clock_divider Documentation

Table of Contents

Introduction

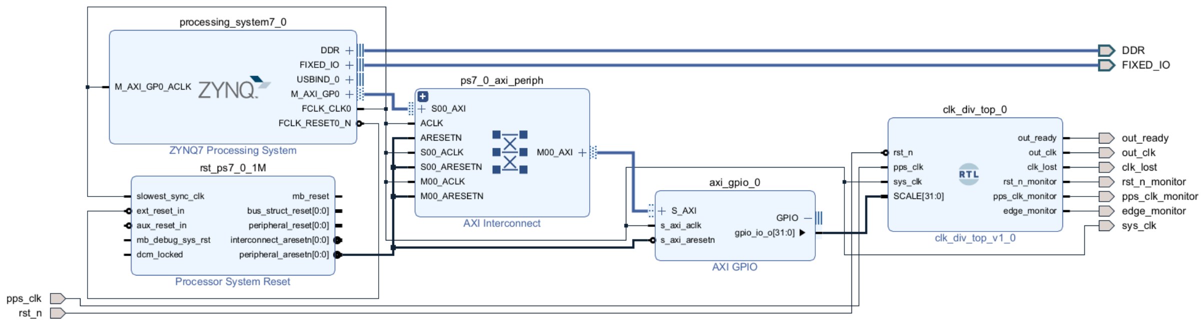
Baseline (Done 7/30/2024)

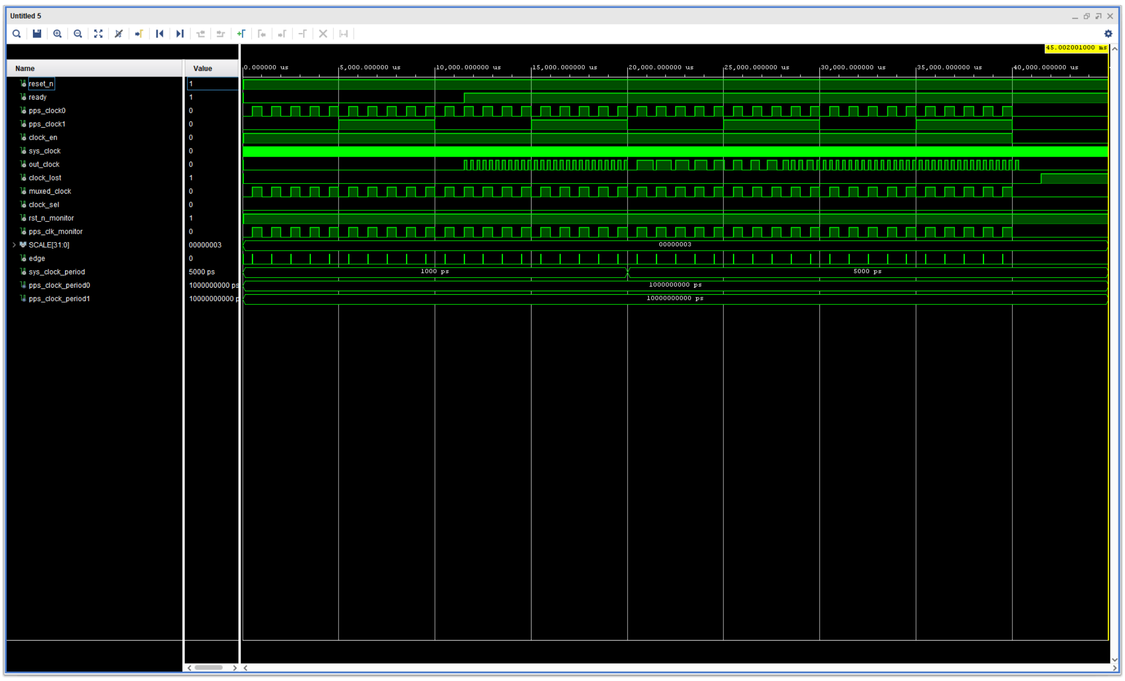
- Location: ./baseline
- Assumption: Pps_clk is absolutely accurate, sys_clk is inaccurate
- Goal: Output a clock based on sys_clk which is synced to pps_clk on every pps_clk rise edge, and can have a frequency that is multiple of pps_clk's.
- How it works:
  - divisor: Calculate **diviser** by taking average of sys_clk count of **4** windows of pps_clk right after reset (or change of **SCALE**) and finalize when the difference between two consecutive **divisor**s are within the **THRESHOLD** of **16**. Doesn't change **divisor** after finalizing and stick to it all the way.
  - synchronization: Since out_clk should only sync at pps_clk rise edge, when out_clk has a frequency of larger than 1 time of the pps_clk, out_clk and pps_clk are asynchronous to each other for all other edges.
    - To make sure out_clk is synced to pps_clk at rise edge, force out_clk to rise at pps_clk rise edge.
    - To prevent the asynchronous edges from overwriting the synching edge (when the **divisor** counts less sys_clk tiks than actual tiks, hence out_clk edge that is supposed to rise with the next pps_clk rise edge would happen before the next pps_clk rise edge), count the number of fall edges in between of two pps_clk rise edges. Force the out_clk to be 0 after **SCALE** number of falls have happened.
- SCALE: Update **SCALE** using an AXI GPIO IP. Output the desired scale from PS to PL via **axi_gpio_0**. Prompt the user to enter an unsigned integer as scale with UART.

