AN12620

Zero-Crossing Detection on Kinetis MCUs

Rev. 0 — November 2019

Application Note

by: NXP Semiconductors

Contents

1 Introduction

This application note describes advanced trigger options of the FlexTimer (FTM) and on-chip comparator peripherals and their interconnection. It explains how to setup and evaluate the FTM for Critical Conduction Mode (CCM) Power Factor Correction (PFC) applications and applications requiring advanced comparation of analog values. The information in this document can save you time and effort. This document is accompanied by a software package with examples. The FreeMASTER real-time debugger is used for application control.

1.1 Related documents

This document is related to these application notes:

- *FlexTimer and ADC Synchronization for Field-Oriented-Control* (document AN4410)
- *FlexTimer and ADC Synchronization* (document AN3731)
- *Features of the FlexTimer Module* (document AN5142)

1.2 Critical Conduction Mode (CCM) Power Factor Correction (PFC)

There are several control methods and topologies of PFC circuits. The basic control methods are the discontinuous mode, the continuous mode, and the CCM. These methods are named according to the current shape profile of the PFC. The CCM control is exceptional, because the boost inductor secondary winding acts as a zero-current detection source for the dedicated MCU peripherals. The second reason is the variable frequency, which is a result of the CCM realized according to the zero-crossing detection evaluation principle.

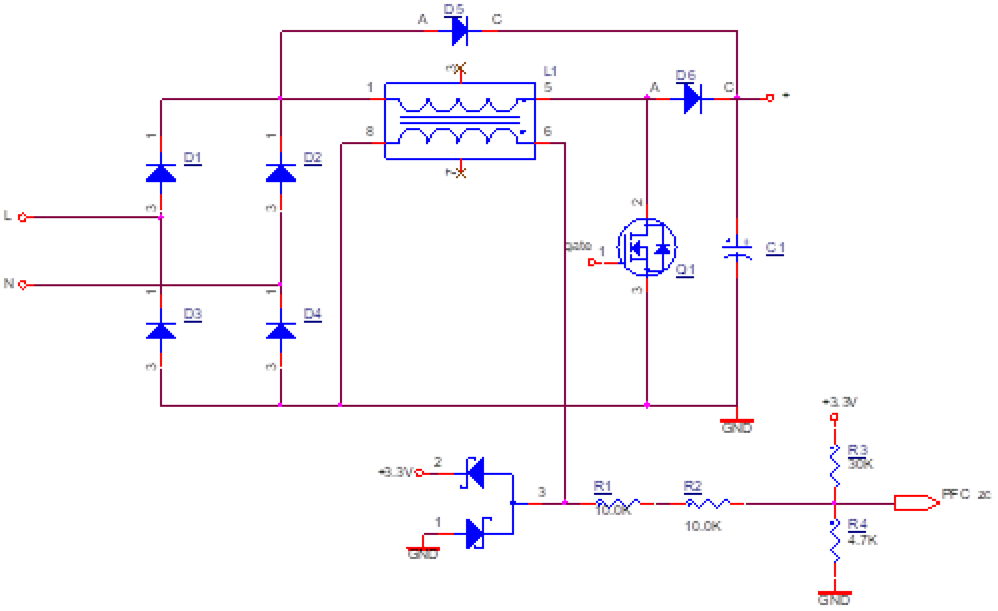

Figure 1. Basic PFC CCM topology

The CCM PFC design must consider the limitation of the varying frequency, especially when the PFC is unloaded or during startup when fake and unwanted zero-crossing pulses may occur.

2 FTM triggering options

This chapter describes advanced triggering possibilities of the FTM module. In common motor-control or power-conversion applications, the PWM FTM can be used as a trigger source for the ADC with or without the Programmable Delay Block (PDB). However, to start and re-init the FTM counter by external signals, sources, or events is a different task.

There are several ways to trigger the FTM counter:

- CMP event
- Fault input
- FTM
- PDB
- Software write

The trigger sources are described in the FTM chip configuration chapter in the reference manual and they are specific for each device or Kinetis series. The FTM hardware triggers are also extended in the System Options Registers (SOP) and in the dedicated FTM peripheral registers. Here is an example of hardware triggers from the reference manual on MKV31:

