Application Report

MSP430FR2xx/FR4xx DCO+FLL Applications Guide

Lixin Chen *MSP430 Applications*

ABSTRACT

MSP430™ microcontroller (MCU) portfolio offers a wide variety of 16-bit MCUs with ultra-low-power and integrated analog and digital peripherals for sensing and measurement applications. This application report introduces internal digitally controlled oscillator (DCO) and frequency-locked loop (FLL) in MSP430FR2xx/FR4xx devices and how to use DCO+FLL in the applications to achieve accurate clock frequency, how to lock factory defined frequency or in-between frequency with software trim, how to fast lock any target frequency within maximum frequency range, and how to use enhanced software trim to cover the temperature and voltage drift. MSP430FR2xx/FR4xx DCO+FLL Code Examples are provided and test results are also discussed.

Table of Contents

List of Figures

List of Tables

Trademarks

MSP430™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

1 Introduction for DCO and FLL in MSP430FR2xx/FR4xx Devices

Clock system is very important for microcontroller (MCU) system. In MSP430FR2xx/FR4xx devices, the clock system (CS) module has six clock sources integrated:

1. DCO – Digitally controlled oscillator
2. XT1CLK – High-frequency or low-frequency oscillator working with external crystal
3. VLOCLK – Very-low-power 10 kHz typical frequency oscillator
4. REFOCLK – Trimmed 32768 Hz typical frequency oscillator
5. MODCLK – 5 MHz typical frequency oscillator
6. FLL – Frequency-locked loop

Please refer to MSP430FR4xx and MSP430FR2xx Family User's Guide for more details about the clock sources and the block diagram. This document will introduce the DCO and FLL and how they work together to achieve accurate target frequency.

1.1 DCO Operation Theory

The DCO is an internal digitally controlled oscillator without using any external components such as a crystal or RC (resistor, capacitor) combination. The DCO frequency can be adjusted by software configuring the DCORSEL, DCO, MOD, and DCOFTRIM bits. Figure 1-1 shows the DCO block diagram for MSP430FR2xx/4xx.

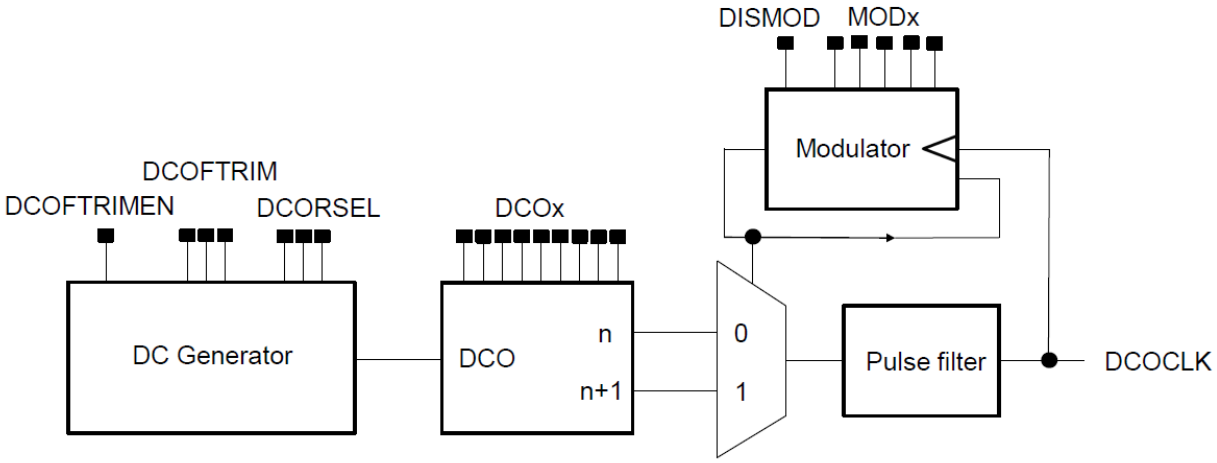

Figure 1-1. DCO Block Diagram for MSP430FR2xx/4xx

MSP430FR2xx/4xx register CSCTL1.DCORSEL defines several frequencies range: 1 MHz, 2 MHz, 4 MHz, 8 MHz, 12 MHz, 16 MHz, 20 MHz, and 24 MHz. Please notice only MSP430FR2x5x supports high frequency range select of 20 MHz and 24 MHz. For DCO frequency setting, the DCOFTRIM 3-bits can be considered a 'coarse' tuning, and the DCOx 9-bits represent more of a 'fine' tuning. DCOFTRIM has 7 trim taps and DCOx has 512 DCO taps.

Please refer to the section 5.13.3 Clock Specifications in MSP430FR2xx/FR4xx device specific data sheet for the typical DCO frequencies with different DCO register settings. Figure 1-2 shows the typical DCO frequency for MSP430FR231x. It shows the frequency gap between two neighboring DCORSEL settings for the same DCOFTRIM configuration when DCO frequency range is less than 8 MHz (DCORSEL=3).

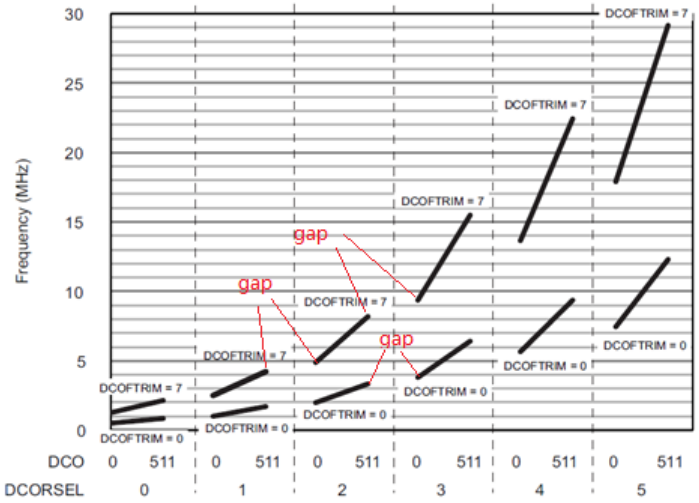

Figure 1-2. Typical DCO Frequency for MSP430FR2311

In MCUs, with the same DCO register setting, silicon production variations, temperature drift and voltage drifts can cause the DCO frequency different. To achieve a stable and well defined target frequency, the DCO frequency can be adjusted by software (see Section 1.4), or can be adjusted and stabilized by FLL which is integrated in MSP430FR2xx/FR4xx devices.

