Preparation

KEK IPNS E-sys
Ryotaro Honda, Yun-Tsung Lai

Preparation

In this exercise, we will actually see how the signal running in FPGA.
We use these two debugging tools:

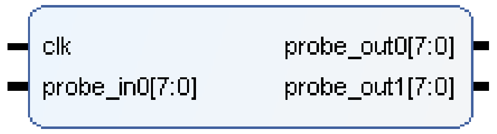
VIO
- IP to implement virtual IO to FPGA.
- Extended IO for interactive debugging.
- Add virtual push-buttons, DIPs, and virtual LEDs.

ILA (Integrated Logic Analyzer)
- Logic analyzer IP to check signals inside FPGA.
- The trigger conditions for data capture can be flexibly changed, and the signal waveform around the trigger's moment can be seen.

Considering the time of the exercise, no simulation will be performed.
Validation with simulation is important.
Utilize simulation in realistic design.

VIO

VIO's component diagram



- Max 256 input output blocks.
- (From IP catalog: up to 64)
- Probe data width: 1-256-bit。
- Synchronized to the clock。

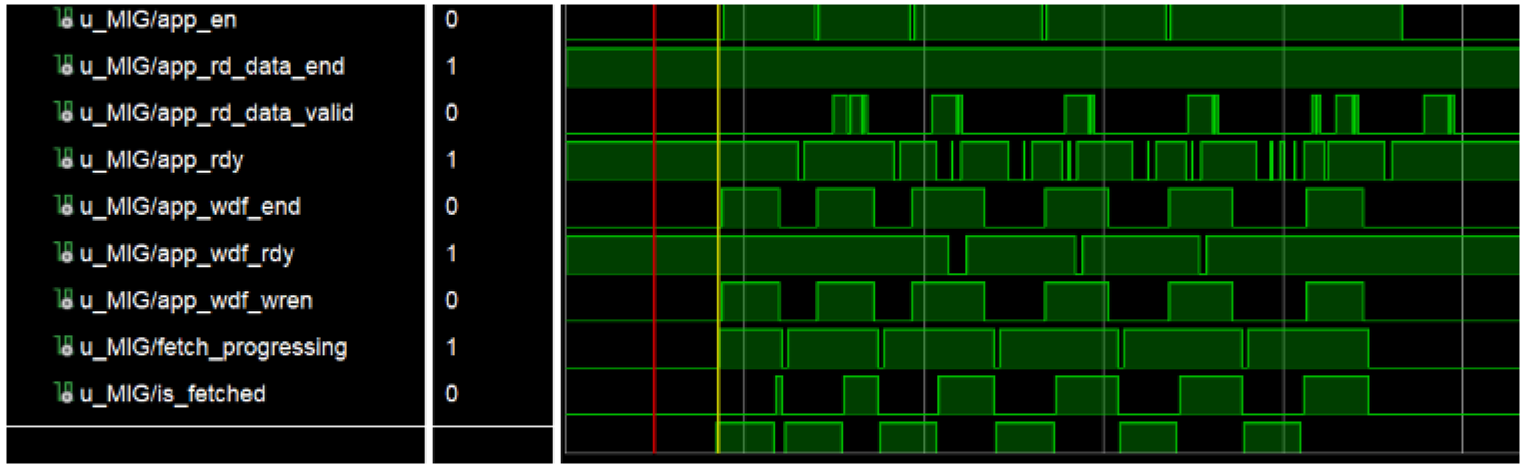
Control and monitor of the VIO is done by the hardware manager via JTAG.

ILA (Integrated Logic Analyzer)

- FPGA embeddable logic analyzer.
- Transition of the digital signal can be observed just like a real Logic analyzer.
- Conditions for data capture (trigger) can be set.
  - Complicated condition is also fine.

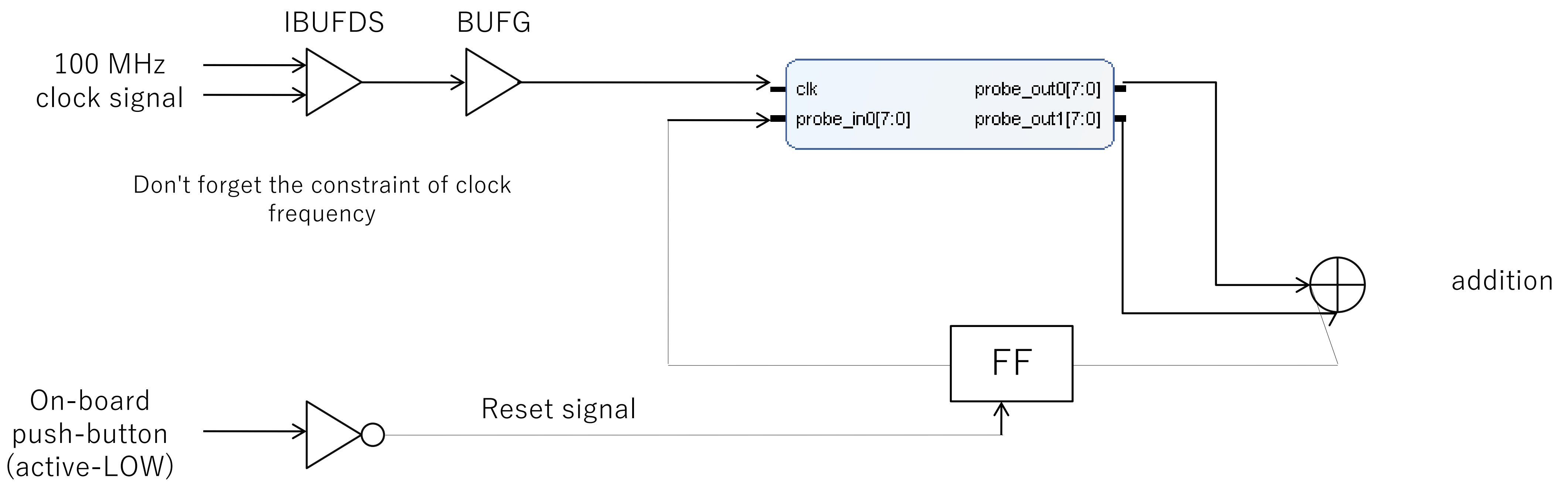
Also possible to generate ILA from the IP catalog, but this time, it will be instantiated by specifying the attribute in the source.
Just like VIO, operation is done with the hardware manager.

