XMC4000

XMC4500 Family

CCU4 and Counting Schemes

Counting Modes and Control

User Guide

Device Guide

<Revision 1.0>, <2012-30-03>

Microcontroller

Edition <2012-30-03>

Published by
Infineon Technologies AG
81726 Munich, Germany

© 2012 Infineon Technologies AG
All Rights Reserved.

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, 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.

Information

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

Warnings

Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.

Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

Confidential

Revision History	
Page or Item	**Subjects (major changes since previous revision)**
<Revision 1.0>, <2012-30-03>	

Trademarks of Infineon Technologies AG

AURIX™, C166™, CanPAK™, CIPOS™, CIPURSE™, EconoPACK™, CoolMOS™, CoolSET™, CORECONTROL™, CROSSAVE™, DAVE™, EasyPIM™, EconoBRIDGE™, EconoDUAL™, EconoPIM™, EiceDRIVER™, eupec™, FCOS™, HITFET™, HybridPACK™, I²RF™, ISOFACE™, IsoPACK™, MIPAQ™, ModSTACK™, my-d™, NovalithIC™, OptiMOS™, ORIGA™, PRIMARION™, PrimePACK™, PrimeSTACK™, PRO-SIL™, PROFET™, RASIC™, ReverSave™, SatRIC™, SIEGET™, SINDRION™, SIPMOS™, SmartLEWIS™, SOLID FLASH™, TEMPFET™, thinQ!™, TRENCHSTOP™, TriCore™.

Other Trademarks

Advance Design System™ (ADS) of Agilent Technologies, AMBA™, ARM™, MULTI-ICE™, KEIL™, PRIMECELL™, REALVIEW™, THUMB™, µVision™ of ARM Limited, UK. AUTOSAR™ is licensed by AUTOSAR development partnership. Bluetooth™ of Bluetooth SIG Inc. CAT-iq™ of DECT Forum. COLOSSUS™, FirstGPS™ of Trimble Navigation Ltd. EMV™ of EMVCo, LLC (Visa Holdings Inc.). EPCOS™ of Epcos AG. FLEXGO™ of Microsoft Corporation. FlexRay™ is licensed by FlexRay Consortium. HYPERTERMINAL™ of Hilgraeve Incorporated. IEC™ of Commission Electrotechnique Internationale. IrDA™ of Infrared Data Association Corporation. ISO™ of INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. MATLAB™ of MathWorks, Inc. MAXIM™ of Maxim Integrated Products, Inc. MICROTEC™, NUCLEUS™ of Mentor Graphics Corporation. Mifare™ of NXP. MIPI™ of MIPI Alliance, Inc. MIPS™ of MIPS Technologies, Inc., USA. muRata™ of MURATA MANUFACTURING CO., MICROWAVE OFFICE™ (MWO) of Applied Wave Research Inc., OmniVision™ of OmniVision Technologies, Inc. Openwave™ Openwave Systems Inc. RED HAT™ Red Hat, Inc. RFMD™ RF Micro Devices, Inc. SIRIUS™ of Sirius Satellite Radio Inc. SOLARIS™ of Sun Microsystems, Inc. SPANSION™ of Spansion LLC Ltd. Symbian™ of Symbian Software Limited. TAIYO YUDEN™ of Taiyo Yuden Co. TEAKLITE™ of CEVA, Inc. TEKTRONIX™ of Tektronix Inc. TOKO™ of TOKO KABUSHIKI KAISHA TA. UNIX™ of X/Open Company Limited. VERILOG™, PALLADIUM™ of Cadence Design Systems, Inc. VLYNQ™ of Texas Instruments Incorporated. VXWORKS™, WIND RIVER™ of WIND RIVER SYSTEMS, INC. ZETEX™ of Diodes Zetex Limited.

Last Trademarks Update 2011-02-24

Table of Contents

1 Counting Modes and Control 5
1.1 Basic Timer Functions 6
1.2 PWM Range 0 – 100% 6

2 Counting Schemes in Detail 8
2.1 Counting Rules 8

3 Reload Schemes in Detail 9
3.1 Reload by Shadow Transfer Rules 9
3.2 Shadow Transfers for Coherent Signal Pattern Update 10
3.2.1 The Global Shadow Transfer Set Enable Register 10
3.3 Shadow Transfer of Compare Register values 10
3.3.1 Compound Shadow Transfers 10

4 Start Counting Scheme 11
4.1 Global Start of CAPCOM Units 11
4.2 Global Start of the CAPCOM Units CCU4 and CCU8 11

5 Getting Started with CCU4 and Counting Schemes 12
5.1 Ultimate Initialization Sequence 12
5.1.1 Ultimate Start-Up Enable 12
5.2 Setting the Timer Counting Modes 13
5.2.1 Initialize and Start the Timer 13
5.2.2 Using the Timer Counting Modes Control Register 13

