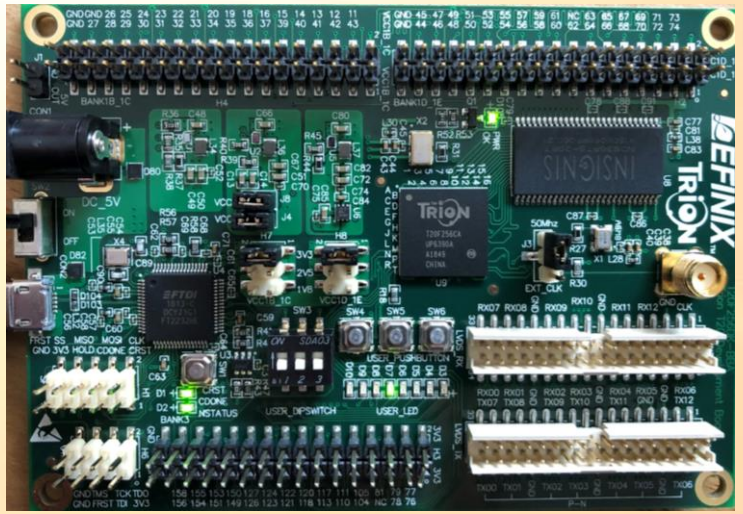
EFINIX®

***Accelerating Your Innovation***

# Trion T20 BGA256 LAB

By Harald Werner

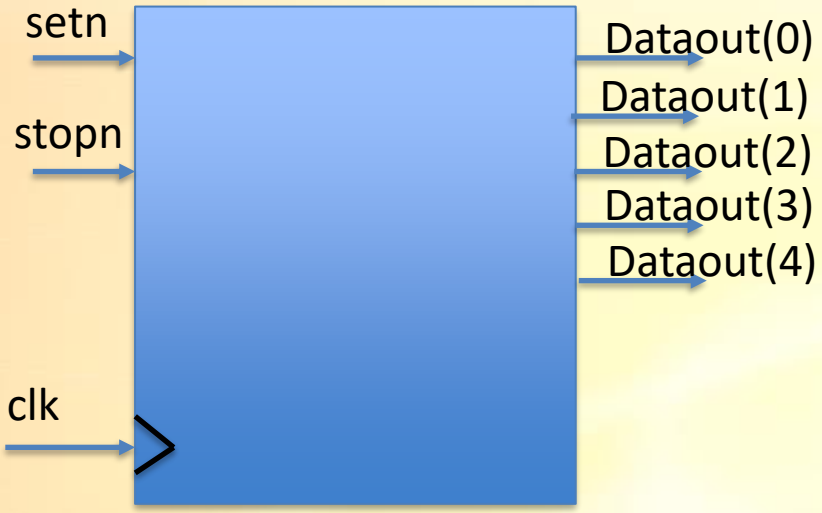
Version: 1.0.4





EFINIX®

# Design (simple up counter with set and stop)




counter.vhd

```
1  -- smal example design for the Trion 20 BGA256 Efinix Eval Board
2  -- By Harald Werner
3  -- 18.03.2020
4  --
5  library ieee;
6  use ieee.std_logic_1164.all;
7  use ieee.std_logic_unsigned.all;
8
9  entity counter is
10 port ( clk      : in std_logic;                    -- clock input.  external 74.25MHz clock use GPIOl_75_PLLIN
11        setn     : in std_logic;                    --  Set signal, low active; sett all outputs to '1' (LED are low active, means all LEDs msut be OFF) GPIOL_02 SW4
12        stopn    : in std_logic;                    -- Stop signal, low active Stop counting GPIOL_04 SW5
13        Dataout  : out std_logic_vector ( 4 downto 0));   -- Output data connected to the LEDs (low active); GPIOR_153,GPIOR_118,GPIOR_117,GPIOR_105,GPIOR_104
14 end counter;
15
16 architecture vers1 of counter is
17
18 signal cnt: std_logic_vector ( 30 downto 0) := (others => '0');
19
20 Begin
21
22 cnt_proc : process(clk, setn)
23         Begin
24           if setn = '0' then
25              cnt    <= (others => '1');
26              dataout <= (others => '1');
27           elsif clk'event and clk = '1' then
28              if stopn = '0' then
29                 cnt <= cnt;
30              else
31                 cnt <= cnt +1;
32              end if;
33              Dataout <= cnt(30 downto 26);   --For the 75MHz external clock use 30 downto 26
34           end if;
35        end process;
36 end vers1;
37
```