An example of waveform displayed by ILA

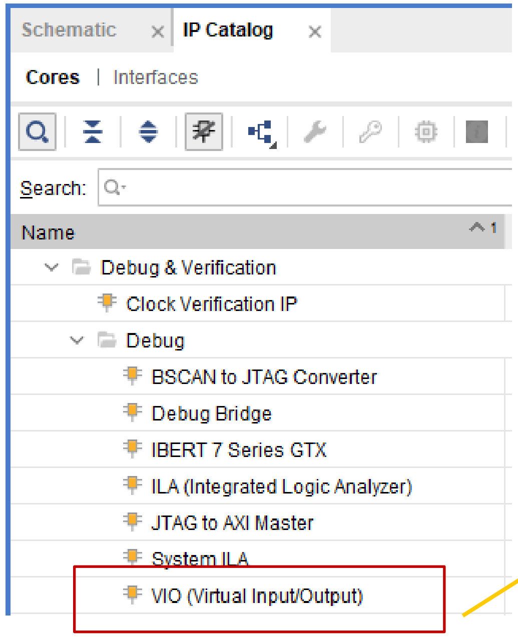


EX0 : VIO’s implementation

Implement the circuit below:



EX0 : VIO's implementation



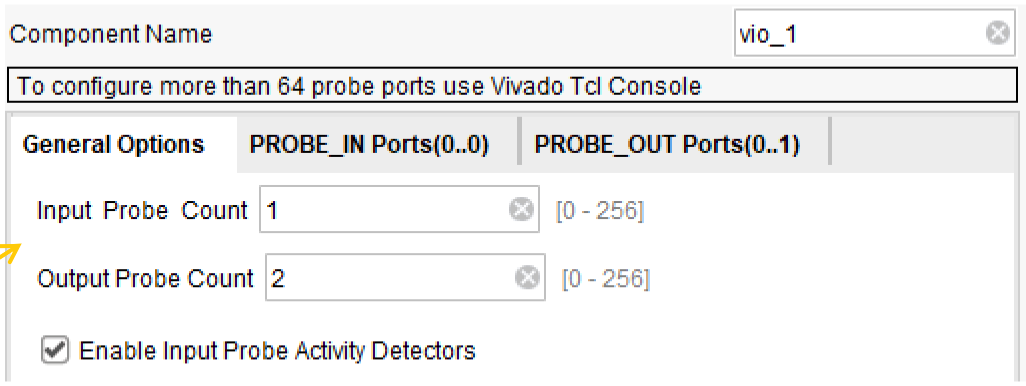
Schematic | IP Catalog

Cores | Interfaces

Search:

Name

- Debug & Verification
  - Clock Verification IP
  - Debug
    - BSCAN to JTAG Converter
    - Debug Bridge
    - IBERT 7 Series GTX
    - ILA (Integrated Logic Analyzer)
    - JTAG to AXI Master
    - System ILA
    - VIO (Virtual Input/Output)



Component Name: vio_1

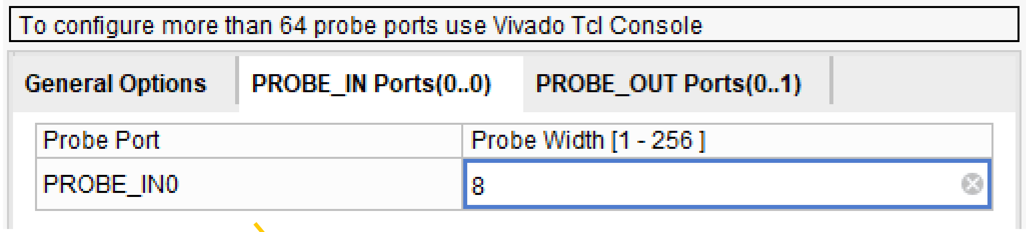
To configure more than 64 probe ports use Vivado Tcl Console

General Options | PROBE_IN Ports(0..0) | PROBE_OUT Ports(0..1)

Input Probe Count: 1 [0 - 256]

Output Probe Count: 2 [0 - 256]

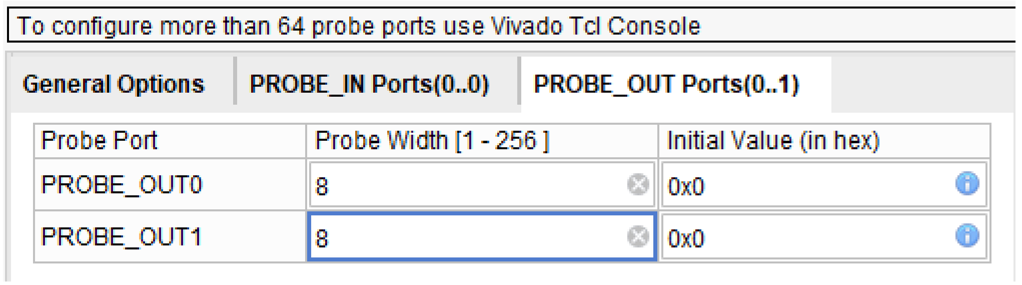
[✓] Enable Input Probe Activity Detectors