List of Figures

Figure 1 CAPCOM4 Basic Slices 5
Figure 2 CAPCOM4 Basic Counting Schemes 6
Figure 3 PWM Range in Edge Aligned or Center Aligned Mode 7
Figure 4 Edge and Center Aligned Counting Mode Schemes 8
Figure 5 Reload Schemes via Shadow Registers (Shadow Transfer Timing) 9
Figure 6 The Shadow Transfer Mechanism (Center Aligned Mode) 11
Figure 7 Synchronous Global Start of Timer Slices on different CAPCOM units (CCU4 or CCU8) 12

1 Counting Modes and Control

Each CAPCOM4 (CCU40/../43) is a multi-purpose timer unit for many use cases. It is designed with repetitive structures with multiple timer slices that have all the same base functionalities. This allows the development of re-usable SW low level drivers. The main function blocks of 1 of the 4 CC4y slices on a CCU4x are shown here. See Figure 1.

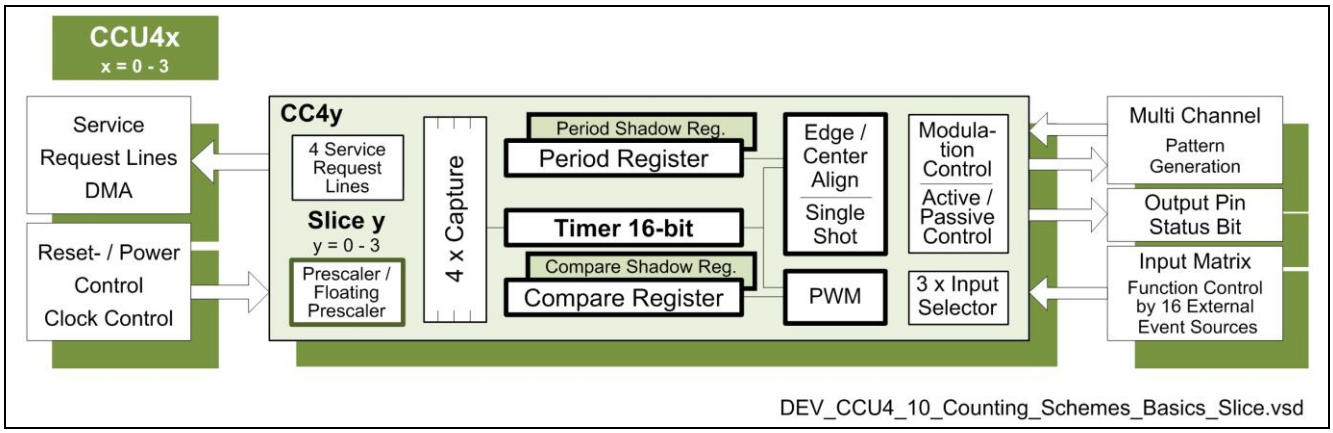

Figure 1 CAPCOM4 Basic Slices

1.1 Basic Timer Functions

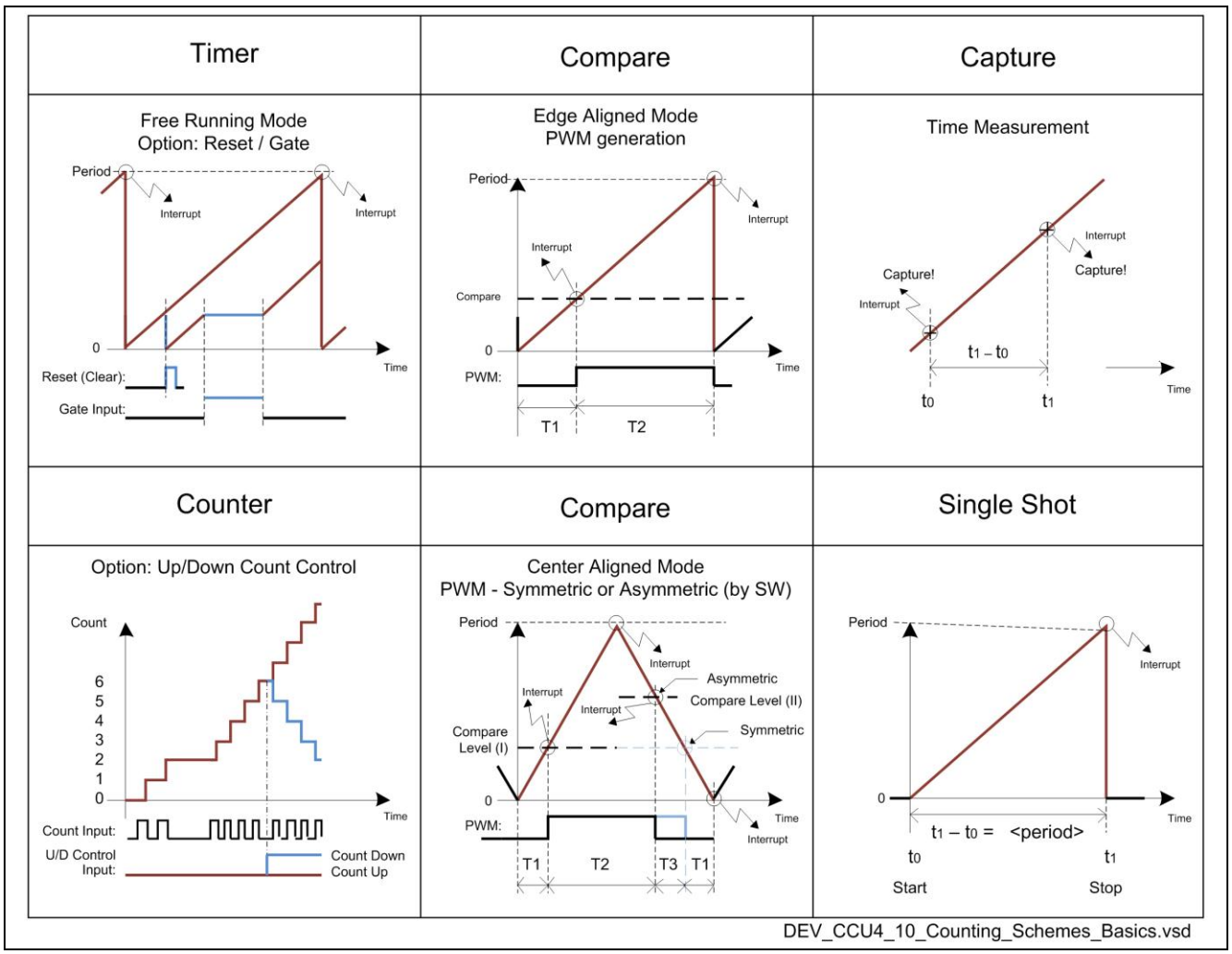

Figure 2 CAPCOM4 Basic Counting Schemes

1.2 PWM Range 0 – 100%

The Compare Rule in all Counting Modes is very simple: As long as the timer register value is equal or greater than the compare register value, the Status Bit (CCST or even named CCU4xSTy) will be set to one (=1). Otherwise it is set to zero (=0). The dynamic PWM range can be set to any value from 0 to 100%.

See Figure 3.