1.2 DCO Difference Between MSP430FR2xx/4xx and MSP430F5xx

There is also DCO in MSP430F5xx. But there is some significant difference from the DCO in the MSP430FR2xx/4xx.

1. DCO tap steps
   - In MSP430F5xx, DCO register is 5 bits and the DCO tap tuning range is from 0 to 31.
   - In MSP430F2xx/4xx, DCO register is 9 bits and the DCO tap tuning range is from 0 to 511 which results in frequency step size much smaller than MSP430F5xx DCO. One benefit from this change is that the clock jitter is smaller for MSP430F2xx/4xx DCO+FLL.
2. Frequency tuning coverage gap
   - For MSP430F5xx DCO, there is no frequency coverage gap across DCO taps for adjacent DCORSEL values. Figure 1-3 shows the Typical DCO Frequency for MSP430F5529. It is observed that the frequency of DCOx = 31 is always greater than DOCx = 0 of next DCORSEL.

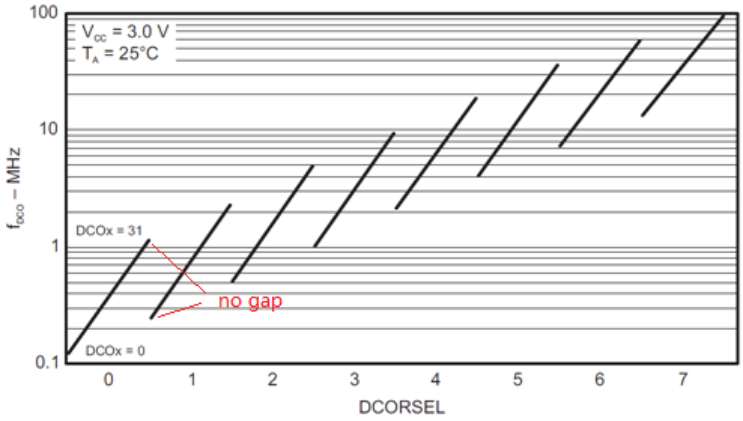

Figure 1-3. Typical DCO Frequency for MSP430F5529

   - For MSP430FR2xx/4xx DCO, there is frequency coverage gap across DCO taps for adjacent DCORSEL values and the same DCOFTRIM configuration. It is obviously observed in the Figure 1-2 when frequency range is less than 8 MHz.

- Please refer to the DCO Frequency section in the specific data sheets for details. Figure 1-4 is a capture from the Table 5-6 of MSP430FR231x Mixed-Signal Microcontrollers.
  - For DCORSEL = 3 and DCOFTRIM = 0, the frequency range is 3.8 MHz ~ 6.5 MHz.
  - For DCORSEL = 2 and DCOFTRIM = 0, the frequency range is 2 MHz ~ 3.2 MHz.
  - There is 0.6 MHz (3.8 – 3.2) frequency gap not covered by both DCORSEL = 2 and DCORSEL = 3. That means the target frequency within 3.2 MHz ~ 3.8 MHz cannot be locked by only tuning DCORSEL and DCO tap with DCOFTRIM = 0.
  - It is similar for case of DCOFTRIM = 7: frequency gap is 1.5 MHz (9.5 – 8).

$f_{DCO, 8 MHz}$	DCO frequency, 8 MHz	DCORSEL = 011b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 0	3.8	MHz
		DCORSEL = 011b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 511	6.5	
		DCORSEL = 011b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 0	9.5	
		DCORSEL = 011b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 511	16	
$f_{DCO, 4 MHz}$	DCO frequency, 4 MHz	DCORSEL = 010b,, DISMOD = 1b, DCOFTRIM = 000b, DCO = 0	2	MHz
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 511	3.2	
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 0	4.8	
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 511	8	

Figure 1-4. DCO Frequency Range for MSP430FR2311

3. DCOFTRIM
   - In MSP430FR2xx/4xx, there is DCOFTRIMEN register and DCOFTRIM register. DCOFTRIMEN bit default setting is DCO software trim disabled.
   - In MSP430FR5xx, there is not DCOFTRIMEN register and DCOFTRIM register.
   - DCOFTRIM can also be used for frequency tuning with DCO tap together. The DCOFTRIM bits can be considered a 'coarse' tuning of the DCO frequency, and the DCO bits represent more of a 'fine' tuning.

1.3 FLL Operation Theory

The FLL is a frequency-locked loop which continuously counts up or down a frequency integrator. The frequency integrator controls the nine bits of the DCO frequency tap to tune the frequency automatically. Figure 1-5 shows the FLL block diagram for MSP430FR2xx/FR4xx.

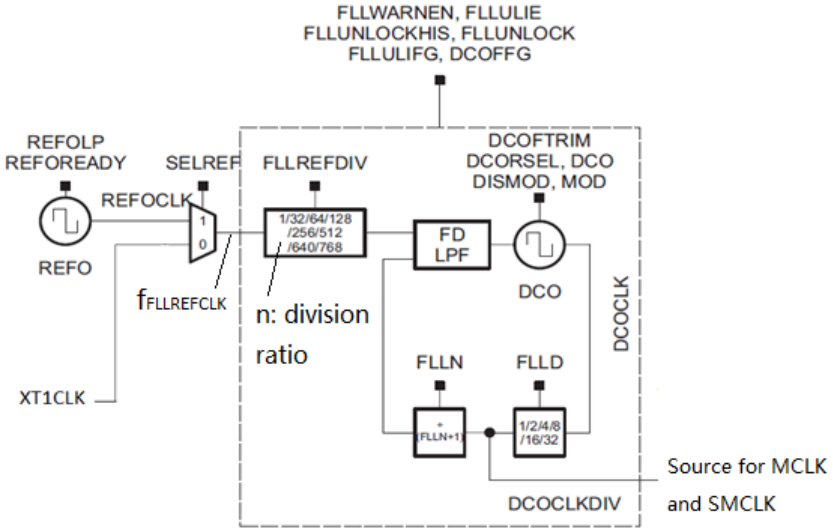

Figure 1-5. FLL Block Diagram for MSP430FR2xx/FR4xx

The FLL compares divided FLL reference (FLLREFDIV output) frequency and the FLLN divided target frequency, and control the DCO taps to increase or decrease in fixed steps based on the comparison result until the FLLN divided target frequency is very close to the divided FLL reference clock frequency.

The FLL is working together with the DCO and the reference clock. The reference clock $f_{FLLREFCLK}$ can be from REFOCLK or XT1CLK, which is selected by SELREF bit in register CSCTL3. There is a divider FLLREFDIV to divide the reference clock for FLL use. Please note the FLLREFDIV is always read and written as 0 (division ratio n=1) if the REFOCLK and XT1CLK is only 32 kHz clock.

