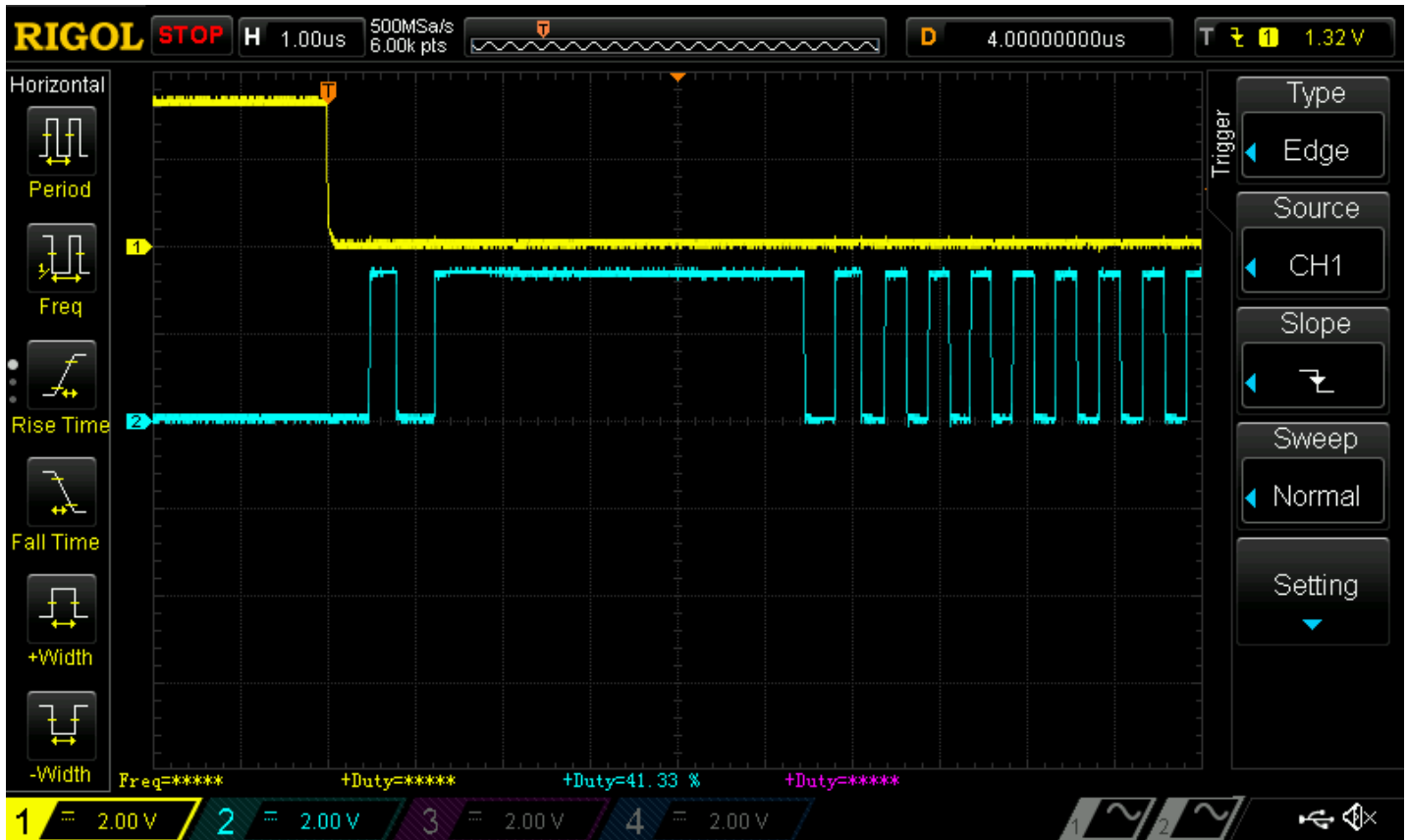


SLG46580

ISSUE 1: Long 2 MHz OSC Settling Time Functional Block Affected: OSC, Counter, Delay

Description:

2MHz OSC has an additional ~ 9 cycles settling period. Higher VDD shows longer settling time.



Channel 1 – OSC Power Down; Channel 2 –OSC output

Such behavior will lead to substantial error in period calculations if the delay time is relatively small.

Workaround:

- Enable Fast Start-up option. Fast Start-up means forcing bias ready at the power-up instead of automatic enabling at OSC event. The standby current consumption difference between Fast start-up disabled and enabled is only an additional 300 nA.
- Use the “Force power on” OSC power control option to make the OSC operate at all times. However, this will cause increased constant current consumption.