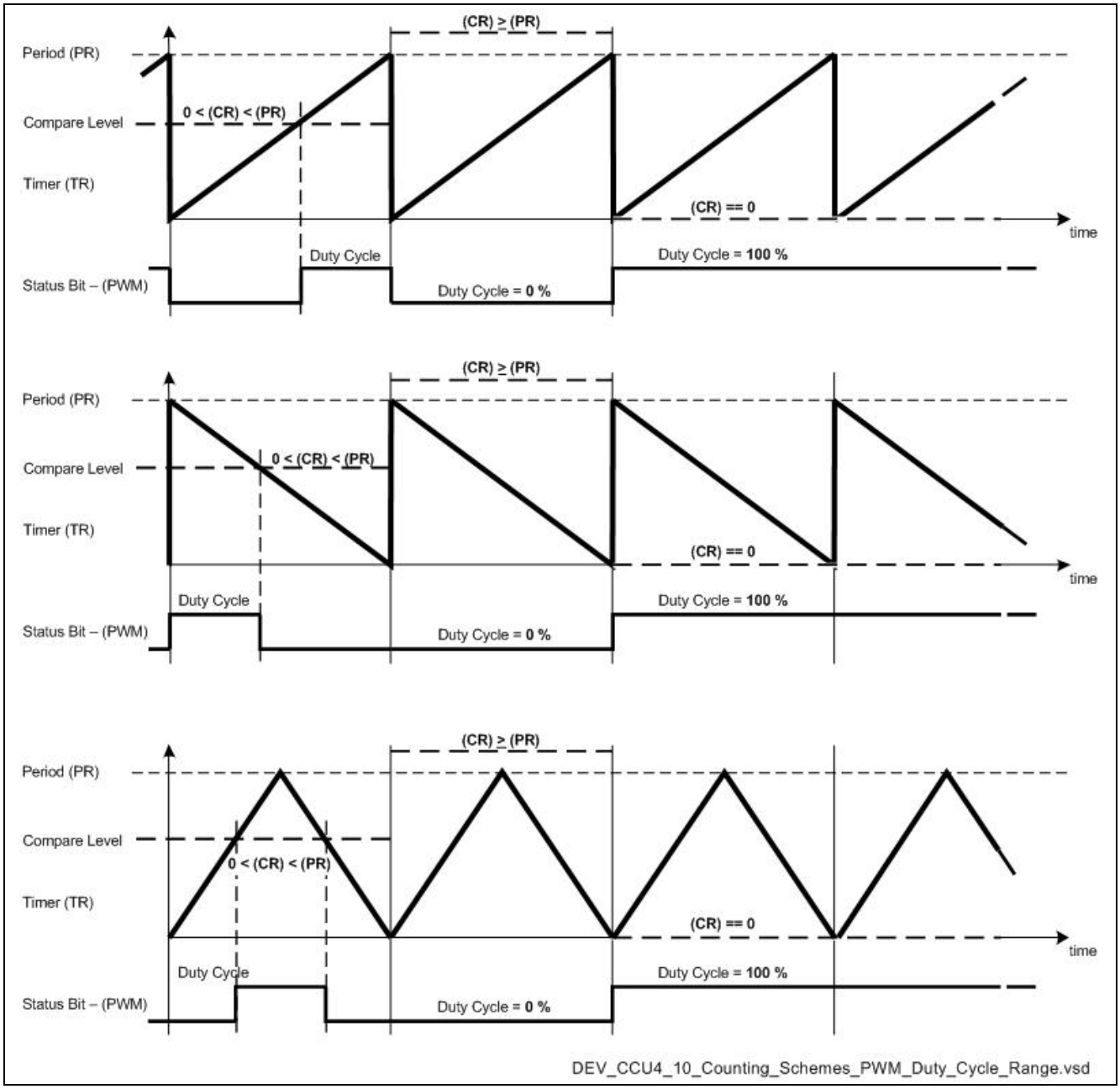

Figure 3 PWM Range in Edge Aligned or Center Aligned Mode

2 Counting Schemes in Detail

2.1 Counting Rules

See Figure 4.

One count cycle after the Counting Direction status bit **CC4yTCST**.CDIR has been changed, the timer counting direction will have been changed at the next count. In Center Aligned Mode this bit will be changed from 0 to 1, one cycle after a Period-Match state is entered – and from 1 to 0, one count cycle after a One-Match is entered.