1.4 DCO Calibration and Stabilization

As talked in Section 1.1, DCO varies lot due to variation of process, drift of temperature and voltage. To get an accurate DCO clock frequency, the DCO must be tuned or calibrated based on a known accurate clock frequency. There are three methods to achieve this.

1. Factory trim: Load factory DCO trim data for DCOFTRIM, DCO, and MOD.
   - In MSP430 device production, there is test program to trim the DCO frequency with external accurate frequency source. The trimmed ready data is saved in non-volatile memory TLV table before shipping to customer. Please refer to MSP430FR4xx and MSP430FR2xx Family User's Guide and device specific data sheet for the TLV introduction.
   - When initializing the device in user code, after selecting the clock source and setting the target frequency, load the trim data from TLV table into the DCO tap register. The accurate DCO clock frequency can be achieved quickly (see Section 3.1 for details).

   Note

   Note, in MSP430FR5xx/6xx devices without FLL, all the defined frequencies have trimmed data in the TLV table. Setting the DCOFSEL register to the target frequency range will automatically load the corresponding factory trim data
2. Software FLL: Software calibration for DCO frequency with accurate reference clock
   - As mentioned in the application report Using the DCO Library, MSP430™ built-in timer module can be used to measure the DCO frequency with SMCLK clock as the timer clock source to count in one or more 32 kHz crystal cycles.
   - Based on the measurement result, adjust the DCO register value to tune the DCO frequency. Because the crystal 32 kHz frequency is accurate, the calibrated DCO clock frequency will also be accurate.
3. FLL module: DCO frequency tuned and stabilized with FLL and accurate reference
   - FLL module automatically calibrates the DCO frequency and stabilizes the frequency across the process, temperature and voltage variation. FLL needs to work with DCO and reference clock together. The FLL module exists in MSP430F4xx, MSP430F5xx, MSP430F6xx, MSP430FG4xx, MSP430FG6xx, MSP430FR2xx, and MSP430FR4xx devices.

2 How to Achieve Accurate Frequency With DCO+FLL

By default after power on, FLL is enabled. In this case, the DCO frequency is stabilized by the FLL operation. FLL operation can be disabled by setting SCG0. When the FLL is disabled, the DCO continues to operate at the current settings defined in registers CSCTL0 and CSCTL1. The DCO frequency can be adjusted manually if desired.

From the MSP430FR2xx/4xx specification, the DCO can start within 10 us. This provides features of fast startup after power on and fast wakeup from low power modes. The typical FLL lock time is 200 ms. Before FLL locked, the MCLK and SMCLK default clock source DCOCLKDIV is not accurate. This will not impact the user code execution. But it will impact the applications execution which needs accurate clock source as soon as possible after power on such as high speed UART communication. Section 4 discusses how to fast lock target frequency with DCO+FLL.

To achieve the accurate frequency, the following configurations are required.

- Disable FLL
- Select the reference clock source for FLL
- Set the DCO range
- Set target frequency by configuring FLL registers FLLN, FLLD
- Enable the FLL and check the FLL lock status

Section 3 discusses the detailed process sequence for the different frequency configuration.

FLL is used to stabilize the DCO clock. With FLL locked, the frequency of DCOCLK and DCOCLKDIV will have accuracy close to $f_{FLLREFCLK}$ — REFOCLK or XT1CLK. The divider (FLLN + 1) and the divider FLLD define the DCOCLK and DCOCLKDIV frequencies.

$$f_{DCOCLKDIV} = (FLLN + 1) \times (f_{FLLREFCLK} \div n) \quad (1)$$

Where:

- $f_{FLLREFCLK}$ – FLL reference clock
- n – division ratio of divider FLLREFDIV (n is fixed 1 if $f_{FLLREFCLK}$ is 32kHz)

The MOD function in the DCO module can be enabled by setting register CSCTL0.MOD. The modulator mixes two DCO frequencies, f_{DCO} and f_{DCO+1} to produce an intermediate effective frequency between f_{DCO} and f_{DCO+1}. With this feature, more frequencies can be generated.

FLL lock status can be checked by FLLUNLOCK bits in CSCTL7 register.

The chapters 3.2.5, 3.2.6, 3.2.7, 3.2.8, and 3.2.9 in MSP430FR4xx and MSP430FR2xx Family User's Guide introduce details about how to set DCO frequency, how to use DCO modulator function, and FLL lock/unlock status detection with DCO+FLL.

When using the DCO+FLL to achieve accurate DCO clock, the reference clock is critical. If the reference clock accuracy is better across the temperature and voltage, the DCO clock frequency accuracy will be better.

The reference clock $f_{FLLREFCLK}$ can be from REFOCLK or XT1CLK, selected by SELREF bit in register CSCTL3. The DCO+FLL stabilized DCO clock accuracy specification can be known from the section 5.13.3 Clock Specifications in MSP430FR2xx/FR4xx device specific data sheet. Figure 2-1 shows the example specification captured from MSP430FR231x Mixed-Signal Microcontrollers.

over recommended operating free-air temperature (unless otherwise noted)

PARAMETER		TEST CONDITIONS	V_{CC}	MIN	TYP	MAX	UNIT
$f_{DCO, FLL}$	FLL lock frequency, 16 MHz, 25°C	Measured at MCLK, internal trimmed REFO as reference	3.0 V	–1.0%		1.0%	
	FLL lock frequency, 16 MHz, –40°C to 85°C			–2.0%		2.0%	
	FLL lock frequency, 16 MHz, –40°C to 85°C	Measured at MCLK, XT1 crystal as reference		–0.5%		0.5%	
f_{DUTY}	Duty cycle	Measured at MCLK, XT1 crystal as reference	3.0 V	40%	50%	60%	
$Jitter_{cc}$	Cycle-to-cycle jitter, 16 MHz	Measured at MCLK, XT1 crystal as reference	3.0 V		0.25%		
$Jitter_{long}$	Long-term jitter, 16 MHz	Measured at MCLK, XT1 crystal as reference	3.0 V		0.022%		
$t_{FLL, lock}$	FLL lock time, 16 MHz	Measured at MCLK, XT1 crystal as reference	3.0 V		200		ms

Figure 2-1. DCO+FLL Specification for MSP430FR2311

3 How to Lock Factory Defined and In-Between Frequency With DCO+FLL

MSP430FR2xx/4xx register CSCTL1.DCORSEL defines several frequencies range: 1 MHz, 2 MHz, 4 MHz, 8 MHz, 12 MHz, 16 MHz, 20 MHz, and 24 MHz. But only the high frequencies 16 MHz and 24 MHz have the factory trim data saved in TLV memory. For the target frequency other than 16 MHz and 24 MHz, there is no factory trim data. And because the frequency range is large, the software trim DCOFTRIM also needs to be adjusted. Then the software trim algorithm is needed. So the FLL lock software drivers for these frequencies are different.