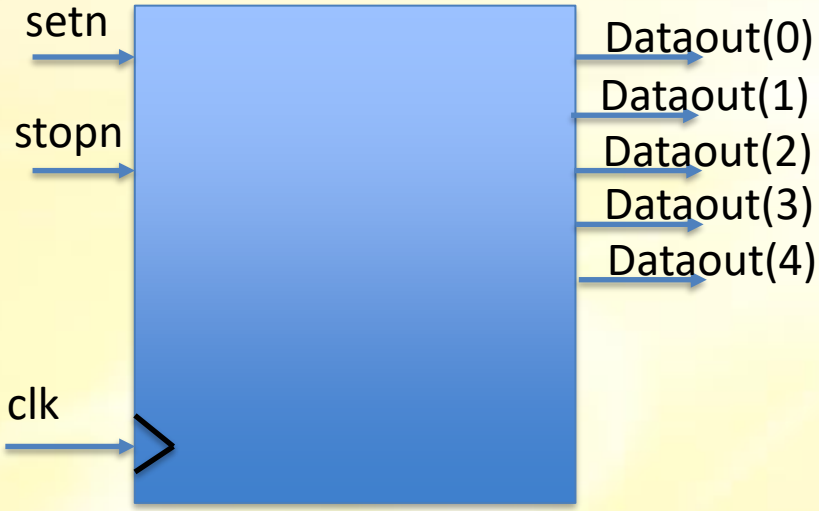
EFINIX®

# Design / Board connection

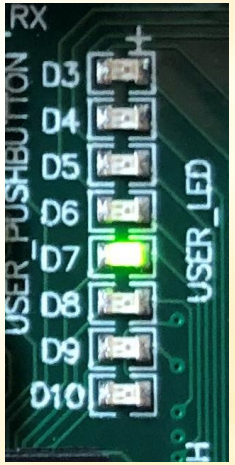

SW4 GPIOL_02

SW5 GPIOL_04

75MHz , GPIOL_75_PLLIN



