

POSIF

Position Interface

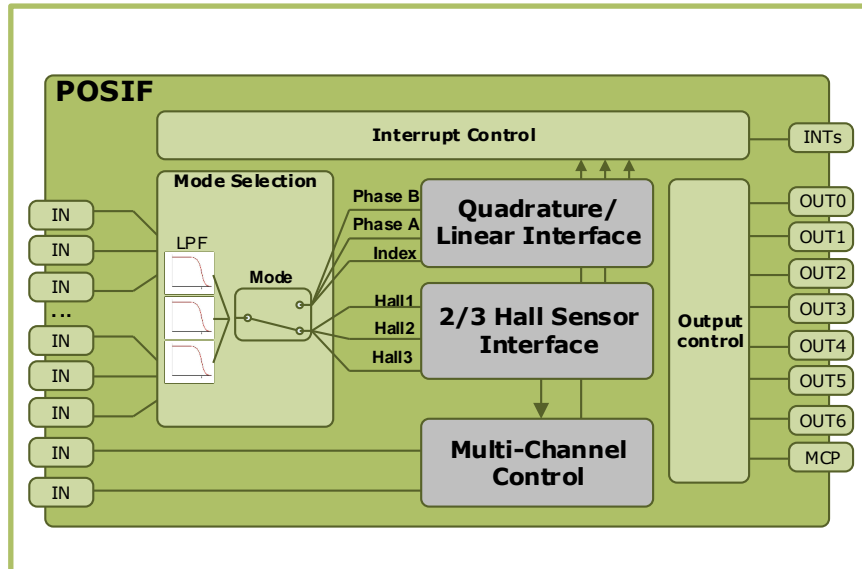
XMC™ Microcontrollers

July 2014



POSIF

Position Interface



Highlights

The POSIF module is the ideal solution for motor control applications using Hall Sensors and Quadrature Decoders. The user can configure freely the type and usage of the resources to perform an optimized mapping to the wanted application.

Key Feature

- > Interface for linear or quadrature rotary encoder
- > Interface for Hall Sensors
- > Stand-alone multi channel control

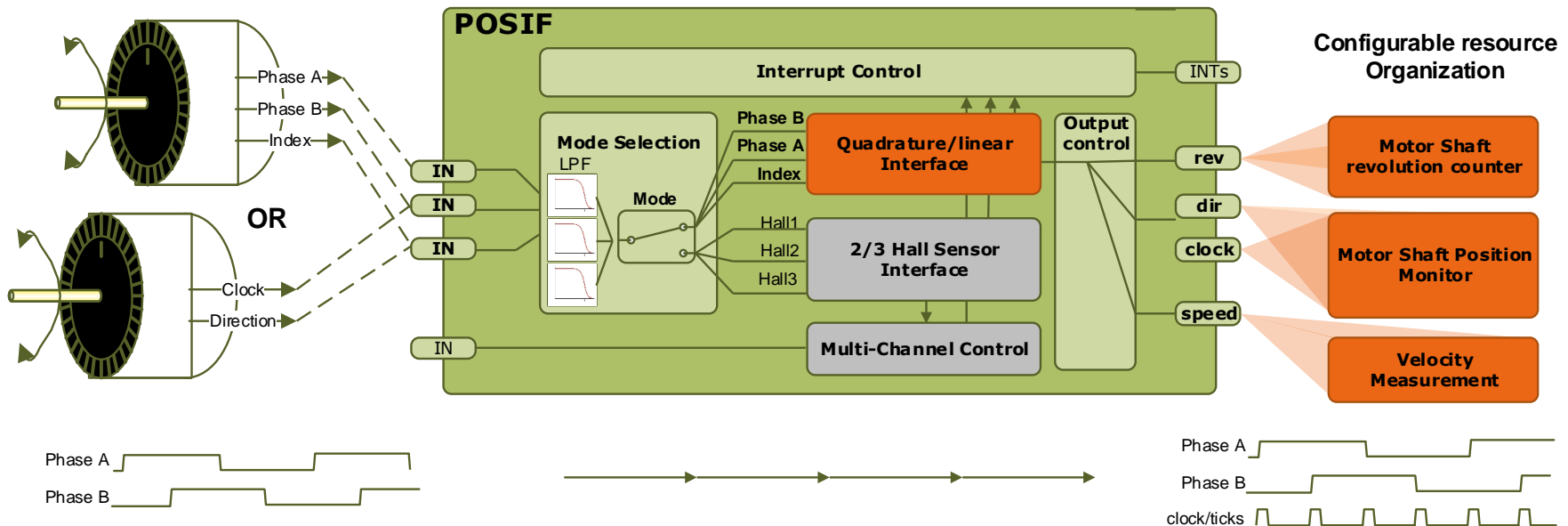
Customer Benefits

- > Application tailored motor position and velocity measurement.
- > Tailored solution for 2 or 3 Hall Sensor applications. Coupling with PWM generation.
- > Perform multi-level modulation for PWM.
- > Tailored modulation development

