TN-Number	Author	Phone	Filename	Date	Page
TN608-04	R.Bachmann	+41 56 486 8103	TN608-04 Time Domain.docx	10.05.2017	1 / 7

Concern

Better Understanding for Filters in the Time Domain

1. Frequency Domain and Time Domain

Working in the frequency domain is very useful in many cases. With a signal frequency we recognize in one number how often something happens. For instance with the frequency of a sinusoidal signal and the corner frequency of a filter we can easily see if a signal will pass the filter or not.

Trigger signals rarely have a sinus shape and it is not simple to understand what frequencies are included in such a signal. Further the difference between the frequency spectrum and the pulse frequency may lead to confusion.

In the time domain one can work with rise times, filter time constants and time delays. All these parameters are easy to understand and illustrate. That is why Triggering is easier to understand in the time domain.

2. Time Constant and Corner Frequency of a Filter

A filter, such as the low and highpass input filters of PI608 Versatile Trigger Unit, is usually specified with its corner frequency. But in the case of triggering the corner frequency often is not a practical specification. The question one asks is "what is the minimal signal rise time when I use a certain filter". A steep signal edge is of interest to achieve a high timing accuracy.

In the time domain a filter is specified by its time constant as it is in the frequency domain with its corner frequency. This time constant defines how fast the output of a filter responds to a signal step at the input. In more detailed words it specifies the time the output needs to get to 63% of the step level at the input.

For a simple filter it is calculated by its corner frequency as followed:

$$\tau = \frac{1}{2 * \pi * f_c}$$

Finally we are not interested in the time to reach 63%. Usually rise time means the time the output needs to get to 90% of the input step level.

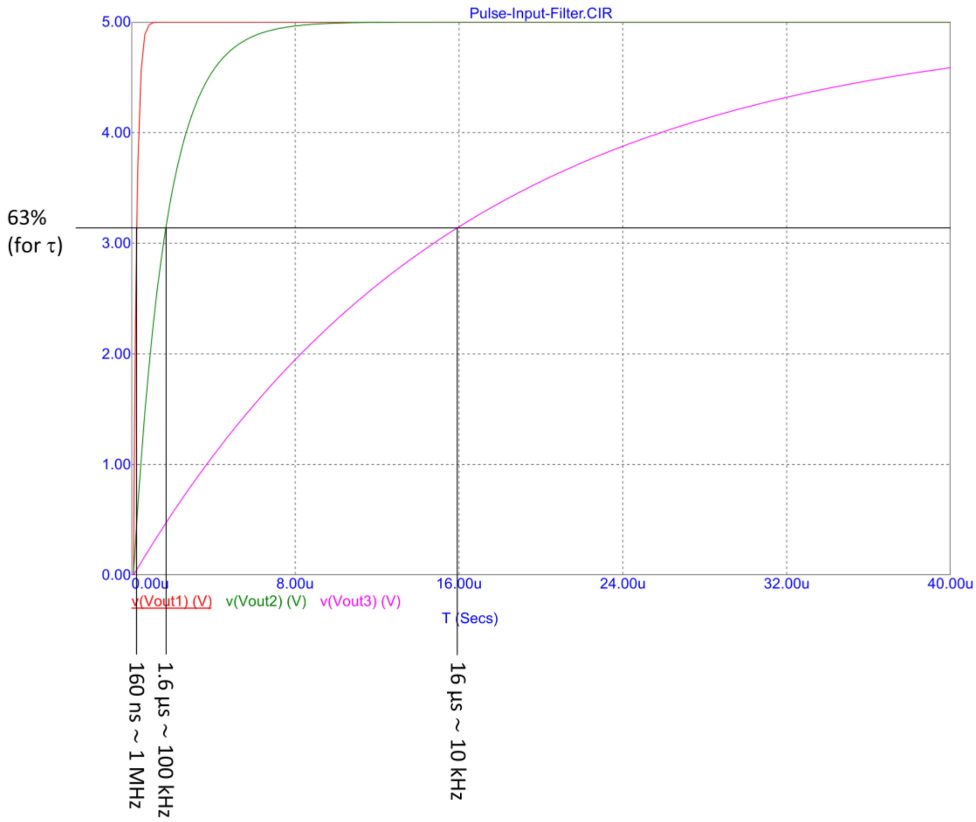
As a rule of thumb it can be calculated as shown below:

$$t_{rise} = 2.3 * \tau$$

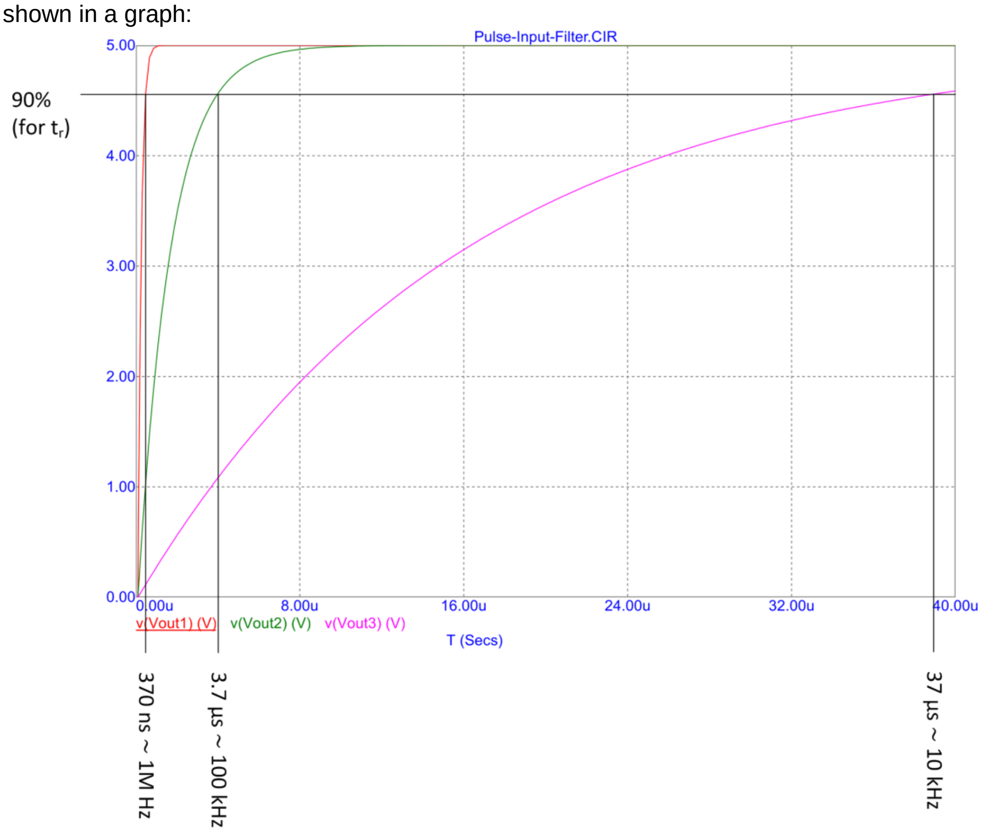
The time constants and minimal rise times of the low pass filters of PI608 Versatile Trigger Unit are:

Corner Frequency (fc)	Time Constant (τ)	Min. Rise Time (t_r)
1 MHz	160 ns	370 ns
100 kHz	1.6 µs	3.7 µs
10 KHz	16 µs	37 µs
1 kHz	160 µs	370 µs
100 Hz	1.6 ms	3.7 ms
10 Hz	16 ms	37 ms

Time constant (τ) shown in a graph:



Rise time (t_r) shown in a graph:



3. Pulses after filtering

Now we take a look to what happens to an ideal pulse signal, when we let it pass through one of the input lowpass filters of PI608 Versatile Trigger Unit.