FTM0 hardware trigger 0 - SIM_SOPT8[FTM0SYNCBIT] or CMP0 Output or FTM1 Match

FTM0 hardware trigger 1 - PDB channel 1 Trigger Output or FTM2 Match

FTM0 hardware trigger 2 - FTM0_FLT0 pin

FTM1 hardware trigger 0 - SIM_SOPT8[FTM1SYNCBIT] or CMP0 Output

FTM1 hardware trigger 1 - CMP1 Output

FTM1 hardware trigger 2 - FTM1_FLT0 pin

FTM2 hardware trigger 0 - SIM_SOPT8[FTM2SYNCBIT] or CMP0 Output

FTM2 hardware trigger 2 - FTM2_FLT0 pin

FTM3 hardware trigger 0 - SIM_SOPT8[FTM3SYNCBIT] or FTM1 Match

FTM3 hardware trigger 1 - FTM2 Match

FTM3 hardware trigger 2 - FTM3_FLT0 pin

2.1 Comparator (CMP) module

The CMP module provides a circuit for comparing two analog input voltages. The CMP circuit is designed to operate in the full range of the supply voltage, which is called rail-to-rail operation. The analog MUX provides a circuit to select an analog input signal from eight channels. One signal is provided by a 6-bit Digital-to-Analog Converter (DAC). The MUX circuit is designed to operate in the full range of the supply voltage.

The CMP features:

- Operational in the entire supply range
- Programmable hysteresis control
- Selectable interrupt on the rising/falling edges or both edges of the CMP output
- Selectable inversion on the CMP output
- Window mode
- Sampled mode
- Digital filter

For more details, see the Comparator chapter in the reference manual.

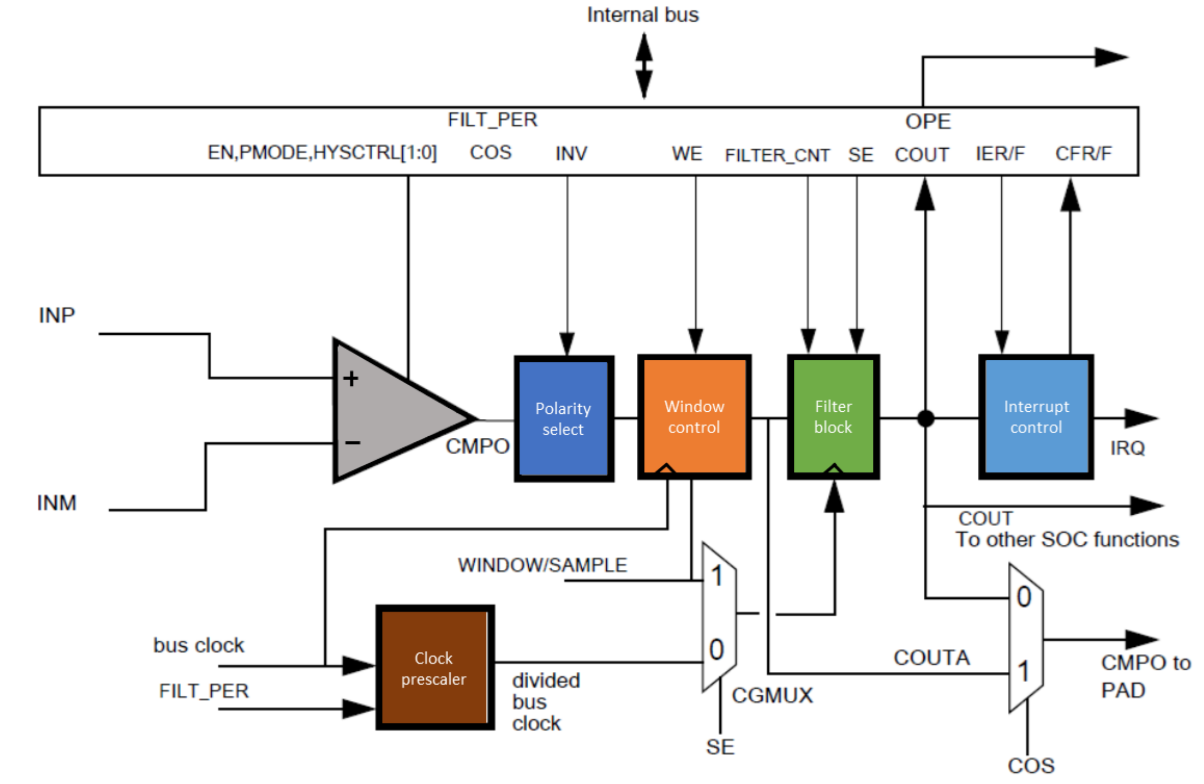

Figure 2. CMP module block diagram

2.2 FTM trigger from CMP

The analog CMP can be used to trigger the FTM counting. Each hardware or software trigger event that is enabled in the FTM registers causes a preset action (counter re-initialization by the CININT value in the described case). In the application example, you can change the CMP DAC value, hysteresis level, PWM duty cycle, and modulo. You can also enable or disable the trigger from CMP to FTM. The CMP output is displayed on the oscilloscope. For evaluation purposes, the RC filter filters the PWM and the feedback to the CMP to see a simple close-loop system (close to a simulation of the PFC coil current). The CMP can be set in a way that its output is on the MCU pin and you can observe and tune the compare event on the oscilloscope.