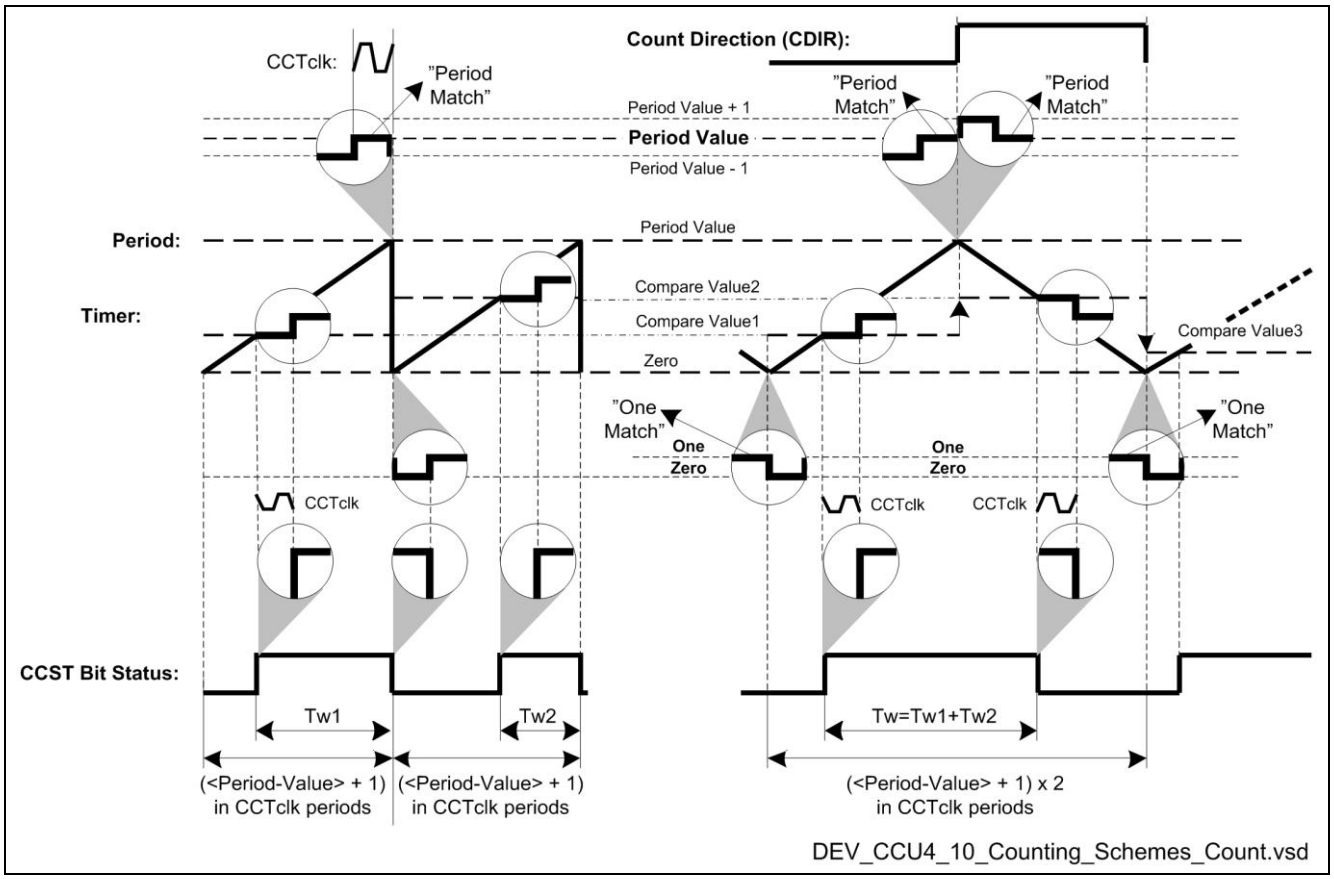

Figure 4 Edge and Center Aligned Counting Mode Schemes

3 Reload Schemes in Detail

3.1 Reload by Shadow Transfer Rules

See Figure 5.

Reload of registers by *Shadow Transfers* from associated shadow registers will occur according to the following:

- In the next clock cycle after a Period Match while counting up
- In the next clock cycle after an One Match while counting down
- Immediately if the timer is stopped and a transfer request was triggered (not illustrated here though)

Note: This implies that it is still due time to Set Transfer Enable requests as late as on a Period Match event, by using a Period Match Interrupt Service with a shorter latency time than the last clock cycle before the Shadow Transfer occurs. Normally the Compare Event Interrupts are used for this purpose. However, it shows that even a compare register value equal to the period register value is accurate for Transfer Enable request settings – in other words: The entire timer period may be used to Set Transfer Enable bits.