Improvements

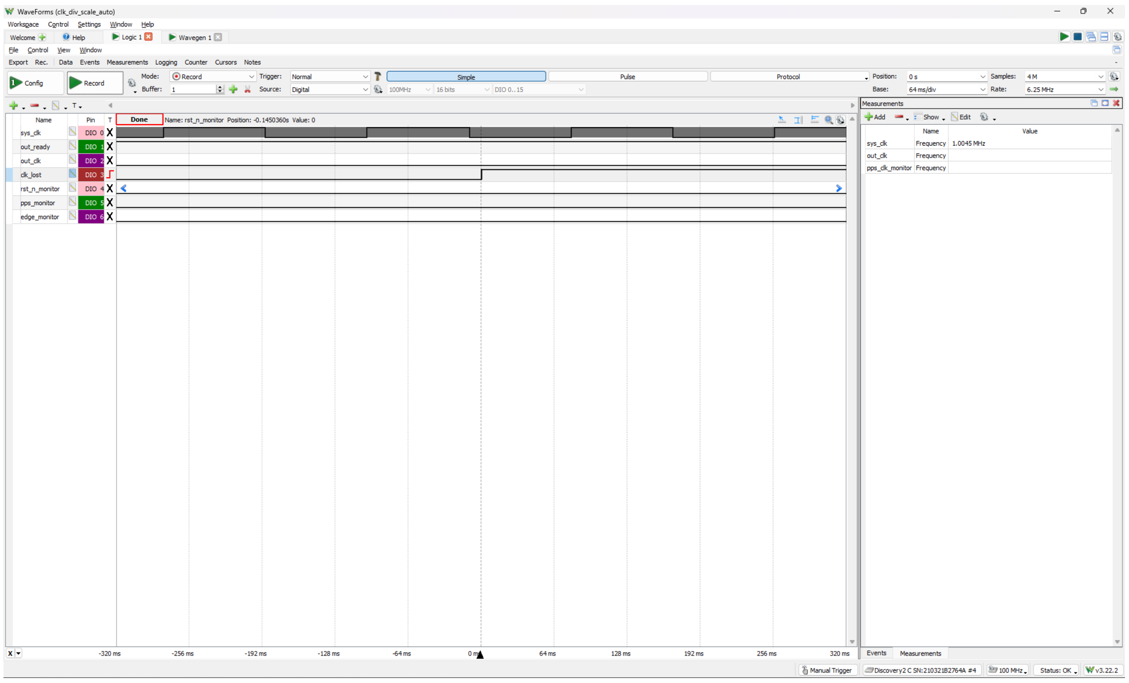
- Sources
  - Files: ./improved/files
  - Vivado Project: ./improved/vivadoProject/clk_div_scale_auto

  - Vitis Project: ./improved/vitisProject/clk_div_scale_auto



- Same assuption and goal
- Adjustments needed:
  - Updating divisor: Sometimes sys_clk may change its frequency due to change in temperature, so I need to adjust **divisor** constantly.In the previous version, divisor actually gets constantly updated, but I want to make sure that the change of divisor is constantly reflected to out_clk.
  - Clock lost & threshold: It's important to detect when the clock is lost, so a warning should be given when the **difference of two consecutive divisors is too high** while also gives some leniency for sys_clk change. Currently, I say clock is lost if the current **divisor** is at least **2 times at large** as the previous one after **out_ready** is high and different of divisors is acceptible if the difference of **divisor** is under **THRESHOLD** of **16**. I need to make sure it works.
  - More customization: Adjustable number of divisor-calculating windows and adjustable SCALE is helpful. It means I need two more 32-bit GPIOs, adjusting Vitis software, and most importantly, automate the scaling of divisor-calculating windows in PL.
- Adjustments done:
  - divisor is proven to keep being updated and is reflected to out_clk. It's hard for the logic analyzer to record that many samples, but from simulation, we can see the change in sys_clk frequency at 20ms is reflected on out_clk.
    - 

  - The maximum difference acceptable between previous and current divisor for out_clk to be ready is 16, and the minimum difference of indicating clock lost is current counter being twice as large as the previous. It works in both simulation and logic analyzer

- 
WaveForms (clk_div_scale_auto)

Mode: Record Trigger: Normal Source: Digital Buffer: 1 Position: 0 s Base: 64 ms/div Samples: 4 M Rate: 6.25 MHz

Name: rst_n_monitor Position: -0.1450360s Value: 0

Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

	Name	Value
sys_clk	Frequency	1.0045 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

  - Using arrays of variable length, I can make the number of windows variable, but only when it's a power of two because we need to divide to get the average, and we use shifting for fast division.

Details

- Pin Mapping (Bank 34):

package_pin	pin_name	JX1_pin_#	CON1_pin_#	block_pin_name	DIO_#
N/A	GND	N/A	1, 2	N/A	GND, GND
T11	JX1_LVDS_0_P	11	5	sys_clk	0
T12	JX1_LVDS_1_P	12	6	out_ready	1
T10	JX1_LVDS_0_N	13	7	out_clk	2
U12	JX1_LVDS_1_N	14	8	clk_lost	3
U13	JX1_LVDS_2_P	17	9	rst_n	W1
V12	JX1_LVDS_3_P	18	10	pps_clk	W2
V13	JX1_LVDS_2_N	19	11	rst_n_monitor	4
W13	JX1_LVDS_3_N	20	12	pps_clk_monitor	5
T14	JX1_LVDS_4_P	23	13	edge_monitor	6

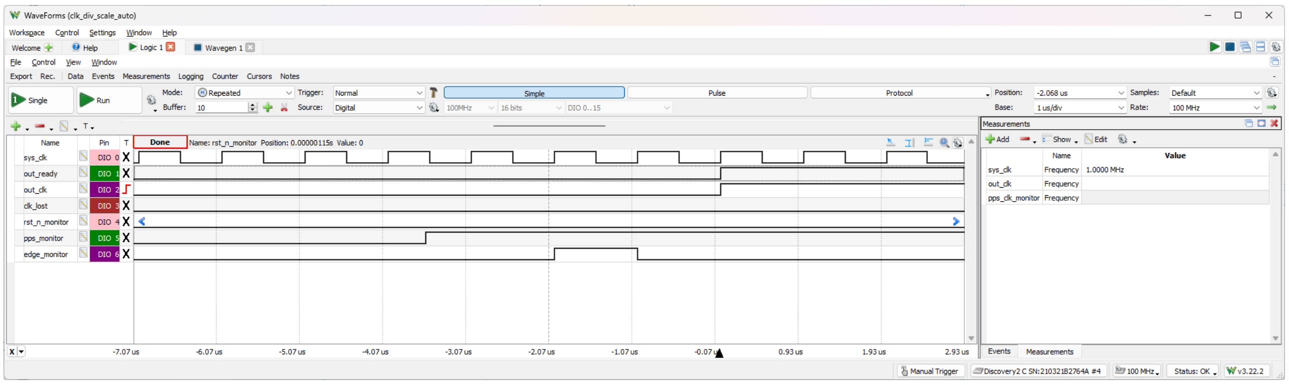
Test Result

1. pps_clk rise edge and first out_clk rise edge always have constant delay

   We can confirm that in the steady state, out_clk rises at the second sys_clk rise edge after the sys_clk rise edge that reads pps_clk high. While at the start, out_clk rises at the third sys_clk rise edge after the sys_clk rise edge that reads pps_clk high because an additional cycle is required to set up out_ready.