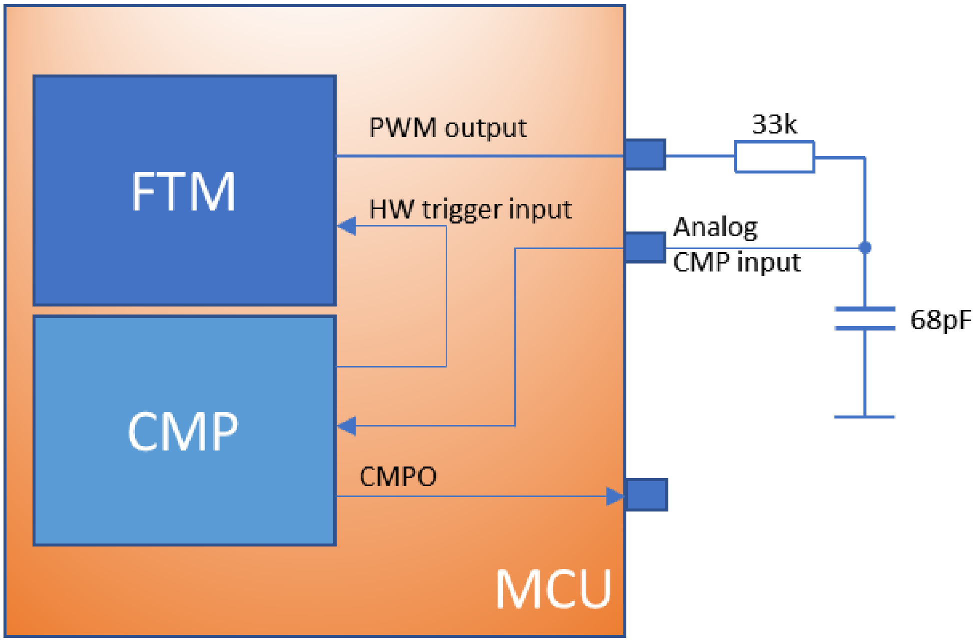

Figure 3. Zero-crossing evaluation circuit

In Figure 4, hardware triggering from CMP to FTM is disabled. The yellow trace is the CMP input voltage (the voltage on the capacitor of the evaluation circuit) and the blue trace is the CMP output. The DAC register value of the CMP module is 16. The FTM counts within its modulo value.

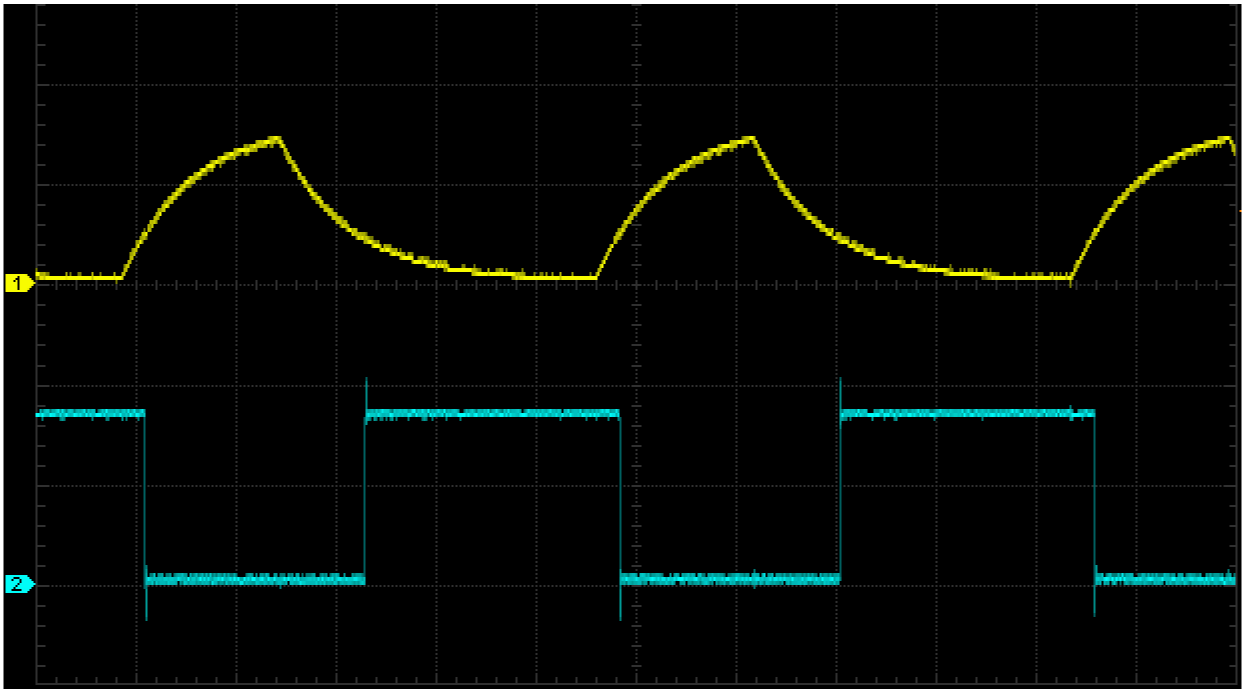

Figure 4. Zero-crossing evaluation–CMP DAC value=16, FTM hardware trigger disabled

In Figure 5, hardware triggering from CMP to FTM is enabled. The DAC register value of the CMP module is 16. The FTM can count within its modulo value, or it can be reinitialized from the CMP event before the modulo is reached.