3.1 FLL Lock Factory Trimmed Frequency

For the maximum frequency 24 MHz MSP430FR2xx/4xx devices, there are two factory trimmed DCO tap data stored in the TLV table for the target frequencies of 16 MHz and 24 MHz. Figure 3-1 shows the addresses of the two trim data (for 16 MHz and 24 MHz) captured from table 6-70 of MSP430FR235x, MSP430FR215x Mixed-Signal Microcontrollers.

Reference and DCO calibration	Calibration tag	1A26h	12h
	Calibration length	1A27h	0Ah
	Internal shared 1.5-V reference factor	1A28h	Per unit
		1A29h	Per unit
	Internal shared 2.0-V reference factor	1A2Ah	Per unit
		1A2Bh	Per unit
	Internal shared 2.5-V reference factor	1A2Ch	Per unit
		1A2Dh	Per unit
	DCO tap settings for 16 MHz, temperature 30°C	1A2Eh	Per unit
		1A2Fh	Per unit
	DCO tap settings for 24 MHz, temperature 30°C (4)	1A30h	Per unit
		1A31h	Per unit

(3) The calibration value is device dependent at 105°C.
(4) This value can be directly loaded into the DCO bits in the CSCTL0 register to get an accurate 24-MHz frequency at room temperature, especially when MCU exits from LPM3 and below. TI also suggests to use a predivider to decrease the frequency if the temperature drift might result an overshoot faster than 24 MHz.

Figure 3-1. DCO Tap Trim Data for MSP430FR2355

For maximum frequency 16 MHz MSP430FR2xx/4xx devices, only 16 MHz DCO tap trim data is available. Please refer to the specific data sheet for details. Please note, the DCO tap trim data in the TLV is only for room temperature 30°C.

There are two kinds of software routine for FLL to lock 24 MHz or 16 MHz frequency.

1. Recommended process in the user's guide – normal FLL lock.
   a. Disable the FLL.
   b. Select the reference clock.
   c. CLear the CSCTL0 register.
   d. Set the DCO range and set FLLN and FLLD for target frequency.
   e. Execute NOP three times to allow time for the settings to be applied.
   f. Enable the FLL.
   g. Poll the FLLUNLOCK bits until FLL is locked.

   Please refer to code example msp430fr231x_CS_03.c in MSP430FR231x Code Examples or other MSP430FR2xx/4xx device code examples.

   This process is to clear the DCO tap before enabling the FLL. The FLL will adjust the DCO tap from zero. Finally the DCO tap will be stabilized at any value of 0~511. So the lock time will be long up to tens of milliseconds or two hundreds milliseconds. But this process is very reliable without frequency overshooting risk.
2. Load the DCO tap TLV trim data into the register CSCTL0.DCO – faster FLL lock.

a. Disable the FLL.
b. Select the reference clock.
c. Clear the CSCTL0 register.
d. Set the DCO range and set FLLN and FLLD for target frequency.
e. Set CSCTL0.DCO = Trimmed DCO tap value TLV.
f. Execute NOP three times to allow time for the settings to be applied.
g. Enable the FLL.
h. Set the MCLK source to REFOCLK or XT1CLK and delay at least 10 cycles.
i. Poll the FLLUNLOCK buts until FLL is locked.
j. Set the MCLK source back to DCOCLKDIV

This process is to load the DCO register with DCO tap TLV trim data directly. It can speed up the FLL lock to target frequency. The MCLK source switch can simplify the wait cycles calculation. To avoid the MCLK overshoot risk and reserve time for FLL logic settled, step h is introduced. An example of this software routine for MSP430FR2355 is available in the MSP430FR2x5x_FLL_FastLock_24MHz-16MHz.c which is in the reference software package (MSP430FR2xx/FR4xx DCO+FLL Code Examples) for this application report.

Based on the test in room temperature, the FLL lock time (from the clock system registers configuration to MCLK output on package pin) is about 750~800 us with this code example on MSP430FR2355 LaunchPad™ Development Kit for MCLK 16 MHz. Figure 3-2 shows the lock time captured on the board. The example code can also be used for other MSP430FR2xx/4xx devices with only the GPIO pins to be checked if available.

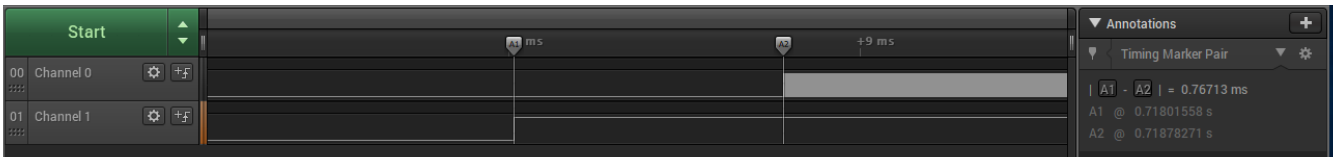

Figure 3-2. FLL Lock Time on MSP430FR2355 LaunchPad™ Development Kit With the Fast Lock Code Example

3.2 FLL Lock Other Defined Frequency and In-Between Frequency With Software Trim

Due to the frequency gap discussed in Section 1.1 and Section 1.2, the target frequency will not be locked if it locates in the frequency gap area and without changing the register DCOFTRIM. We need to involve the DCOFTRIM for frequency coarse tuning and DCO tap for fine tuning. FLL cannot automatically tune the DCOFTRIM to achieve the target frequency. So the software trim solution is needed. Except factory trimmed frequency 16 MHz and 24 MHz, the other factory defined frequencies such as 1 MHz, 2 MHz, 4 MHz, 8 MHz, and 12 MHz also need software trim function to lock because the default DCOFTRIM may not be the best setting for the frequency due to the variation of process, temperature and voltage.

3.2.1 Recommended Program Sequence of the Software Trim

Chapter 3.2.11 in MSP430FR4xx and MSP430FR2xx Family User's Guide introduces detailed about the software trim. Following recommended program sequence is just a capture from it.

