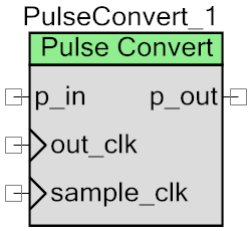
Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

Pulse Converter

1.0



Features

- Terminals for out_clk and sample_clk for configurability of sample rate and output pulse width.

General Description

The Pulse Converter component produces a pulse of known width when a pulse of any width is sampled on p_in.

When to Use a Pulse Converter

Use the Pulse Converter to interface pulse events from a fast domain to a slow domain, or when a specific pulse width must be guaranteed:

- Safely responding to the DMA nrq event in a slow domain.
- Converting a variable width pulse to a known width.

Input/Output Connections

This section describes the various input and output connections for the Pulse Converter.

p_in – Input

The signal connected to the p_in input is sampled for a pulse with sample_clk.

out_clk – Input

The out_clock input determines the width of the generated pulse on p_out, and also determines the maximum rate at which pulses on p_in can be acknowledged. Note: out_clk and sample_clk should be synchronous to a common clock.

sample_clk – Input

The sample_clock input determines the sampling rate of the p_in input, and hence the minimum acceptable pulse width for p_in. Note: out_clk and sample_clk should be synchronous to a common clock. Sample_clk must be at least as fast as out_clk.

p_out – Output

The p_out output pulses high for one cycle of out_clk when a pulse of any width is detected on p_in.

Component Parameters

The Pulse Converter component has no parameters.

Functional Description

The Pulse Converter samples the p_in input using sample_clk. When a rising edge is detected, the logical high value is first stored in the sample_clk domain, and then transitioned to the out_clk domain, producing a pulse with width equal to one cycle of out_clk.

The Pulse Converter provides a safe way to react in a slow domain to level or pulse events from a fast domain without the possibility of missing events.

Figure 1. Catching Fast Domain Events in a Slow Domain

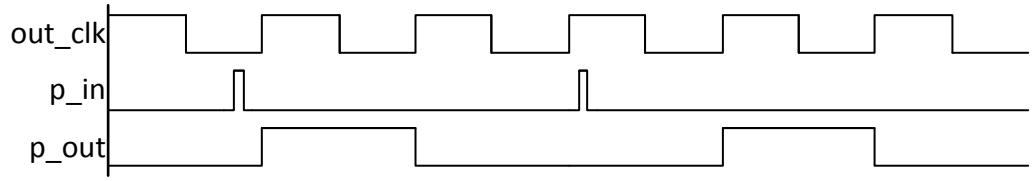


Figure 1 shows an example of the Pulse Converter being used to convert pulses from a fast domain into pulses that can be acknowledged in the slow domain clocked by out_clk. This assumes that sample_clk is sufficiently fast to sample the pulses on p_in.

The Pulse Converter can also be used to guarantee a specific pulse width, even when the input pulse width may vary.

Figure 2. Guaranteeing Specific Pulse Width

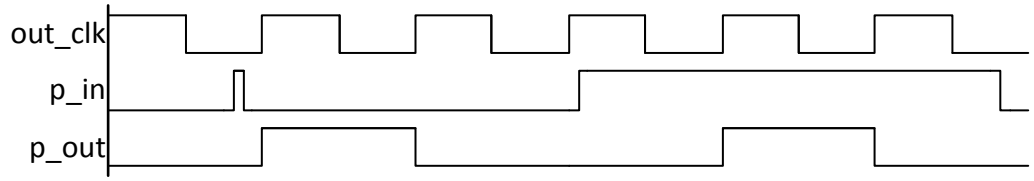


Figure 2 shows an example of the Pulse Converter being used to generate pulses of a specific pulse width on p_out even though the pulse width on p_in varies. This assumes that sample_clk is sufficiently fast to sample the pulses on p_in.

The Pulse Converter is not intended to handle pulses that occur more often than the frequency of out_clk. If multiple pulses occur on p_in before p_out has had a chance to pulse, the extra pulses will be swallowed, and only one pulse will be seen on p_out.

Figure 3. Rapid Pulses Being Swallowed

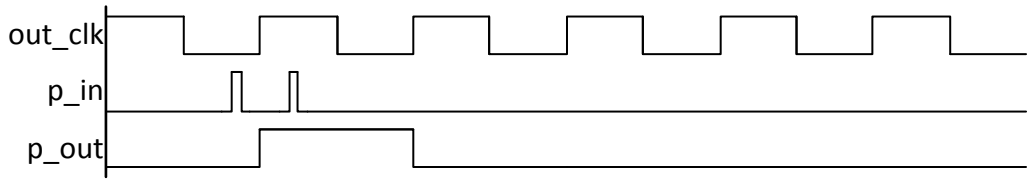


Figure 3 shows the result of multiple pulses occurring on p_in before p_out has had a chance to acknowledge the first one. This is usually not the intended behavior, so care should be taken to ensure that pulses are not lost.

There is a delay between the time a pulse begins on p_in and the time the corresponding pulse beings on p_out. Depending on when the pulse on p_in occurs during the cycle of out_clk, the delay could be as small as $period_{sample_clk}$.

Resources

Configuration	Resource Type					
	Datapath Cells	Macrocells	Status Cells	Control Cells	DMA Channels	Interrupts
Pulse Converter	–	3	–	–	–	–

MISRA Compliance

This section describes the MISRA-C:2004 compliance and deviations for the component. There are two types of deviations defined: project deviations – deviations that are applicable for all PSoC Creator components and specific deviations – deviations that are applicable only for this component. This section provides information on component specific deviations. The project deviations are described in the MISRA Compliance section of the *System Reference Guide* along with information on the MISRA compliance verification environment.

The Pulse Converter component does not have any C source code APIs.

Component Changes

Version 1.0 is the first release of the Pulse Converter Component.

© Cypress Semiconductor Corporation, 2012-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC (“Cypress”). This document, including any software or firmware included or referenced in this document (“Software”), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress’s patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage (“Unintended Uses”). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.

