NXP Semiconductors
Application Note

Document Number: AN12080
Rev. 0, November, 2017

The Clock Monitor Unit (CMU)

Initializing the CMU on MPC56xx & MPC57xx Devices

by: NXP Semiconductors

Contents

1 Introduction

A part's clocks are critical to its safe operation. They need to be accurate and correctly programmed and if there is any variation in their frequency then the application should be notified. Because of this, most NXP safety critical parts contain several Clock Monitor Units (CMU).

The main CMU of each device serves three purposes:

- Measures the frequency of the internal oscillator (IRCOSC)
- Monitors the external oscillator (XOSC) clock
- Monitors a selected clock

2 Overview

Main features of a CMU:

- Frequency meter: measure the IRCOSC frequency using the XOSC as a reference.
- XOSC clock monitoring: monitor the existence of the XOSC by comparing it to the IRCOSC clock.
- Selected clock monitoring: monitor the frequency of a selected clock against a user programmed upper and lower frequency boundary using IRCOSC clock ÷ 4 as a reference.
- Event generation for various failures detected.

The majority of the MPC56xx and MPC57xx parts have more than one CMU unit. The device has a CMU_0 that is capable of measuring the IRCOSC, XOSC monitoring, and monitoring one of the other system clocks. The device then contains other CMU_*n* blocks that are capable of monitoring other important system clocks. See the following table for examples of the CMU use for the MPC5643L and MPC5744P devices:

Table 1. CMU Use

Device	Module	Monitored clocks
MPC5643L	CMU_0	• System clock • XOSC • IRCOSC
	CMU_1	• Motor control clock
	CMU_2	• FlexRay Clock

Table continues on the next page...

Table 1. CMU Use (continued)

Device	Module	Monitored clocks
MPC5744P	CMU0	• MOTC_CLK • XOSC • IRCOSC
	CMU1	• SYS_CLK
	CMU2	• PBRIDGE*n*_CLK
	CMU3	• ADC_CLK
	CMU4	• SENT_CLK

The user can use the CMU_0 to verify the exact frequency of the IRCOSC clock. Since this clock can be used as the reference for other system clocks or protocol clocks it could be critical to know the exact frequency to ensure proper protocol timing.

The CMU_0 can also be used to monitor the XOSC. This feature only checks that the XOSC > (IRCOSC clock) ÷ 2^{RCDIV} where the RCDIV is a user selectable value of 0, 1, 2, or 3 in the CMU's Control Status Register (CMU_CSR). This feature does not measure the frequency of the XOSC. Assuming an IRCOSC clock of 16 MHz, the CMU only checks that the XOSC is greater than either 16 MHz, 8 MHz, 4 MHz, or 2 MHz.

The user can closely monitor any of the selected system clocks that are hooked up to one of the part's many CMU's. With this feature the user can set an upper and lower limit for the CMU to compare the selected clock's frequency.

When a failure is detected in a CMU, by either the selected clock monitor or the XOSC monitor, the CMU can notify different modules in the system depending on the device's configuration. For some devices, this could be the Reset Generation Module (MC_RGM), the Mode Entry (MC_ME) module, and the Fault Control and Collection Unit (FCCU) module. The user has the option to program the system's reaction to a CMU fault through the modules that get notified. Some of the options available depending on what modules are available in the device are generating a reset, entering SAFE mode, or generating an interrupt.

In the Figure 1. CMU_0 Block Diagram on page 3 below, the clock selection signal, controlled by the clock selection (CKSEL1) bits of the CMU_CSR, allows the user to select from several different reference sources. In truth the only option that is ever implemented is the IRCOSC. In some devices, it is hooked up to one of the inputs making the other ones invalid and in others it is hooked up to all of them.