SCALE = 1

first R


WaveForms (clk_div_scale_auto)

Mode: Repeated Trigger: Normal Buffer: 10 Source: Digital Position: -2.068 us Base: 1 us/div Samples: Default Rate: 100 MHz

Done Name: rst_n_monitor Position: 0.00000115s Value: 0

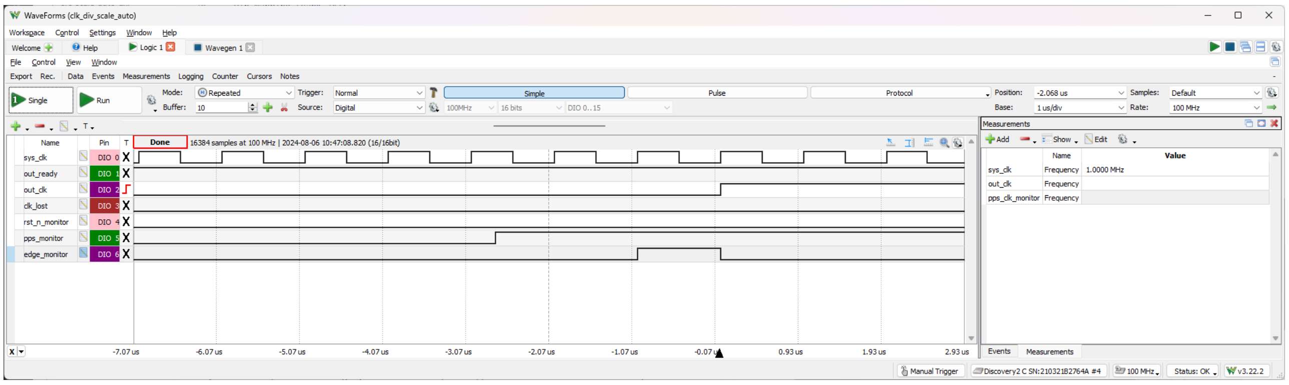
Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

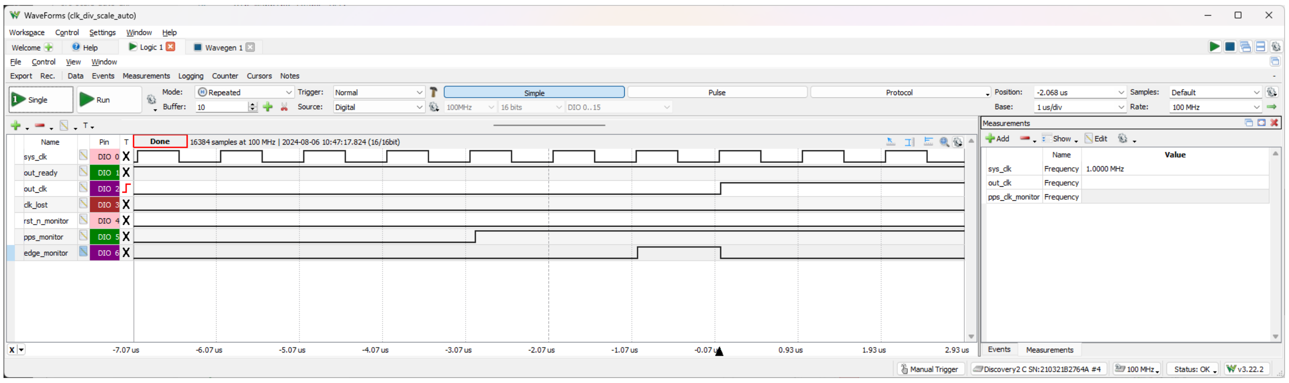
	Name	Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

Manual Trigger Discovery2 C SN:210321B2764A #4 100 MHz Status: OK v3.22.2

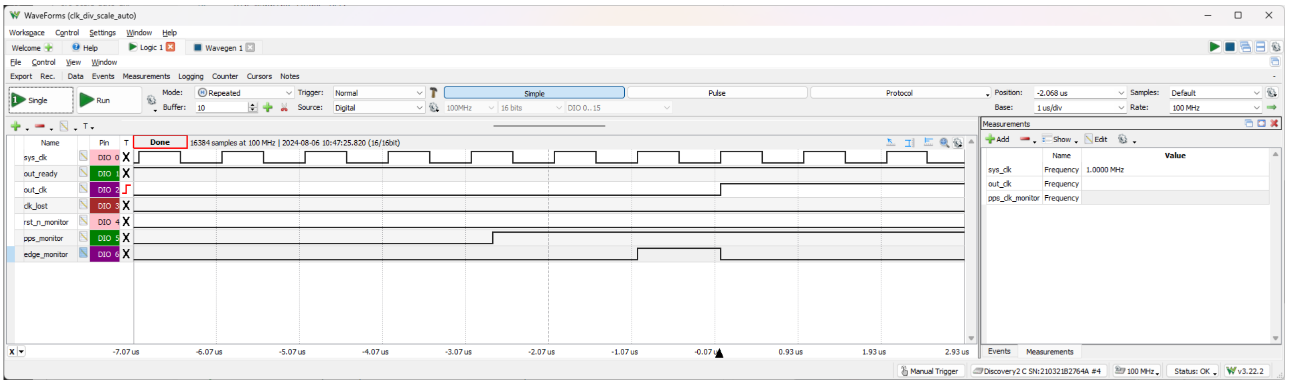
following Rs


Done 16384 samples at 100 MHz | 2024-08-06 10:47:08.820 (16/16bit)

	Name	Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	


Done 16384 samples at 100 MHz | 2024-08-06 10:47:17.824 (16/16bit)