POSIF

Interface for linear or quadrature rotary encoder

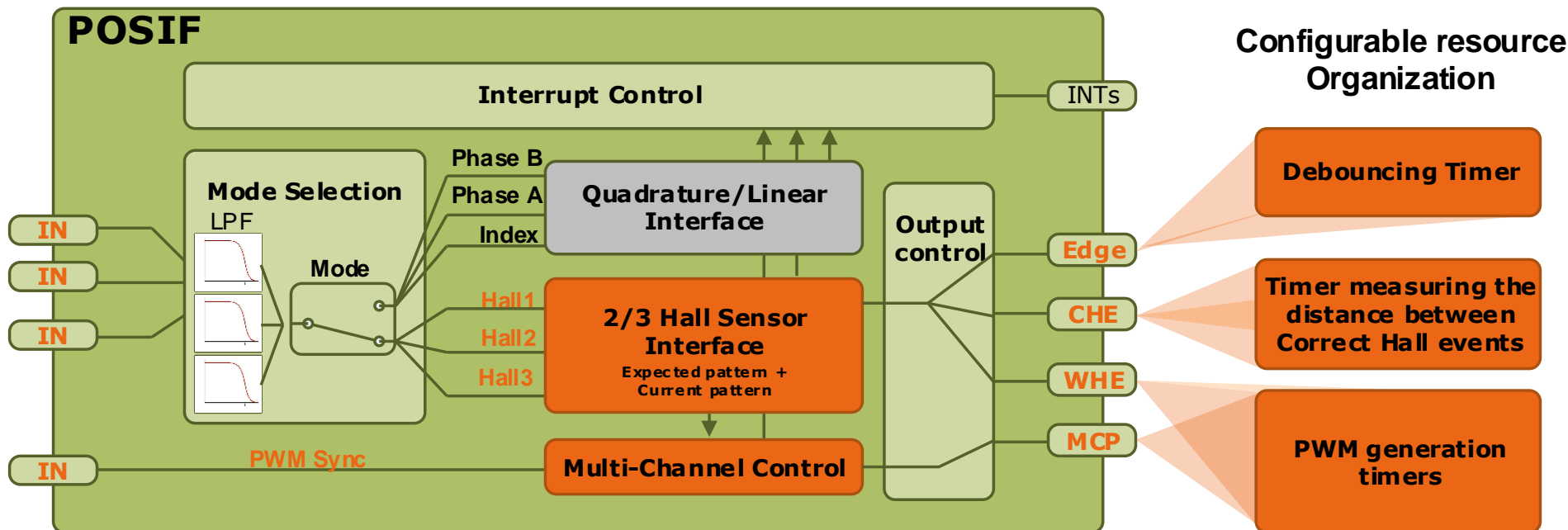
- › Linear or Quadrature Interface
- › Input signal filtering
- › Position monitoring (tick counting + direction)
- › Velocity Monitoring (time between ticks or/and elapsed time for a number of ticks)
- › Revolution Monitoring



POSIF

Interface for Hall Sensors

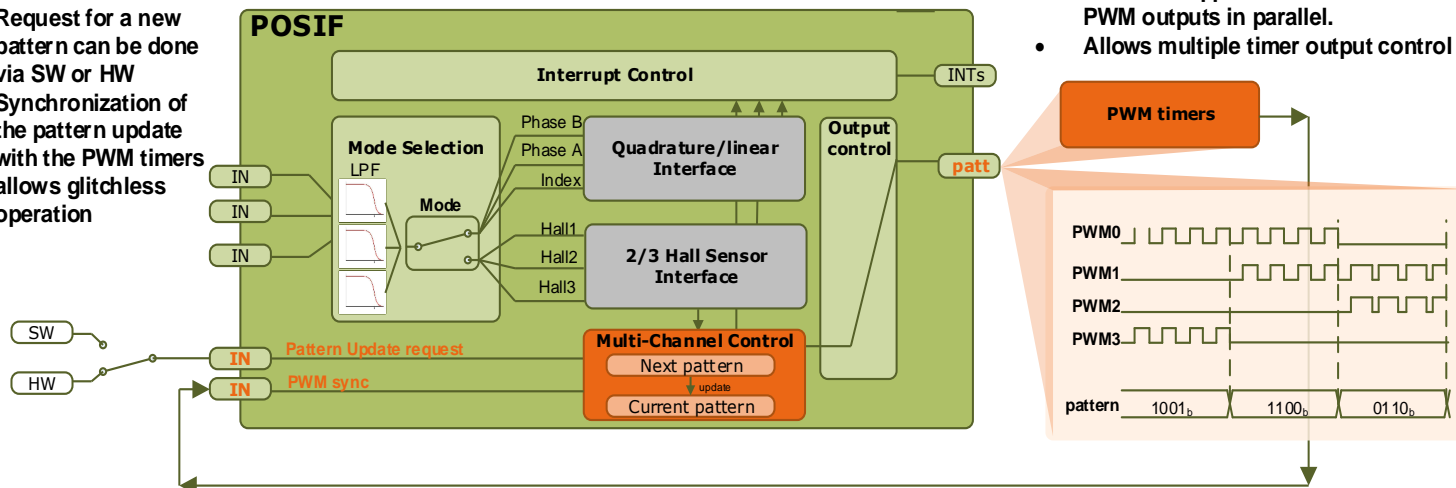
- › Current and Expected pattern can be easily updated via SW
- › Flexible Input signal debouncing/filtering
- › Time measurement between **C**orrect **H**all **E**vents
- › Programmable error handling (**W**rong **H**all **E**vent)
- › Synchronization with the PWM generation