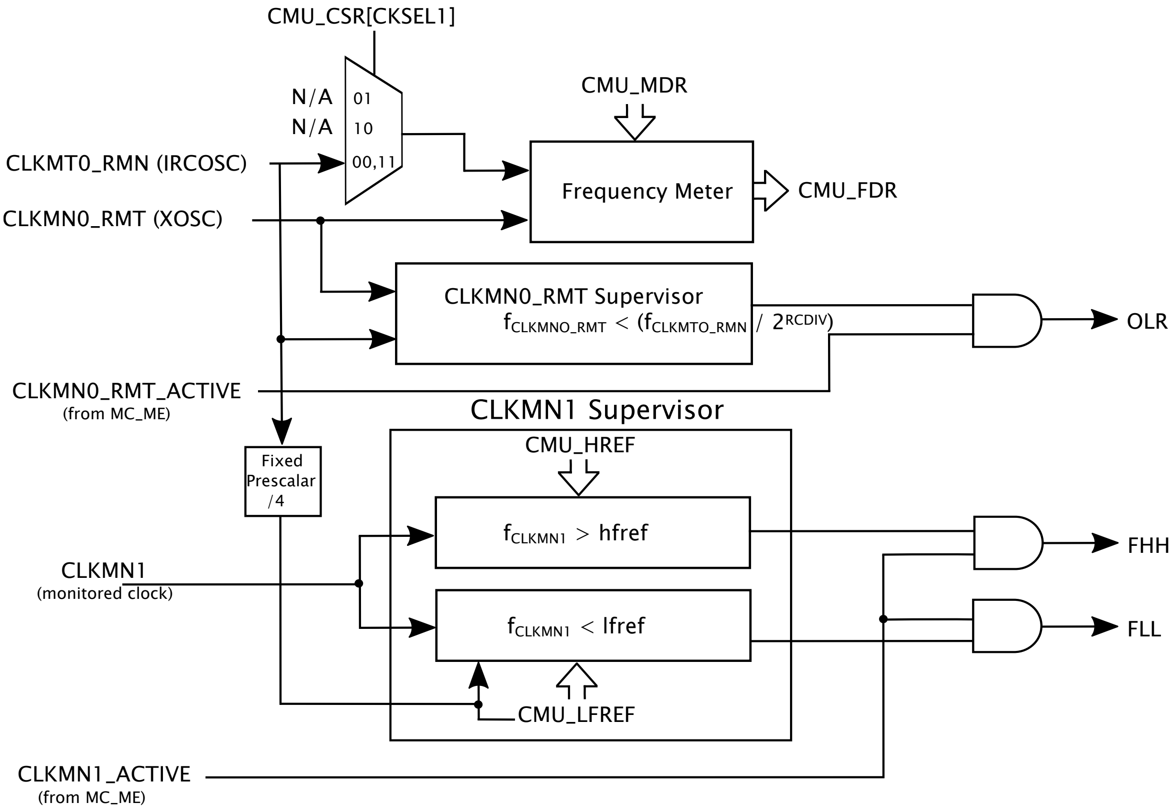

Figure 1. CMU_0 Block Diagram

In the Figure 2. CMU_n Block Diagram on page 3 below, the blocks for the frequency meter and the blocks for monitoring the XOSC are absent. The CMU_*n* modules do not support these functions.

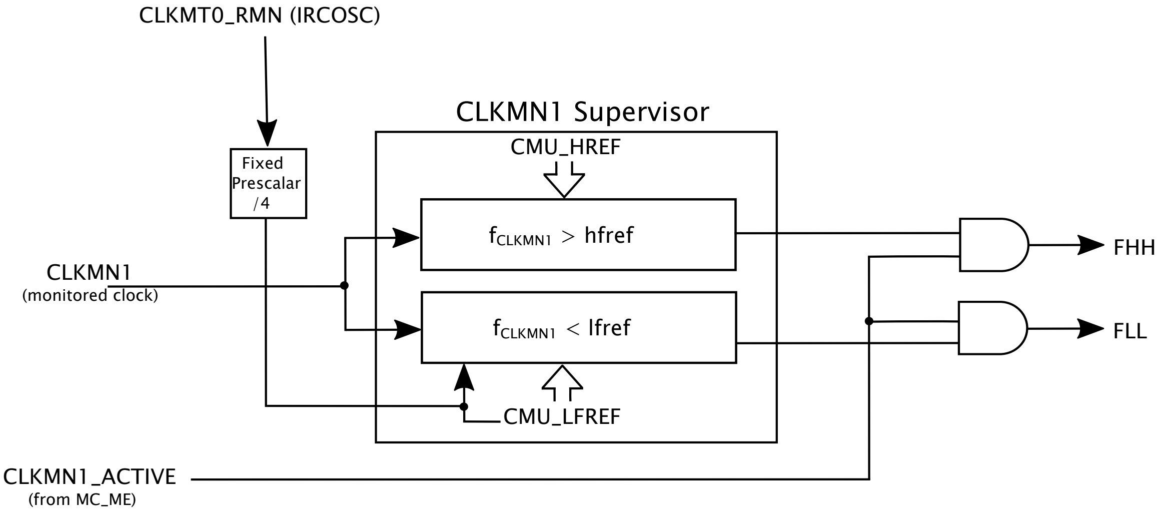

Figure 2. CMU_*n* Block Diagram

Different device manuals might refer to the CMU signals differently. MPC56xx manuals usually reference the IRCOSC and the XOSC signals directly while it is more common in the MPC57xx devices to use the more formal signal names such as CLKMN0_RMT and CLKMNMT0_RMN. The Table 2. Signal Description on page 4 table below describes which clock sources are connected to which CMU signals.