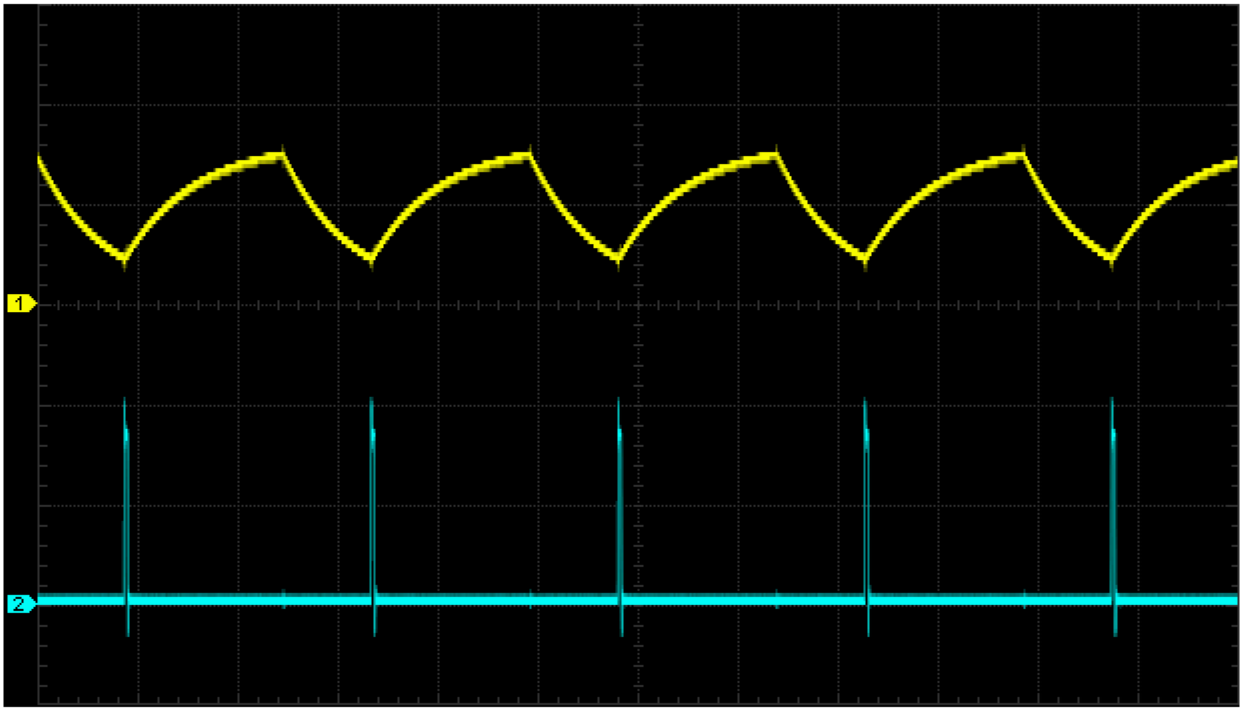

Figure 5. Zero-crossing evaluation–CMP DAC value=16, FTM hardware trigger enabled

In Figure 6, hardware triggering from CMP to FTM is enabled. The DAC register value of the CMP module is 0. The FTM can count within its modulo value or it can be reinitialized from the CMP event before the modulo is reached.