ISSUE 2: Possible Glitch on ACMP Output

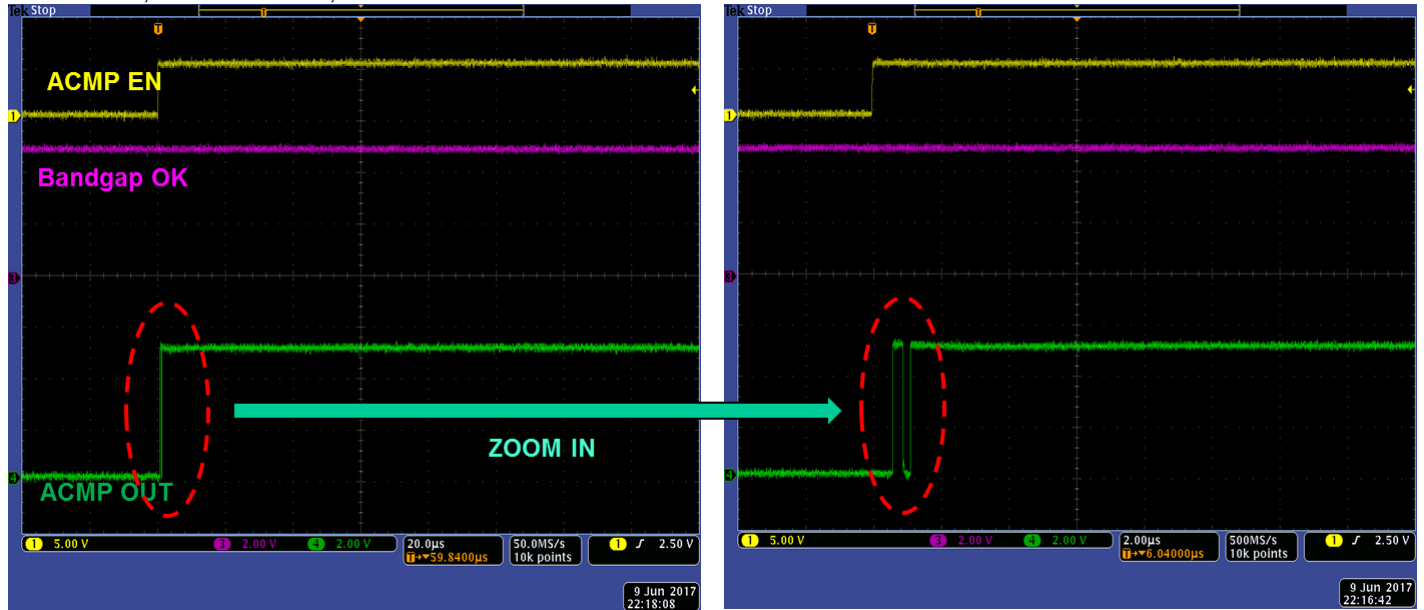
Functional Block Affected: ACMP

Description:

After power-up, if LDO is enabled earlier than ACMP, its output may generate a glitch.

SLG46580

VDD = 5.0 V, ACMP IN+ > IN-, One of LDOs enabled



Because BG_OK is already released by LDO, there is no gating signal. Depending on the ACMP VREF condition & the positive input value, it is possible for the ACMP to have a glitch. Subsequently, the other ACMP will not have an issue unless the customer repeats LDO enable first and ACMP later during power stable.

Workaround:

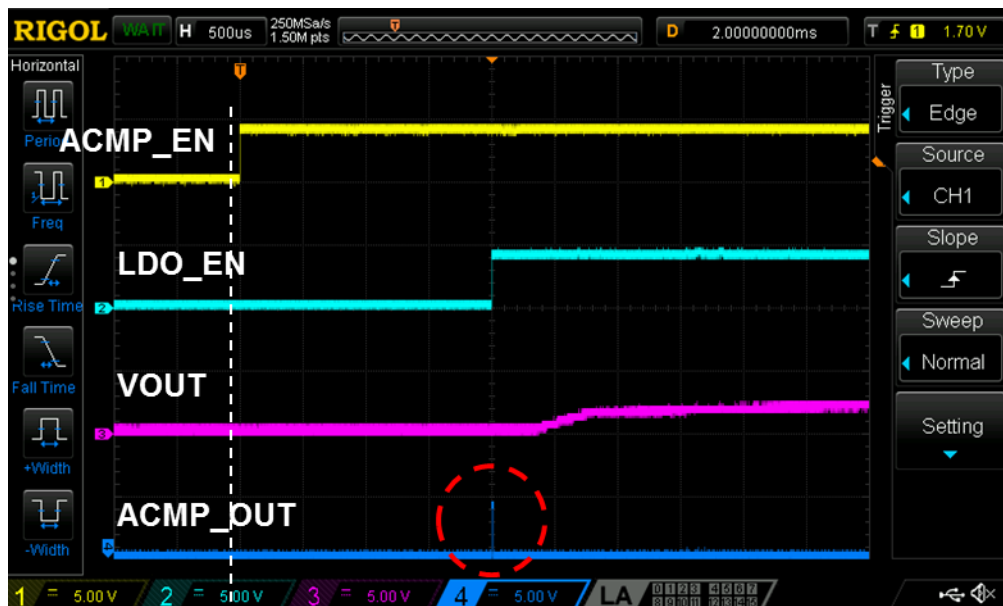
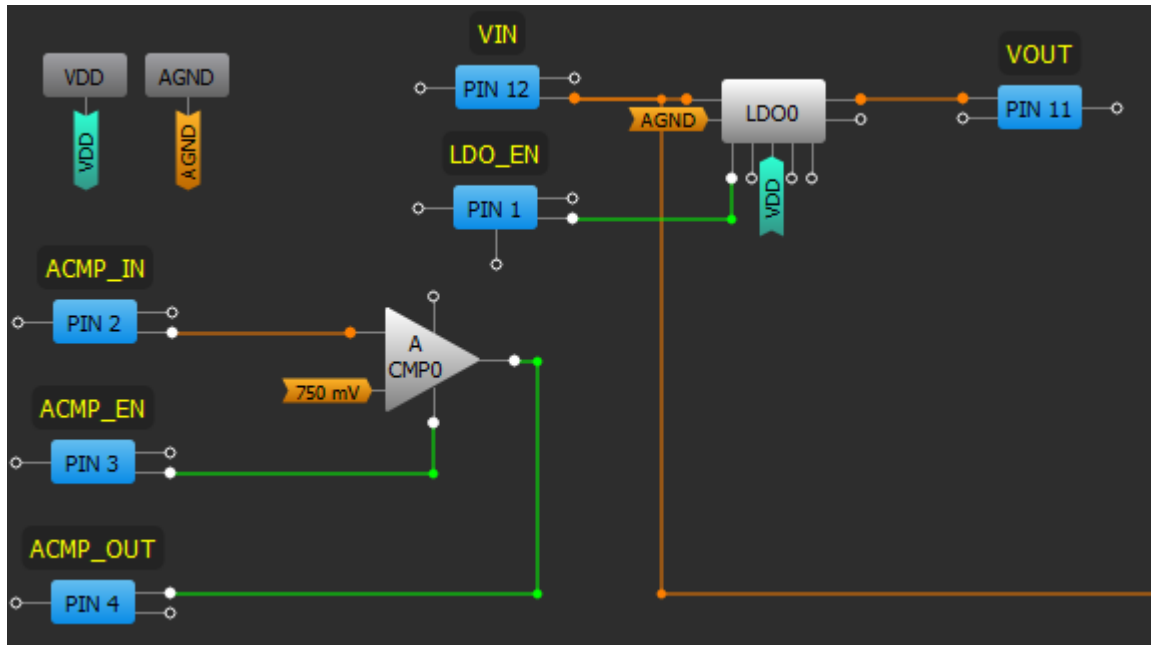
- Use both edge delay on the output to filter out the glitch.

