins Not Bonded Out on MSP430FR413x and MSP430FR203x Packages

Unbonded Pins for FR413x and FR203x Package G56(1)	Unbonded Pins for FR413x and FR203x Package G48(1)
P5.6/L38	P5.6/L38
P5.7/L39	P5.7/L39
P6.6/L22	P6.6/L22
P6.7/L23	P6.7/L23
P7.6/L6	P7.6/L6
P7.7/L7	P7.7/L7
P8.0/SMCLK/A8	P8.0/SMCLK/A8
P8.1/ACLK/A9	P8.1/ACLK/A9
	P8.2/TA1CLK
	P8.3/TA1.2
	P5.4/L36
	P5.5/L37
	P6.4/L20
	P6.5/L21
	P7.4/L4
	P7.5/L5

(1) See the device-specific data sheet for detailed information.

8.2.2 Capacitive Touch I/O

The main difference in the capacitive touch implementation between the G2xx and the FR4xx lies in the selection of the port pins and the internal wiring of the capacitive touch I/O to the timer.

In the F2xx and G2xx devices with the pin oscillator feature, the selection of the pin oscillator is enabled with the PxSEL and PxSEL2 registers. In the FR4xx devices, there is one register, CAPTIO0CTL, to select a port and a specific pin in that port to be used as capacitive touch I/O.

For example, the CAPTIOPOSELx field in the Capacitive Touch IO x Control (CAPTIO0CTL) register can select port 1. The CAPTIOPISELx field in the same register can select pin 5. Hence, pin 1.5 is designated as a capacitive touch I/O.

For the CAPTIO0CTL register, the selected capacitive touch I/O is hard wired (internally connected) to a device-specific timer.

Note

In the FR4xx family, some MCUs do not have the capacitive touch feature shared on GPIO pins. See the device-specific data sheet for details.

8.3 Analog-to-Digital Converters

8.3.1 ADC10 to ADC

Compared to ADC10 in F2xx, the ADC module in the FR4xx has been redesigned for lower power and also includes some new features. Some of the significant differences are:

- The ADC module in FR4xx devices supports both 10-bit ADC and 12-bit ADC configurations.
- The ADC10 internal reference is no longer a part of the ADC module in FR4xx. The VREF generation block is in the PMM module (see Section 7.1.3).
- If the reference buffer is used for the ADC reference voltage, and the conversion rate is below 50 kbps, setting ADCSR = 1 reduces the current consumption of the buffer approximately 50%.
- The data transfer controller (DTC) is used for automatic storage of conversion results in the F2xx family. In the FR4xx family, the DTC is not supported.
- In FR4xx, up to 10 input channels are available externally and 2 channels are available internally. The 1.2-V V_{REF} can be output to a device-specific external channel. The on-chip temperature sensor can be internally connected to channel A12. The 1.5-V V_{REF} can be internally connected to channel A13. For detailed connection, see the *10-Bit Analog-to-Digital Converter (ADC)* section in the device-specific data sheet.
- In FR4xx, a new added window comparator block allows the ADC module to monitor analog signals without CPU interaction. The window comparator can generate interrupts when specific thresholds have been reached.
- The interrupt vector register ADCIV has six interrupt flag sources including three from the window comparator function.
- Increased options for the ADC clock dividers and controlling the sampling rate in FR4xx.
- ADC10OSC is renamed to MODOSC in the FR4xx family.
- In the MSP430FR413x and MSP430FR203x devices, the ADC pin selection is set in the SYSCFG2 register. For other FR2xx devices, ADC pin selection is set in the PxSEL0 and PxSEL1 registers.
- Because there is only one pair of power supply pins (DVCC and DVSS) in FR4xx devices, to achieve good ADC performance, board design should avoid system noise:
  - Place the decoupling capacitors as close as possible to DVCC.
  - Select the reference voltage carefully.
  - Do not route high-frequency toggled digital signals close to power lines or ADC input signals.
  - Do not toggle I/O pins while the ADC is working.
  - Find more design guidance for the ADC on www.ti.com.