The pulse signal is specified with:

- repetition rate: 1 kHz
- pulse width: 25 µs
- Level: 0V / 5V

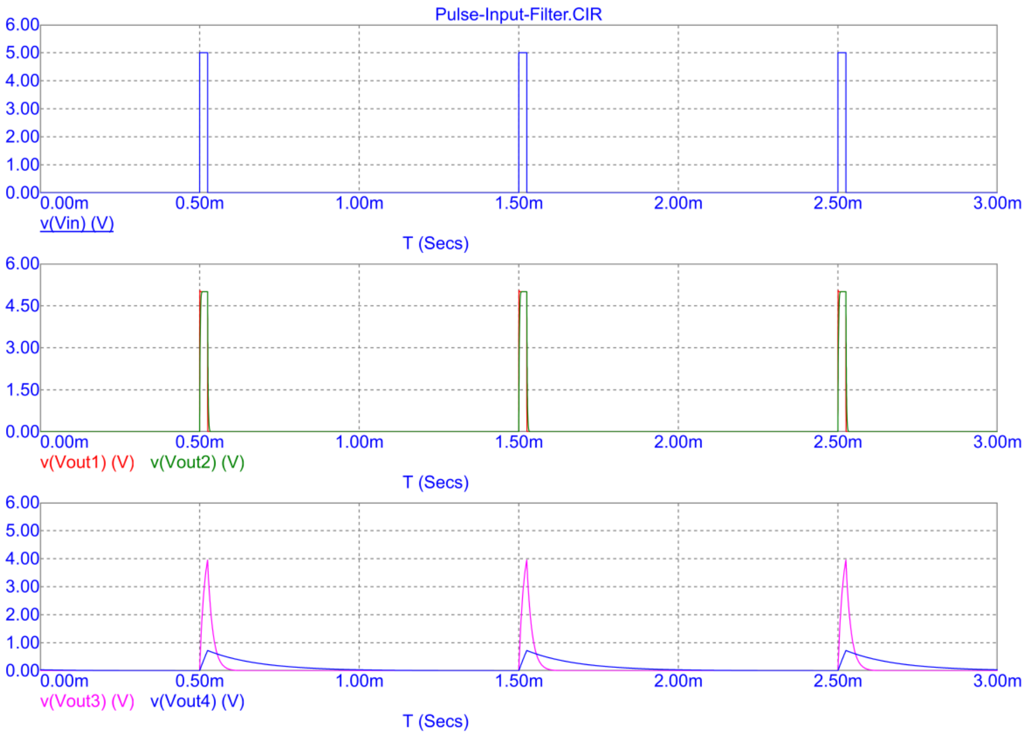
Unfiltered:

red:
tr = 0.39 µs
(fc = 1 MHz)

green:
tr = 3.9 µs
(fc = 100 kHz)

pink:
tr = 39 µs
(fc = 10 kHz)

blue:
tr = 390 µs
(fc = 1 kHz)



It is easy to see that a filter with a rise time longer than the pulse with will change the shape of the pulse dramatically. The pulse becomes flat and delayed. Proper triggering may become impossible.

This also shows that the pulse frequency is no information for the setup of the input filter. The important information is the needed rise time or if not know the pulse width.

4. Time shift by Filtering of a Square Pulse

Now we zoom into the graphs of chapter '3 Pulses after filtering ' to see what happens with which filter time constant to the pulse. In the example below a trigger level of 3V is set. It is marked by the orange line.