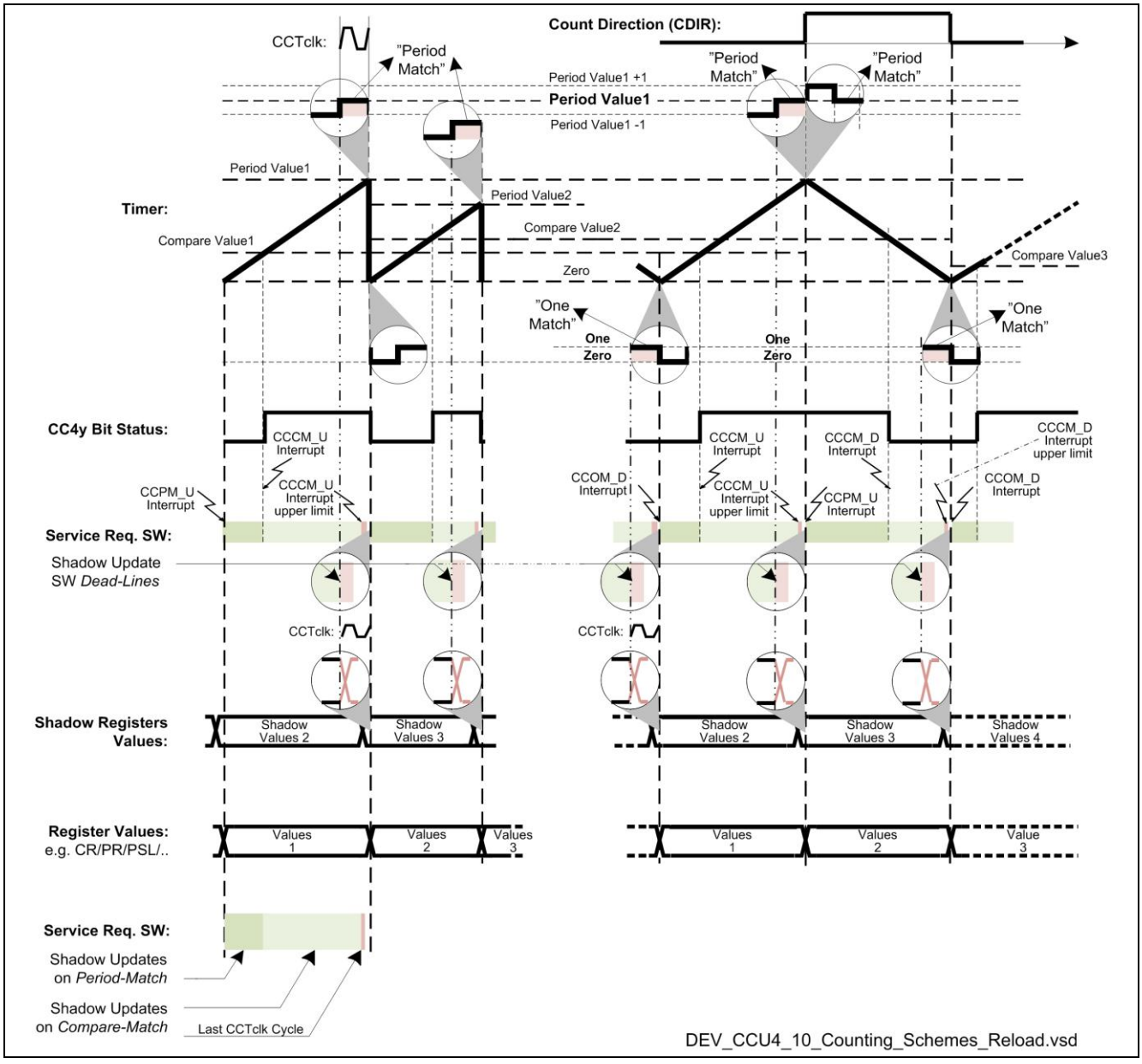

Figure 5 Reload Schemes via Shadow Registers (Shadow Transfer Timing)

3.2 Shadow Transfers for Coherent Signal Pattern Update

All CAPCOM4 timers, in any slice configuration, no matter how complex or for what signal pattern, are assured *coherent update* by HW of all concerned timer function parameters – by update values in shadow registers that, on *a global* preset request, are transferred simultaneously to all function registers at a **Period-** or **One-Match**.

3.2.1 The Global Shadow Transfer Set Enable Register

There is one, global register, **GCSS**, carrying *all* enable-flags that have to be preset by SW to selectively activate the targeted Shadow Transfer Requests, which will be cleared by HW after the transfer. By this manner total realtime correctness is achieved also from the logic operations aspect – essential for safe power switching.

3.3 Shadow Transfer of Compare Register values

The compare values that are targeted for an update operation have to be written into the **CC4yCRS** shadow registers - *AND the corresponding* **Slice Transfer Set Enable** bits, e.g. **SySE** in **GCSS**, *must be preset – latest within the clock cycle of* **Period Match** (in *Edge Aligned Mode*) or **Period/One Match** (in *Center Aligned Mode*)

See Figure 5.

3.3.1 Compound Shadow Transfers

Beside the *Compare* (CR) values, there are also the timer *Period* register (PR) and the PWM *Active/Passive control* bit (PSL) that are updated simultaneously on the **SySE** flag. *Dithering* or *Floating Prescaler* values (not shown here though) may get simultaneous update too by the **SyDSE** & **SyPSE** request flags.

See Figure 6.