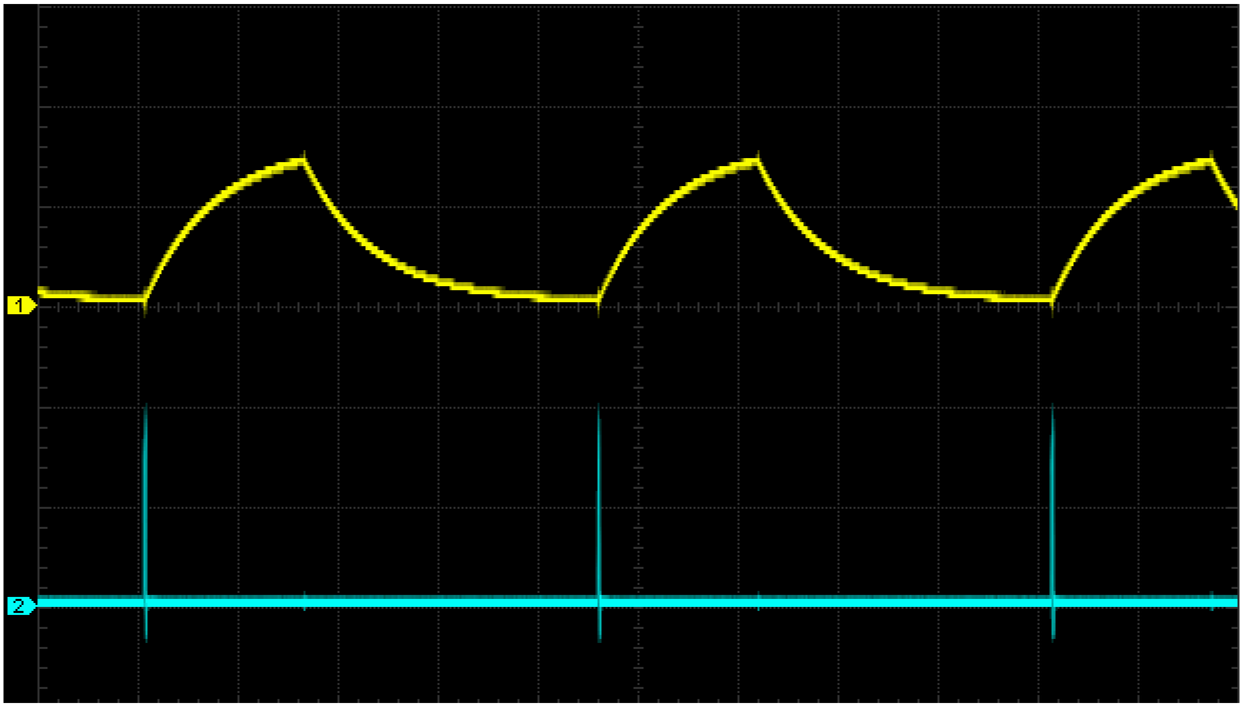

Figure 6. Zero-crossing evaluation–CMP DAC value=0, FTM hardware trigger enabled

2.3 CMP window option

There is a requirement to use the window for zero-crossing events and this feature can be enabled in the CMP module. This window is defined by the PDB module and its Pulse-Out Delay (POD) defined by the PODLY1 and PODLY2 registers. The Zero-Crossing Detection (ZCD) events are evaluated within this window. In most applications, the PDB module is used to trigger the ADC. For example, the main trigger is created by the PWM FTM and the PDB creates a small delay and pre-triggers for both ADC engines (if the device supports dual ADCs). The secondary function of the PDB that can be enabled is to define the window for the CMP unit. The compare event of the CMP unit can be used as an FTM hardware trigger. The compare event can be enabled to the pin for observation purposes. One drawback is that the window defined by the POD cannot be simply observed because it is not connected to the MCU pinout. One possibility is to set and define some of the free FTM PWM outputs to simulate the PDB POD. On some devices (such as KV1x), the PDB has a clock different from the FTM. On KV1x, the PDB has its clock divided by three, so the different timing values must be considered.