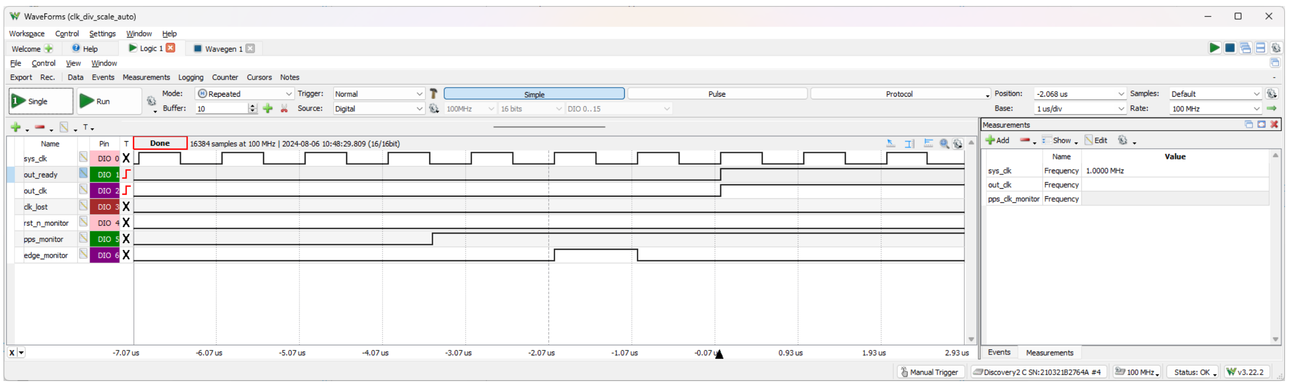
	Name	Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	


Done 16384 samples at 100 MHz | 2024-08-06 10:47:25.820 (16/16bit)

	Name	Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

SCALE = 2

first R


WaveForms (clk_div_scale_auto)

Mode: Repeated   Trigger: Normal   Position: -2.068 us   Samples: Default
Buffer: 10   Source: Digital   Base: 1 us/div   Rate: 100 MHz

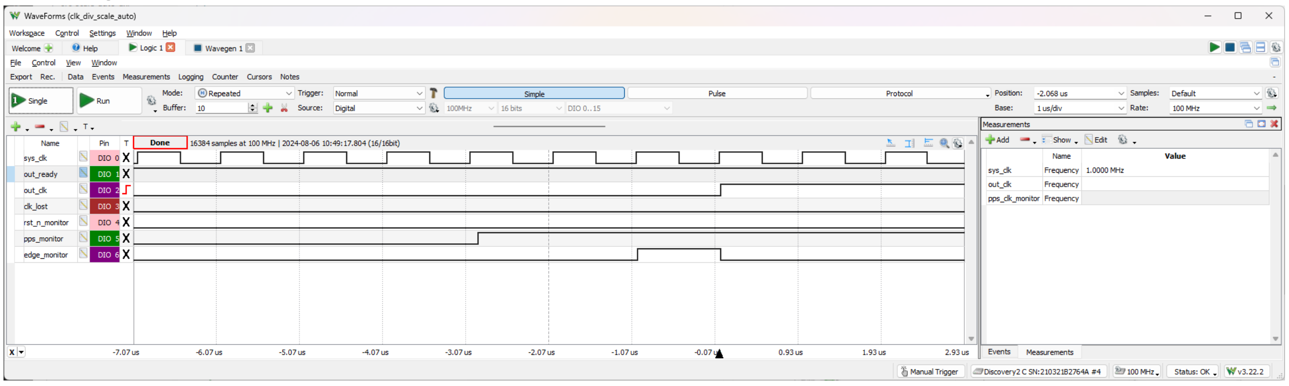
Done   16384 samples at 100 MHz | 2024-08-06 10:48:29.809 (16/16bit)

Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Name		Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

Manual Trigger   Discovery2 C SN:210321B2764A #4   100 MHz   Status: OK   v3.22.2

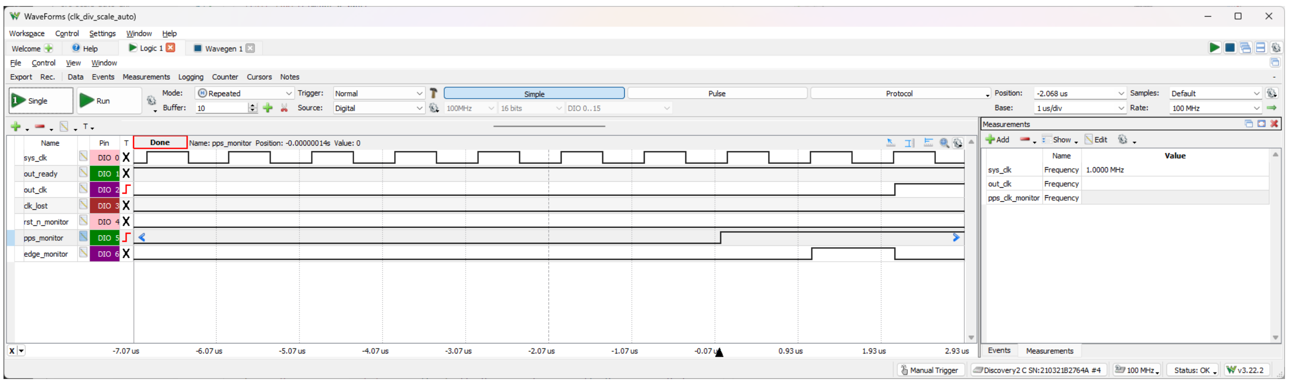
following Rs


WaveForms (clk_div_scale_auto)

Mode: Repeated   Trigger: Normal   Position: -2.068 us   Samples: Default
Buffer: 10   Source: Digital   Base: 1 us/div   Rate: 100 MHz

