XMC4000

XMC4500 Family

CCU4 and External Events Control

Dynamic Control of Timer Functions on System Events

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 Dynamic Control of Timer Functions on System Events 5
1.1 Synchronized Control of CAPCOM Units on External Events 5
1.2 External Control Basics 5
1.3 External Events Control of Timer Functions 6
1.4 Selection of External Events Control Sources 6
1.5 Selection of External Events Control of Input Functions 6
1.5.1 Extended Slice Input Functions 6
1.6 External Control by Timer Events 6
1.7 Top-Level Control of Event Requests to/from a Timer Slice 6

2 Getting Started with CCU4 and External Events Control 8
2.1 Getting Started with the CAPCOM4 Unit CCU4x 8
2.1.1 Using the CAPCOM4 External Events Control Setup Registers 8
2.2 Using the External Events Control Setup Registers 8
2.2.1 External Event Control Register Bitfields 9
2.3 Example: Start a Timer by External Event Control 9

List of Figures

Figure 1 Timer Slice Input Functions Control on External Events via the System Interconnect Matrix 5
Figure 2 Principle Block Diagram illustrating External Event Control of a CC4y Timer Slice 6
Figure 3 Block Diagram of the External Event Control Implementation 7
Figure 4 Initialization of External Event Control Registers 8

1 Dynamic Control of Timer Functions on System Events

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 is shown here. See Figure 1.

1.1 Synchronized Control of CAPCOM Units on External Events

The manner of External Events Control distribution to CCUs (i.e. including CCU8 as well) opens for advanced applications with synchronized timer control - e.g. in Drive & Power Control, such as 3-Level Inverters requiering 12 synchronized PWMs. Just the fantacy may limit the realizable topography or timing pattern complexity range.

1.2 External Control Basics

A slice may have its input functions controlled by *external* sources. The *external source(s)*, *active mode(s)* and *input function(s)* should be mapped to the 3 inputs of the slice in the **CC4yINS** and **CC4yCMC** registers. Function mode extending alternatives can be added by selections in the **CC4yTC** Timer Slice Control register.