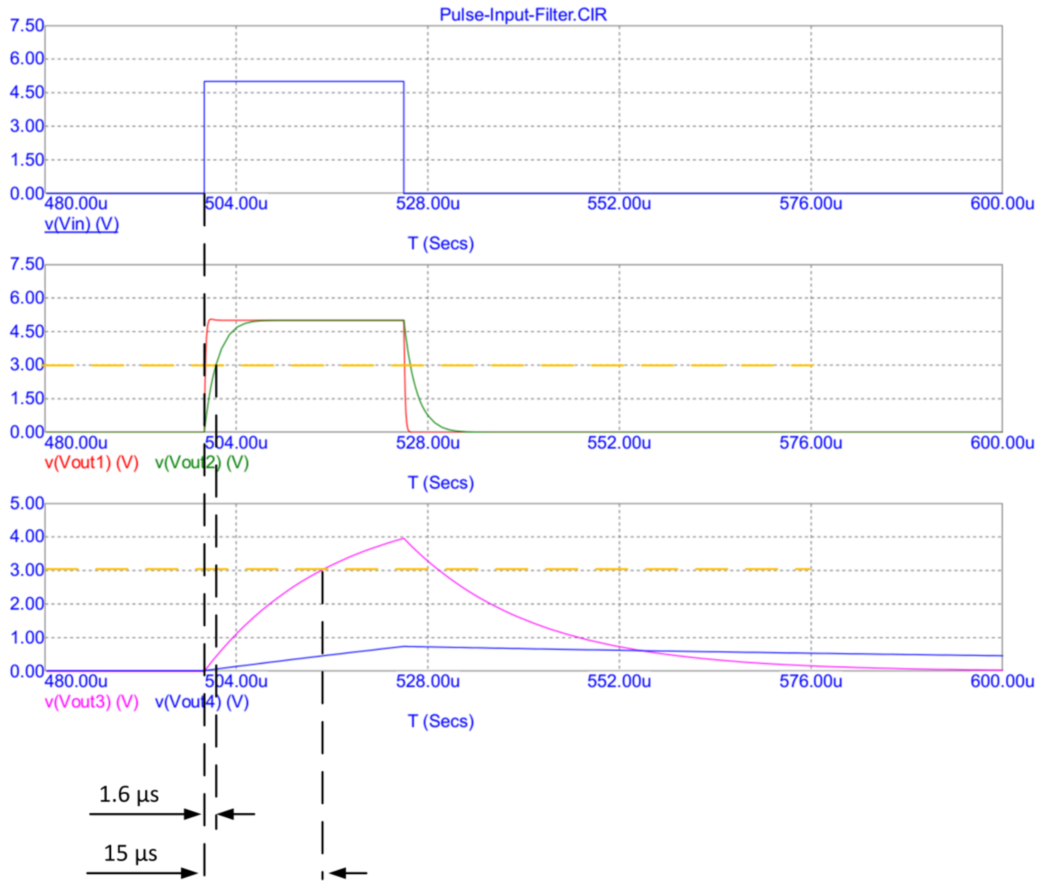
Unfiltered:

red:
tr = 0.39 µs
(fc = 1 MHz)

green:
tr = 3.9 µs
(fc = 100 kHz)

pink:
tr = 39 µs
(fc = 10 kHz)

blue:
tr = 390 µs
(fc = 1 kHz)



As we can see a filter with a rise time of 3.9 µs (100 kHz) makes a delay of 1.6µs. That can be already quite a lot for a 25 µs pulse.

5. Time Shift of the Signal of a reluctance Sensor

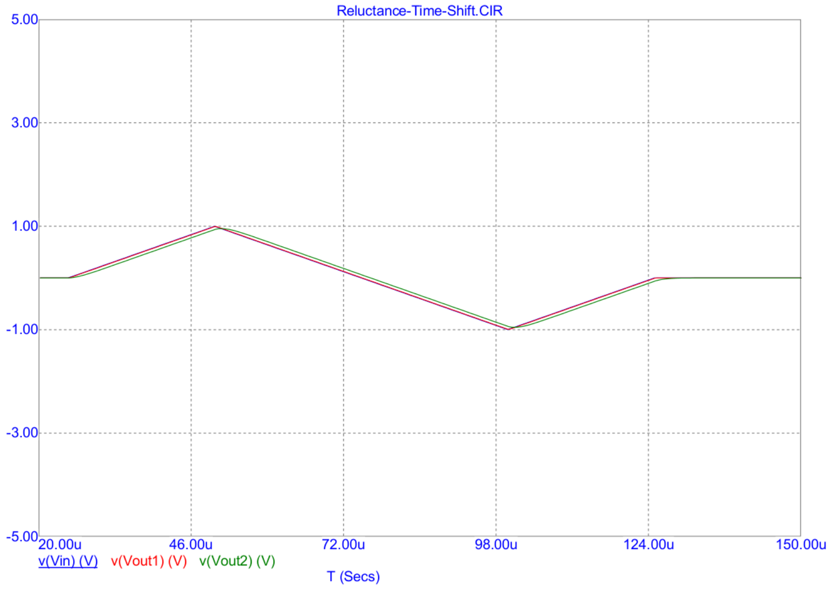
The signal of reluctance sensors usually is not as steep as the pulse signals described in the previous chapters. But signal quality may suffer from the same issues when the use of filters is not well thought-out. In the graphs below we see the signal of a reluctance sensor used at a rotating machine.