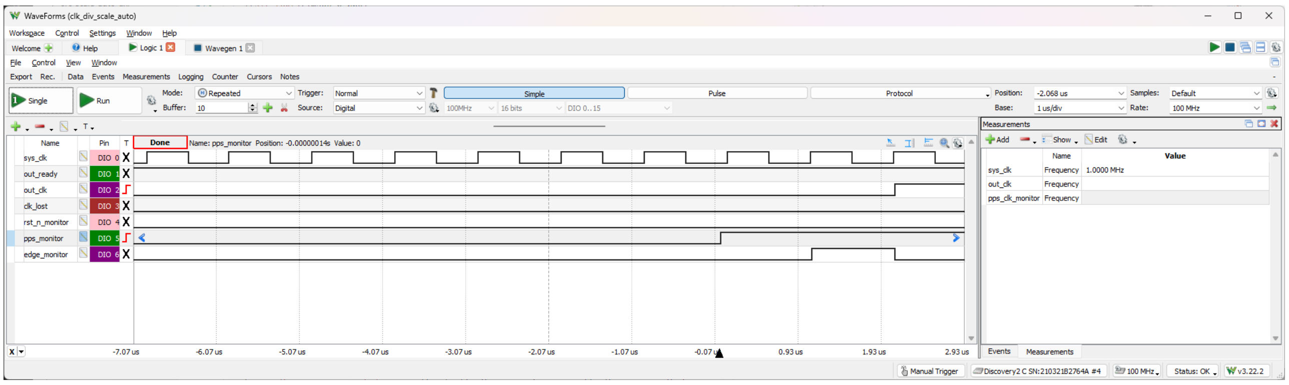
Done   16384 samples at 100 MHz | 2024-08-06 10:49:17.804 (16/16bit)

Name		Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	


WaveForms (clk_div_scale_auto)

Done   Name: pps_monitor Position: -0.00000014s Value: 0

Name		Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

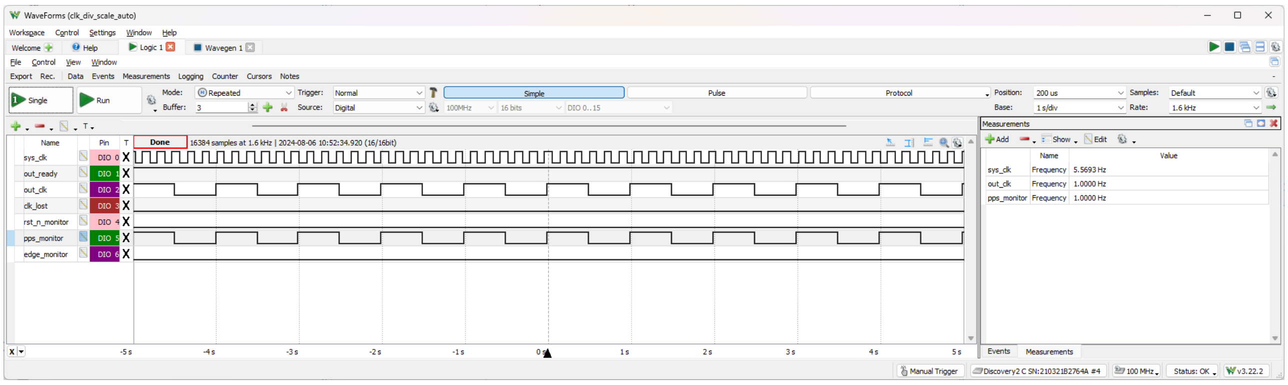

WaveForms (clk_div_scale_auto)

Done   Name: pps_monitor Position: -0.00000014s Value: 0

Name		Value
sys_clk	Frequency	1.0000 MHz
out_clk	Frequency	
pps_clk_monitor	Frequency	

2. out_clk is multiplied by the destinated amount correctly

SCALE = 1


WaveForms (clk_div_scale_auto)
Workspace Control Settings Window Help
Welcome Help Logic 1 Wavegen 1
File Control View Window
Export Rec. Data Events Measurements Logging Counter Cursors Notes
Single Run
Simple Pulse Protocol
-5 s -4 s -3 s -2 s -1 s 0 s 1 s 2 s 3 s 4 s 5 s
Events Measurements

Mode: Repeated Trigger: Normal
Buffer: 3 Source: Digital 100MHz 16 bits DIO 0..15
Position: 200 us Samples: Default
Base: 1 s/div Rate: 1.6 kHz

Done 16384 samples at 1.6 kHz | 2024-08-06 10:52:34.920 (16/16bit)

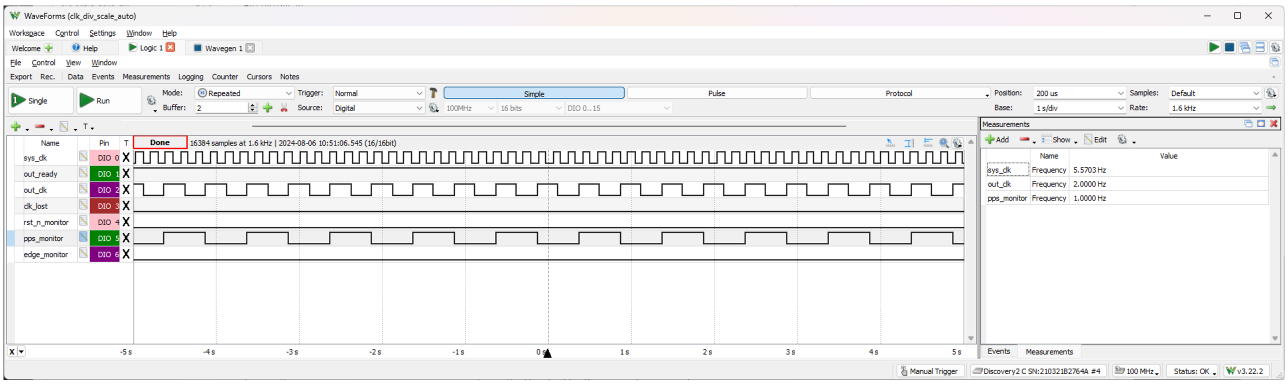
Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

	Name	Value
sys_clk	Frequency	5.5693 Hz
out_clk	Frequency	1.0000 Hz
pps_monitor	Frequency	1.0000 Hz

Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

SCALE = 2


WaveForms (clk_div_scale_auto)
Workspace Control Settings Window Help
Welcome Help Logic 1 Wavegen 1
File Control View Window
Export Rec. Data Events Measurements Logging Counter Cursors Notes
Single Run
Simple Pulse Protocol
-5 s -4 s -3 s -2 s -1 s 0 s 1 s 2 s 3 s 4 s 5 s
Events Measurements