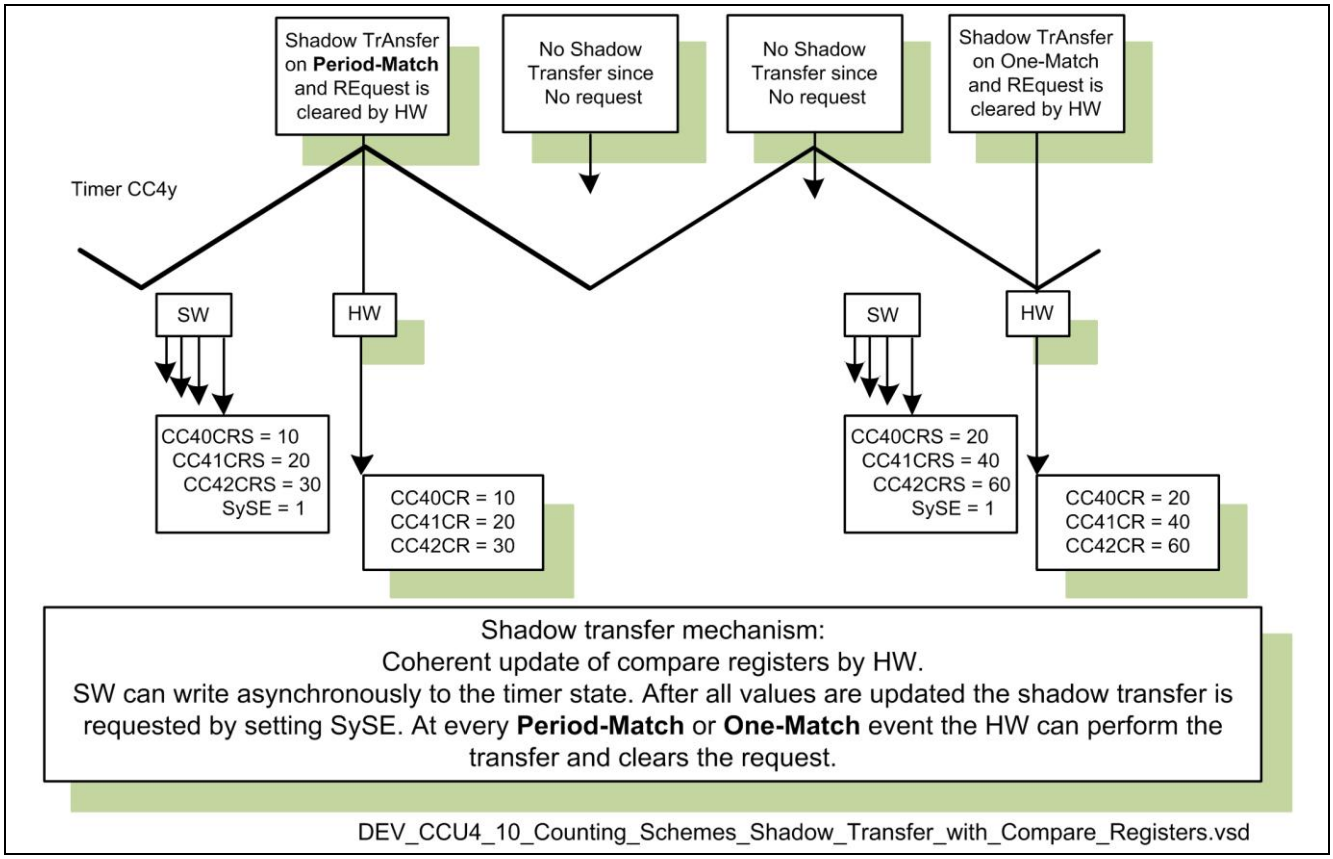

Figure 6 The Shadow Transfer Mechanism (Center Aligned Mode)

4 Start Counting Scheme

There are two ways to start a timer, either *directly* by SW setting the Timer Run Bit Set (**TRBS**) - or *indirectly* by HW when a considered event occurs in an external unit that may be selected for this, according to the **Top-Level Connection Matrix** for **External Events Control** – by *CAN, ADC, USIC, IO, CCU4/8, ERU1, POSIF* etc.

4.1 Global Start of CAPCOM Units

There is a way to get a synchronized start of CAPCOM Units, both CCU40/-41/-42,/-43 and CCU80/-81:

- **Global Start by SW**, by the CCUx Global Start Control bits in the Global Start Control register **CCUCON**
- **Global Start by HW**, indirectly, by **External Events Control** using the registers **CC4yINS** and **CC4yCMC**

4.2 Global Start of the CAPCOM Units CCU4 and CCU8

Global start of nearly any number of timers, independently of CAPCOM unit belonging, can to be done! By this manner a *synchronization of the timers* will be achieved and *all timings can be controlled in parallel* – such as output pattern generation of many different kinds - for example in Multi Channel control.

See Figure 7.