1. Disable the FLL.
2. Select the reference clock.
3. Set DCO range, enable DCO frequency trim.
4. Set FLLN and FLLD for target frequency.
5. Execute NOP three times to allow time for settings to be applied.
6. Enable th FLL.
7. Set DCO tap = 256.
8. Clear the DCO fault flag (DCOFFG).
9. Wait to allow the FLL lock status (FLLUNLOCK) to be stable. The minimum recommended wait time is 24 divided by the FLL reference clock frequency (for example, if the divided reference clock = 32768 Hz, wait time = 0.733ms). Note: 24 can be changed to 12 to reduce the lock time.
10. Poll the FLLUNLOCK bits and DCOFFG bit until FLL is locked without DCO fault.

11. Read DCO tap value, calculate the delta value between the DCO tap and 256.
12. Record the CSCTL0 and CSCTL1 registers value when the DCO tap is closer to 256 than the last recorded one.
13. Increase or decrease DCOFTRIM based on the relative position of DCO tap with respect to the midrange (if DCO tap < 256, DCOFTRIM – 1; if DCO tap ≥ 256, DCOFTRIM + 1).
14. Repeat the process starting from step 7 in a loop until the DCO tap values cross 256 for two adjacent DCOFTRIM settings.
15. Reload the recorded CSCTL0 and CSCTL1 for which DCOTAP is closest to 256.

The software time algorithm is to tune the DCOFTRIM from default value until the DCO tap values for FLL locks cross the 256, where 256 is the midrange of DCO tap range 0 ~ 511. Because the DCO tap is automatically tuned by FLL, the FLL locked DCO tap will be varied with temperature and voltage variation. It is recommended the DCO tap is locked close to the midrange to ensure the FLL always locked for large variations of temperature and voltage.

An implementation of this software routine is available in the code example msp430frxx_cs07.c which is in the reference software package (MSP430FR2xx/FR4xx DCO+FLL Code Examples) for this application report. Alternatively, the Driver Library API also implements this routine. To be clearer and easier to understand, there is a flow chart showed as Figure 3-3

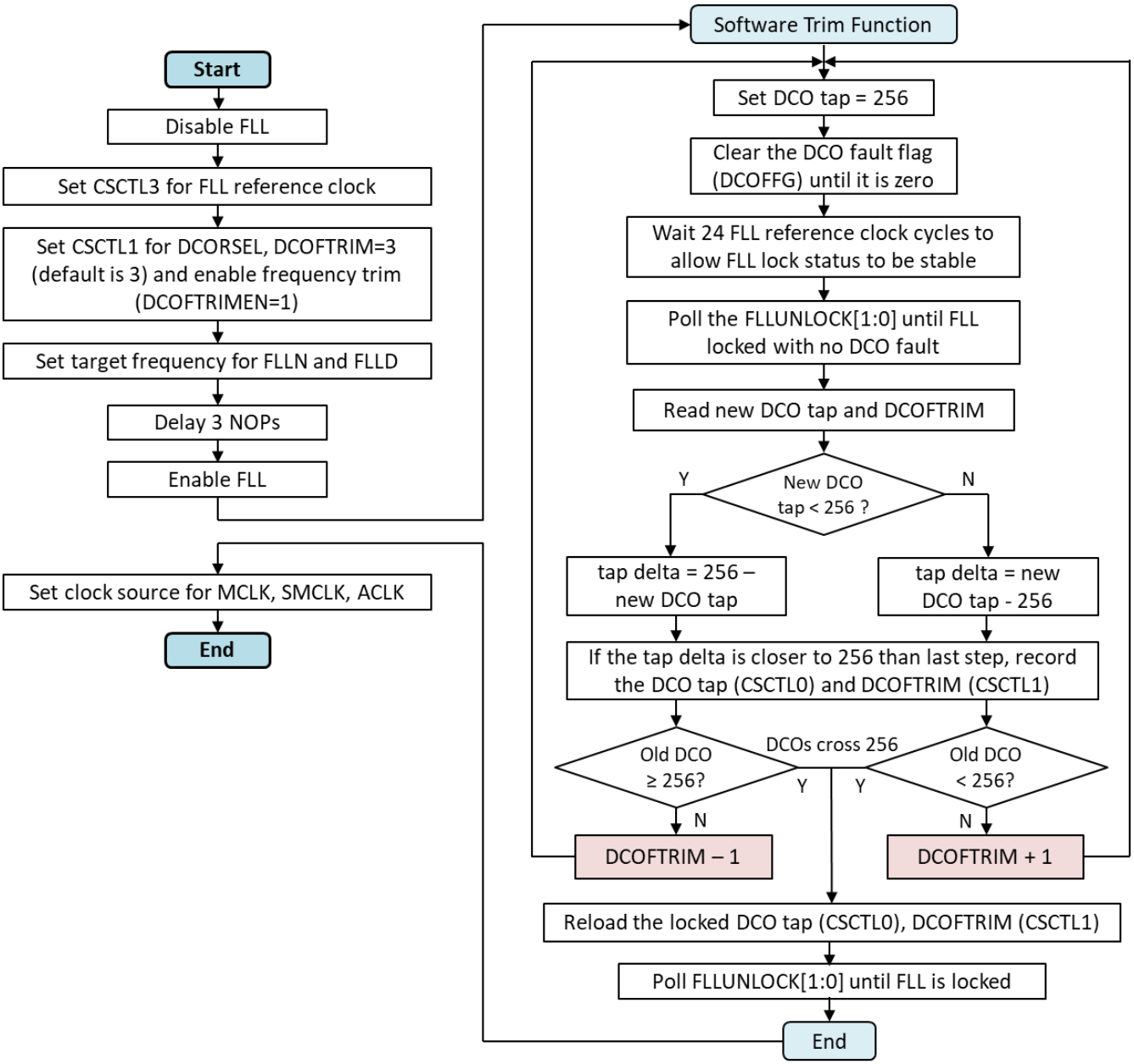

Figure 3-3. DCO+FLL Software Trim Flow Chart

3.2.2 How to Set the DCORSEL

The code example msp430frxx_cs07.c doesn't mention how to set the DCORSEL - DCO range select for in-between frequency. User may be confused for the selection. For example, how to set the DCORSEL for target frequency 5.5 MHz for MSP430FR2311? In Figure 1-4, both DCORSEL = 2 and DCORSEL = 3 can be set since the DCOFTRIM will be adjusted for frequency tuning in the software trim function. For this case, it is required to find the best setting for the DCORSEL. Otherwise, the final locked DCO tap may not be in the midrange so that there is risk for DCO tap tuned close to the DCO tap boundary of 0 or 511.