Some register names have been changed to include the added functionality. For more information when porting firmware, see the MSP430FR4xx and MSP430FR2xx family user's guide.

8.4 Communication Modules

8.4.1 USI to eUSCI

The USI module that is available on some F2xx devices is architecturally different from the eUSCI module. The USI module is built primarily on a shift register that is used in conjunction with a counter to shift out data bits. Any protocol-specific aspects for SPI or I^2C communication are implemented in software. Hence, it can be said that the implementation of the USI module is a combination of equal parts firmware and hardware.

In contrast, the eUSCI module is almost completely hardware based. The application firmware is only required to configure the module based on the protocol being used and then access interrupts to receive or transmit data. Hence, with regards to migrating firmware from the USI to the eUSCI, code cannot be reused. It is instead recommended to see the code examples provided online in the device product folder that show easy setup of the eUSCI module and handling of interrupts. In addition, MSP430 driver library APIs are available as part of the MSP430Ware™ software to help with easy module configuration and use.

The eUSCI module handles all communication-specific implementation details in hardware, which allows the application to be better power-optimized and to service data transmission and reception more efficiently. The USI module supports SPI and I^2C protocols. The eUSCI module supports SPI, I^2C, and UART protocols.

8.4.2 USCI to eUSCI

The architecture and the internal state machine of the eUSCI in the FR4xx family are very similar to the USCI module in the F2xx family. However, there are many new features added in the eUSCI as well as changes made to the existing features. While most of the code is still compatible, TI recommends reviewing the register names. Table 8-2 shows most of the significant differences between the families. For more detailed information, see Migrating from the USCI module to the eUSCI module [8].

Table 8-2. Comparison of USCI and eUSCI Modules

Parameter or Feature	USCI (F2xx)	eUSCI (FR4xx)
UART		
Enhanced baud rate generation	No	Yes
TXEPT interrupt (similar to USART)	No	Yes
Start edge interrupt	No	Yes
Selectable glitch filter	No	Yes
Interrupt vector generator	No	Yes
SPI		
Enhanced baud rate generation	No	Yes
Maximum baud rate	4 to 6 MHz	5 MHz(1)
Interrupt vector generator	No	Yes
I^2C		
Preload of transmit buffer	No	Yes
Clock low timeout	No	Yes
Byte counter	No	Yes
Multiple slave addressing	No	Yes
Address bit mask	No	Yes
Hardware clear of interrupt flags	Yes	No
Interrupt vector generator	No	Yes

(1) Calculated based on SPI timing with another MSP430FR4133 device in slave mode. For the formula to calculate the maximum baud rate, see the device-specific data sheet.

The eUSCI_A module provides support for SPI (3 or 4 pin), UART, enhanced UART, and IrDA. The eUSCI_B module provides support for SPI (3 or 4 pin) and I^2C.

Another significant difference for the eUSCI module in most FR4xx devices (except for the MSP430FR231x, MSP430FR211x, MSP430FR231x, MSP430FR235x, MSP430FR215x, MSP430FR267x, and MSP430FR247x)

is that there is no ACLK available for the clock source. If the eUSCI clock source does not support ACLK, the eUSCI cannot work in LPM3 mode. See the clock distribution table in the device-specific data sheet for details. A workaround to allow UART or I^2C to work in LPM3 mode is to route the ACLK output to the UCA0CLK or UCB0CLK pin externally. This workaround costs two GPIO pins, and it is available only with package that include an output for the ACLK signal. In MSP430FR231x devices, clock distribution is changed so that ACLK is available for the eUSCI module.

The FR4xx devices have eUSCI_A and eUSCI_B modules. See the device-specific data sheet for the number of instances of each module. Table 8-3 summarizes the pin configurations for the communication interfaces.