To configure more than 64 probe ports use Vivado Tcl Console

General Options | PROBE_IN Ports(0..0) | PROBE_OUT Ports(0..1)

Probe Port	Probe Width [1 - 256]
PROBE_IN0	8



To configure more than 64 probe ports use Vivado Tcl Console

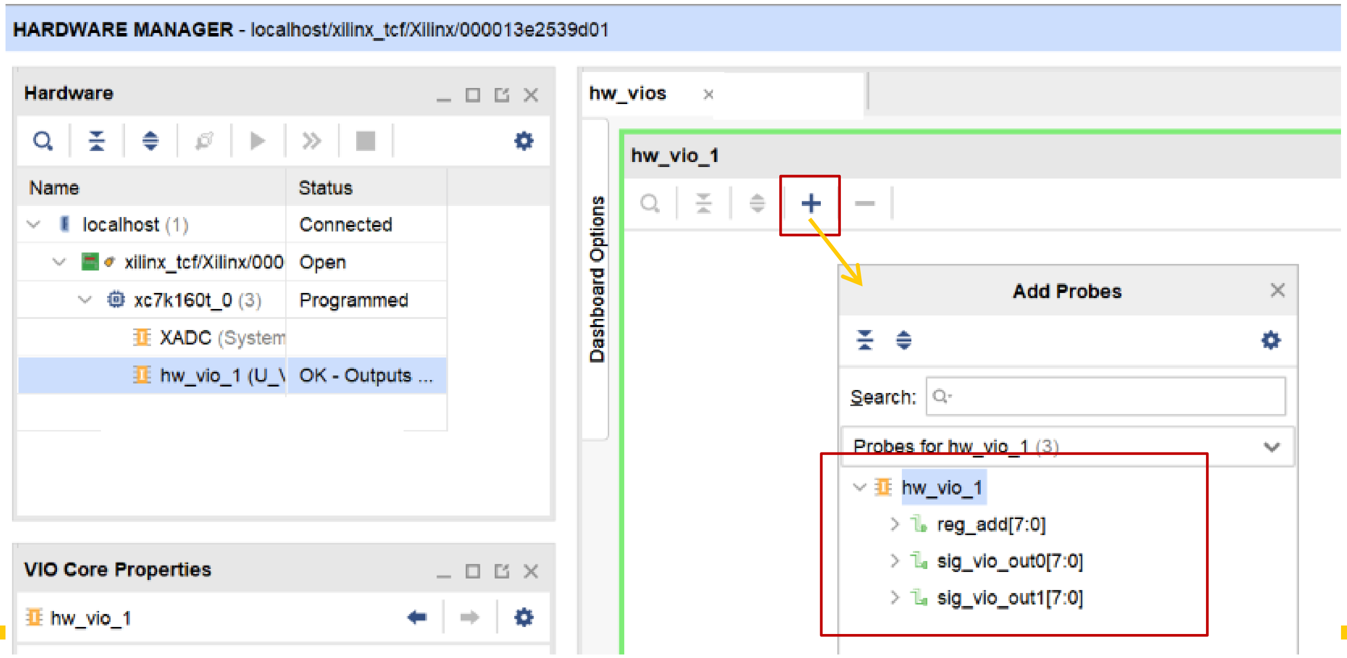
General Options | PROBE_IN Ports(0..0) | PROBE_OUT Ports(0..1)

Probe Port	Probe Width [1 - 256]	Initial Value (in hex)
PROBE_OUT0	8	0x0
PROBE_OUT1	8	0x0

EX0 : VIO's implementation

- After HDL is written, directly generate the bit stream.
  - At the synthesis result, you will notice that something called "dbg_hub" is embedded without permission.
  - VIO and ILA communicate with external side via this hub.

- Start Hardware manager, and write both .bit file and .ltx file to FPGA.
  - .ltx file has the embedded debugging tool's information.
  - If you forget it, you cannot access it with the hardware manager.



HARDWARE MANAGER - localhost/xilinx_tcf/Xilinx/000013e2539d01

Hardware

Name	Status
localhost (1)	Connected
xilinx_tcf/Xilinx/000	Open
xc7k160t_0 (3)	Programmed
XADC (System	
hw_vio_1 (U_\	OK - Outputs ...

VIO Core Properties

hw_vio_1

hw_vios

hw_vio_1

Dashboard Options

Add Probes

Search:

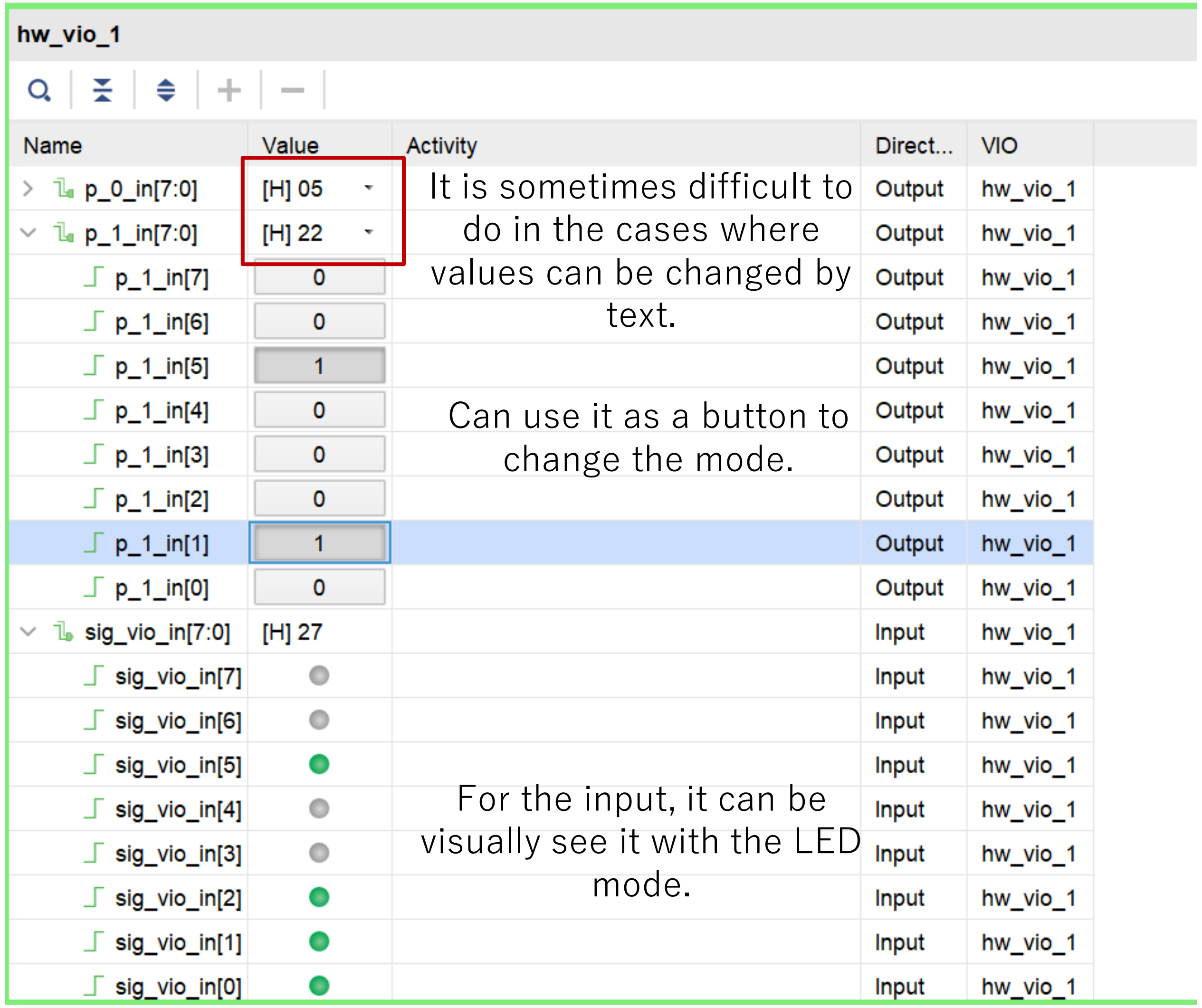
Probes for hw_vio_1 (3)

- hw_vio_1
  - reg_add[7:0]
  - sig_vio_out0[7:0]
  - sig_vio_out1[7:0]

Such display should be shown.

Select the VIO probes to display.

EX0 : VIO's implementation



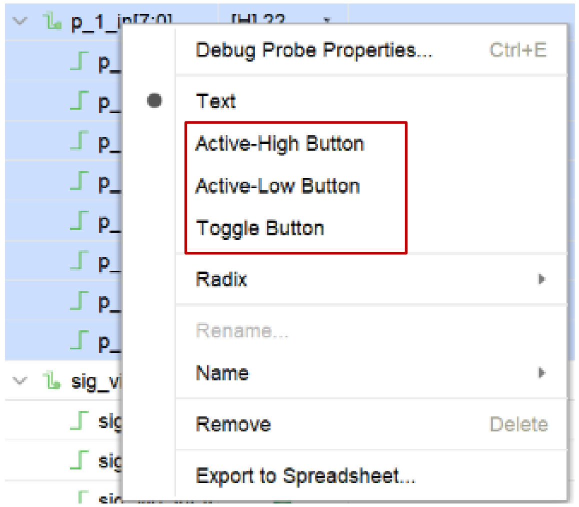
hw_vio_1

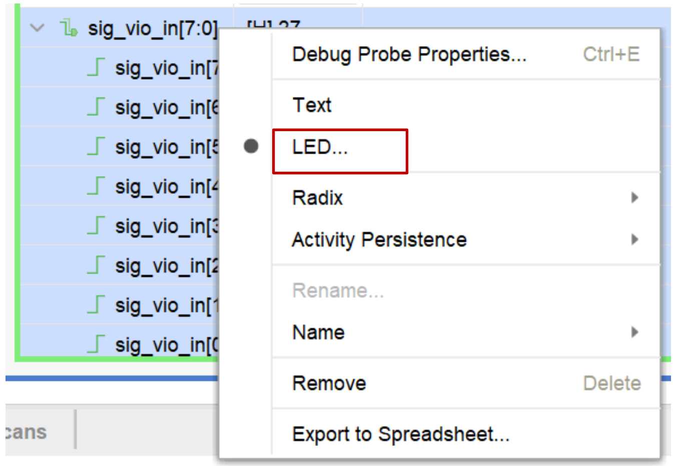
Name	Value	Activity	Direct...	VIO
p_0_in[7:0]	[H] 05		Output	hw_vio_1
p_1_in[7:0]	[H] 22		Output	hw_vio_1
p_1_in[7]	0		Output	hw_vio_1
p_1_in[6]	0		Output	hw_vio_1
p_1_in[5]	1		Output	hw_vio_1
p_1_in[4]	0		Output	hw_vio_1
p_1_in[3]	0		Output	hw_vio_1
p_1_in[2]	0		Output	hw_vio_1
p_1_in[1]	1		Output	hw_vio_1
p_1_in[0]	0		Output	hw_vio_1
sig_vio_in[7:0]	[H] 27		Input	hw_vio_1
sig_vio_in[7]	⚪		Input	hw_vio_1
sig_vio_in[6]	⚪		Input	hw_vio_1
sig_vio_in[5]	🟢		Input	hw_vio_1
sig_vio_in[4]	⚪		Input	hw_vio_1
sig_vio_in[3]	⚪		Input	hw_vio_1
sig_vio_in[2]	🟢		Input	hw_vio_1
sig_vio_in[1]	🟢		Input	hw_vio_1
sig_vio_in[0]	🟢		Input	hw_vio_1

It is sometimes difficult to do in the cases where values can be changed by text.