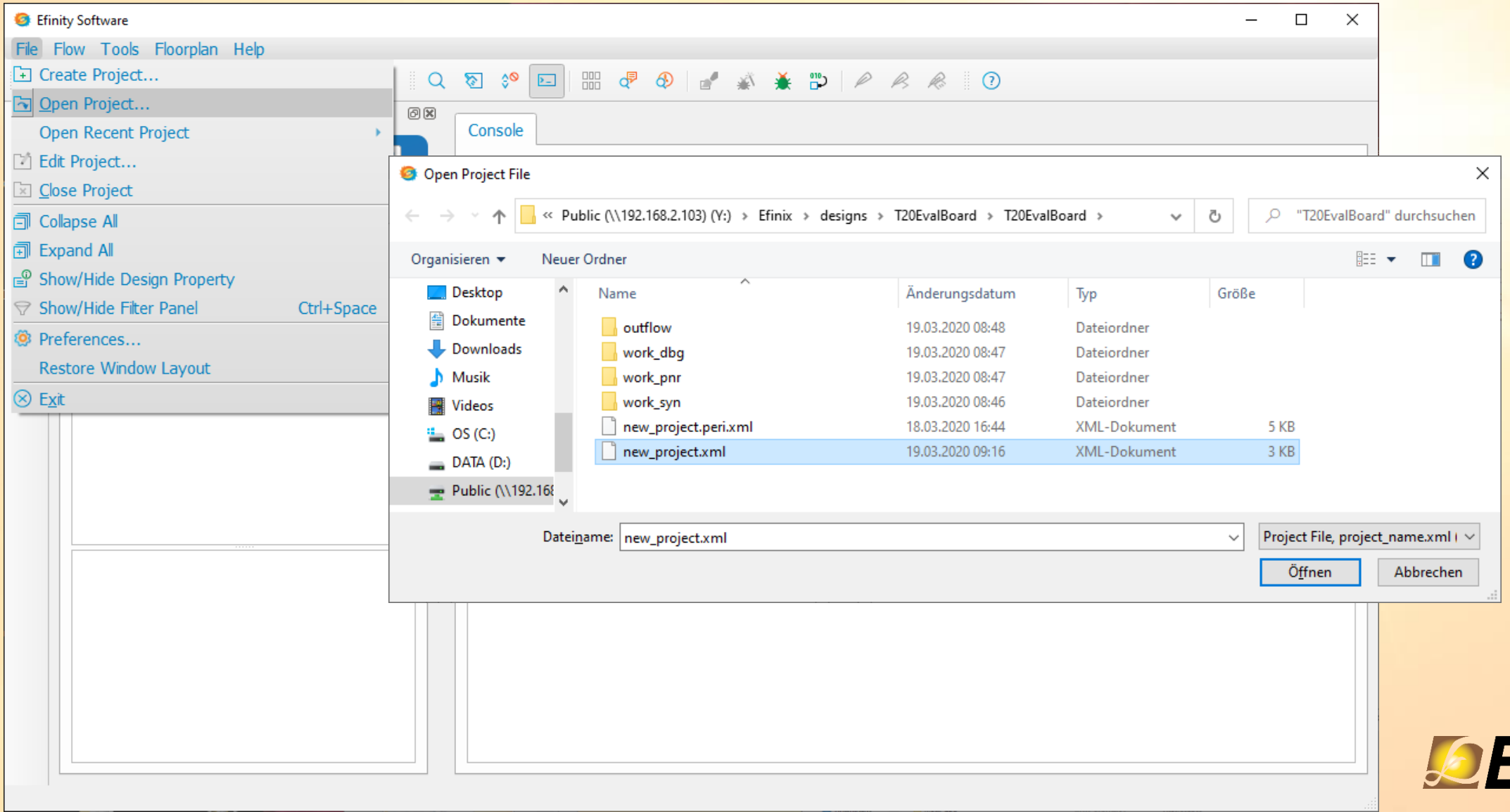

- LED D3 GPIOR_104
- LED D4 GPIOR_105
- LED D6 GPIOR_117
- LED D7 GPIOR_118
- LED D8 GPIOR_153



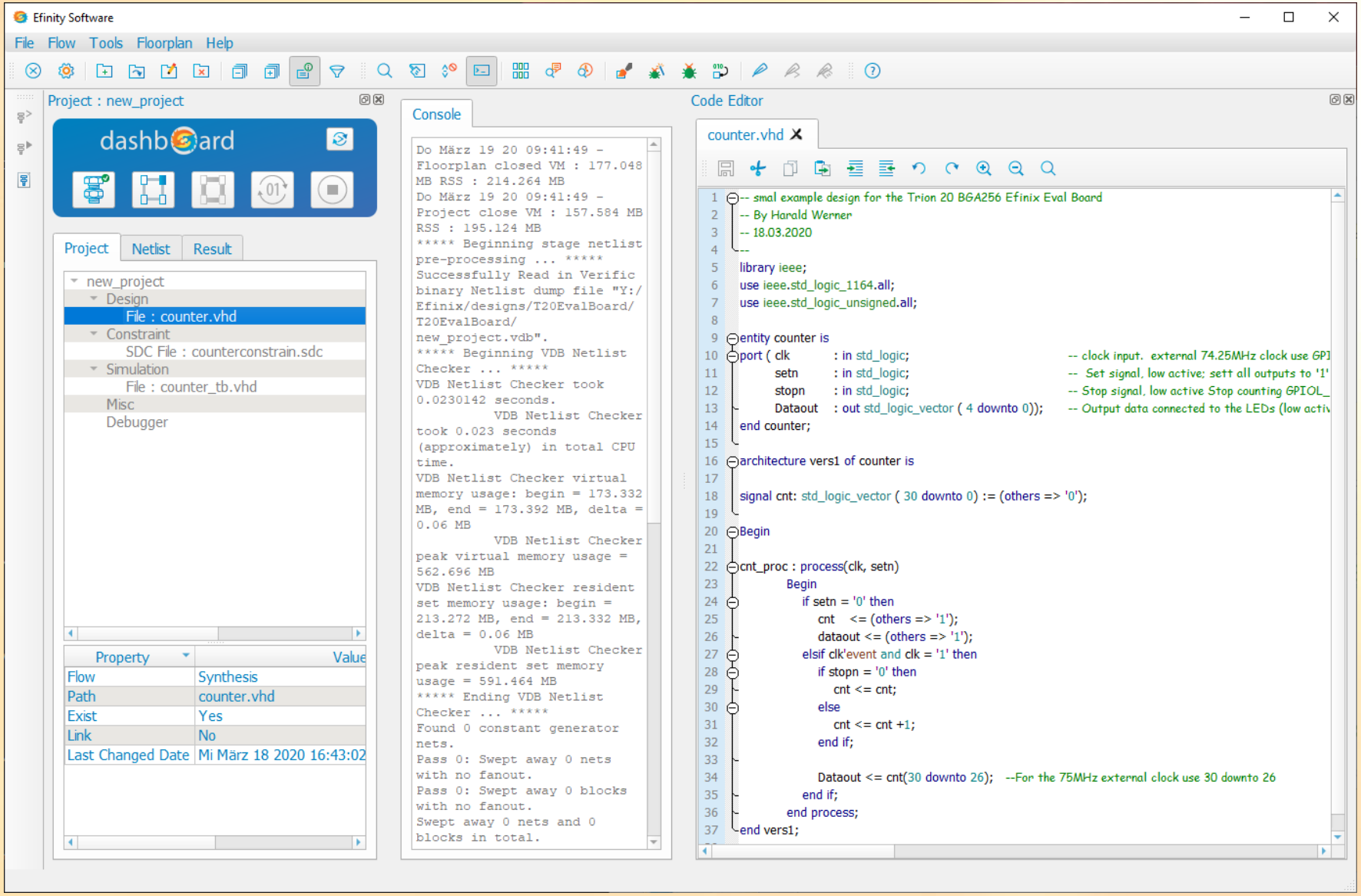



EFINIX®