In MSP430FR231x devices, a pin remapping function is available for eUSCI_B0. The USCIBRMP bit in SYSCFG2 register controls eUSCI_B0 pins remapping from P1.0–P1.3 to P2.2–P2.5. Only one port can be selected and valid at one time.

In MSP430FR211x devices, the pin remapping functions are available for eUSCI_A and Timer_B. The USCIARMP bit in the SYSCFG3 register controls eUSCI_A pin remapping from P1.4–P1.7 to P1.0–P1.3. The TBRMP bit in the SYSCFG3 register controls Timer_B output pin remapping from P1.6–P1.7 to P2.0–P2.1. Only one port can be selected and valid at one time.

Table 8-3. FR4xx eUSCI Pin Configurations

eUSCI_A0	**Pin of FR413x or FR203x**	**Pin of FR2433, FR263x, or FR253x**	**Pin of FR231x**	**Pin of FR21xx or FR2000**	**Pin of FR235x or FR215x**	**Pin of FR267x or FR247x**	**UART**	**SPI**
	P1.0	P1.4	P1.7	P1.7, P1.3	P1.7	P1.4[1], P5.2[2]	TXD	SIMO
	P1.1	P1.5	P1.6	P1.6, P1.2	P1.6	P1.5[1], P5.1[2]	RXD	SOMI
	P1.2	P1.6	P1.5	P1.5, P1.1	P1.5	P1.6[1], P5.0[2]	–	SCLK
	P1.3	P1.7	P1.4	P1.4, P1.0	P1.4	P1.7[1], P4.7[2]	–	STE
eUSCI_A1	**Pin of FR413x or FR203x**	**Pin of FR2433, FR263x, or FR253x**	**Pin of FR231x**	**Pin of FR21xx or FR2000**	**Pin of FR235x or FR215x**	**Pin of FR267x or FR247x**	**UART**	**SPI**
	Not available	P2.6	Not available	Not available	P4.3	P2.6	TXD	SIMO
		P2.5			P4.2	P2.5	RXD	SOMI
		P2.4			P4.1	P2.4	–	SCLK
		P3.1			P4.0	P3.1	–	STE
eUSCI_B0	**Pin of FR413x or FR203x**	**Pin of FR2433, FR263x, or FR253x**	**Pin of FR231x**	**Pin of FR21xx or FR2000**	**Pin of FR235x or FR215x**	**Pin of FR267x or FR247x**	**I^2C**	**SPI**
	P5.0	P1.0	P1.0, P2.2	Not available	P1.0	P1.0[1], P5.6[2]	–	STE
	P5.1	P1.1	P1.1, P2.3		P1.1	P1.1[1], P5.5[2]	–	SCLK
	P5.2	P1.2	P1.2, P2.4		P1.2	P1.2[1], P4.6[2]	SDA	SIMO
	P5.3	P1.3	P1.3, P2.5		P1.3	P1.3[1], P4.5[2]	SCL	SOMI

Table 8-3. FR4xx eUSCI Pin Configurations (continued)

	Pin of FR413x or FR203x	Pin of FR2433, FR263x, or FR253x	Pin of FR231x	Pin of FR21xx or FR2000	Pin of FR235x or FR215x	Pin of FR267x or FR247x	I²C	SPI
eUSCI_B1	Not available	Not available	Not available	Not available	P4.4	P2.7(1), P5.4(2)	–	STE
					P4.5	P3.5(1), P5.3(2)	–	SCLK
					P4.6	P3.2(1), P4.4(2)	SDA	SIMO
					P4.7	P3.6(1), P4.3(2)	SCL	SOMI

(1) This is the mappable default function that is controlled by the USCIBRMP or USCIARMP bit of the SYSCFG2 or SYCFG3 register. Only one selected port is valid at any time.
(2) This is the mappable function that is controlled by the USCIBRMP or USCIARMP bit of the SYSCFG2 or SYCFG3 register. Only one selected port is valid at any time.