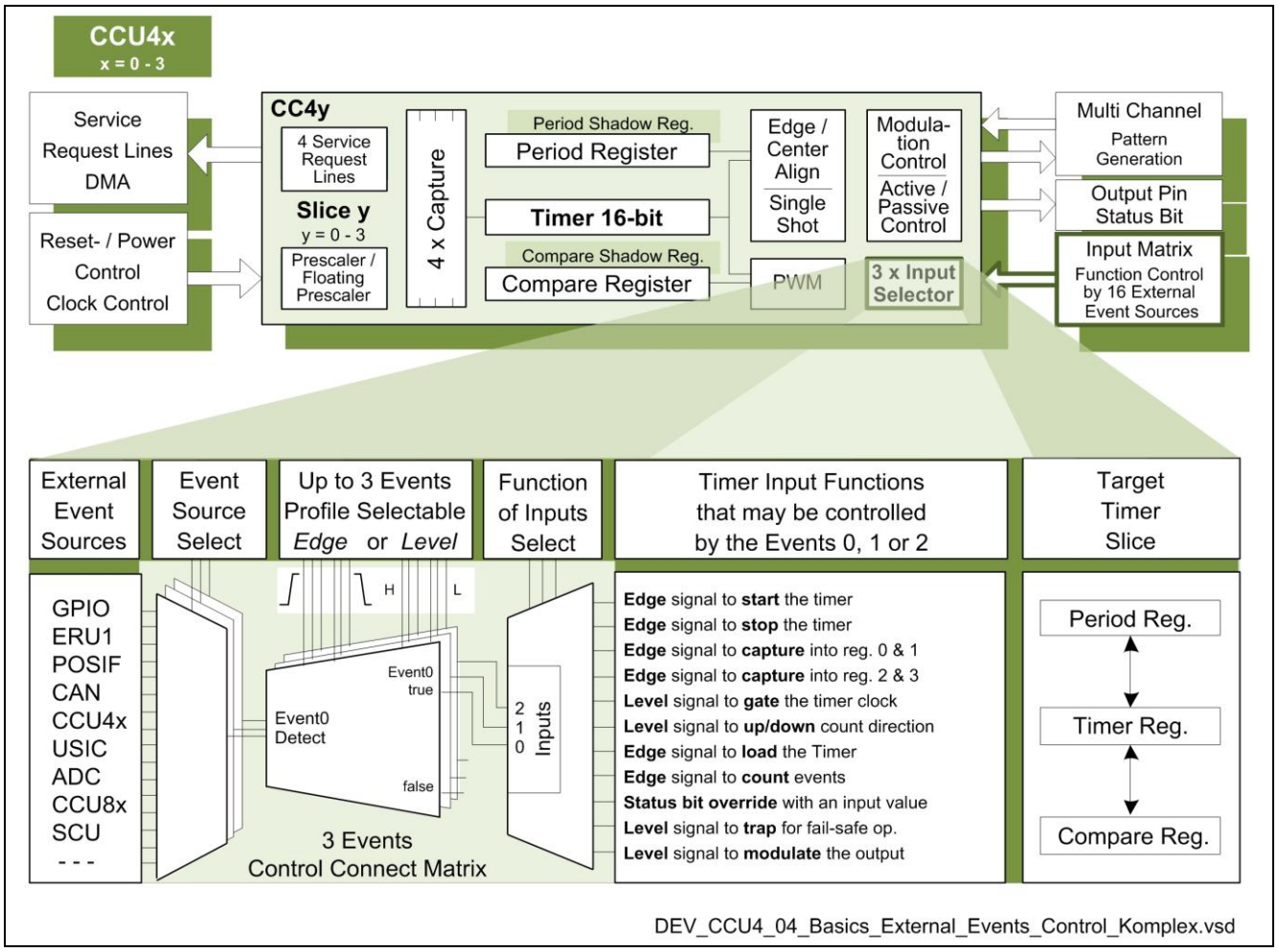

Figure 1 Timer Slice Input Functions Control on External Events via the System Interconnect Matrix

1.3 External Events Control of Timer Functions

An External Event Control request is a *considered edge* or *level event signal* from a peripheral unit or a GPIO - and is distributed to CCU4xCC4y slices input selection stages via a comprehensive matrix. A slice that by any of its 3 events setups can identify a *considered source-event profile* will be controlled by a selected input function!

1.4 Selection of External Events Control Sources

CCU4xCC4y Input Functions can be linked to external trigger requests – from such as the following sources: *GPIO*, *ERU*, *POSIF*, *CAN*, *CCU4x*, *USIC*, *ADC*, *CCU8x or SCU*. **Pin Connections** are given by the **Top-Level Interconnect** matrix and the **CC4yINS**[P:A] *Input Select* vector - and *Function Select* by the **CC4yCMC** register.

Note: A CC4y internal event is regarded as External Event too – i.e. a CC4y can control itself by its own events.

1.5 Selection of External Events Control of Input Functions

There are 11 Timer Input Functions (as e.g. *Start the Timer*), controlable by external events via 3 selectable input lines with configurable *source-event profile* conditions to the Timer Slices CC4y (y=0-3) of a CCU4x unit (x=0-3) for *Start, Stop, Capture0-3, Gate, Up/Down, Load, Count, Bit Override, Trap & Modulate Output* control.

Note: The Input Functions are, due to nature, controlled by either event edge or event level signals.

1.5.1 Extended Slice Input Functions

There are some Extended Input Functions too, in the register **CC4yTC** for the options *Flush/Start, Flush/Stop* or just *Flush* the timer – and for an *Extended Capture Mode* option that by a read access register (**ECRD**) setup simplifies administration of capture registers and full-flags, when more than one slice is used in Capture mode.