ISSUE 3: Possible Glitch on ACMP Output

Functional Block Affected: ACMP

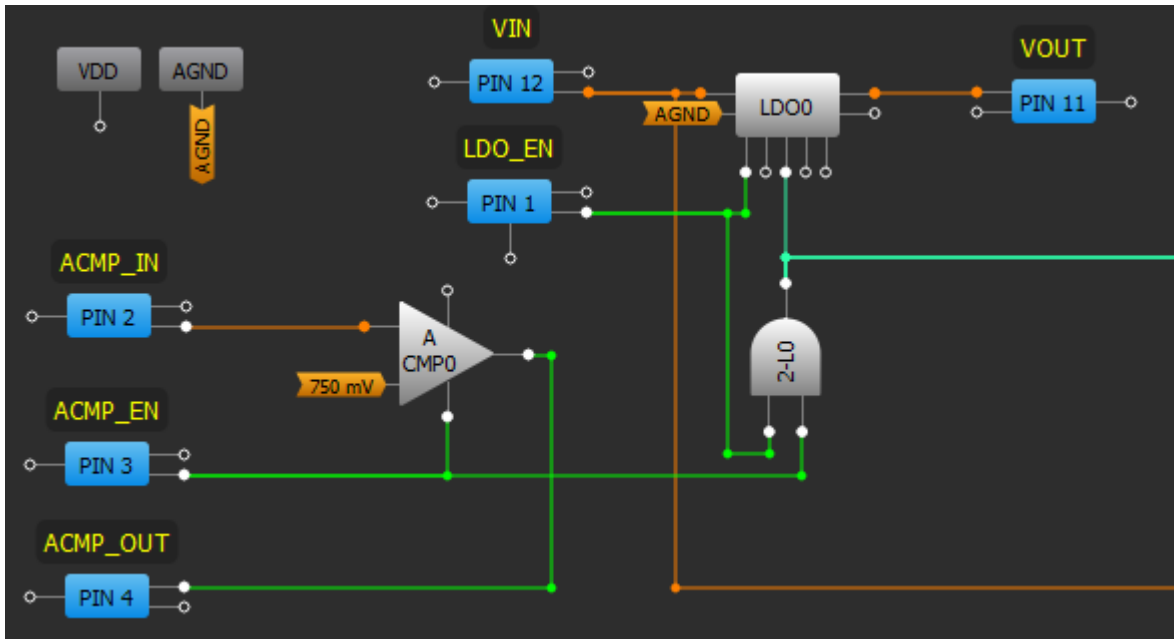
Description:

If an LDO's Low Power Mode is selected (for example tied to VDD) a glitch may appear on the ACMP's output after the first LDO is enabled. When the first LDO is turned on, the Low Power Mode switch causes a drop in the internal bandgap voltage that is used to derive the ACMP reference voltages.



Workaround:

- Use both edge delay on the output to filter out the glitch.
- Use some logic to avoid turning on Low Power Mode before enabling LDO and powering up the ACMP. Example is shown below.