Stand-alone multi channel control

- › Multi-Channel Control can work in stand-alone (without any of the other two modes)
- › With multi-channel control it is possible to control a pattern (max. 16 bits) that is controlling the PWM outputs (of CCU4/CCU8)
- › Pattern can be updated on-the-fly completely synchronous with the PWM timers
- › The pattern is completely controlled by SW, allowing any type of PWM output control

- Request for a new pattern can be done via SW or HW
- Synchronization of the pattern update with the PWM timers allows glitchless operation



- Pattern is applied to the Timers PWM outputs in parallel.
- Allows multiple timer output control

POSIF System Integration

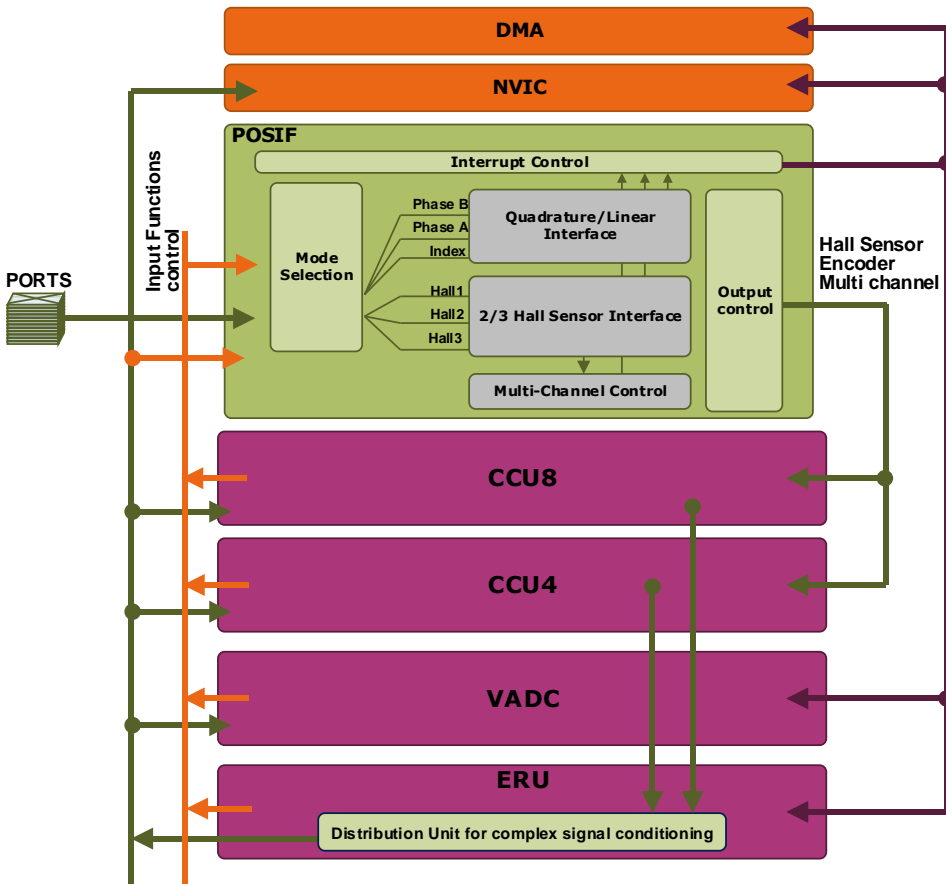
XMC4100	XMC4200	XMC4400	XMC4500	XMC4700
•	•	•	•	•

XMC1100	XMC1200	XMC1300
		•

Interrupts/
Service
Requests

The POSIF system integration offers several advantages:

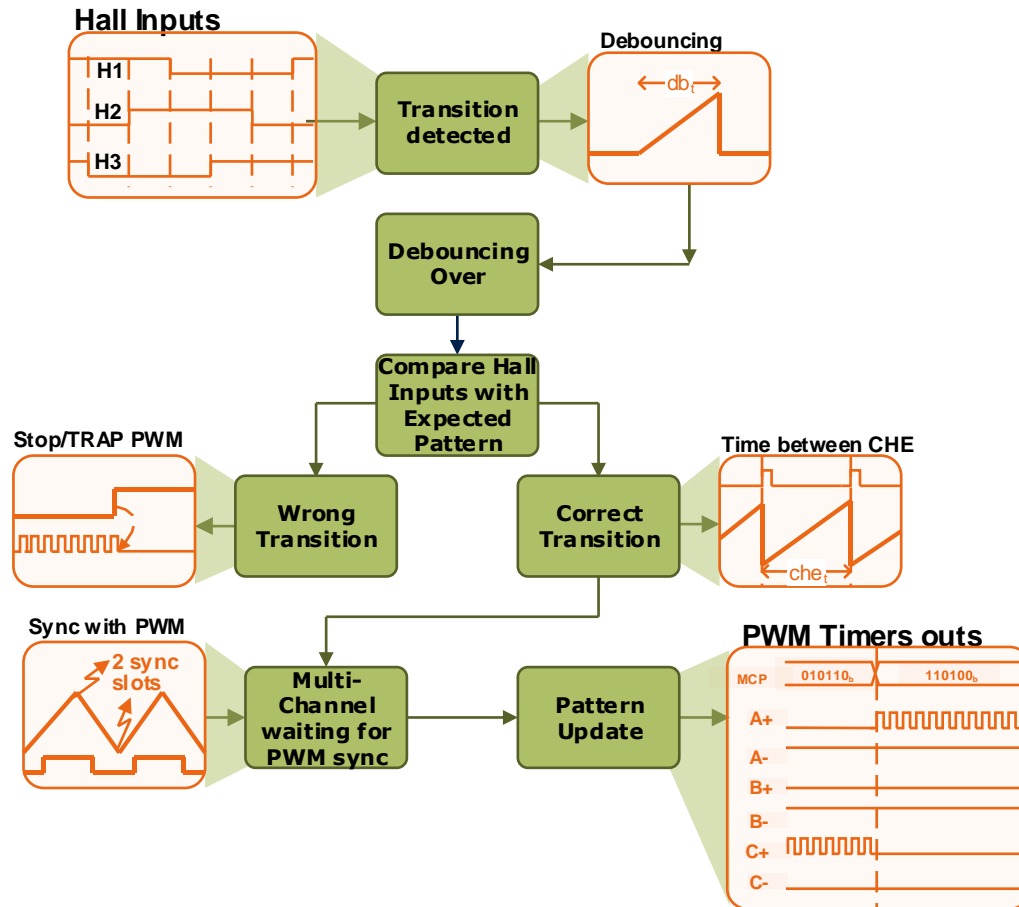
- › Distribution bus from CCU4/CCU8 over the ERU for complex signal conditioning application cases
- › Distribution of control logic to all of the Timer units (CCU4/CCU8)
- › Connectivity to the ADC to perform signal monitoring (or emulation) for motor control
- › Target applications
 - Motor Control
 - Power Conversion
 - Human Machine Interface
 - Connectivity
 - General Purpose



**Several components may be present or not depending on the device*

Application Example

Hall Sensor Control BLDC (1/2)



In Brief

Standard BLDC motor control

Overview

For a standard BLDC motor control application, **the POSIF is the monitoring the Hall Sensor outputs**. In every transition of the Hall Sensors a **debouncing** timer is started.