Table 2. Signal Description

Signal	I/O	Description
CLKMN0_RMT (Always XOSC)	I	Monitored Clock Signal 0/Metered Clock Signal Reference — Receives a clock signal that the CMU compares to a specified low-limit frequency to determine whether the frequency of the clock signal is greater than the specified limit. Also provides a reference clock signal for all metered clock signals.
CLKMN1	I	Monitored Clock Signal 1 — Receives a clock signal that the CMU compares to specified low limit and high limit frequencies to determine whether the frequency of the clock signal is between the specified limits.
CLKMT0_RMN (Always IRCOSC)	I	Metered Clock Signal 0/Monitored Clock Signal Reference — Receives a clock signal that the CMU measures against a reference clock frequency. Also provides a reference clock signal for all monitored clock signals.

3 Procedure

3.1 Measuring the IRCOSC, also called frequency metering

The frequency meter, which is part of CMU_0, measures the IRCOSC (CLKMT0_RMN) using a known frequency, the XOSC (CLKMN0_RMT). Knowing the exact frequency of the IRCOSC allows the user to make other timing corrections if necessary. The measurement starts when the start frequency measurement (SFM) bit of the Control Status Register (CMU_CSR) is set. The measurement duration is given by the measurement duration (MD) bits of the Measurement Duration Register (CMU_MDR) in terms of IRCOSC clock cycles with a width of 20 bits. The SFM bit is cleared by the hardware after the frequency measurement is done and the count is loaded in the measured frequency (FD) bits of the Frequency Display Register (CMU_FDR). The IRCOSC measurement be can derived as follows:

$$f_{\mathrm{sel}} = f_{\mathrm{CLKMN0\_RMT}} \times \frac{\mathrm{CMU\_MDR[MD]}}{\mathrm{CMU\_FDR[FD]}}$$

Steps:

1. Program the CMU_MDR[MD] value.
2. Program the CMU_CSR[CKSEL1] bit to the CLKMT0_RMN you wish to measure.
3. Enable the monitor by setting the CMU_CSR[SMF] bit.
4. When the CMU_CSR[SMF] bit is clear read the CMD_FDR[FD] value and calculate the IRCOSC frequency.

3.2 Monitoring the XOSC clock

The XOSC (CLKMN0_RMT) clock is automatically monitored by CMU_0. If the XOSC is determined on and stable, signaled by CLKMN0_RMT_ACTIVE signal, and the frequency of the XOSC meets the following condition:

$$f_{\mathrm{XOSC\_CLK}} < f_{\mathrm{IRCOSC\_CLK}} \div 2^{\mathrm{CSR[RCDIV]}}$$

then an oscillator less than (OLR) error event has occurred and the CMU_0 takes the following steps:

1. The oscillator less than interrupt bit (OLRI) gets set in the CMU Interrupt Status Register (CMU_ISR[OLRI]).
2. A failure event is signaled to the system which can react as programmed by the user. In many devices the MC_RGM, MC_ME, and FCCU are notified, which in turn can generate a ‘functional' reset, a SAFE mode request, or an interrupt.

NOTE

For the MPC56xx devices the XOSC monitor may produce a false event when $f_{XOSC\_CLK} < 2 \times f_{IRCOSC\_CLK} \div 2^{CSR[RCDIV]}$ due to an accuracy limitation of the compare circuitry.

NOTE

For the MPC57xx devices the f_{XOSC_CLK} (CLKMN0_RMT) must be greater than $f_{IRCOSC\_CLK} \div 2^{CSR[RCDIV]}$ (CLKMT0_RMN) by at least 0.5 MHz in order to guarantee correct XOSC (CLKMN0_RMT) monitoring.

3.3 Monitoring the system clock or any other CMU_*n* clock

This section applies to the device's main system clock, usually monitored by CMU_0, and all monitored clocks (CLKMN1) monitored by the other CMU_*n*s on the device.

The monitored clock (CLKMN1) frequency can be monitored by setting the Clock Monitor Enable (CME) bit in the CMU_CSR. The clock monitoring starts as soon as CMU_CSR[CME] = 1. This monitor can be disabled at any time by writing CMU_SCR[CME] = 0.