# 1. Open Efinity new_project.xml

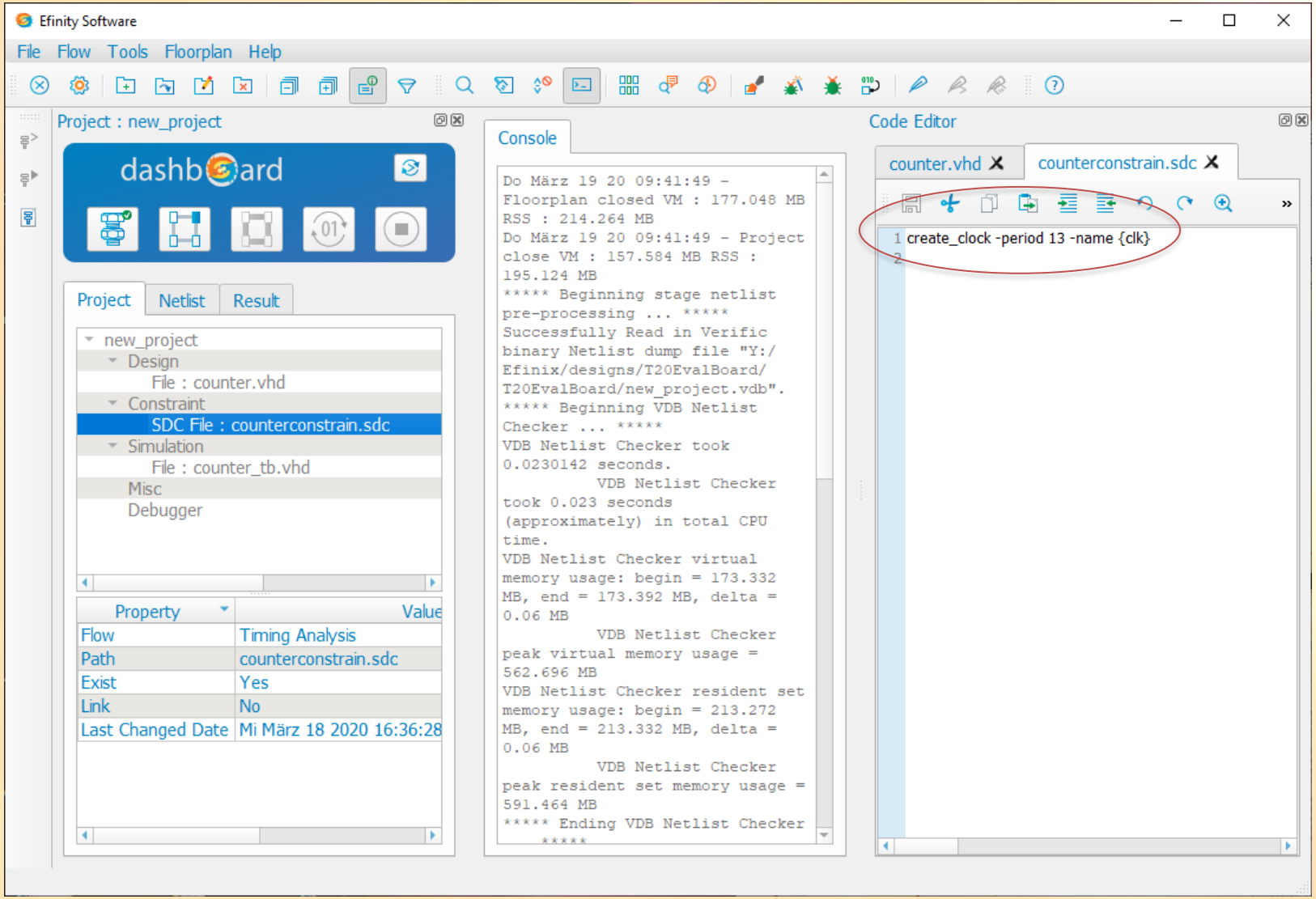


# Double click on counter.vhd




- The counter.vhd will show up in the Code Editor