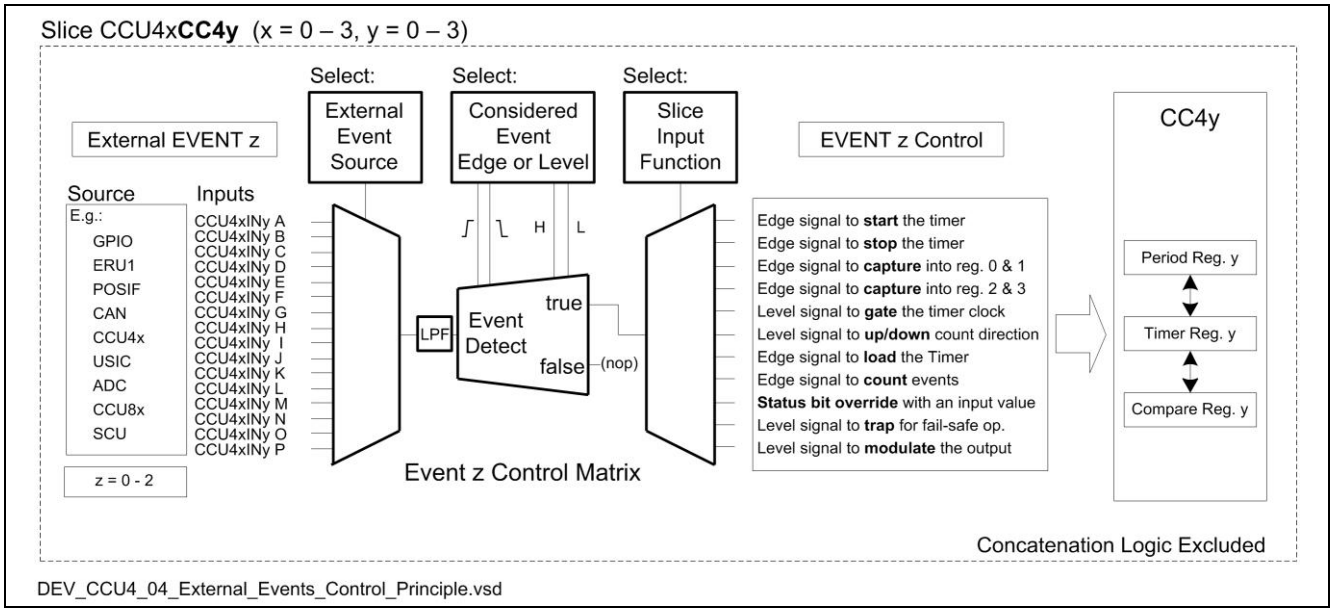

Figure 2 Principle Block Diagram illustrating External Event Control of a CC4y Timer Slice

1.6 External Control by Timer Events

A timer event may *trigger external actions* via the **Top-Level Interconnect** matrix or *request for an Interrupt*. Each CAPCOM4 has four **Service Request Lines** and each slice has a dedicated output signal **CC4ySR[3...0]** selectable to a line by **CC4ySRS.** I.e. timer slice events may request for direct peripheral actions or an interrupt.

1.7 Top-Level Control of Event Requests to/from a Timer Slice

Top-Level control means also *conditional* control of events requests between a slice and other action providers. The Event Request Unit (**ERU1**) *plus* the **Top-Level Interconnect** matrix may *combine, control and link* event signals according to *user defined* request-to-action event patterns - e.g. invoke I/O states, Time Windowing etc.