Can use it as a button to change the mode.

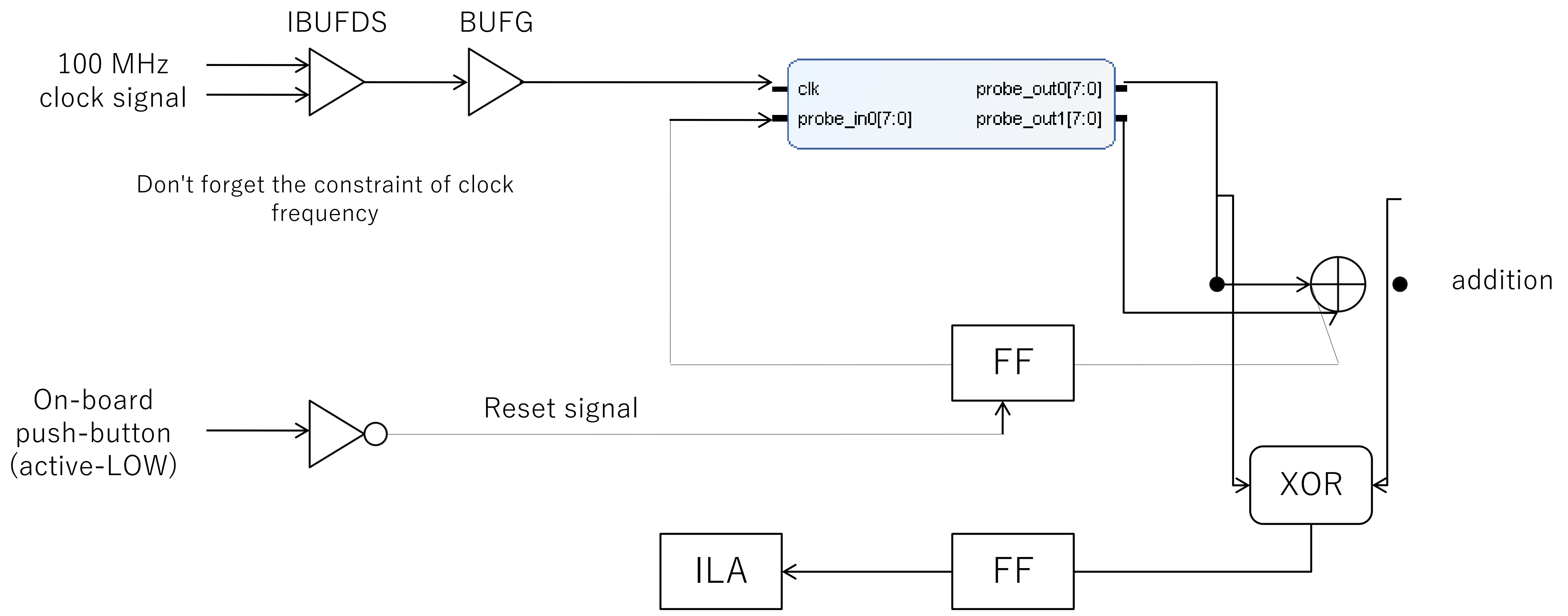
For the input, it can be visually see it with the LED mode.





EX: ILA implementation

Implement the circuit below:



Try to add this part

EX: ILA implementation

In the HLD code, use Vivado's function to route to the ILA by specifying attributes

Verilog

Declare a signal with mark_debug

```verilog
(* mark_debug = "true" *) reg [7:0]counter;
```

```verilog
always@(posedge clk_sys) begin
   if(rst_sys) begin
      counter[7:0]   <= 8'd0;
   end else begin
      counter[7:0]   <= counter[7:0] + 8'd1;
   end
end
```

VHDL

Specify attributes in the declaration part (before begin)

```vhdl
signal reg_xor         : std_logic_vector(7 downto 0);

-- debug
attribute mark_debug  : string;
attribute mark_debug of reg_xor  : signal is "true";
```

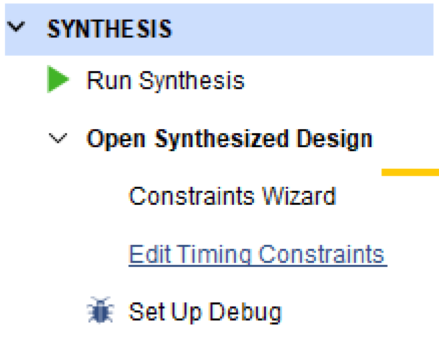
```vhdl
U_BUF : process(rst_sys, clk_sys)
begin
  if(rst_sys = '1') then
    reg_add   <= (others => '0');
  elsif(clk_sys'event and clk_sys = '1') then
    reg_add   <= std_logic_vector(signed(sig_vio_out0)
                                + signed(sig_vio_out1));

    reg_xor   <= sig_vio_out0 xor sig_vio_out1;
  end if;
end process;
```