# Double click on the timing constrain counterconstrain.sdc

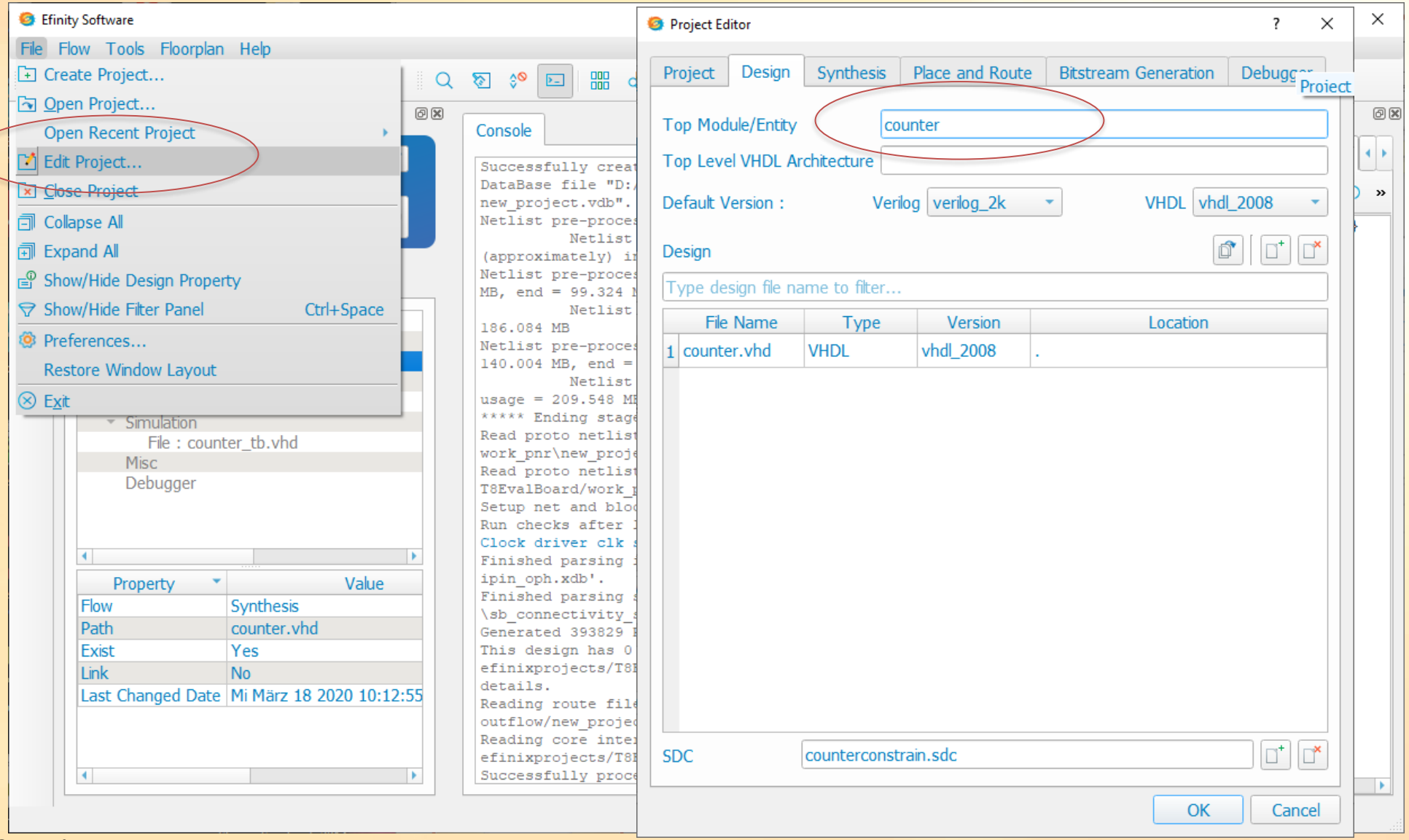



The counterconstrain.sdc will show up and you can see the 13ns period constrain for the 75MHz clock