For the software trim recommended program sequence introduced in Section 3.2.1, since the requirement is to exit the while loop after DCO taps cross the midrange 256, there will be risk that the software trim function will not be ended if the target frequency is close to the boundary of the defined DCORSEL frequency range while the DCORSEL setting is incorrect.

$f_{DCO, 4MHz}$	DCO frequency 4 MHz	DCORSEL = 010b,, DISMOD = 1b, DCOFTRIM = 000b, DCO = 0	T	3.0 V	2	MHz
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 511	T	3.0 V	3.4	
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 0	T	3.0 V	5	
		DCORSEL = 010b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 511	T	3.0 V	8.2	
$f_{DCO, 2MHz}$	DCO frequency 2 MHz	DCORSEL = 001b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 0	T	3.0 V	1	MHz
		DCORSEL = 001b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 511	T	3.0 V	1.7	
		DCORSEL = 001b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 0	T	3.0 V	2.5	
		DCORSEL = 001b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 511	T	3.0 V	4.2	
$f_{DCO, 1MHz}$	DCO frequency 1 MHz	DCORSEL = 000b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 0	T	3.0 V	0.5	MHz
		DCORSEL = 000b, DISMOD = 1b, DCOFTRIM = 000b, DCO = 511	T	3.0 V	0.85	
		DCORSEL = 000b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 0	T	3.0 V	1.2	
		DCORSEL = 000b, DISMOD = 1b, DCOFTRIM = 111b, DCO = 511	T	3.0 V	2.1	

Figure 3-4. DCO Frequency Range for MSP430FR2355

For example, Figure 3-4 shows several DCO frequencies range for MSP430FR2355. For the target frequency 3.2 MHz, either DCORSEL=1 (frequency range is 1 MHz to 4.2 MHz) or DCORSEL=2 (frequency range is 2 MHz to 8.2 MHz) can be selected. If setting DCORSEL=1, the recommend software trim function will not exit to the main function because the FLL locked DCO taps are always greater than 256 so as to no chance for DOC taps to cross 256. If setting DCORSEL=2, the software trim function will work normally as expected.

The recommended rule for DCOREL configuration is: If the target DCO frequency is in-between of the DCORSEL defined frequencies, the recommended DCORSEL setting is the one for which the defined frequency is closer to the target frequency.

- Set DCORSEL to n, when FTarget_frequency – FDefined_frequency[n] < FDefined_frequency[n+1] – FTarget_frequency
- Set DCORSEL to n+1, when FTarget_frequency – FDefined_frequency[n] > FDefined_frequency[n+1] – FTarget_frequency
  - Where n is the DCORSEL setting within 0 ~ 7.

For above case target frequency 5.5 MHz, the DCORSEL = 2 is for defined frequency 4 MHz, and DCORSEL=3 is for defined frequency 8 MHz. It is observed that the 4 MHz is closer to 5.5 MHz than 8 MHz. So the DCORSEL=2 should be selected.

4 How to Fast Lock Target Frequency With DCO+FLL

The typical FLL lock time for DCO+FLL is about 200 ms. Some applications request the accurate clock to be available within 1~2 ms after power on or reset. It is a challenge for DCO+FLL since the FLL lock takes time. But it is possible to achieve this feature by loading the custom calibrated DCO tap value into the register directly.

4.1 Fast Lock Factory-Trimmed Frequency

For factory-trimmed high frequency, the fast lock solution has been discussed in Section 3.1. But there are some notes to be taken cared.

1. Avoid overshoot issue before the FLL lock. This has been mentioned in the steps of Section 3.1: switch MCLK source to REFOCLK or XT1CLK and delay at least 10 cycles.
2. Package shift impact: the TLV data in the specific data sheet may have some deviation to the target defined frequency due to package stress effect. Please refer to the errata number CS14 in MSP430FR2111 Device Erratasheet. This issue is only found in MSP430FR2111 in the MSP430FR2xx/4xx family devices.

4.2 Fast Lock Other Defined Frequency and In-Between Frequency

It is also available to lock the other factory defined frequency and in-between frequency by loading the calibrated DCO registers value directly.

Since there is no factory DCO tap trim value for the target frequency, it is recommended to store the FLL locked DCO tap, DCOFTRIM and DCORSEL values into the nonvolatile FRAM memory after the first software trim of DCO+FLL. This DCO calibration can be done at factory test or on site. There should be a calibration flag set which is also stored in the FRAM memory. When the product powers on or hardware resets again, the stored DCO registers value can be loaded directly from FRAM memory to speed up the FLL lock if the calibration flag is set.

Because the factory test environment may be different from the real application environment, it is recommended to develop on-site calibration function in the firmware. User can trigger the software trim function on-site and store the calibrated DCO tap, DCOFTRIM and DCORSEL values into the nonvolatile FRAM memory and then set the calibration flag.

An example of this software routine for MSP430FR2355 is available in MSP430FR2x5x_FastLock_InBetweenFreq.c which is in the reference software package (MSP430FR2xx/FR4xx DCO+FLL Code Examples) for this application report.

In the example code, the software trim is called for the DCO calibration. After software trim FLL locked, store the DCO registers CSCTL0, CSCTL1 and the calibration flag into information FRAM memory. The calibration flag will be checked at next time MSP430™ resets. If the flag is set, the code will load the calibration DCO registers CSCTL0 and CSCTL1 from FRAM directly and check FLL lock status after a recommended delay.

To shorten the software trim execution time, the condition of (newDcoDelta < FLL_Softtrim_DCOTapDelta0) is added to the variable end-loop setting conditions. FLL_Softtrim_DCOTapDelta0 is the threshold setting for the DCO tap delta to midrange 256. If the new DCO tap is within the range of (256+/- FLL_Softtrim_DCOTapDelta0), it can be regarded the DCOFTRIM is tuned to the best value. FLL_Softtrim_DCOTapDelta0 is set to 50 in the example code which is based on the test results of adjacent DCOFTRIM settings.

There are 3 target frequencies can be selected in the example code: 3.2 MHz, 1 MHz, and 8 MHz. Micro TARGET_FREQ_3_2MHz, TARGET_FREQ_1MHz, and TARGET_FREQ_8MHz are defined for each frequency. XT1CLK and REFOCLK are also available to be selected as the FLL reference clock source through the micro definition CRYSTALXT1_FOR_FLLREF. The default setting for the example code is 3.2 MHz target frequency and REFOCLK as the FLL reference clock source.

After download the code into the MSP430FR2355, the first time FLL lock time will be long to 91.5 ms for the DCO+FLL calibration and storing calibration data into FRAM. When the board is reset again, the FLL lock time will be less than 757 us due to the loading of the DCO calibration data from FRAM. Please find the lock time test results on MSP430FR2355 LaunchPad™ Development Kit in the Figure 4-1 and Figure 4-2.