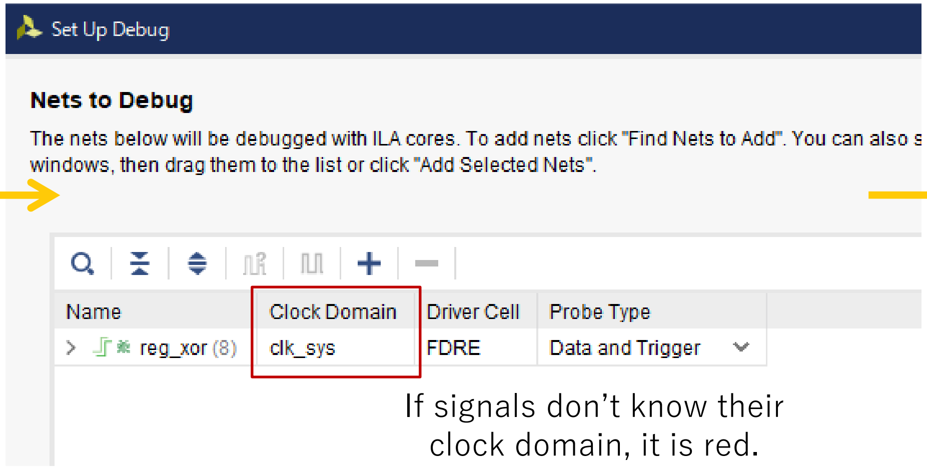
The place where the actual assignment to reg_xor is written
(do not write anything here)

EX: ILA implementation

Electronics System Group

Just do synthesis as it is, the debug core is embedded in the synthesized design.



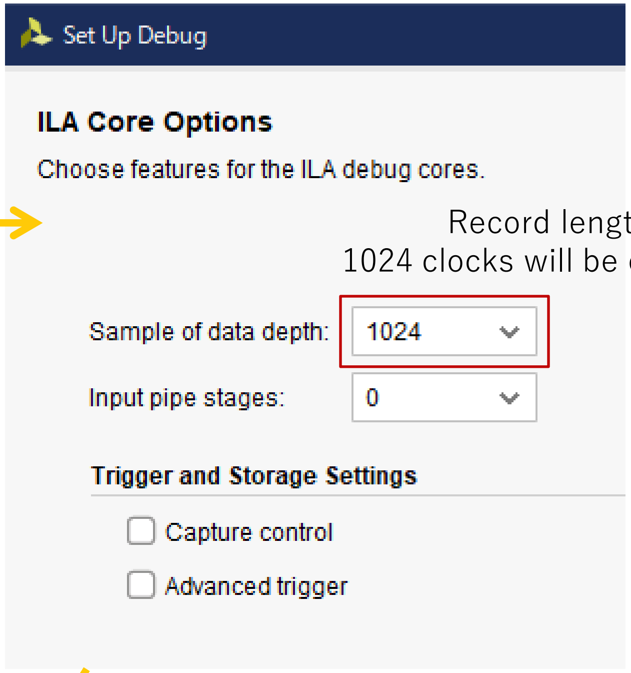


Nets to Debug

The nets below will be debugged with ILA cores. To add nets click "Find Nets to Add". You can also s
windows, then drag them to the list or click "Add Selected Nets".

Name	Clock Domain	Driver Cell	Probe Type
reg_xor (8)	clk_sys	FDRE	Data and Trigger

If signals don't know their clock domain, it is red. Has to specify.



ILA Core Options

Choose features for the ILA debug cores.

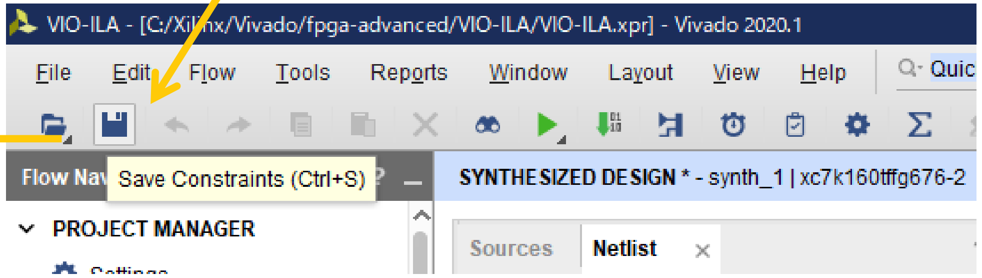
Record length. 1024 clocks will be captured.