8.5 Timer and IR Modulation Logic

Except for the clock source difference described in the following paragraph, there are few differences in the timer modules for the F2xx and FR4xx families. Unlike most FR4xx devices that implement Timer_A, MSP430FR231x, MSP430FR21xx, MSP430FR2000, MSP430FR235x, MSP430FR215x, MSP430FR267x, and MSP430FR247x devices implement Timer_B. For the Timer_B in these devices the control bits TBxTRGSEL are added for TBxOUTH trigger source selection. TI recommends stopping the timer before modifying its operation. A delay of at least 1.5 timer clocks is required to resynchronize before restarting the timer if the timer clock source is asynchronous to MCLK.

Except for in the enhanced clock system of FR235x, FR215x, FR267x, and FR247x, the VLO is not a clock source for ACLK in FR4xx devices, and VLO cannot be a clock source and capture input for the timer module in FR4xx devices. This is a significant difference for the Timer_A clock source and capture input between FR4xx and F2xx. Therefore, the VLO cannot be calibrated and used to output accurate pulse as described in Using the VLO library. Because the FR4xx devices have the RTC output routed to the timer capture input, there is a workaround to implement the same function in FR4xx devices:

1. Set the VLO as the RTC clock source. The RTC outputs a pulse based on the VLO frequency.
2. Configure the RTC output as the timer capture input.
3. Set the timer clock to ACLK or SMCLK, which provide good accuracy.
4. Measure the VLO frequency with the accurate ACLK or SMCLK.
5. Adjust the RTCMOD register so that an accurate period interrupt can be triggered for the MCU to generate an accurate pulse on a GPIO. See the details in VLO calibration on MSP430FR4xx and MSP430FR2xx family.

New IR modulation logic has been added to the SYS module of the FR4xx family. This logic combines two timers' outputs to easily generate accurately modulated IR waveforms. Both ASK and FSK modulations can be implemented. Two other inputs to this logic are UCA0TXD/UCA0SIMO and the IRDATA bit in the SYSCFG1 register. This makes it possible to generate the modulation data by hardware using eUSCI_A or by software using IRDATA.

For more information, see the Infrared Modulation Function section in the MSP430FR4xx and MSP430FR2xx family user's guide and the application note Infrared remote control implementation with MSP430FR4xx.

8.6 Backup Memory

A backup memory module is available in the FR4xx family devices except for the MSP430FR203x. The Backup Memory provides up to 256 bytes that are retained during LPM3.5. The size of the Backup Memory varies by device (see the device-specific data sheet for details). The backup memory can be used for data storage when LPM3.5 power mode is used in the application. In MSP430FR4xx devices, the LCD memories are also retained during LPM3.5, which can be regarded as the extra backup memory if application needs more memory retained. See Section 7.3.1 for details.

8.7 Hardware Multiplier (MPY32)

A hardware multiplier module (MPY32) is available in the MSP430FR235x, MSP430FR215x, MSP430FR267x, and MSP430FR247x MCUs. The MPY32 module supports 8-bit, 16-bit, 24-bit, and 32-bit operations, which are fully utilized by the C-compiler when building code. The module also supports fractional number mode and saturation mode; however, these features must be accessed by directly manipulating the hardware multiplier's memory-mapped control registers or through use of a suitable software library.

MSPMATHLIB is an accelerated floating-point math library. This library delivers faster computation for most commonly used math functions. To learn more, see MSPMATHLIB: an optimized MSP430 library of floating-point scalar math functions.

8.8 RTC Counter

A Real-Time Clock (RTC) counter module is available in FR4xx family devices. The RTC counter is a 16-bit counter that is functional in active mode (AM), LPM0, LPM3, and LPM3.5 low-power modes. The RTC counter accepts multiple clock sources, selected by control register settings, to generate timing from less than 1 µs up to many hours. One example use of the RTC counter module is for a software calendar. Combined with the LFXT clock and LPM3.5, an ultra-low-power RTC application can be achieved. The base address for the RTC registers is different for FR413x and FR203x devices compared to other FR4xx devices. See the device-specific data sheet for details.