Mode: Repeated Trigger: Normal
Buffer: 2 Source: Digital 100MHz 16 bits DIO 0..15
Position: 200 us Samples: Default
Base: 1 s/div Rate: 1.6 kHz

Done 16384 samples at 1.6 kHz | 2024-08-06 10:51:06.545 (16/16bit)

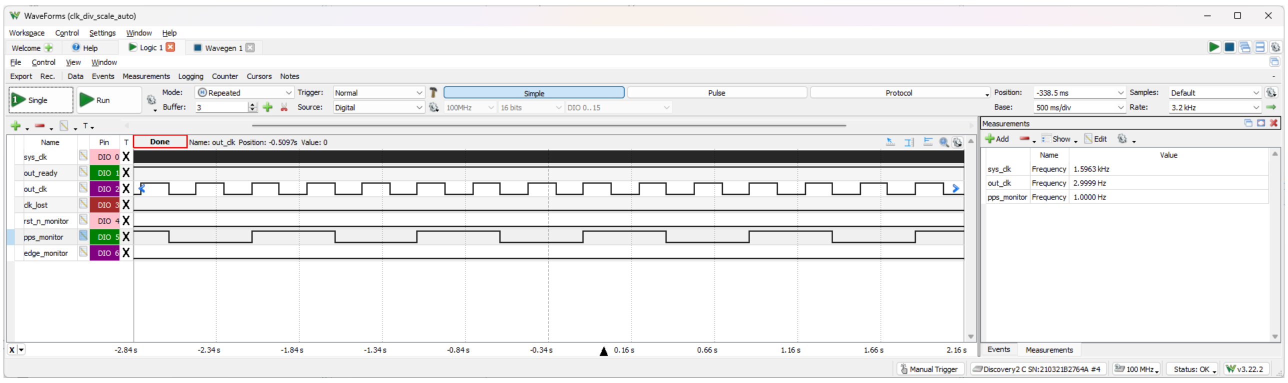
Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

	Name	Value
sys_clk	Frequency	5.5703 Hz
out_clk	Frequency	2.0000 Hz
pps_monitor	Frequency	1.0000 Hz

Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

SCALE = 3


WaveForms (clk_div_scale_auto)
Workspace Control Settings Window Help
Welcome Help Logic 1 Wavegen 1
File Control View Window
Export Rec. Data Events Measurements Logging Counter Cursors Notes
Single Run
Simple Pulse Protocol
-2.84 s -2.34 s -1.84 s -1.34 s -0.84 s -0.34 s 0.16 s 0.66 s 1.16 s 1.66 s 2.16 s
Events Measurements

Mode: Repeated Trigger: Normal
Buffer: 3 Source: Digital 100MHz 16 bits DIO 0..15
Position: -338.5 ms Samples: Default
Base: 500 ms/div Rate: 3.2 kHz

Done Name: out_clk Position: -0.5097s Value: 0

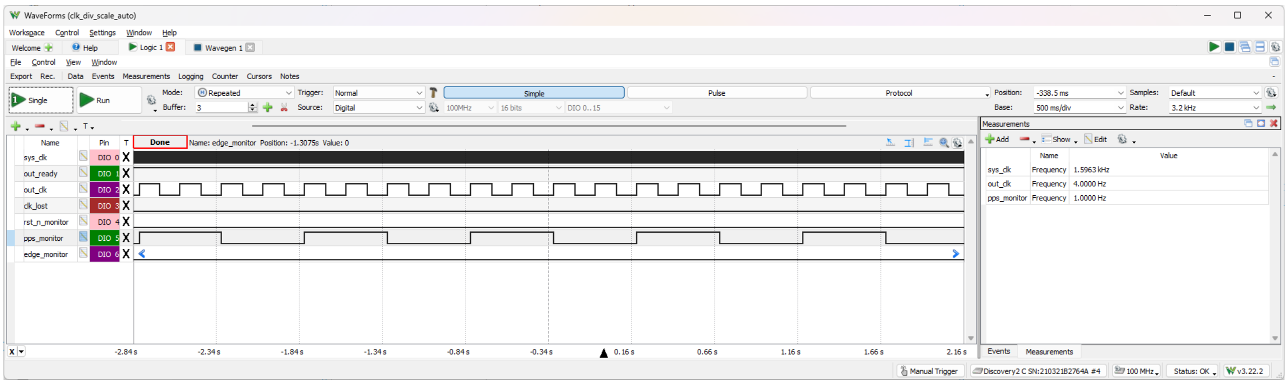
Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

	Name	Value
sys_clk	Frequency	1.5963 kHz
out_clk	Frequency	2.9999 Hz
pps_monitor	Frequency	1.0000 Hz

Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

SCALE = 4


WaveForms (clk_div_scale_auto)
Workspace Control Settings Window Help
Welcome Help Logic 1 Wavegen 1
File Control View Window
Export Rec. Data Events Measurements Logging Counter Cursors Notes
Single Run
Simple Pulse Protocol
-2.84 s -2.34 s -1.84 s -1.34 s -0.84 s -0.34 s 0.16 s 0.66 s 1.16 s 1.66 s 2.16 s
Events Measurements

Mode: Repeated Trigger: Normal
Buffer: 3 Source: Digital 100MHz 16 bits DIO 0..15
Position: -338.5 ms Samples: Default
Base: 500 ms/div Rate: 3.2 kHz