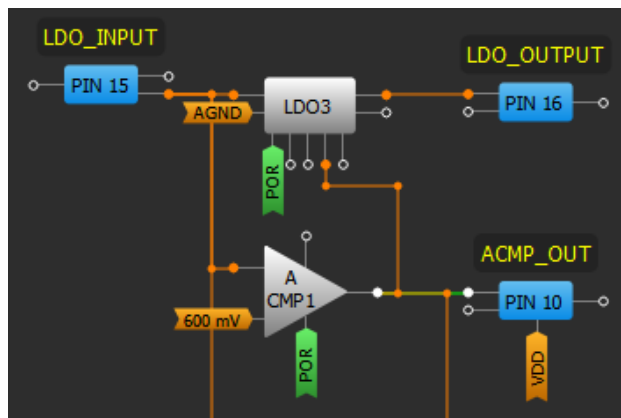
ISSUE 4: ACMP's Erroneous Behavior when Internally Tied to the LDO's Input

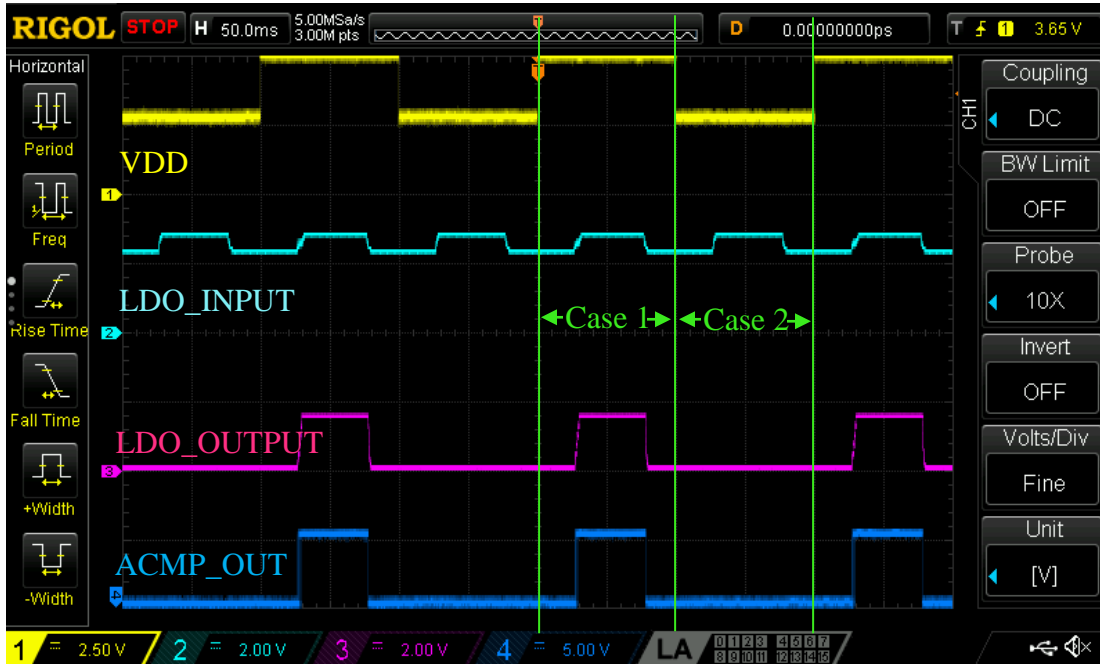
Functional Blocks Affected: ACMP and LDO

Description:

If an LDO's VIN pin (LDO_INPUT) has an internal connection (depicted with an orange wire) to the ACMP's IN+ port, the ACMP's output may give an erroneous result with respect to the LDO's input voltage. The switch between the LDO's VIN pin and the ACMP's IN+ port is implemented using an NMOS-only based transmission gate with its gate controlled by logic supplied from VDD. As a result, this switch is unable to pass LDO input voltages that are close to VDD. This can result in the ACMP's IN+ port seeing a lower voltage than the actual voltage present at LDO_INPUT. This lower input voltage may cause the output of the ACMP to behave unexpectedly.

If the LDO's UVLO feature is enabled, the ACMP's output will cause unexpected LDO behavior that results from the previously described transmission gate voltage drop.





Case 1: Proper Behavior at VDD = 5V;

With a 5V VDD, the NMOS transmission gate passes the 2.25 and 2.75V LDO_INPUT pin voltages to the ACMP's IN+ port. When the input voltage exceeds the 2.4V IN- threshold of the ACMP, the LDO is enabled.

Case 2: Erroneous Behavior at VDD = 2.75V;

When the VDD is reduced to 2.75V, the NMOS transmission gate is unable to pass LDO_INPUT's full magnitude to the ACMP's IN+ port. Since the IN+ port of the ACMP doesn't exceed the 2.4V IN- threshold, the ACMP's output stays low and keeps the LDO disabled.

Workaround:

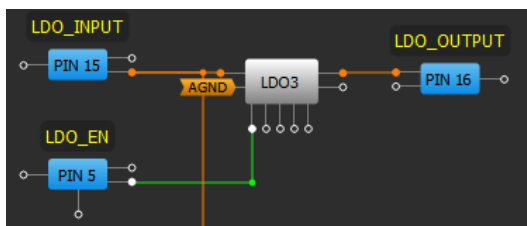
- If the GreenPAK's VDD is always powered up before the LDO's VIN input, you can externally short the LDO's VIN to another dedicated analog input pin for the desired ACMP.