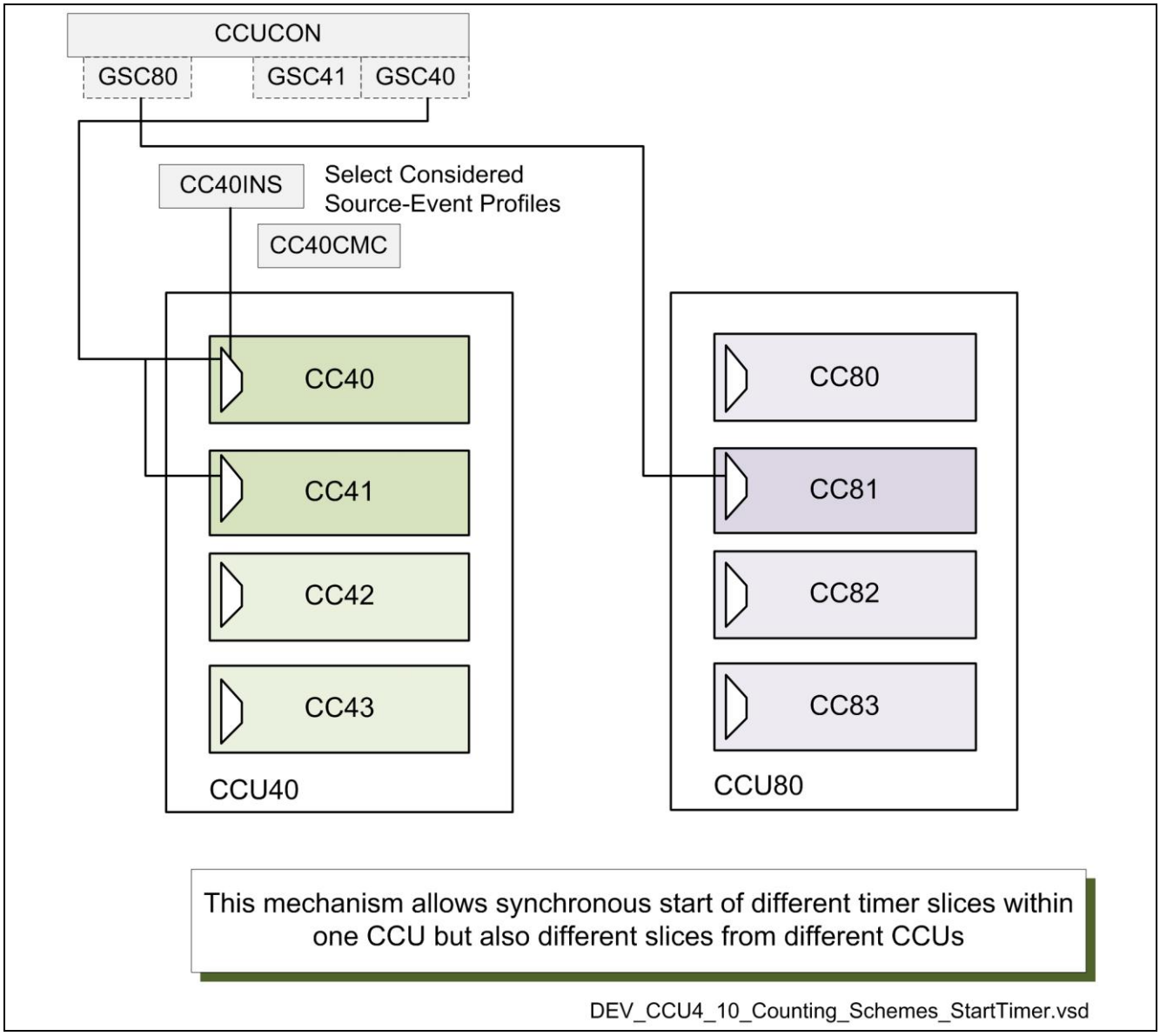

Figure 7 Synchronous Global Start of Timer Slices on different CAPCOM units (CCU4 or CCU8)

5 Getting Started with CCU4 and Counting Schemes

5.1 Ultimate Initialization Sequence

Before start and execution of a timer slice SW first time, the CAPCOM4 must have been initialized appropriately according to a specific **CAPCOM4 Initialization Sequence:** 1. *Apply Reset*, 2. *Release Reset*, 3. *Enable Clock*, 4. *Enable Prescaler Block*, 5. *Configure Global Control*, 6. *Configure Slice(s) Functions*, *Interrupts* and *Start-up*.

5.1.1 Ultimate Start-Up Enable

In the last part of the ultimate **CAPCOM4 Initialization Sequence** the startup value(s) for a specific Compare Channel Status of the Timer Slice(s) may be configured by the respective **GCSS.SyTS** bit. After that the default *IDLE mode has to be removed* from the Timer Slice(s) in register **GIDLC** and Start or Global Start can be done!

All these topics are covered in detail by the Reference Manual: <<< Link to 23.6.1 Initialization Sequence >>>

5.2 Setting the Timer Counting Modes

The Timer Counting Mode is *Edge Aligned* by default, i.e. **CC4yTC.**TCM=0_B. The *Center Aligned* mode will be set by TCM= 1_B. Single Shot Mode is set by the **CC4yTC.**TSSM bit. *Start* (and even *Stop*) may be controlled by External Events, added by extended functionalities e.g. *Start, Extended Start respective Stop, Extended Stop*.

5.2.1 Initialize and Start the Timer