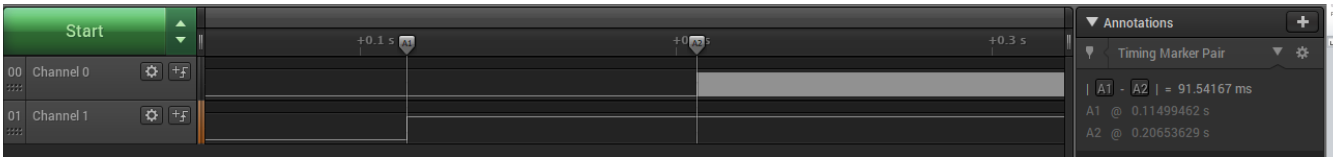

Figure 4-1. First Power-on FLL Lock Time for MSP430FR2355 With the Fast Lock Code Example

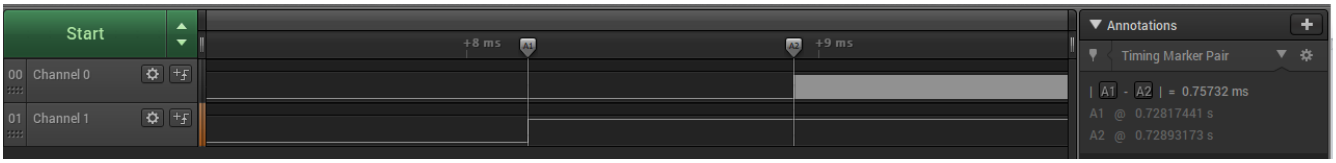

Figure 4-2. Subsequence Power-on or Reset FLL Lock Time for MSP430FR2355 With the Fast Lock Code Example

Figure 5-1 in Section 5.2 shows the flow chart of the fast lock example code.

5 FLL Lock the Target Frequency Over Temperature and Voltage Drift

5.1 DCO+FLL Clock Accuracy Drift With Temperature and Voltage

After FLL locked, the DCO clock frequency will track the reference clock frequency automatically by FLL. If the reference clock frequency drifts with the temperature and voltage, the DCO clock frequency will drift. For the application in which the temperature and voltage drift is considered, the FLL reference clock drift performance needs to be considered for the clock accuracy.

If REFOCLK as the FLL reference, the REFO drift specification df_{REFO}/d_T (REFO frequency temperature drift) and df_{REFO}/d_{VCC} (REFO frequency supply voltage drift) in the specific data sheet need to be checked.

Because the crystal clock frequency is very stable over the temperature and voltage, it is recommended to use the crystal frequency (XT1CLK) as the FLL reference clock if there is big variation for temperature or voltage in the product application environment and the clock accuracy requirement is high.

5.2 Enhanced Software Trim for FLL Lock Over the Temperature and Voltage Drift

As mentioned in Section 3.2.1, the DCO tap is automatically tuned by FLL. If the temperature and voltage drift, the FLL locked DCO tap will also drift. From limited tests, the DCO drift varies a lot among the MSP430FR2xx/4xx devices. It can be regarded as 40 to 160 DCO taps for temperature drift from -40°C to 85°C.

Following scenario should be considered:

- The product with MSP430FR2xx/4xx powers on at room temperature with DCO+FLL locked.
- The working environment temperature changes to -40°C after several hours.
- The DCO tap may drift far away from the midrange 256 – increased and close to 511.
- When the DCO tap drifts out of range (0~511), the FLL will be unlocked.

This is the same issue as errata number CS11 in MSP430FR4133 Device Erratasheet.

For the working environment without too much voltage and temperature drift (i.e. no risk for the DCO tap drift out of range), the original software trim introduce in Section 3.2 can work well. But for the working environment with temperature and voltage drift a lot, it is necessary to consider the FLL lock failed possibility if the DCO+FLL accuracy is important to the system.

The solution to make FLL always locked is to monitor the DCO tap and execute the software trim again if DCO tap drifts to range boundary.

- If the DCO tap is close to the lowest boundary (DCO = 0) or the highest boundary (DCO = 511) and the difference to the boundary is less than defined threshold taps (minimum 50 taps is recommended to keep DCO tap reserve margin for further drift of temperature and voltage), it is recommended to change the DCOFTRIM and rerun the software trim execution.

Original software trim introduced in Section 3.2 doesn't consider the case of DCOFTRIM at its boundary 0 or 7. If the DCO tap drift is large or the DCORSEL is wrongly set, there will be risk software trim subroutine will not return to main function. So it is necessary to enhance the software trim function to cover the large variation of temperature and voltage.

- If the DCOFTRIM is already at boundary 0 or 7 while FLL locked DCO tap is close to the boundary and less than defined threshold taps, it is recommended to change DCORSEL and set DCOFTRIM to default value 3 before the software trim execution.

Here is the solution for the DCO monitor and enhanced software trim.

1. Monitor the DCO tap.
   - If the DCO tap < 50 or (511- DCO tap) < 50, call the enhanced software trim function.
2. Enhance software trim function.
   - Configuration is same as software trim function in Figure 3-3.
   - Add DCOFTRIM boundary check and DCORSEL tuning to cover large variation.
     - If the DCO tap < 256 and DCOFTRIM > 0, decrease DCOFTRIM by 1.
     - If the DCO tap < 256 and DCOFTRIM = 0, decrease DCORSEL by 1 and set DCOFTRIM = 3.

  - If the DCO > 256 and DCOFTRIM < 7, increase DCOFTRIM by 1.
  - If the DCO > 256 and DCOFTRIM = 7, increase DCORSEL by 1 and set DCOFTRIM = 3.
- Return different values for different lock conditions.

An example of this software routine for MSP430FR2355 is available in the MSP430FR2x5x_FastLock_InBetweenFreq+PVTVariation(EnhancedSoftTrim+DCOMonitor).c which is in the reference software package (MSP430FR2xx/FR4xx DCO+FLL Code Examples) for this application report. It is based on the example code for Section 4.2, adding the enhanced software trim and DCO monitor functions. Figure 5-1 shows the flow chart for this example code.