ISSUE 5: Non-Zero LDO Output Voltage Step during LDO Enable

Functional Block Affected: LDO

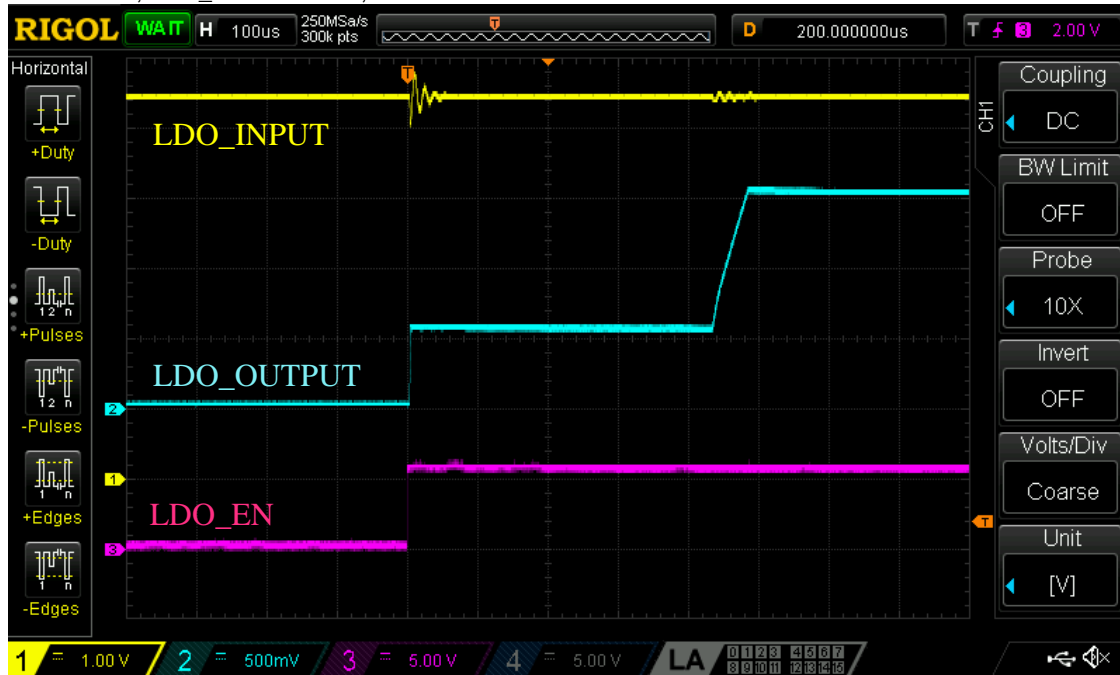
Description:

When a GreenPAK's first LDO is enabled, the LDO's VOUT will step to a non-zero voltage during its 500 μ s wait time. The magnitude of this step is directly proportional to the LDO's VIN and the GreenPAK's VDD. With a large LDO input voltage and VDD, this step can exceed 500mV.



SLG46580

VDD = 5.5V, LDO_INPUT = 5.5V, All other LDOs Disabled



Workaround:

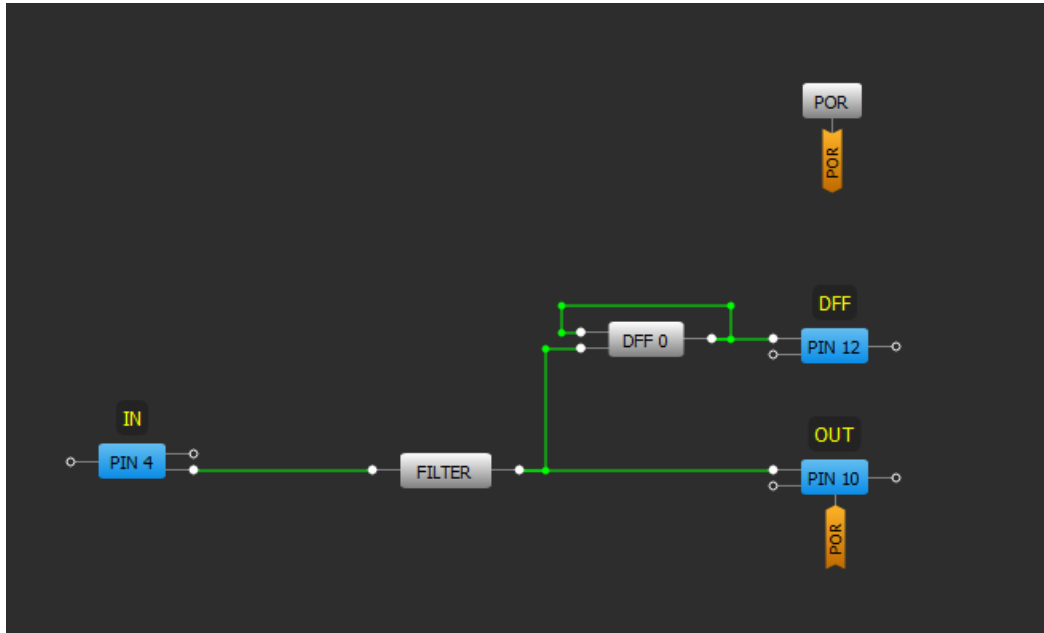
- If your GreenPAK design has an unused LDO, you can reduce the magnitude of this voltage step by keeping an unused LDO enabled in high power mode.