Lower Speed:

blue:
unfiltered

red:
tr = 0.39 µs
(fc = 1 MHz)

green:
tr = 3.9 µs
(fc = 100 kHz)

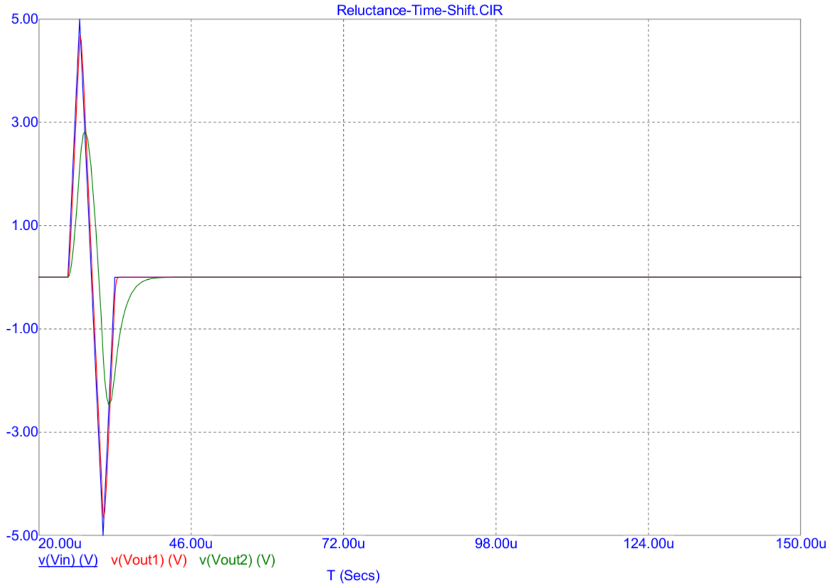


Higher Speed:

blue:
unfiltered

red:
tr = 0.39 µs
(fc = 1 MHz)

green:
tr = 3.9 µs
(fc = 100 kHz)



See the difference of the green line at lower speed and higher speed.

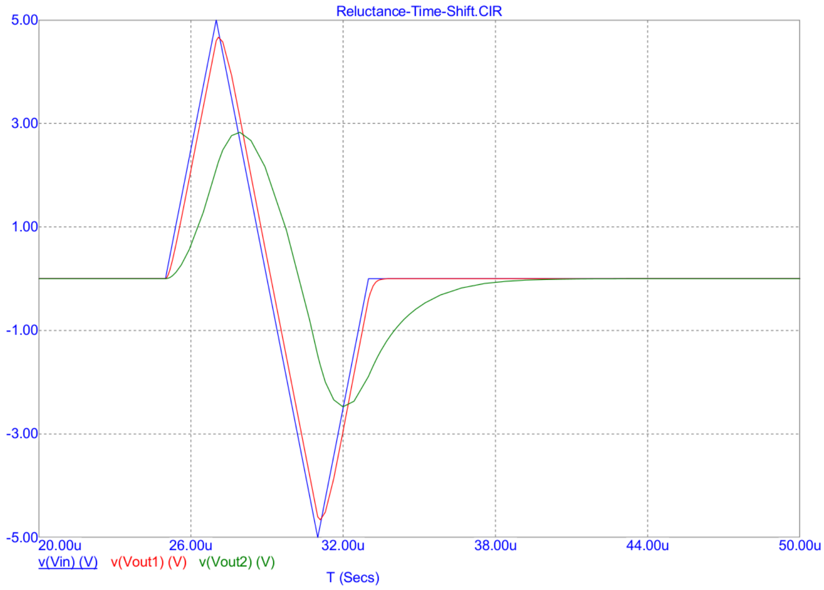
The signal of a reluctance sensor changes with speed in both dimensions, time and amplitude. That means its edge is getting steeper based on growing amplitude and shorter signal duration.
A filter with a time constant, that does not affect the signal at low speed, changes the signal shape at high speed.
In the end, the user measures a time delay at high speed, but not at low speed. That may lead to irritating timing results.