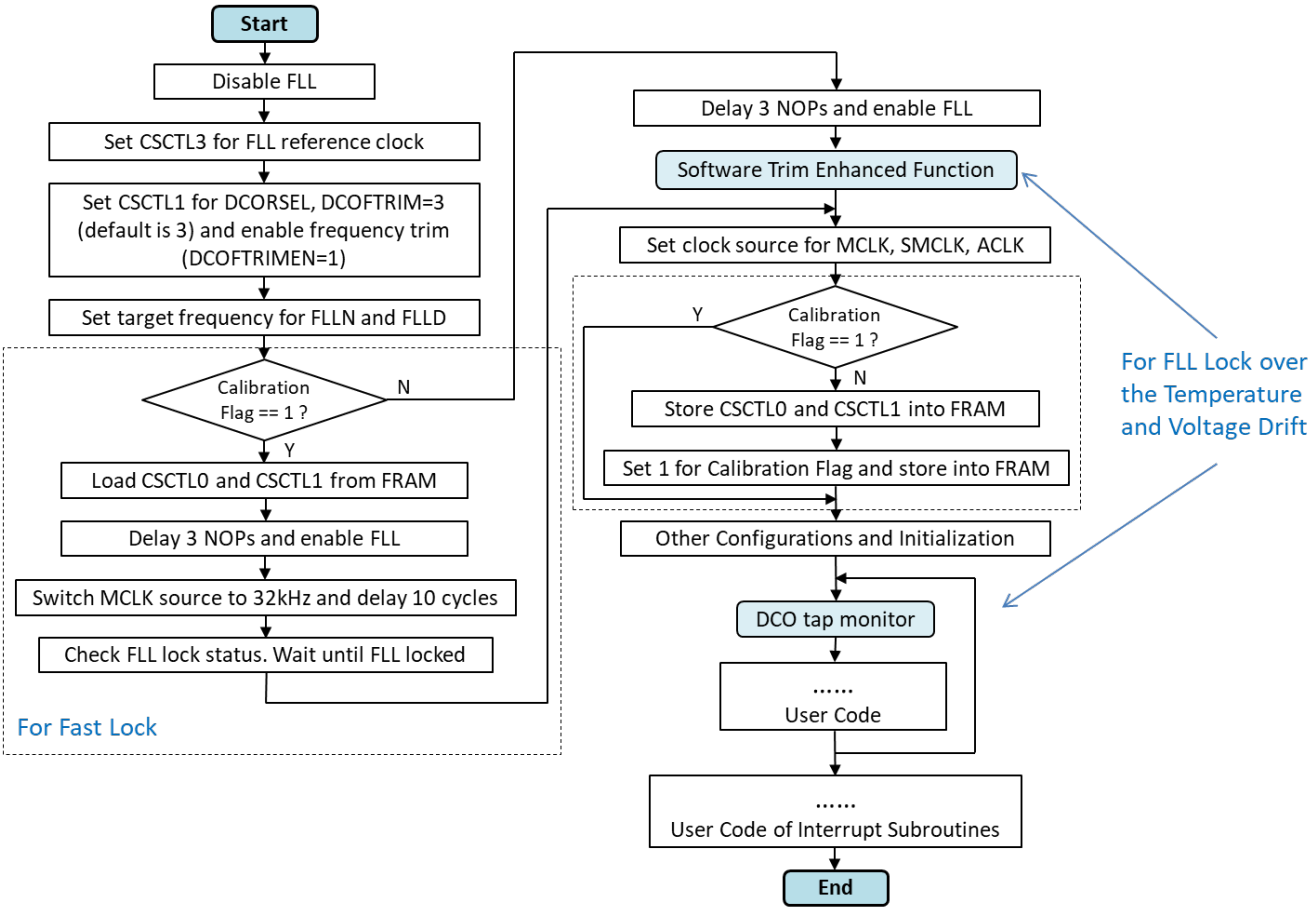

Figure 5-1. Flow Chart for DCO+FLL Fast Lock, Enhanced Software and DCO Monitor

Normally, for the software trim function in Section 3.2, if the DCORSEL is set correctly and the FLL is locked, there is no need to change the DCOFTRIM because the DCO taps cross the midrange 256 which gives enough margin for DCO tap drifting to boundary to cover the temperature variation. But if the FLL locked DCO tap is close to the boundary 0 or 511 at the power-on and the temperature and voltage variation is large for which there is risk for DCO tap to drift out of range, the DCO monitor and enhanced software trim are needed to be considered.

The enhanced software trim and DCO monitor subroutines can be used to cover the process, temperature and voltage variation of the MSP430FR2xx/4xx device. It can also be used as the workaround for the issue CS11 in MSP430FR4133 Device Erratasheet.

In addition, this enhanced software trim can also fix the subroutine no exit issue described in Section 3.2.2 – target frequency 3.2 MHz and DCORSEL=1. In the example code, the target frequency is 3.2 MHz and the DCORSEL=1 is set. From the test result, the DCORSEL is tuned to 2 and the DCO tap is close to midrange 256.

6 Summary

The DCO+FLL in MSP430FR2xx/4xx are a little different from other MSP430™ families. Due to the smaller DCO tap step size, the DCOCLK jitter is smaller. But this also introduces DCO frequency tuning coverage gap for FLL. In addition, because the default DCOFTRIM value may not be the best value for the target frequency, the DCOFTRIM tuning is requested while FLL doesn't implement it automatically in hardware. So the software trim solution is introduced. For specific applications to cover the large variations from temperature and voltage for which there is risk for DCO tap to drift out of range, the DCO monitor and enhanced software trim are necessary to be considered.

With the enhanced software trim, any frequency less than device maximum frequency can be locked for the DCO+FLL. This provides the flexibility for the applications which need to generate specific target frequency. For the fast lock requirement applications, code examples are provided by loading the custom calibrated DCO settings into the registers directly which were stored in the built-in nonvolatile FRAM memory after the first software trim of DCO+FLL.

References

1. MSP430FR235x, MSP430FR215x Mixed-Signal Microcontrollers
2. MSP430FR231x Mixed-Signal Microcontrollers
3. MSP430FR413x Mixed-Signal Microcontrollers
4. MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers
5. MSP430FR4133 Device Erratasheet
6. MSP430FR2111 Device Erratasheet
7. MSP430FR4xx and MSP430FR2xx Family User's Guide
8. Using the DCO Library
9. Using the VLO Library
10. Controlling the DCO Frequency of the MSP430F11x
11. Programmable Frequency Locked Loop Using MSP430™ MCUs
12. MSP430FR235x, MSP430FR215x Code Examples
13. MSP430FR231x Code Examples
14. MSP430FR243x, MSP430FR253x, MSP430FR263x Code Examples
15. MSP430FR2355 LaunchPad™ Development Kit

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), 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, 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 (www.ti.com/legal/termsofsale.html) 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.

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