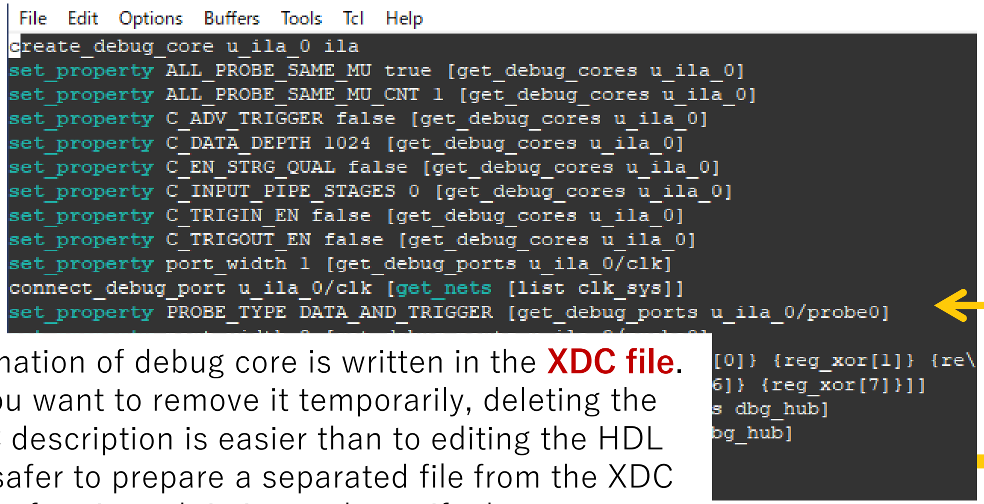
Sample of data depth: 1024

Input pipe stages: 0

Trigger and Storage Settings

- [ ] Capture control
- [ ] Advanced trigger


VIO-ILA - [C:/Xilinx/Vivado/fpga-advanced/VIO-ILA/VIO-ILA.xpr] - Vivado 2020.1
File
Edit
Flow
Tools
Reports
Window
Layout
View
Help
Quic
Flow Nav
Save Constraints (Ctrl+S)
SYNTHESIZED DESIGN * - synth_1 | xc7k160tffg676-2
PROJECT MANAGER
Settings
Sources
Netlist

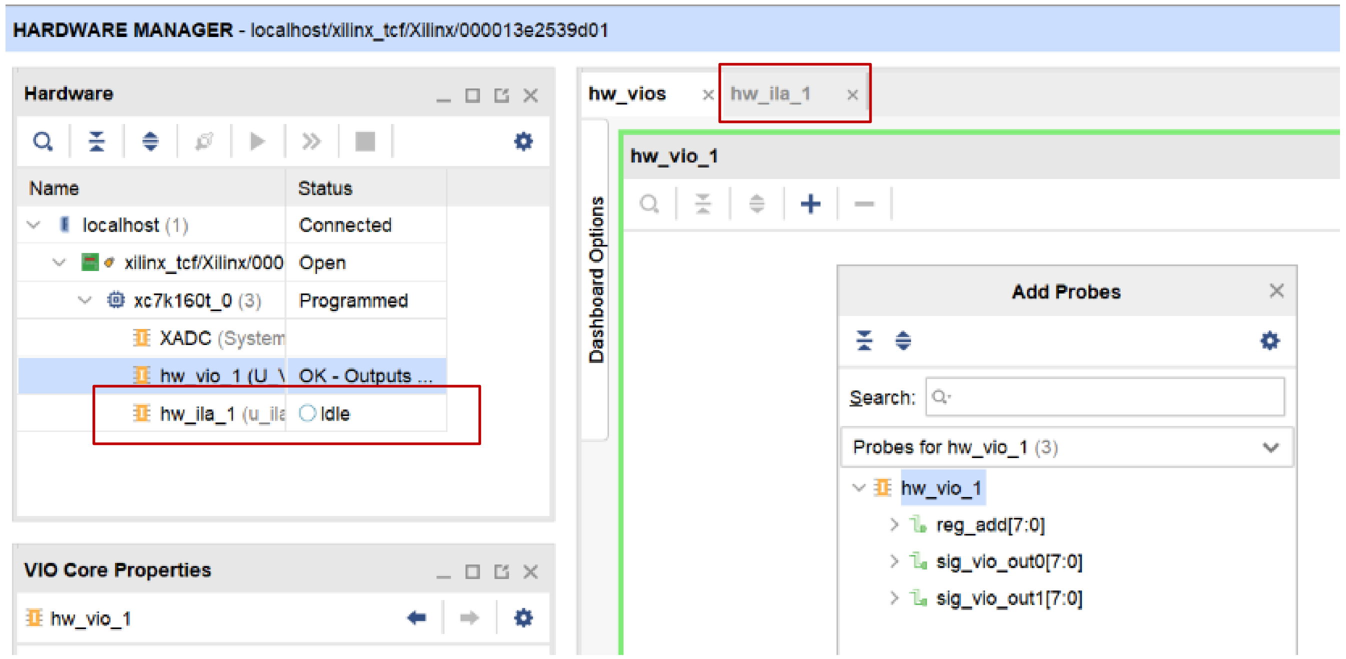


```
create_debug_core u_ila_0 ila
set_property ALL_PROBE_SAME_MU true [get_debug_cores u_ila_0]
set_property ALL_PROBE_SAME_MU_CNT 1 [get_debug_cores u_ila_0]
set_property C_ADV_TRIGGER false [get_debug_cores u_ila_0]
set_property C_DATA_DEPTH 1024 [get_debug_cores u_ila_0]
set_property C_EN_STRG_QUAL false [get_debug_cores u_ila_0]
set_property C_INPUT_PIPE_STAGES 0 [get_debug_cores u_ila_0]
set_property C_TRIGIN_EN false [get_debug_cores u_ila_0]
set_property C_TRIGOUT_EN false [get_debug_cores u_ila_0]
set_property port_width 1 [get_debug_ports u_ila_0/clk]
connect_debug_port u_ila_0/clk [get_nets [list clk_sys]]
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe0]
                                                 [0]} {reg_xor[1]} {re\
                                                 6]} {reg_xor[7]}]]
                                                 s dbg_hub]
                                                 bg_hub]
```

Information of debug core is written in the **XDC file**.
If you want to remove it temporarily, deleting the XDC description is easier than to editing the HDL
It is safer to prepare a separated file from the XDC file for pin and timing and specify the target.

EX: ILA implementation

- XDC has been updated, but the constraints are read after the synthesis is done. Therefore, we can move on to Place & Route.
- Also, launch the hardware manager, and write both the .bit and .ltx files to the FPGA.