Done Name: edge_monitor Position: -1.3075s Value: 0

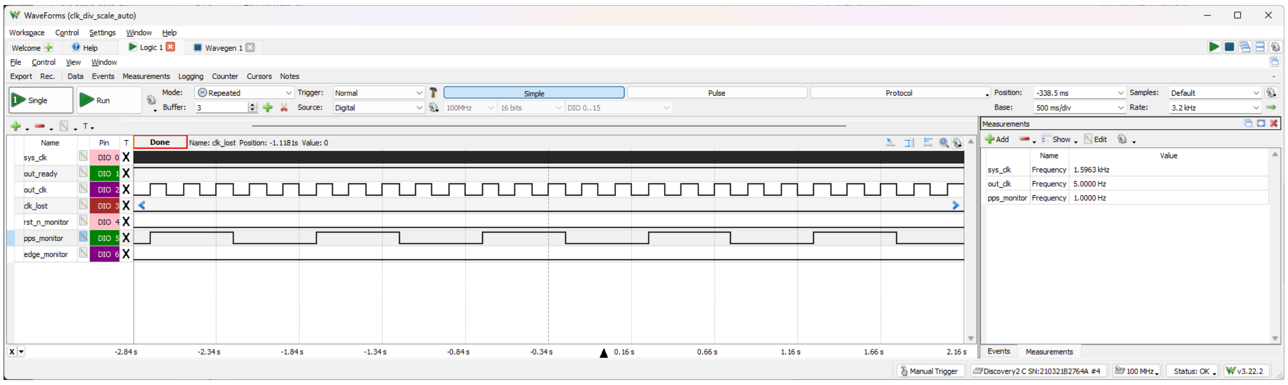
Name	Pin
sys_clk	DIO 0
out_ready	DIO 1
out_clk	DIO 2
clk_lost	DIO 3
rst_n_monitor	DIO 4
pps_monitor	DIO 5
edge_monitor	DIO 6

Measurements

	Name	Value
sys_clk	Frequency	1.5963 kHz
out_clk	Frequency	4.0000 Hz
pps_monitor	Frequency	1.0000 Hz

Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

SCALE = 5



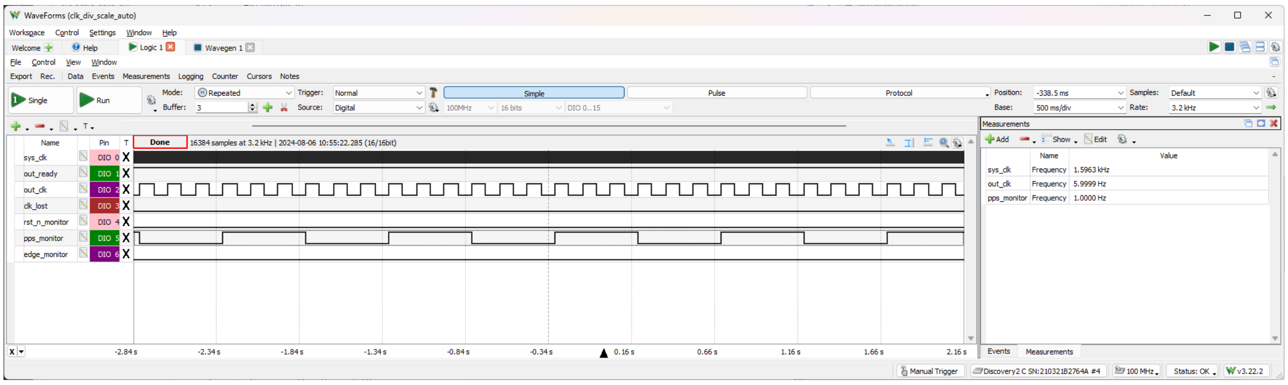
Name: clk_lost Position: -1.1181s Value: 0

Mode: Repeated
Buffer: 3
Trigger: Normal
Source: Digital
Position: -338.5 ms
Base: 500 ms/div
Samples: Default
Rate: 3.2 kHz

Measurements

	Name	Value
sys_clk	Frequency	1.5963 kHz
out_clk	Frequency	5.0000 Hz
pps_monitor	Frequency	1.0000 Hz

SCALE = 6



16384 samples at 3.2 kHz | 2024-08-06 10:55:22.285 (16/16bit)

Mode: Repeated
Buffer: 3
Trigger: Normal
Source: Digital
Position: -338.5 ms
Base: 500 ms/div
Samples: Default
Rate: 3.2 kHz

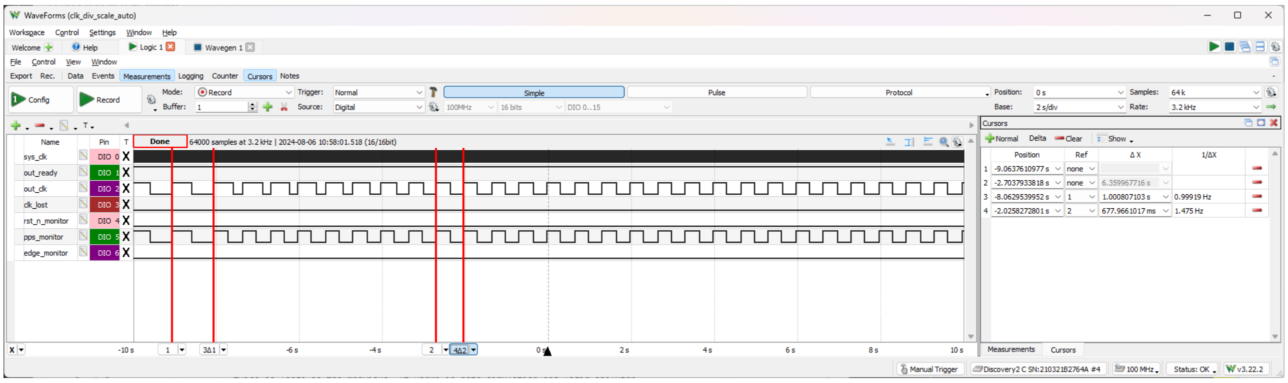
Measurements

	Name	Value
sys_clk	Frequency	1.5963 kHz
out_clk	Frequency	5.9999 Hz
pps_monitor	Frequency	1.0000 Hz

3. when sys_clk is changed, it can still sync to pps_clk with new divisor and outputs the correct clock (tested by adjusting pps_clk)

SCALE = 1

changed pps_clk from 1 Hz to 1.5 Hz



64000 samples at 3.2 kHz | 2024-08-06 10:58:01.518 (16/16bit)

Mode: Record
Buffer: 1
Trigger: Normal
Source: Digital
Position: 0 s
Base: 2 s/div
Samples: 64 k
Rate: 3.2 kHz