A Zoom into the “Higher Speed” plot shows the shape change of the sensor signal more clearly:

blue:
unfiltered

red:
tr = 0.39 µs
(fc = 1 MHz)

green:
tr = 3.9 µs
(fc = 100 kHz)



6. Limited Signal Rise Time and Filters

Taking a closer look to the chapters before shows that the reluctance sensor signal with 2 µs rise time was almost not affected by the filter 0.39 µs rise time, but it was much affected by the one with 3.9µs.

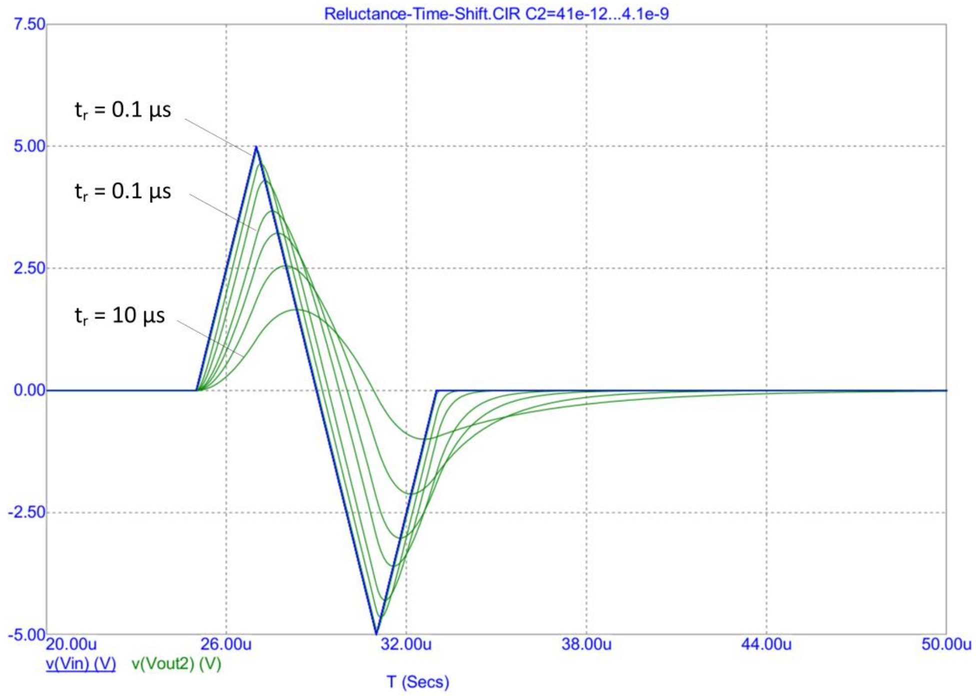
The total rise time is the geometrically addition of the signal rise time and the filter rise time.

$$t_{rise}(tot) = \sqrt{t_{rise}(signal)^2 + t_{rise}(filter)^2}$$

Signal Rise Time	Filter Rise Time	Total Rise Time
[µs]	[µs]	[µs]
2	0.1	2.00
2	0.5	2.06
2	1.0	2.24
2	2.0	2.83
2	3.0	3.61
2	5.0	5.39
2	10.0	10.2

Looking to the table above we see what that means: If there is a big difference between two rise times (signal and filter), the longer one dominates. If they are similar, the result is some higher than the longer rise time depending on the geo-metrical addition.

Let's show the table on the previous page also in a graph:



7. Avoiding Timing Problems by Using the HOLDOFF

The HOLDOFF function is a powerful feature provided by PI608 Versatile Trigger Unit. It works in the time domain by nature. It locks the output of PI608 Versatile Trigger Unit for a specified time after triggering. Doing that does not affect the shape of the sensor signal at all, but helps to ignore all influences after a trigger event, that may lead to mistriggering.

Read more about how to use the HOLDOFF function in the white paper:
"Blocking parasitic triggering with the HOLDOFF function"

Read more about a comparison of the HOLDOFF function and filtering in the white paper:
"The Use of Filters and Holdoff"