ISSUE 6: FILTER Cell does not Filter Out Repetitive Glitches

Functional Block Affected: FILTER

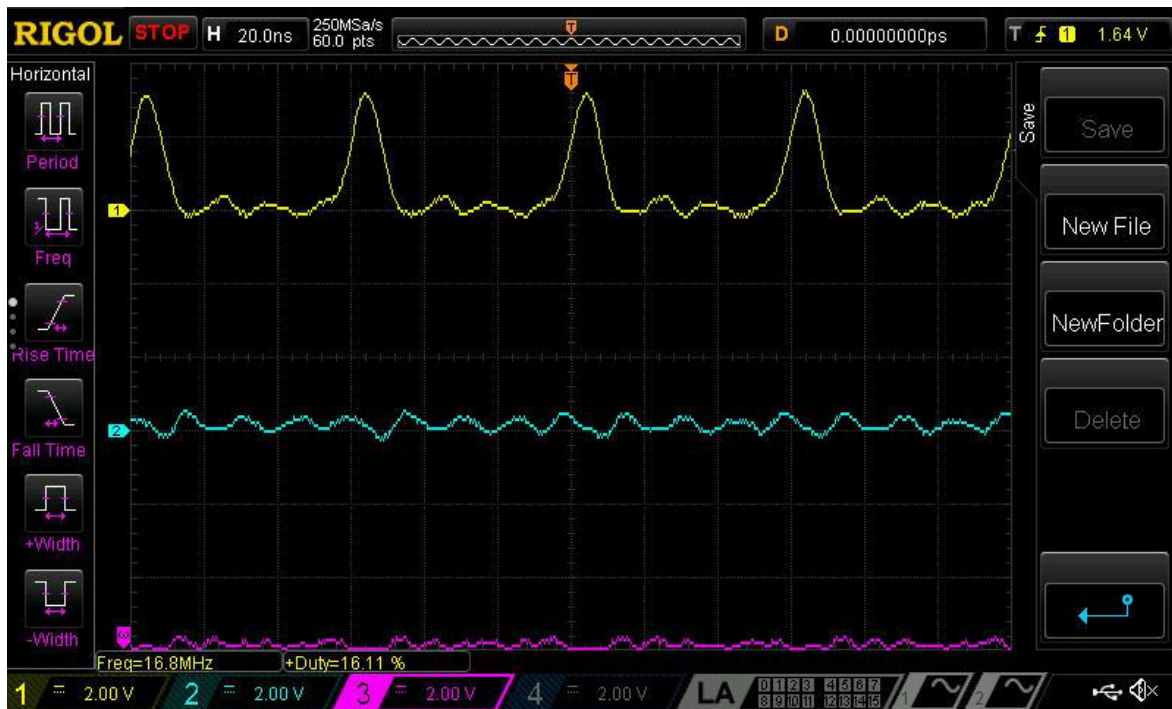
Description:

If the FILTER cell's input signal contains multiple consecutive pulses within short time intervals, the FILTER cell may not filter the input pulses as expected. The errant behavior applies only to repeated input pulses and depends on both their frequency and duty cycle.



Channel 1 (yellow/top line) – PIN#4 (IN)
 Channel 2 (light blue/2nd line) – PIN#10 (OUT)
 Channel 3 (magenta /3rd line) – PIN#12 (DFF)

1. Period is 60ns. Pulse width is 10ns DC=16.7% (Correct functionality)



SLG46580

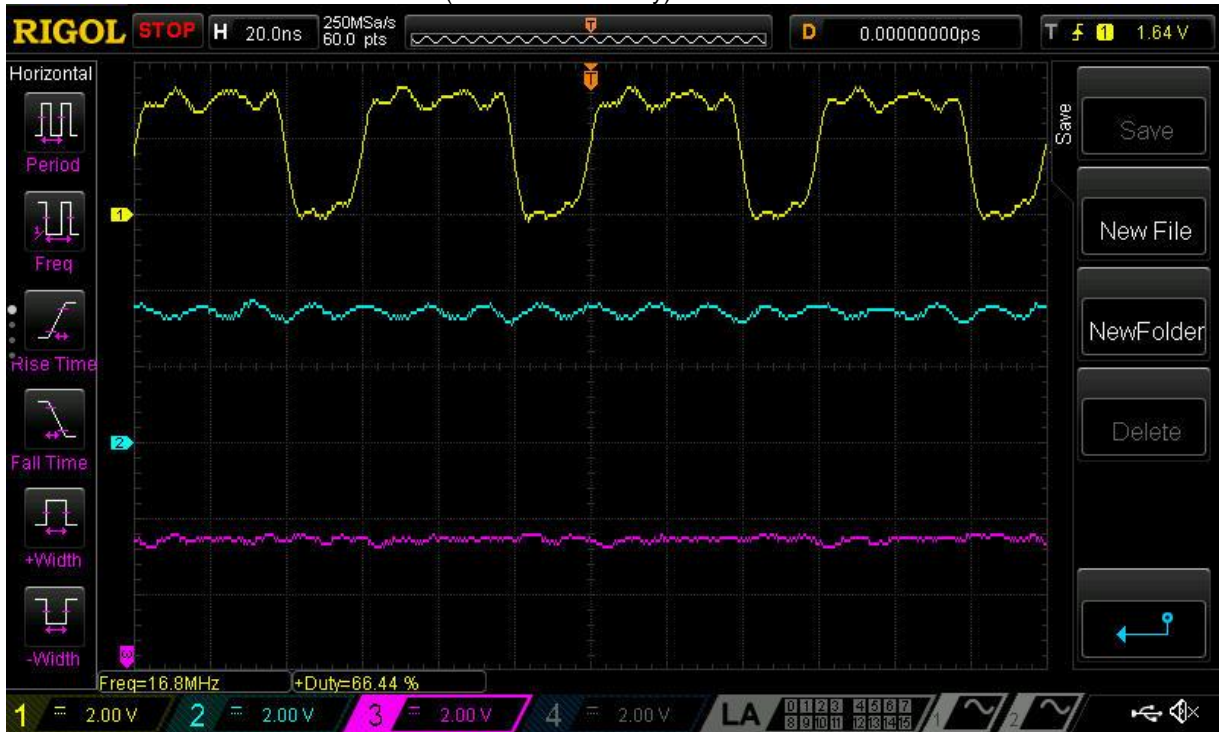
2. Period is 60ns. Pulse width is 20ns DC = 33.3% (Incorrect functionality)