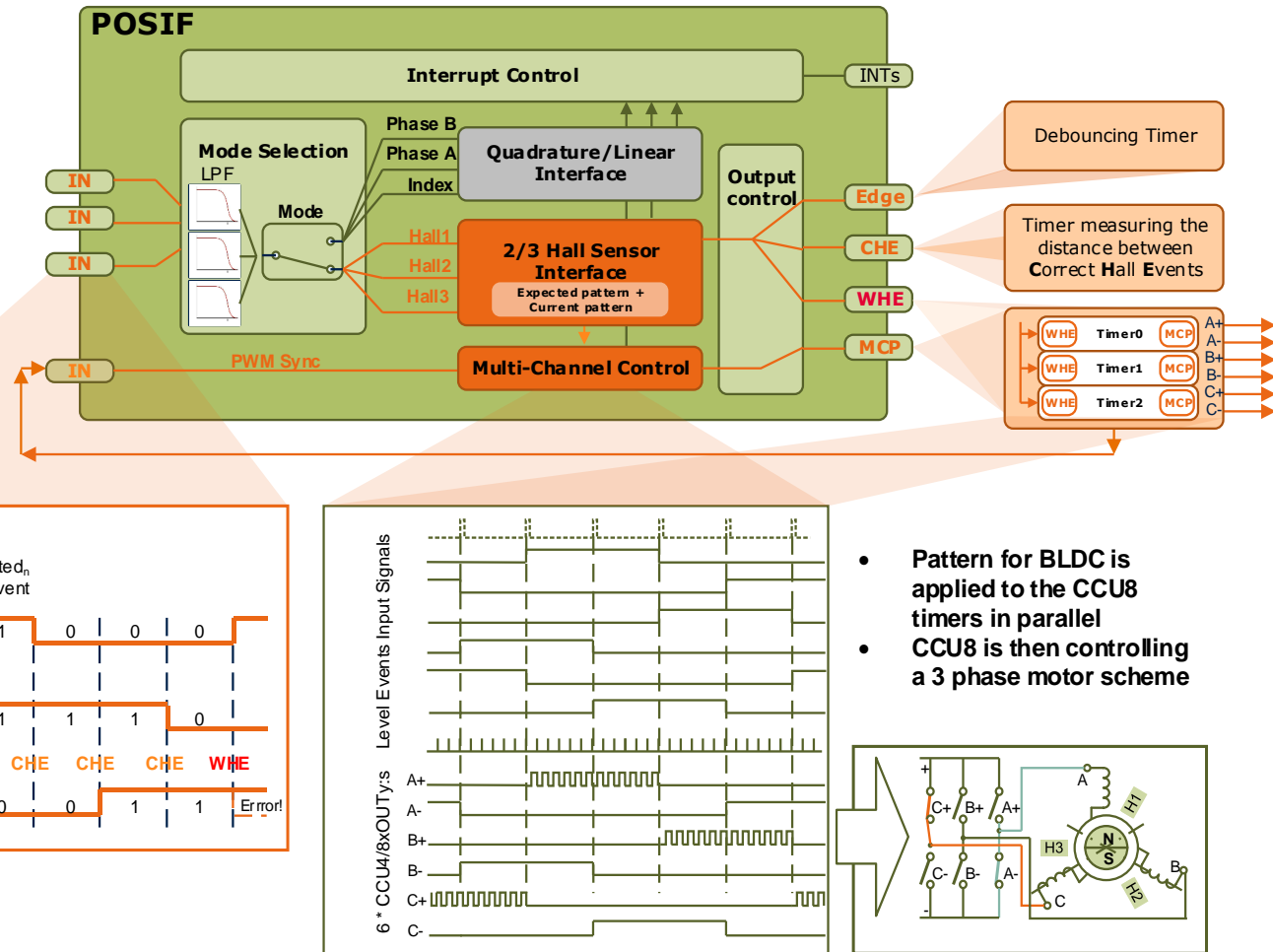
If after the debouncing time the transition is the expected one, than the Hall Sensor Control logic "informs" the multi-channel control that a pattern update is possible.

After this, **the multi-channel control waits for a sync signal from the PWM** timers to update the multi-channel pattern.

Application Example

Hall Sensor Control BLDC (2/2)

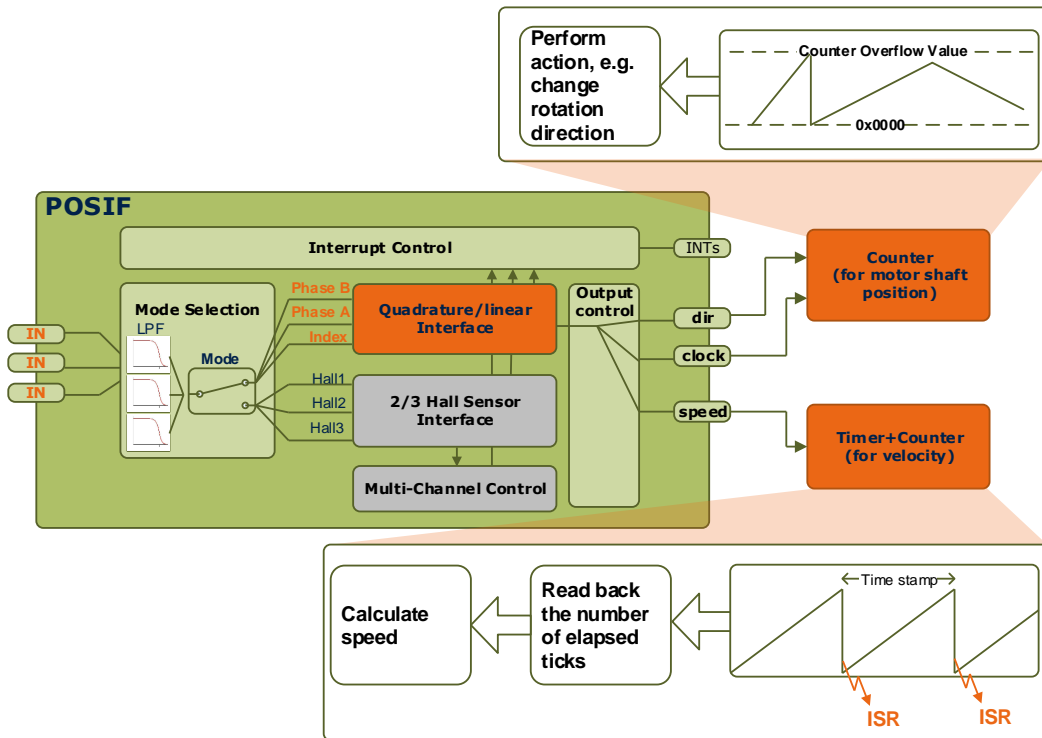
- > The Hall Sensor logic inside the POSIF is monitoring the input pattern against the expected pattern
- > A timer is used to debounce the input signals.
- > If a CHE occurs, then the PWM timers will signalize when the POSIF updates the Multi-Channel pattern
- > Wrong Hall Event can be used to generate a TRAP for the PWM timers