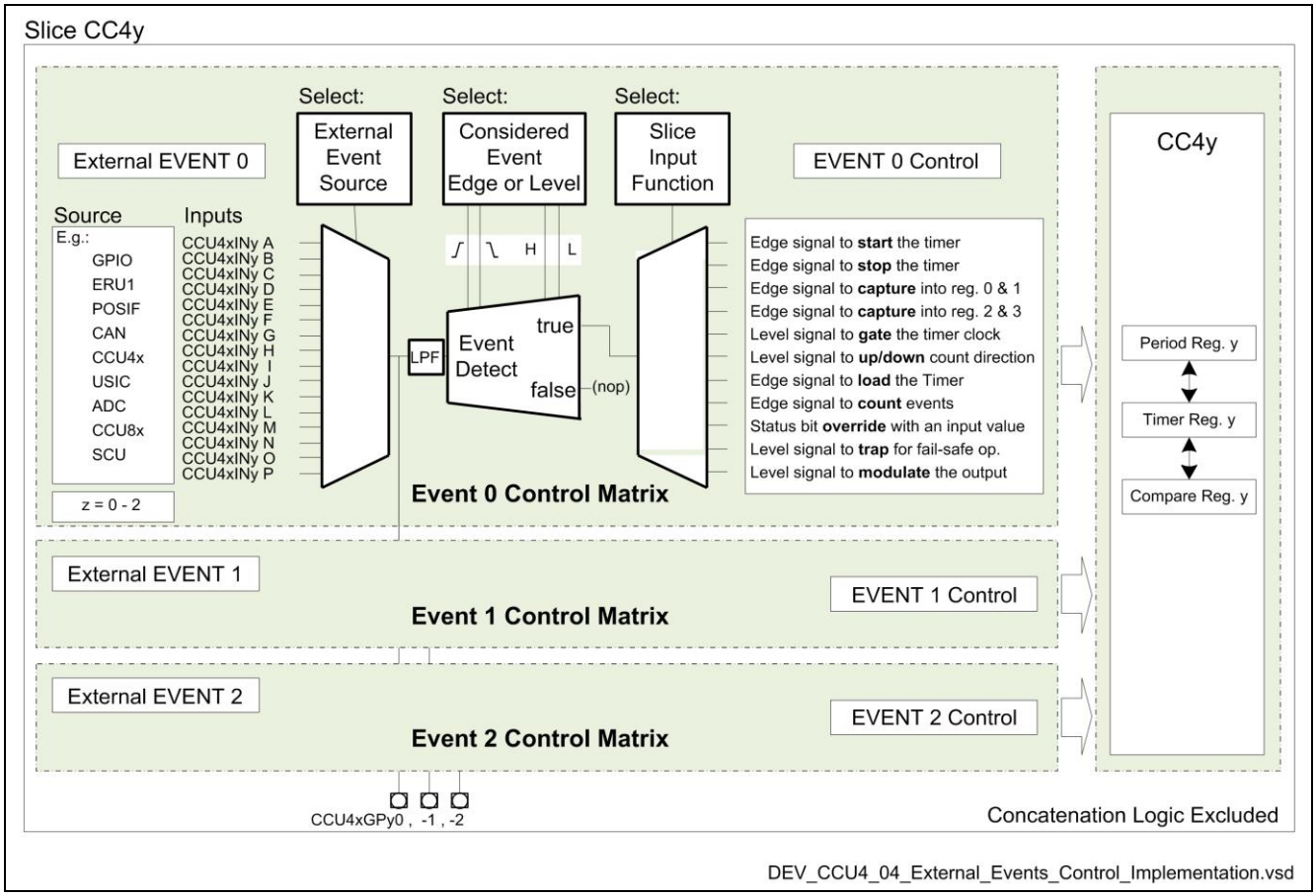

Figure 3 Block Diagram of the External Event Control Implementation

2 Getting Started with CCU4 and External Events Control

2.1 Getting Started with the CAPCOM4 Unit CCU4x

In the following examples **CAPCOM4 Ultimate Initialization Sequence** is assumed to have been done!

2.1.1 Using the CAPCOM4 External Events Control Setup Registers

There are mainly just one Input Selection Register plus one Input Function Control Register per CCU4x (x=0-3) to setup the entire functionality of External Event Control of all CC4y Slices (y=0-3). There are some extended Input functions available though in the **CC4yTC** register, e.g. the options *Flush/Start* or *Flush/Stop* or just *Flush*.

- **CC4yINS** Input Selection Register for considered event profile (i.e. source and event type)
- **CC4yCMC** Input Function Selection Register

Initializations and functions are determined by writing to the respective register **bitfields**.

See Figure 4.