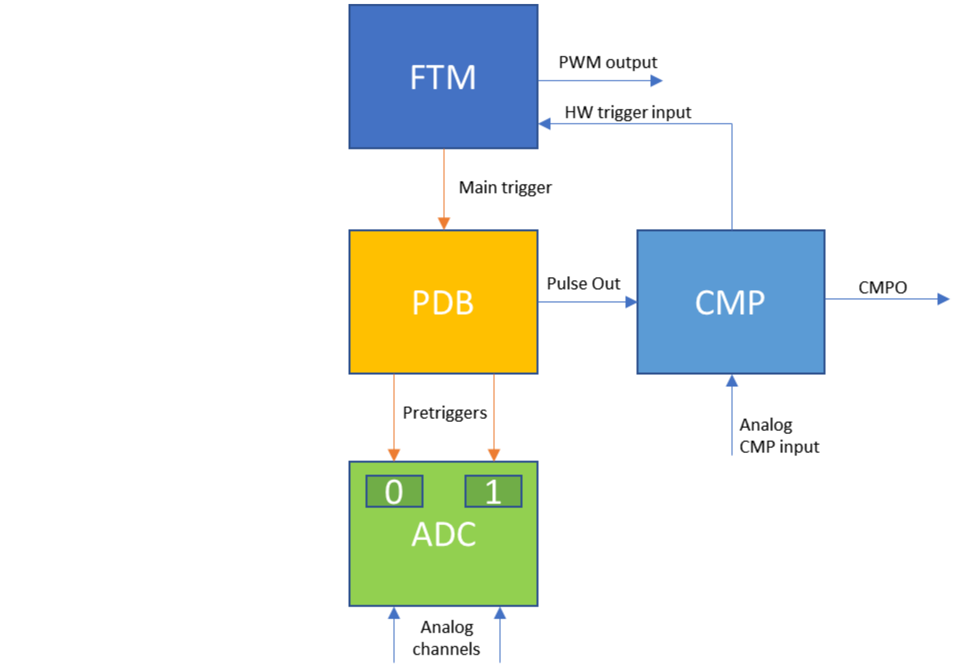

Figure 7. MCU modules interconnection

In Figure 8, the window is ON by PODLY1, because the CMP unit is connected to the FTM and the PDB is triggered from the FTM. When a compare event is detected, PODLY2 never happens when a zero-crossing is detected due to the retrigger of the FTM and PDB (which is the source of PODLY1 and PODLY2). If a zero-crossing is not detected, the FTM restarts by overflowing its modulo. If an early zero-crossing is detected, it is moved to the start of the window.

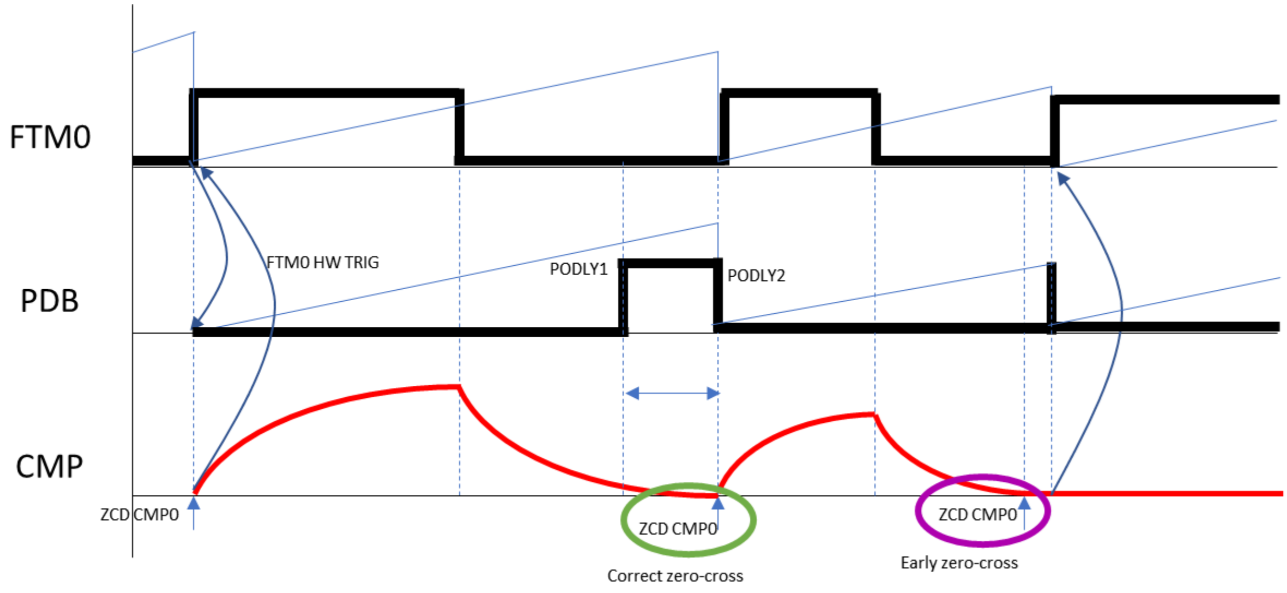

Figure 8. CMP window function evaluation

2.4 Selectable trigger window

For specific PFC CCM requirements, when the minimal and maximal off times are required to mask the unwanted zero-cross events, the PDB window is not usable. This is done by disabling the hardware trigger path from the CMP to the FTM by software and reenabling it in the required time in Figure 9 (Toffmin). Because the PFC frequency is usually higher than the motor PWM, the demand on frequent ISRs to secure the required functionality may load the MCU significantly. The interrupt function call latency may also cause issues. The DMA utilization can be considered here. A late ZCD is secured by the FTM module. All these features keep the CCM performance in the defined frequency, timing, and dynamic limits.