3. Period is 60ns. Pulse width is 30ns DC=50% (Incorrect functionality)



4. Period is 60ns. Pulse width is 40ns DC=66.67% (Correct functionality)



Workaround:

Currently, there is no workaround for this issue. The FILTER block correctly filters isolated glitches, but it shouldn't be used to filter repetitive, high frequency input signals.

ISSUE 7: Incorrect I²C Reads of the 8-bit Counter Registers

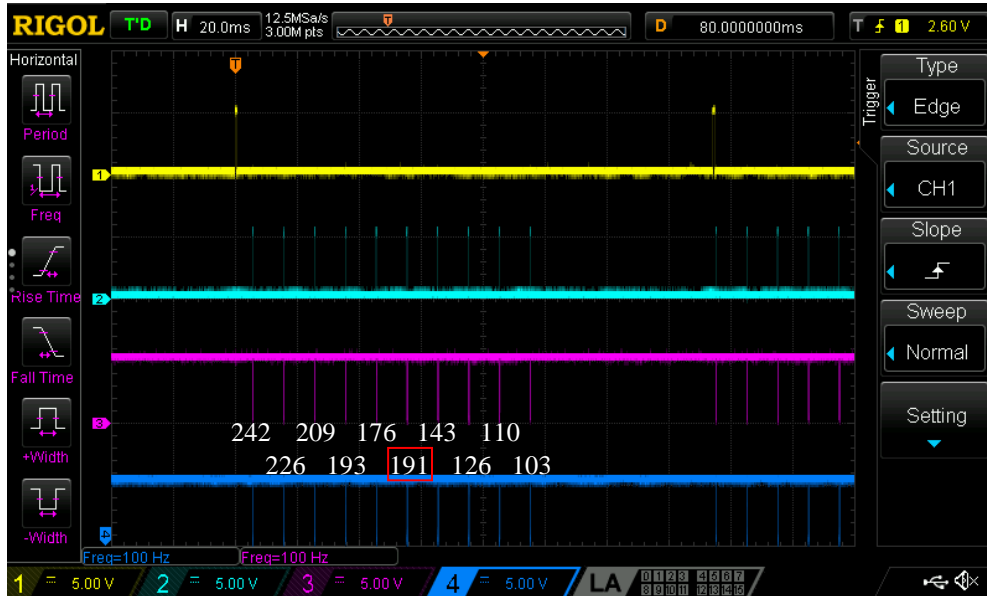
Functional Blocks Affected: CNT2/DLY2 and CNT4/DLY4

Description:

Asynchronous interaction between the CNT/DLY clock input and the I²C latch signal (generated by an I²C read command of the CNT/DLY block's count value) can result in an incorrect I²C data read. The CNT/DLY block will count accurately, but the count value transferred into the block's I²C read register might be loaded incompletely if the I²C latch signal and the clock input occur at about the same time.

The example data capture below shows ten periodic I²C reads of CNT2/DLY2 configured to count down at about 16 clocks per read. The sixth read sample erroneously shows a value greater than that of the fifth. The seventh sample reads as if the previous I²C error never occurred - the difference from the fifth sample (176) to the seventh (143) is 33 clocks or 16 clocks + 17 clocks as expected.

- Channel 1 (yellow/top line) – PIN#2 (CNT2/DLY2 Out)
- Channel 2 (light blue/2nd line) – PIN#1 (I²C Read Triggers)
- Channel 3 (magenta /3rd line) – PIN#8 (I²C SCL)
- Channel 3 (dark blue /4th line) – PIN#9 (I²C SDA)



Workaround:

If the possibility of incorrect I²C data reads can't be accommodated for by external software checks, one can guarantee proper operation by stopping the CNT/DLY block's clock during I²C reads through one of the following methods: by disabling the oscillator block, by reconfiguring the CNT/DLY block's clock source, or by gating an external clock using a LUT (Look-up Table) in the signal matrix. After disabling the CNT/DLY block's clock, the count registers can be read without error. Please note that this workaround will add the I²C read and processing time to the counter's overall clock period.

The best workaround depends on the resource constraints of the application. If the oscillator block doesn't clock other logic elements within the design, a matrix output can be used to manually power down the oscillators for the I²C read. When the CNT/DLY block's clock source is routed internally from the oscillator block, I²C commands can temporarily reconfigure the CNT/DLY block's clock source registers to select "Ext. CLK. (From Matrix)." This action will disable the clock by connecting it to ground. If the CNT/DLY block is clocked from the signal matrix, a LUT can be used to gate the clock during an I²C read.