Application Example Hall Sensor Control: Detailed Block Diagram

Application Example

Quadrature Decoder – fixed time stamp (1/2)



Overview

The Quadrature Decoder Interface of the POSIF, together with the flexible set of Timers present in each derivative, can be used to:

- › implement a tailored solution to monitor the motor shaft position
- › implement a robust velocity measurement algorithm

In this application case, **one counter is monitoring the motor shaft position.**

The velocity is monitored by storing/capturing the elapsed number of ticks within a fixed time windows.

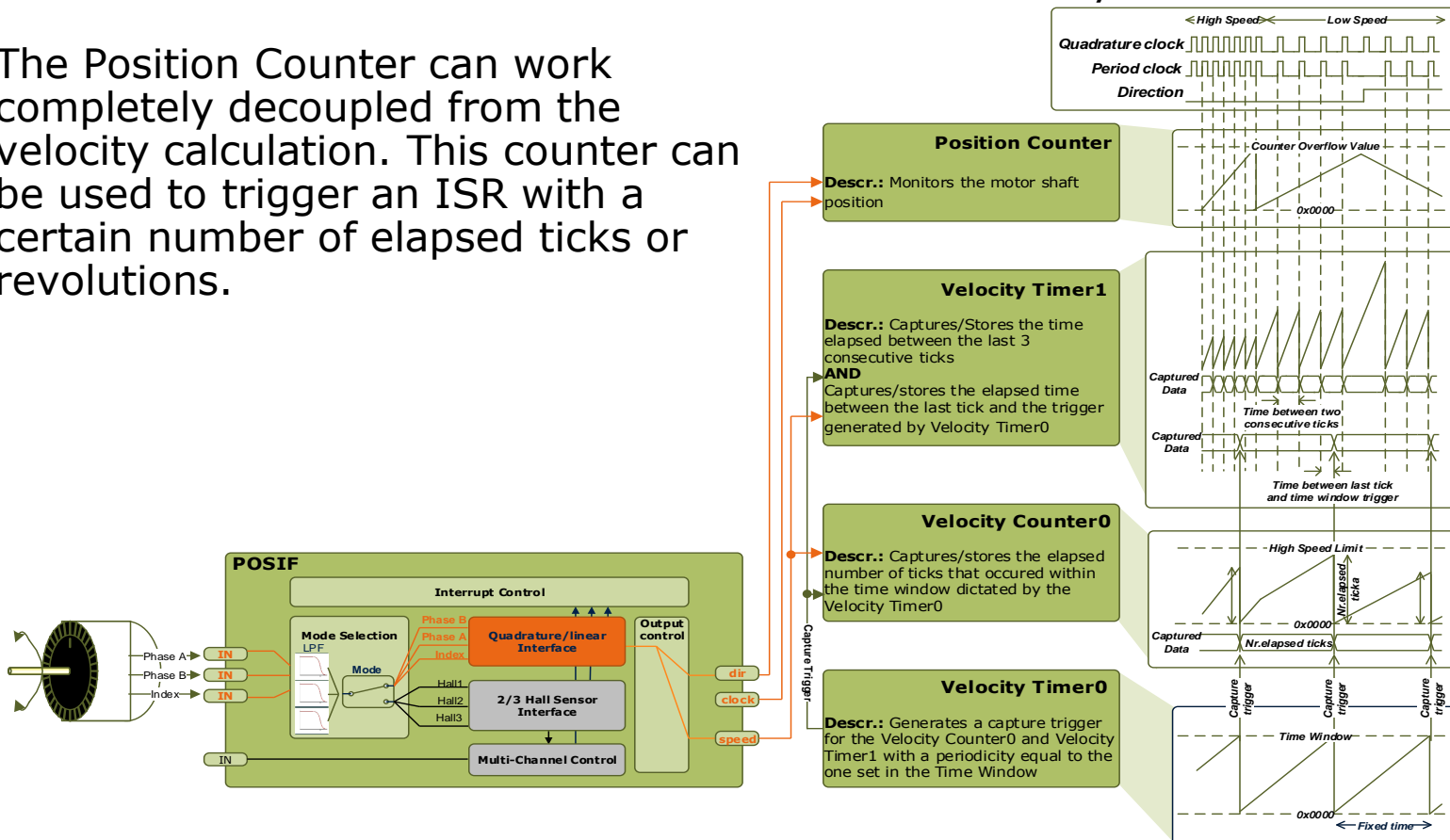
In Brief

Position monitoring plus velocity calculation within a fixed timestamp

Application Example

Quadrature Decoder – fixed time stamp (2/2)