MSP430FR231x, MSP430FR21xx, MSP430FR2000, MSP430FR235x, MSP430FR215x, MSP430FR267x, and MSP430FR247x devices support the control bit RTCCKSEL in the SYSCFG2 register. This bit is not available in other FR4xx devices. The RTCCKSEL bit can select ACLK or SMCLK as the clock source for the RTC.

8.9 Interrupt Compare Controller (ICC)

In the FR4xx family, FR235x devices include the ICC module The ICC supports a hardware-based nested interrupt mechanism. The ICC allows all maskable interrupts to be serviced based on both software configured priority and vector table priority.

Features of the ICC module include:

- Four-level configurable priority for each maskable interrupt source
- Real-time hardware nested interrupt capability
  - Lower-priority interrupt requests cannot preempt higher-priority interrupts.
  - Higher-priority interrupts can preempt lower-priority interrupts.
- Reduces design effort to develop a preemptive scheduler or RTOS
- Can be enabled and disabled in the control register in the SYS module

8.10 LCD

The LCD_E module is supported in the FR4xx family, and the LCD_C module is supported in the FR6xx family. For the detailed design differences for the LCD modules of MSP430 devices, see Designing With MSP430™ MCUs and segment LCDs.

Table 8-4 summarizes the key differences between the LCD_E and LCD_C modules.

Table 8-4. Comparison of LCD_E and LCD_C

Feature	LCD_E (FR4xx)	LCD_C (FR6xx)
Supported types of LCDs	Static, 2-, 3-, 4-, 5-, 6-, 7, 8-mux	Static, 2-, 3-, 4-, 5-, 6-, 7, 8-mux
LCD bias modes	1/3 bias	1/2 bias and 1/3 bias
Flexible configuration for COM and segment pins	Yes	NO
LCD clock selection	ACLK, XT1, VLO	ACLK, VLO
Interrupt capabilities	Yes (three sources)	Yes (4 sources)
Dual memory display	Yes	Yes
Charge pump voltage with external voltage reference	Programmable (15 levels)	Programmable (15 levels)
Low-power waveforms mode	Yes	Yes
LCD blinking memory	Yes	Yes
External pins	R13, R23, R33, LCDCAP0, LCDCAP1	R03, R13, R23, R33

Table 8-4. Comparison of LCD_E and LCD_C (continued)

Feature	LCD_E (FR4xx)	LCD_C (FR6xx)
LPM3.5	Supported	Not supported
Maximum LCD voltage ($V_{LCD,typ}$)	3.44 V	3.44 V
Number of LCD pins	Up to 4×60 or 8×56	Up to 4×50 or 8×46

8.11 Smart Analog Combo (SAC)

There are two operational amplifiers available in the MSP430FR231x devices of the FR4xx family. One is included in the smart analog combo (SAC) module and the other one is the transimpedance amplifier (TIA).

The SAC module has four different configurations from low level to high level: SAC-L1 (OA: operation amplifier), SAC-L2 (PGA: programmable gain amplifier) and SAC-L3 (DAC: digital-to-analog converter), according to the feature set from minimum to maximum. The MSP430FR231x devices support SAC-L1, which includes only one operational amplifier. Unlike in F2xx, no feedback resistor is included in the MSP430FR231x devices, so external resistors must be added for the amplifier circuit. SAC supports both rail-to-rail input and rail-to-rail output.

TIA is an amplifier that converts current to voltage. Similar to SAC-L1, TIA does not integrate a feedback resistor. One external feedback resistor is needed for the current-to-voltage conversion. TIA supports 1/2-rail input and rail-to-rail output. For the MSP430FR231x devices with PW16 package, there is one dedicated low-leakage pin. This pin supports picoamp-level low-leakage input (5 pA typical) that improves the accuracy of current conversion.