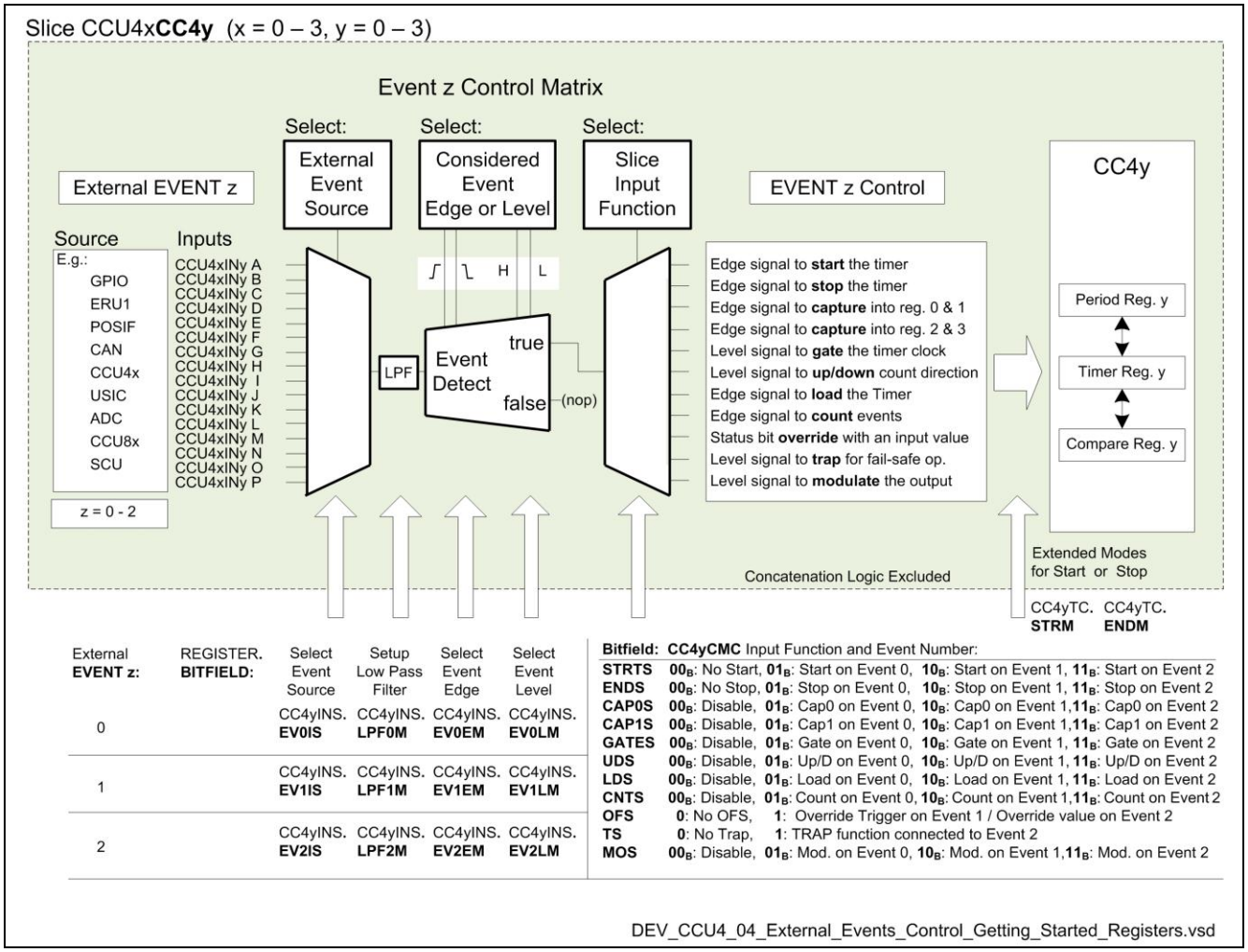

Figure 4 Initialization of External Event Control Registers

2.2 Using the External Events Control Setup Registers

The *Input Selection* register **CC4yINS** maps *External-Event-Sources* and *Edge/Level Mode* conditions to the *Event[2:0] Input Lines*. In turn the *Connection Matrix Control* register **CC4yCMC** maps the three *Event[2:0] Input Lines* to any of the 11 *Input Functions*. The *Timer Control* register **CC4yTC** enables different *Mode Alternatives*.

2.2.1 External Event Control Register Bitfields

1. **CC4yCMC.**<function> Input *function* selection and Event Number (x=0-2) Input Line Selection bitfield
2. **CC4yINS.**<Event-*x* bitfields> Event-x *source* EVxIS (**CC4yINS**[P:A]), *mode* EVxEM or EVxLM bitfields
3. **CC4yTC.**<Mode bitfields> Timer Mode selections e.g. Modes/*Extended modes, Counting Schemes etc*

Note: When using the index 'x' in event nomenclature, it should be understood as 'Actual Event[2:0] Number '.

Note: When using the index 'x' in CCU4x nomenclature, it should be understood as 'Actual CCU4 Unit Number'.

Note: The Ultimate Event Number 'x' for the Input Function is x == <function>-1 (when <function> != 0). <function> == 0 means "disabled".

2.3 Example: Start a Timer by External Event Control

```
// Assume a falling edge transition on GPIO Pin P1.3 as a "considered" external
// event control signal to start Timer CC40 in CAPCOM Unit CCU40.
// Define this event as "Event2". (The alternatives Event0..1 would do as well!):
CC40CMC |= 0x00000003; // Start Timer CC40 by External Event 2 (Bitfield STRTS).

// Setup Timer CC40 External Event Control Input Selection register (CC40INS)
// to activate GPIO Pin P1.3 as selected source for Event 2 in the Bitfield EV2S:
EV2S = 0; // where "0" is the input number (from 0-15) for the input "CCU40IN0[A]"
          // and represents Pin P1.3 according to the Top-level Connection Matrix.
EV2EM = 2; // set signal active on falling edge on Event 2, as "considered".
// See reference: Table 26-10 Pin Definition and I/O Functions.
```

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

Published by Infineon Technologies AG