If the monitored clock is greater than the High Frequency Reference (HFREF) value programed into the High Frequency Reference Register (CMU_HFREFR) and the monitored clock is enabled, then:

- A frequency higher than high reference (FHH) event is detected and CMU_ISR[FHHI] is set
- A failure event is signaled to the system which can react as programmed by the user. In many devices the MC_RGM, MC_ME, and FCCU are notified, which in turn can generate a ‘functional' reset, a SAFE mode request, or an interrupt.

If the monitored clock is less than the Low Frequency Reference (LFREF) value programmed into the Low Frequency Reference Register (CMU_LFREFR) and the monitored clock is enabled, then:

- A frequency lower than low reference (FLL) event is detected and CMU_ISR[FLLI] is set
- A failure event is signaled to the system which can react as programmed by the user. In many devices the MC_RGM, MC_ME, and FCCU are notified, which in turn can generate a ‘functional' reset, a SAFE mode request, or an interrupt.

On the MPC56xx devices only, if the monitored clock is less than the reference clock frequency (FIRCOSC_CLK ÷ 4) and the monitored clock is enabled, then:

- a frequency lower than reference clock (FLC) event is detected and CMU_ISR[FLCI] is set
- A failure event is signaled to the system which can react as programmed by the user. In many devices the MC_RGM, MC_ME, and FCCU are notified, which in turn can generate a ‘functional' reset, a SAFE mode request, or an interrupt.

NOTE

For the MPC56xx devices the system clock monitor may produce a false event when the monitored clock is less than 2 x $f_{IRCOSC\_CLK} \div 2^{CSR[RCDIV]}$ due to an accuracy limitation of the compare circuitry.

The following equations are used to calculate HFREF and LFREF:

$$\mathrm{HFREF} = \frac{f_{CLKMN1}}{(f_{IRCOSC} \div 4)} \times 16$$

$$\mathrm{LFREF} = \frac{f_{CLKMN1}}{(f_{IRCOSC} \div 4)} \times 16$$

Steps:

1. Program the CMU_HFREFR[HFREF] and CMU_LFREFR[LFREF] values.
2. Enable the monitor by setting the CMU_CSR[CME] bit.

3.4 Requirements

A key point that always needs to be considered is that the user should check a device's data sheet for its IRCOSC variation. For any device, there are specifications for IRCOSC variation. It is necessary to take this into account when calculating the HFREF and LFREF values. Failure to adhere to the allowable variation detailed in a device's data sheet could result in false failures. The two tables below have excerpts from the MPC5643L and the MPC5744P data sheet.

Table 3. MPC5643L 16 MHz RC oscillator electrical characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Δ_{RCMVAR}	Fast internal RC oscillator variation over temperature and supply with respect to f_{RC} at T_A = 25 °C in high-frequency configuration	-	-6	-	6	%

Table 4. MPC5744P Internal RC Oscillator electrical specificaitons

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
δF_{var_noT}	IRC frequency variation without temperature compensation	T_J < 150 °C	-8	-	8	%
		T_J < 165 °C	-10	-	10	

An example of determining the $HFREF_{Actual}$ and $LFREF_{Actual}$ for the MPC5744P device is as follows. The f_{IRCOSC_CLK} = 16 MHz with an accuracy of +/- 8%. In order to monitor a f_{CLKMN1} = 200 MHz the ideal reference value is:

$HFREF_{Ideal}$ = (200 / (16 / 4)) * 16

$HFREF_{Ideal}$ = (200 / 4) * 16

$HFREF_{Ideal}$ = 50 * 16

$HFREF_{Ideal}$ = 800.

The $HFREF_{Ideal}$ and $LFREF_{Ideal}$ are the same. The $HFREF_{Actual}$ and $LFREF_{Actual}$ are:

$HFREF_{Actual}$ = $HFREF_{Ideal}$ + ($HFREF_{Ideal}$ * 0.08)

$HFREF_{Actual}$ = 800 + (800 * 0.08)

$HFREF_{Actual}$ = 800 + 64

$HFREF_{Actual}$ = 864

$LFREF_{Actual}$ = $LFREF_{Ideal}$ - ($LFREF_{Ideal}$ * 0.08)

$LFREF_{Actual}$ = 800 - (800 * 0.08)

$LFREF_{Actual}$ = 800 - 64

$LFREF_{Actual}$ = 736

4 Code Example

See the attached Green Hills project for MPC5744P example code.

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support