After the Ultimate Start procedures have been executed, the SW may 1) initialize the slice timer **CC4yTIMER** or just clear it with a **CC4yTCCLR**.TCC bit=1 command and 2) start the timer with a Set Timer Run command in the **CC4yTCSET**.TRBS bit. The timer may be stopped at any time by SW with a **CC4yTCCLR**.TRBC command.

5.2.2 Using the Timer Counting Modes Control Register

There is practically just one and only register that is concerned here – namely the **CC4yTC** register bitfields:

1. **CC4yTC.**CMOD — CAPCOM Mode selection bit may be used: 0 → *Compare* / 1 → *Capture* mode
2. **CC4yTC.**TCM — Timer **Counting Mode selection** bit: 0 → *Edge-Aligned* / 1 → *Center-Aligned*
3. **CC4yTC.**TSSM — Set (**if intended**!) *Single Shot* Mode by this bit
4. **CC4yPR.**PR — Write the timer count cycle upper limit into the *period* register

Options:

5. **CC4yCRS.**CRS — Write the wanted *Compare Level* value into the compare *shadow* register
6. **CC4yPSL.**PSL — *PASSIVE/ACTIVE* control bit: (*CCU4xOUTy)* = (*CCUxSTy*) xor (*CC4yPSL.PSL*)
7. **CC4yTC.**STRM — *Extended Start* function on *External Start* when using External Events Control
8. **CC4yTC.**ENDM — *Extended Stop* function on *External Stop* when using External Events Control
9. **CC4yTIMER** — Initialize a timer start value (or *flush* by the **CC4yTCCLR**.TCC bit=1 command)
10. **CC4yTCSET**.TRBS — Start Timer by SW

w w w . i n f i n e o n . c o m

Published by Infineon Technologies AG