Unlike in F2xx, there are no dedicated analog power or ground pins in MSP430FR231x devices. If a GPIO needs to drive high current, TI recommends that you do not use the SAC or TIA module at the same time. Using both at the same time affects the power supply of SAC and TIA, which leads to lower performance. For the SAC and TIA layout, keep the traces short and avoid vias. The clock signals and high-frequency toggling pins should be far way from the SAC and TIA pins. The rich internal connections between different modules can also help optimize the layout and reduce pin use. For the internal module connections, see the device-specific data sheet.

MSP430FR235x devices implement the high-level SAC configuration SAC-L3, and there are four SAC-L3 modules in MSP430FR235x devices. Each SAC-L3 can be configured to OA, PGA, and DAC plus OA. With the rich internal connections for peripherals modules SAC-L3 (12-bit ADC, eCOMP, and Timer_B), many signal chain applications can be implemented with MSP430FR235x. For detailed information about internal connection and SAC-L3, see the MSP430FR235x, MSP430FR215x mixed-signal microcontrollers data sheet and the MSP430FR4xx and MSP430FR2xx family user's guide.

8.12 Comparator

Compared to Comparator_A+ in F2xx, the enhanced comparator (eCOMP) module in the FR4xx has been redesigned for low power and also includes some new features. Some of the significant differences are:

- The eCOMP module integrates a 6-bit DAC for reference voltage input. The DAC has a dual-buffer on-chip reference voltage selection. The dual buffer can set different values to have two different DAC output voltages as the eCOMP reference input. In F2xx, the voltage reference generator can generate a fraction of the device V_{CC} or a fixed transistor threshold voltage of approximately 0.55 V.
- The eCOMP can function in high-power (high speed) or low-power (low speed) mode according to the power mode selection. This module also has different step hysteresis configurations for better comparison performance.
- The input short switch can be used to short the Comparator_A+ inputs. This can be used to build a simple sample-and-hold for the comparator. The input short switch it is not available in eCOMP.
- For Comparator_A+, the pin input and output buffers can be controlled by software. This feature is not supported in eCOMP.
- There are two interrupt flags in eCOMP, while there is only one interrupt flag in Comparator_A+. With one more interrupt flag, eCOMP can support one more interrupt mode – dual edge mode (either rising or falling edge can be triggered).
- The eCOMP interrupt flag is cleared by writing 1. The Comparator_A+ interrupt flag is cleared by writing 0.

The eCOMP module is available on select FR4xx devices. See the device-specific data sheet for details.

9 ROM Libraries

FR235x and FR215x devices include the MSP430 driver library and FFT library in ROM. FR267x and FR247x devices include the MSP430 driver library in ROM. MSP430 software libraries in ROM are tested to work with Code Composer Studio™ IDE and IAR Embedded Workbench® IDE toolchains.

- For the ROM image to be compatible between CCS and IAR toolchains, there are certain project properties restrictions. More details are provided in the Call conv attribute wiki page.
- To use DriverLib in ROM, #include "rom_driverlib.h". Header file checks continue to provide helpful hints at build time until the user application adheres to __cc_rom.
- To use FFTLib in ROM, #include "DSPLib.h". FFTLib is a subset of MSP430 software library DSPLib.
- For more information, see the MSP430 driver library for MSP430FR2xx_4xx ROM readme and MSP430 DSP library ROM readme in the MSP430Ware software.

The library ROM image is located above the 64KB memory address. Application code using ROM must be large code model (20-bit address pointer rather than 16-bit address pointer).