- With this setup it is possible to configure the ISR timestamp for velocity calculation. Every time that an ISR is triggered by the "Velocity Timer0" the SW reads back the stored number of ticks from Velocity Counter0.
- The Position Counter can work completely decoupled from the velocity calculation. This counter can be used to trigger an ISR with a certain number of elapsed ticks or revolutions.



Application Example Quadrature Decoder fixed time stamp: Detailed Timing Diagram

Application Example

Quadrature Decoder – Enhanced Position and Velocity (1/2)

Overview

The flexible resource utilization between the POSIF and the CCU4/CCU8 modules, gives the possibility of having multiple timers/counters to monitor different variables.

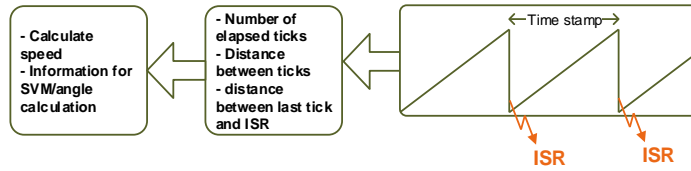
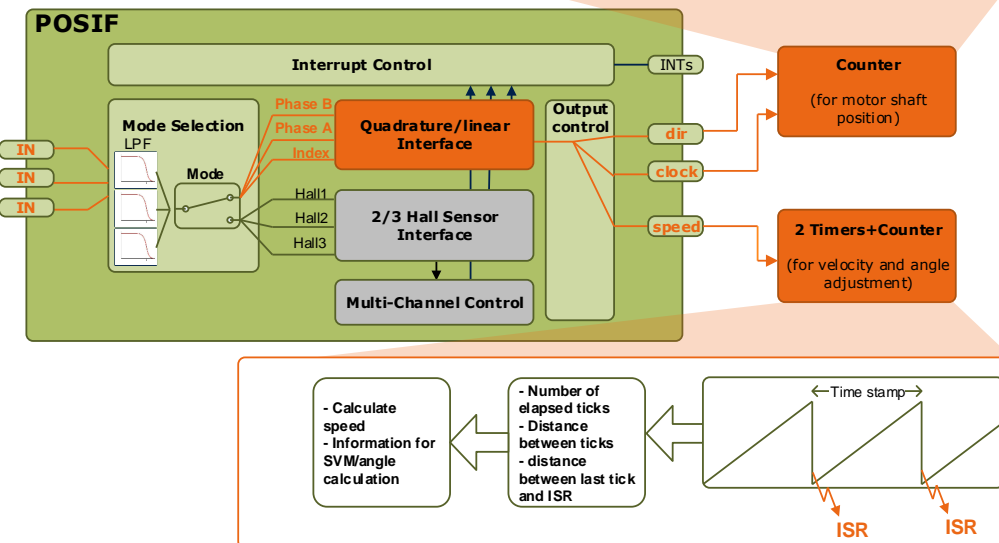
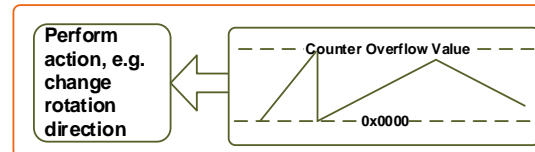
In this application case, a **position counter** is used to monitor the motor shaft position.

Besides that one additional counter and two timers are used to monitor:

- › the number of elapsed ticks between a fixed timestamp
- › the time distance between ticks
- › the jitter between the last tick and the ISR