HARDWARE MANAGER - localhost/xilinx_tcf/Xilinx/000013e2539d01

Hardware

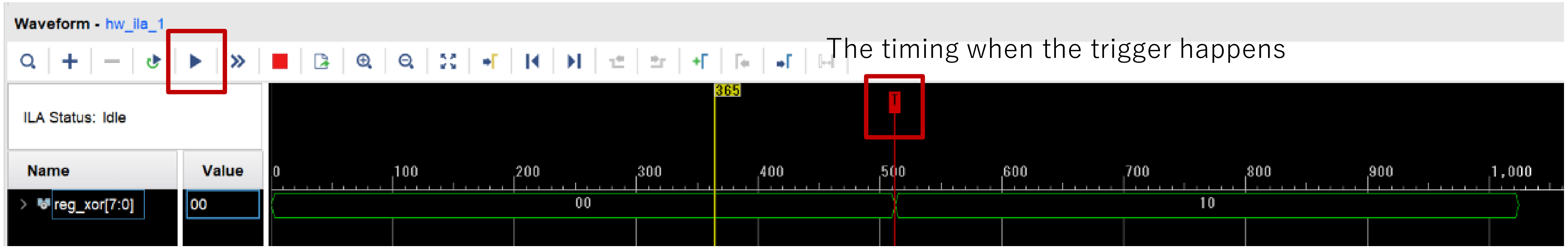
Name	Status
localhost (1)	Connected
xilinx_tcf/Xilinx/000	Open
xc7k160t_0 (3)	Programmed
XADC (System	
hw_vio_1 (U_V	OK - Outputs ...
hw_ila_1 (u_ila	Idle

VIO Core Properties

hw_vio_1

EX: ILA implementation

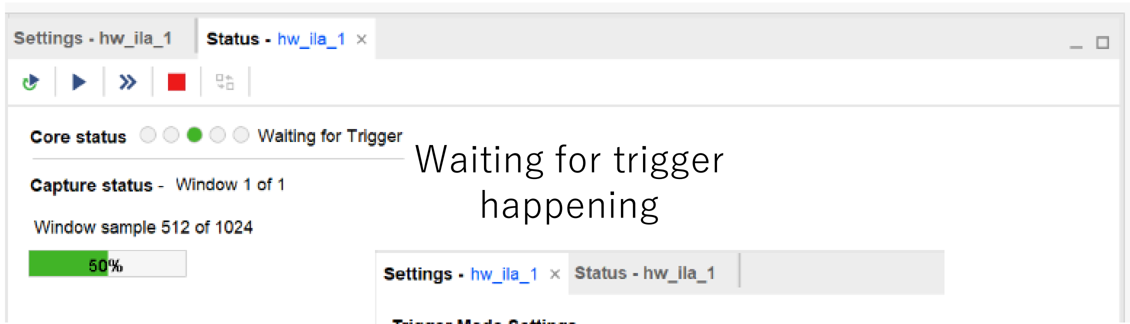
Start capture


Waveform - hw_ila_1

ILA Status: Idle

Name	Value
reg_xor[7:0]	00

The timing when the trigger happens


Settings - hw_ila_1 Status - hw_ila_1

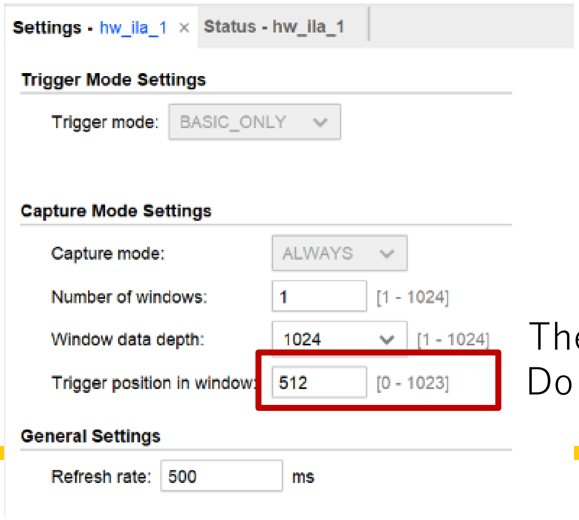
Core status Waiting for Trigger

Capture status - Window 1 of 1

Window sample 512 of 1024

50%

Waiting for trigger happening


Settings - hw_ila_1 Status - hw_ila_1

Trigger Mode Settings

Trigger mode: BASIC_ONLY

Capture Mode Settings

Capture mode: ALWAYS

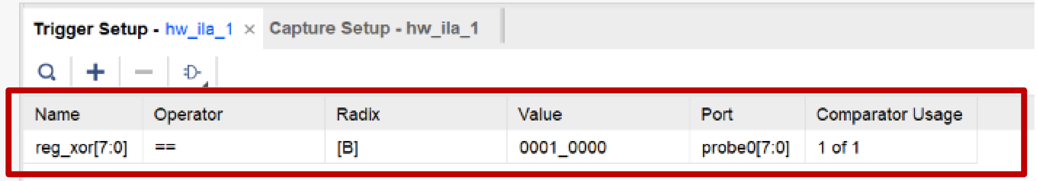
Number of windows: 1 [1 - 1024]

Window data depth: 1024 [1 - 1024]

Trigger position in window: 512 [0 - 1023]

General Settings

Refresh rate: 500 ms


Trigger Setup - hw_ila_1 Capture Setup - hw_ila_1

Name	Operator	Radix	Value	Port	Comparator Usage
reg_xor[7:0]	==	[B]	0001_0000	probe0[7:0]	1 of 1

Trigger condition: reg_xor=0x10

If no trigger condition, the waveform will be shown immediately.

The position of the trigger condition:
Do you want to see the signal before or after the trigger?