ISSUE 8: Inaccurate Data Transfer between the RTC's Shadow Buffer and the RTC's Counter Registers

Functional Block Affected: RTC

Description:

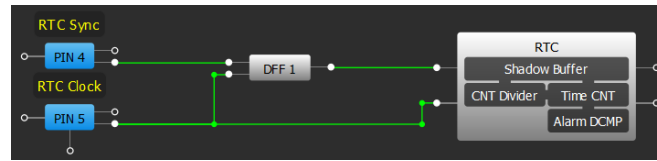
The SLG46580's I²C feature uses an internal shadow buffer to read from and write to the RTC's count registers. The data transfer between the count registers and the shadow buffer can be triggered through either the RTC block's Sync input or I²C.

Issue 2 describes an issue related to asynchronously clocking and latching data for I²C reading in various CNT/DLY blocks. Similar behavior affects the RTC block. When triggered by an I²C read, the data transfer from the counter registers to the shadow buffer should return the correct data, but when the I²C block triggers a data write to the counter registers or when the Sync input triggers the data transfer, a simultaneous rising edge on the Clock input might corrupt the data transfer.

Workaround:

As described in Issue 2, one can guarantee proper operation with I²C by disabling the clock of the RTC block during I²C reads and writes. This can be done by disabling the oscillator clocking the RTC or by gating the matrix clock using a LUT.

Alternatively, if the Sync input is used, one can synchronize the Clock and Sync inputs using a DFF as shown below. This method requires the RTC's Sync input to have an active high pulse width that exceeds 1.5 times the period of the Clock input.



ISSUE 9: Invalid I²C Data Return for Initial “Current Address Read” or “Sequential Read” after an I²C Write

Functional Block Affected: I²C

Description:

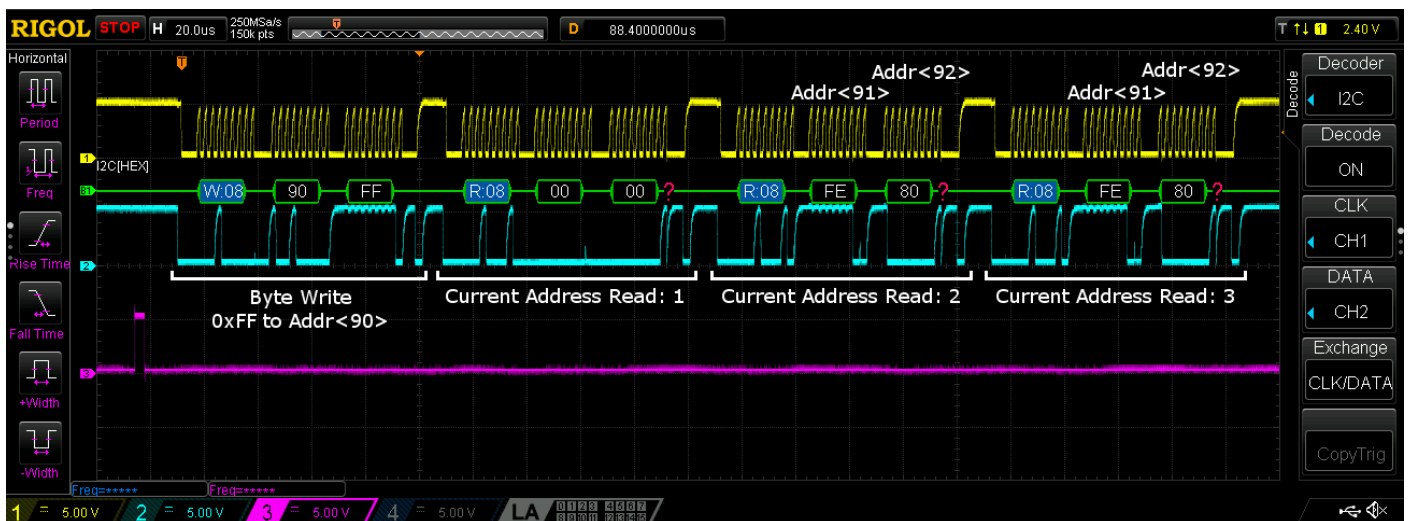
The first “Current Address Read” or “Sequential Read” command following an I²C “Byte Write” or a “Sequential Write” command will produce incorrect data. Additional read commands will return the expected data. See the waveform below for more information.

Channel 1 (yellow/top line) – PIN#8 (SCL)

Channel 2 (light blue/2nd line) – PIN#9 (SDA)

Channel 3 (magenta /3rd line) – I2C Software Trigger

Note: In the GreenPAK test design, Addr<91> and Addr<92> expect FE and 80 respectively.



Workaround:

If possible, use the “Random Read” command as described in the datasheet for SLG46580. This command will output the correct data.

If you expect consecutive reads of the same register, we recommend sending a “Random Read” command to the register preceding the register of interest. After the “Random Read” command finishes, the chip’s register pointer will increment to the desired register and the following “Current Address Read” or “Sequential Read” commands will produce the correct data. Note that the “Current Address Read” and “Sequential Read” commands don’t increment the GreenPAK’s register pointer.

ISSUE 10: ACMP Additional IN- Leakage Current

Functional Blocks Affected: ACMP, PIN

Description:

The SLG46580 has an additional leakage current through the PIN connected to the ACMP IN- input when all of the ACMPs are powered down. Typically, leakage through the PIN connected to IN- is much less than 1 μ A. But when the ACMP is powered down and voltage is applied to the PIN, the leakage current may grow up to several μ A (depending on the VDD and voltage applied).

SLG46580

Workaround:

Currently there is no workaround for this issue.

SLG46580

RoHS Compliance

Renesas Electronics Corporation's suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.