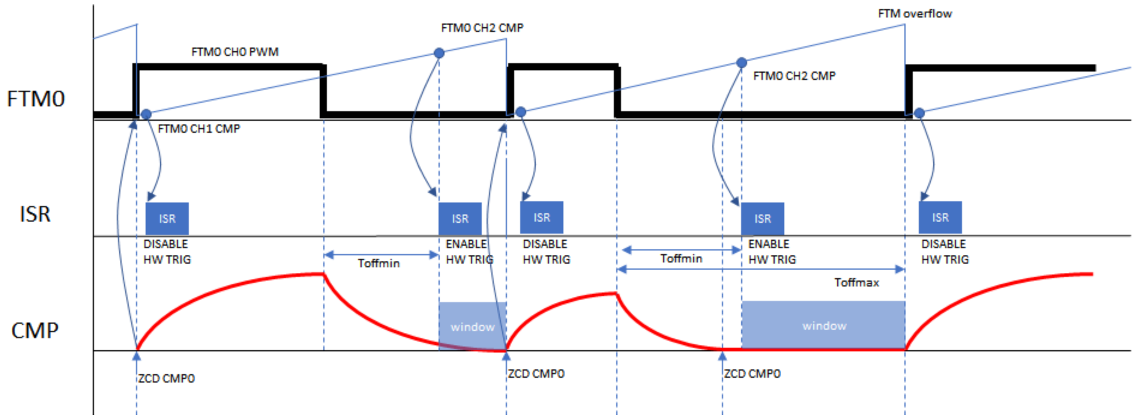

Figure 9. Selectable window with ISR

2.5 Selectable trigger window with DMA support

The DMA significantly offloads the MCU in case of frequent interrupts. However, this and the previous approach have a small disadvantage. When the required duty cycle is changed, the Toffmin and Toffmax compare values must be recalculated. The next chapter shows how to keep those events on their pre-defined values by the interconnection and implementation of another FTM module.

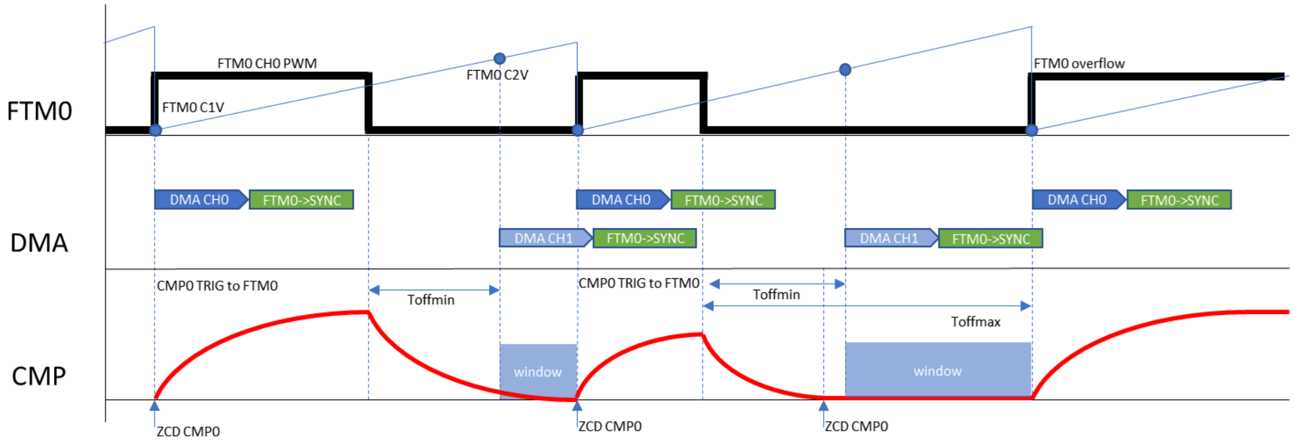

Figure 10. Selectable ZC window with DMA

2.6 Selectable trigger window with DMA and second FTM

Using another FTM to keep Toffmin and Toffmax independent from the main FTM duty cycle saves some computation time and the solution is based on the MCU peripherals, which is usually the best option. To enable the distribution of the FTM compare trigger (FTM match) to another FTM or peripherals (PDB…), select the required compare trigger in the FTMx_EXTTRIG register. The interconnection and the timing diagram are described in the following figures. This option is not available on all devices. See the reference manual *FTM Hardware Triggers* chapter and the SIM_SOPT8 register description. This configuration is possible on KV11Z (for example).