In Brief

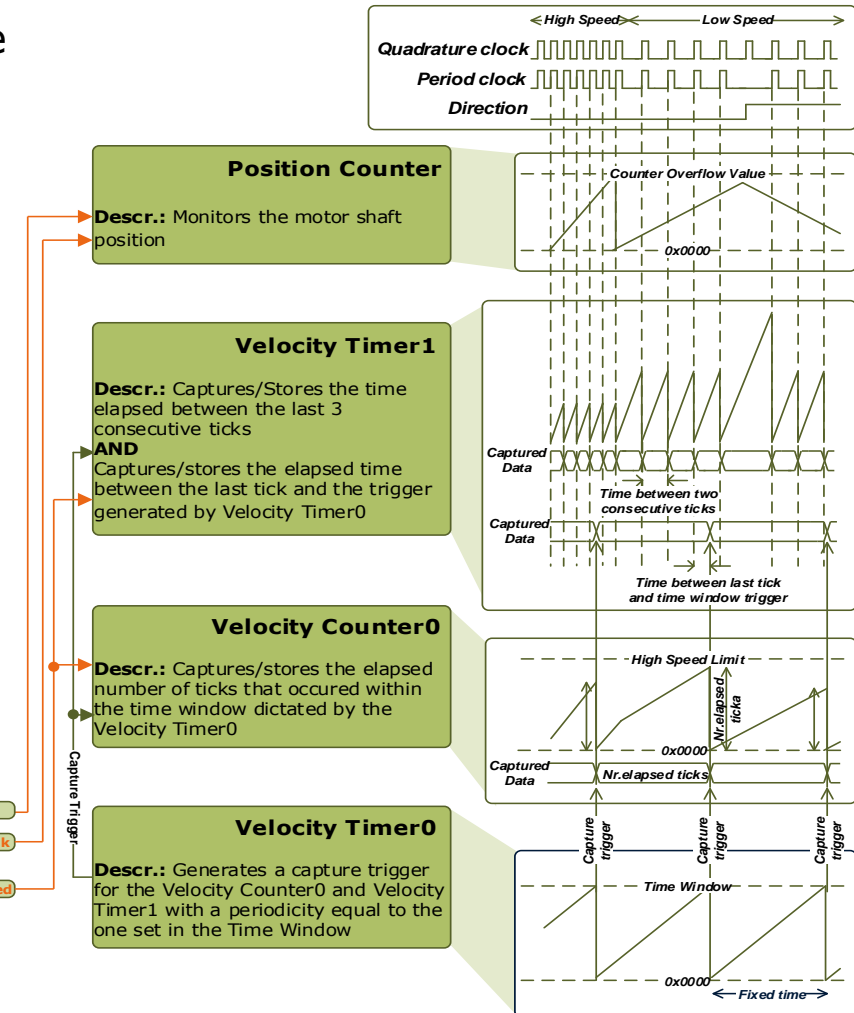
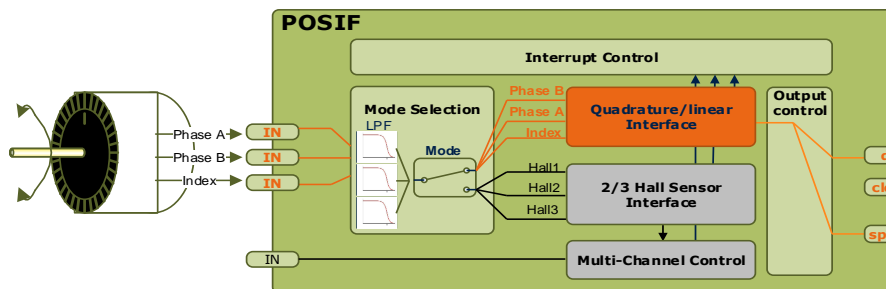
- › Enhanced position and velocity monitoring



Application Example

Quadrature Decoder – Enhanced Position and Velocity (2/2)

- With this setup it is possible to configure the ISR timestamp for velocity calculation. E.g. calculate the velocity every 2 ms.
- In each ISR** the SW has available the **number of elapsed ticks**, the time **distance between two consecutive ticks**, the distance between the **ISR and the last tick**.
- All these variables can be used not only to calculate the accurate velocity but also for angle calculation that can be feedback to the SVM calculation routine



Application Example Enhanced Position and Velocity: Block Diagram

Support material:

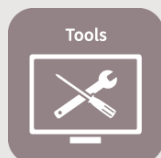
Collaterals and Brochures



- › Product Briefs
- › Selection Guides
- › Application Brochures
- › Presentations
- › Press Releases, Ads

› www.infineon.com/XMC

Technical Material



- › Application Notes
- › Technical Articles
- › Simulation Models
- › Datasheets, MCDS Files
- › PCB Design Data

- › www.infineon.com/XMC
- › [Kits and Boards](#)
- › [DAVE™](#)
- › [Software and Tool Ecosystem](#)

Videos



- › Technical Videos
- › Product Information Videos

- › [Infineon Media Center](#)
- › [XMC Mediathek](#)

Contact



- › Forums
- › Product Support

- › [Infineon Forums](#)
- › [Technical Assistance Center \(TAC\)](#)

Disclaimer

The information given in this training materials is given as a hint for the implementation of the Infineon Technologies component only and shall not be regarded as any description or warranty of a certain functionality, condition or quality of the Infineon Technologies component.

Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this training material.

All the images used in the trainings are free for commercial use or free for use with attribution and were designed by Freepik.



Part of your life. Part of tomorrow.

