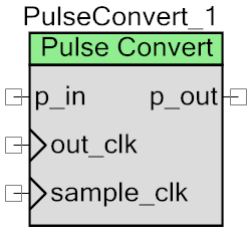
Pulse Converter
1.0



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

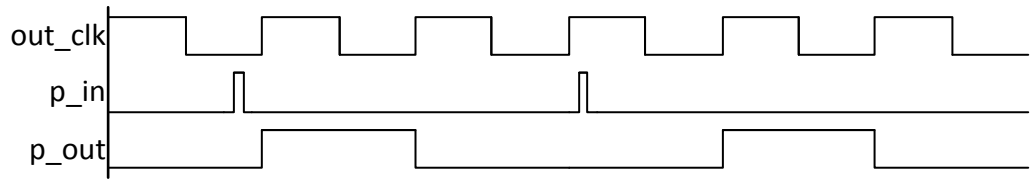


Figure 1 shows an example of the Pulse Converter being used to convert pulses from a fast domain into pulses that can be acknowledged in the slow domain clocked by out_clk. This assumes that sample_clk is sufficiently fast to sample the pulses on p_in.

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

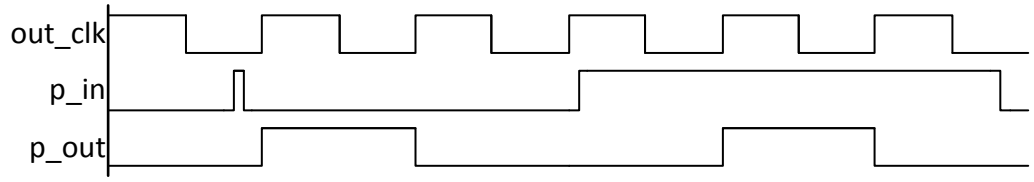


Figure 2 shows an example of the Pulse Converter being used to generate pulses of a specific pulse width on p_out even though the pulse width on p_in varies. This assumes that sample_clk is sufficiently fast to sample the pulses on p_in.

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

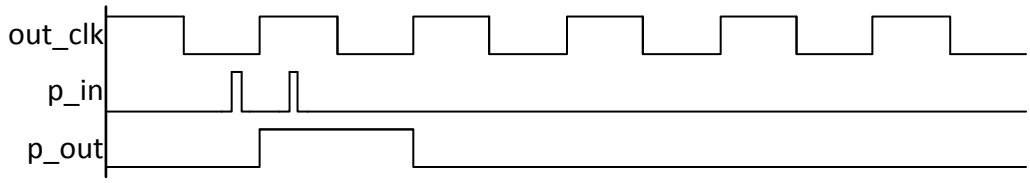


Figure 3 shows the result of multiple pulses occurring on p_in before p_out has had a chance to acknowledge the first one. This is usually not the intended behavior, so care should be taken to ensure that pulses are not lost.

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.