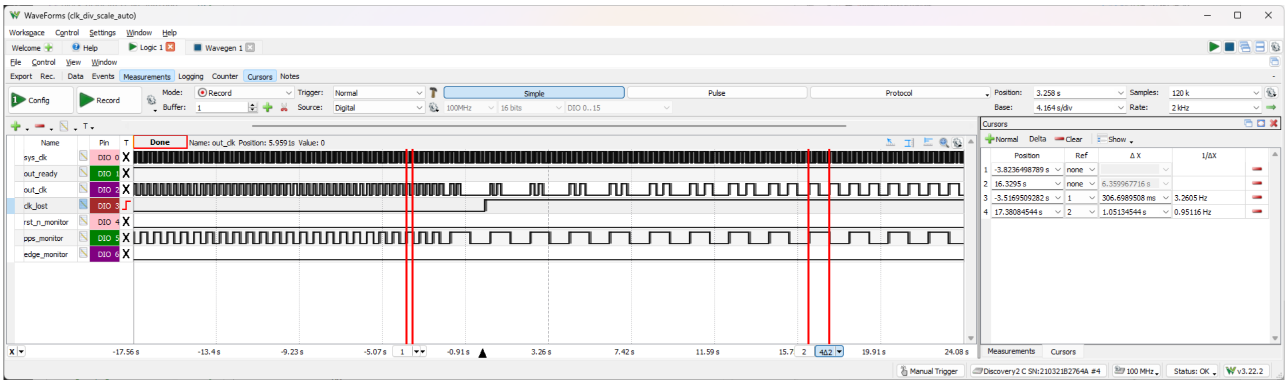
Cursors

	Position	Ref	Δ X	1/ΔX
1	-9.0637610977 s	none		
2	-2.7037933818 s	none	6.359967716 s	
3	-8.0629539952 s	1	1.000807103 s	0.99919 Hz
4	-2.0258272801 s	2	677.9661017 ms	1.475 Hz

took 8 cycles to get the new frequency

SCALE = 2

changed pps_clk from 1.5 Hz to 0.5 Hz


WaveForms (clk_div_scale_auto)

Mode: Record
Trigger: Normal
Buffer: 1
Source: Digital
Position: 3.258 s
Samples: 120 k
Base: 4.164 s/div
Rate: 2 kHz

Done Name: out_clk Position: 5.9591s Value: 0

	Position	Ref	Δ X	1/ΔX
1	-3.8236498789 s	none		
2	16.3295 s	none	6.359967716 s	
3	-3.5169509282 s	1	306.6989508 ms	3.2605 Hz
4	17.38084544 s	2	1.05134544 s	0.95116 Hz

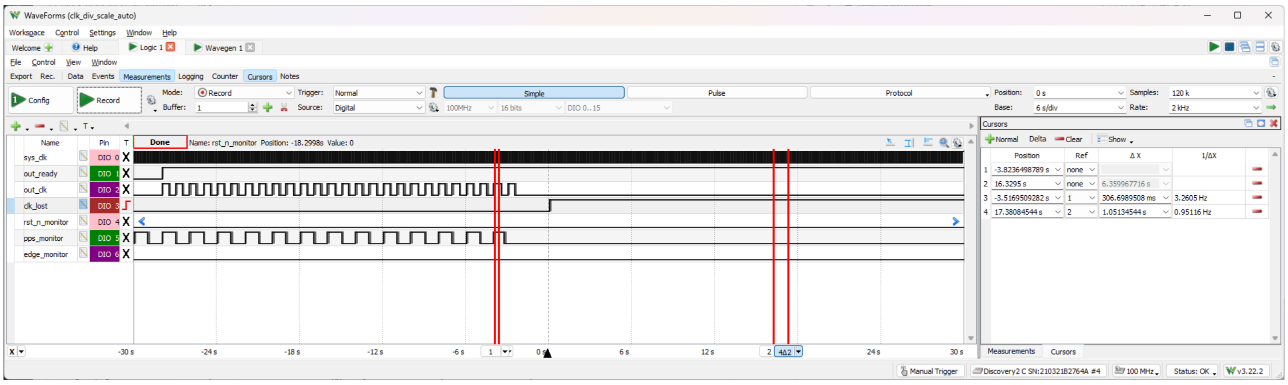
Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

clock lost is triggered because new pps_clk frequency is at least twice as slow, but eventually out clock is adjusted

4. clk_lost is triggered when pps_clk is lost

   SCALE = 2

   changed pps_clk from 0.5 Hz to off


WaveForms (clk_div_scale_auto)

Mode: Record
Trigger: Normal
Buffer: 1
Source: Digital
Position: 0 s
Samples: 120 k
Base: 6 s/div
Rate: 2 kHz

Done Name: rst_n_monitor Position: -18.2998s Value: 0

	Position	Ref	Δ X	1/ΔX
1	-3.8236498789 s	none		
2	16.3295 s	none	6.359967716 s	
3	-3.5169509282 s	1	306.6989508 ms	3.2605 Hz
4	17.38084544 s	2	1.05134544 s	0.95116 Hz

Manual Trigger | Discovery2 C SN:210321B2764A #4 | 100 MHz | Status: OK | v3.22.2

   out_clk would also stop due to lack of pps_clk

Reference

- Forums
- Guides for Xilinx Tools
  - Installing Vivado, Vitis, and Digilent Board Files
  - Getting Started with Vivado and Vitis for Baremetal Software Projects
  - Getting Started with Vivado for Hardware-Only Designs
  - Using a Peripheral with a Hierarchical Block in Vivado IPI and Vitis
  - Getting Started with Zynq Servers
  - Using Digilent Pmod IPs in Vivado and Vitis (Under Construction)
- Vitis Tutorials V2023.1
  - Versal Platform Creation Quick Start V2023.1
- Microzed and AES-MBCC-BRK-G Carrier Board
  - Microzed Hardware User Guide V1.7
  - Microzed RevB Schematic

  - MicroZed Getting Started User Guide
  - UG-AES-MBCC-BRK-G Hardware User Guide V1.2
  - MBCC-BKO RevA Schematic

- UART

  - HOW TO INSTALL CP210X VIRTUAL COM PORT DRIVERS / YAESU RADIOS
  - Tera Term application
  - Setting correct baud rate

- Embedded Design Tutorial

  - Design Example 1: Using GPIOs, Timers, and Interrupts
  - Using the GP Port in Zynq Devices
  - Vitis Unified Getting Started and Featured Tutorials
  - Vitis Unified Software Platform Documentation: Embedded Software Development (UG1400)

- Waveforms

  - time group