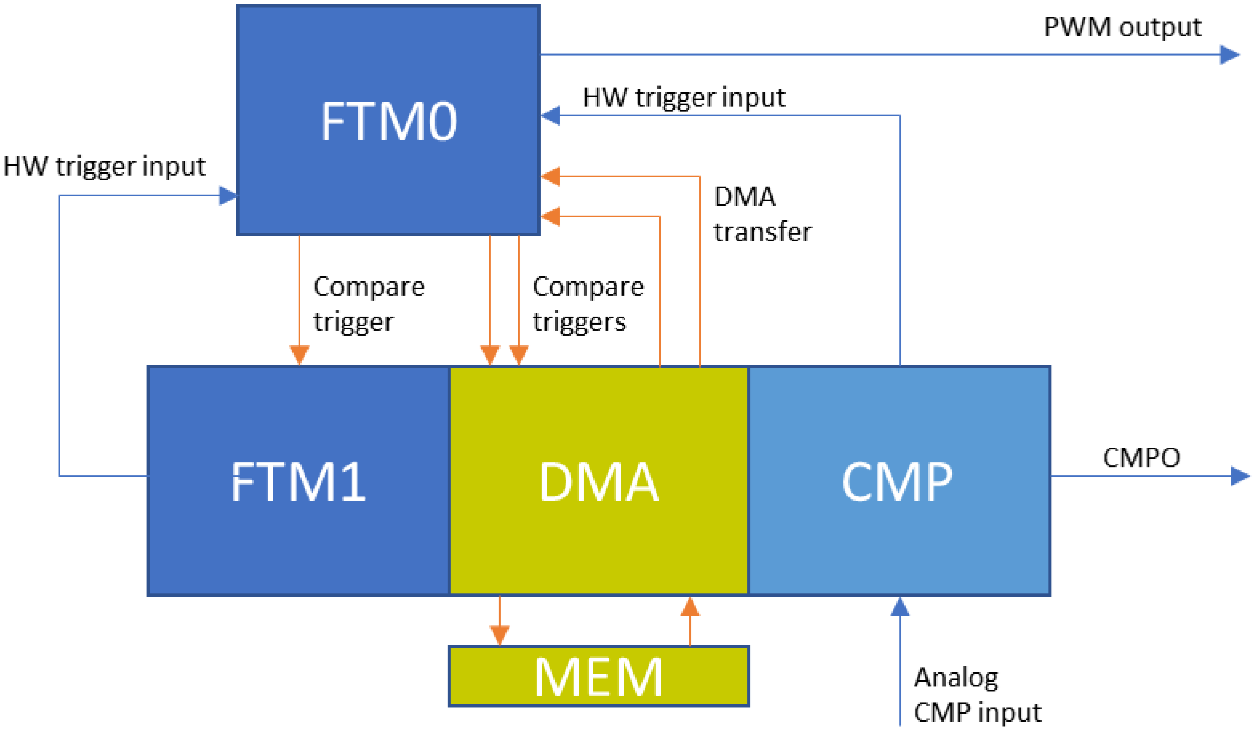

Figure 11. Modules' interconnection

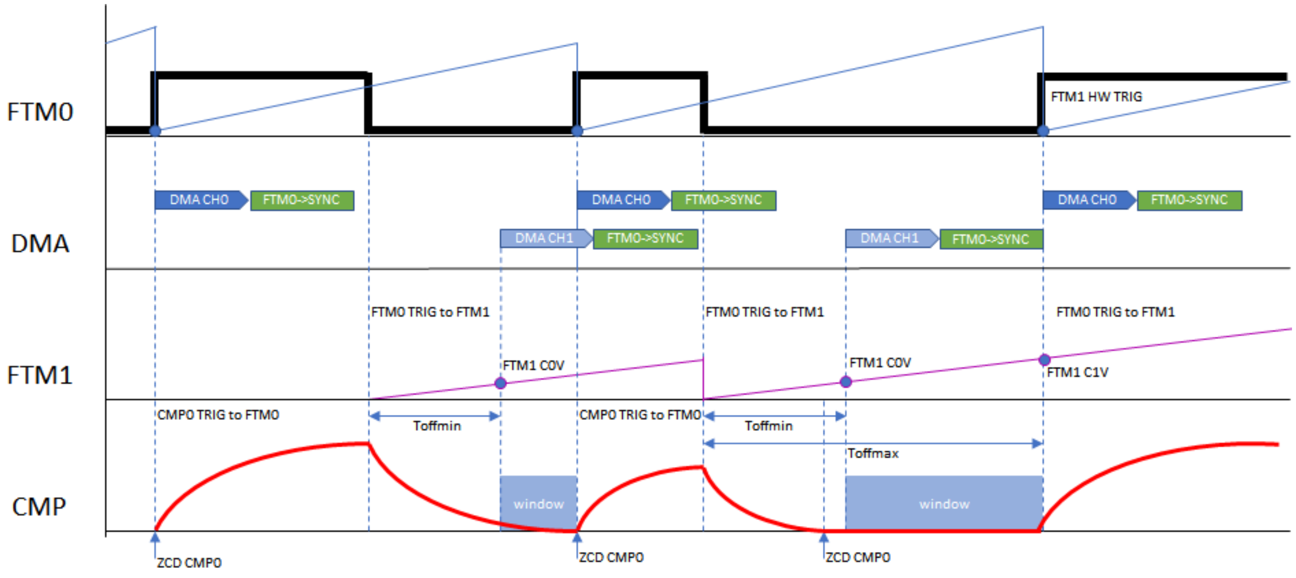

Figure 12. Selectable window-timing diagram

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: November 2019
Document identifier: AN12620