Benefits of ROM library usage:

- Code execution at clock speeds that exceed 8 MHz is faster from of ROM than from FRAM. Without FRAM wait states, code execution performance is limited only by the processor clock, which is generally faster than other subsystems. Executing code from RAM gives comparable performance, but the available RAM size is typically more limited.
- More nonvolatile storage (FRAM) available in the device is left for application code.

10 Conclusion

This application report describes many of the key features change and new modules in the MSP430FR4xx family compared to the MSP430F2xx family. While this document is intended to be comprehensive, there may be minor differences between the F2xx and the FR4xx families that have not been covered here. For details of a given device, the device-specific data sheet is always the best source of information. For module functionality and use, see the MSP430FR4xx and MSP430FR2xx family user's guide. For any bugs and workarounds in the FR4xx family devices, see the device-specific errata sheet.

11 References

1. MSP430FR4xx and MSP430FR2xx family user's guide
2. MSP430x2xx family user's guide
3. MSP430F22x2, MSP430F22x4 mixed-signal microcontrollers
4. MSP430FR413x mixed-signal microcontrollers
5. MSP430FR231x mixed-signal microcontrollers
6. MSP430FR211x mixed-signal microcontrollers
7. MSP430FR235x mixed-signal microcontrollers
8. Migrating from the USCI module to the eUSCI module
9. Maximizing FRAM write speed on the MSP430FR573x
10. Migrating from the MSP430F2xx family to the MSP430FR57xx family
11. Migrating from the MSP430F2xx and MSP430G2xx families to the MSP430FR58xx, FR59xx, 68xx, 69xx family
12. Code Composer Studio™ IDE for MSP430™ user's guide
13. MSP430 hardware tools user's guide
14. MSP430 32-kHz crystal oscillators
15. MSP430 FRAM quality and reliability
16. MSPMATHLIB: an optimized MSP430 library of floating-point scalar math functions

12 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision F (March 2019) to Revision G (August 2021)	Page
• Updated the numbering format for tables, figures and cross-references throughout the document	3

Changes from May 5, 2018 to March 26, 2019	Page
• Updated Section 1, *Introduction*	3
• Changed Table 2-1, *Comparison of Features*	4
• Changed Table 2-2, *Comparison of Memory Maps*	4
• Changed Table 3-1, *FRAM Protection Features*	6
• Changed the note that begins "Unlike in F2xx devices, VLOCLK..." in Table 7-1, *Comparison of FR4xx and F2xx Clock Systems*	14
• Added the last sentence to the first list item in Section 8.2.1, *Digital Input/Output*	20
• Added MSP430FR267x and MSP430FR247x in the paragraph that begins "Another significant difference for the eUSCI module..." in Section 8.4.2, *USCI to eUSCI*	23
• Added "Pin of FR267x or FR247x" column to Table 8-3, *FR4xx eUSCI Pin Configurations*	23
• Added MSP430FR267x and MSP430FR247x in the first paragraph of Section 8.5, *Timer and IR Modulation Logic*	25
• Added "Except for in the enhanced clock system of FR235x, FR215x, FR267x, and FR247x" to the beginning of the second paragraph of Section 8.5, *Timer and IR Modulation Logic*	25
• Added MSP430FR267x and MSP430FR247x to the first paragraph of Section 8.7, *Hardware Multiplier (MPY32)*	26
• Added MSP430FR267x and MSP430FR247x to the last paragraph of Section 8.8, *RTC Counter*	26
• Added SMCLK as an option for the RTCCKSEL bit in the last paragraph of Section 8.8, *RTC Counter*	26
• Changed the title of Section 8.11, *Smart Analog Combo (SAC)*	27
• Updated the list of SAC levels in the second paragraph of Section 8.11, *Smart Analog Combo (SAC)*	27
• Added the paragraph that begins "MSP430FR235x devices implement the high-level SAC..." in Section 8.11, *Smart Analog Combo (SAC)*	27
• Added FR267x and FR247x in the first paragraph of Section 9, *ROM Libraries*	28

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265