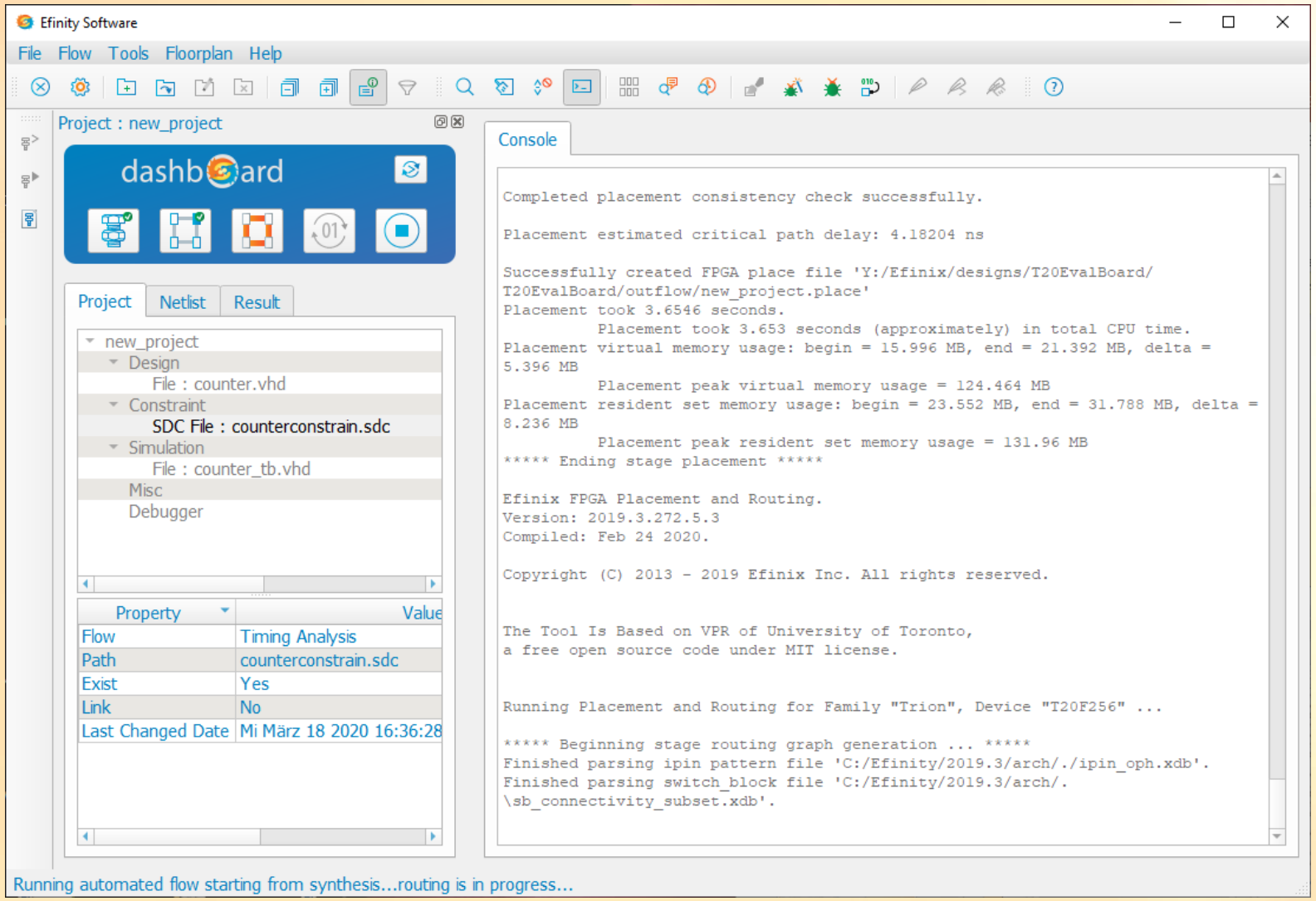
EFINIX®

# Set the top entity name




Enter counter to top Module/Entity and click OK

# Run the whole flow



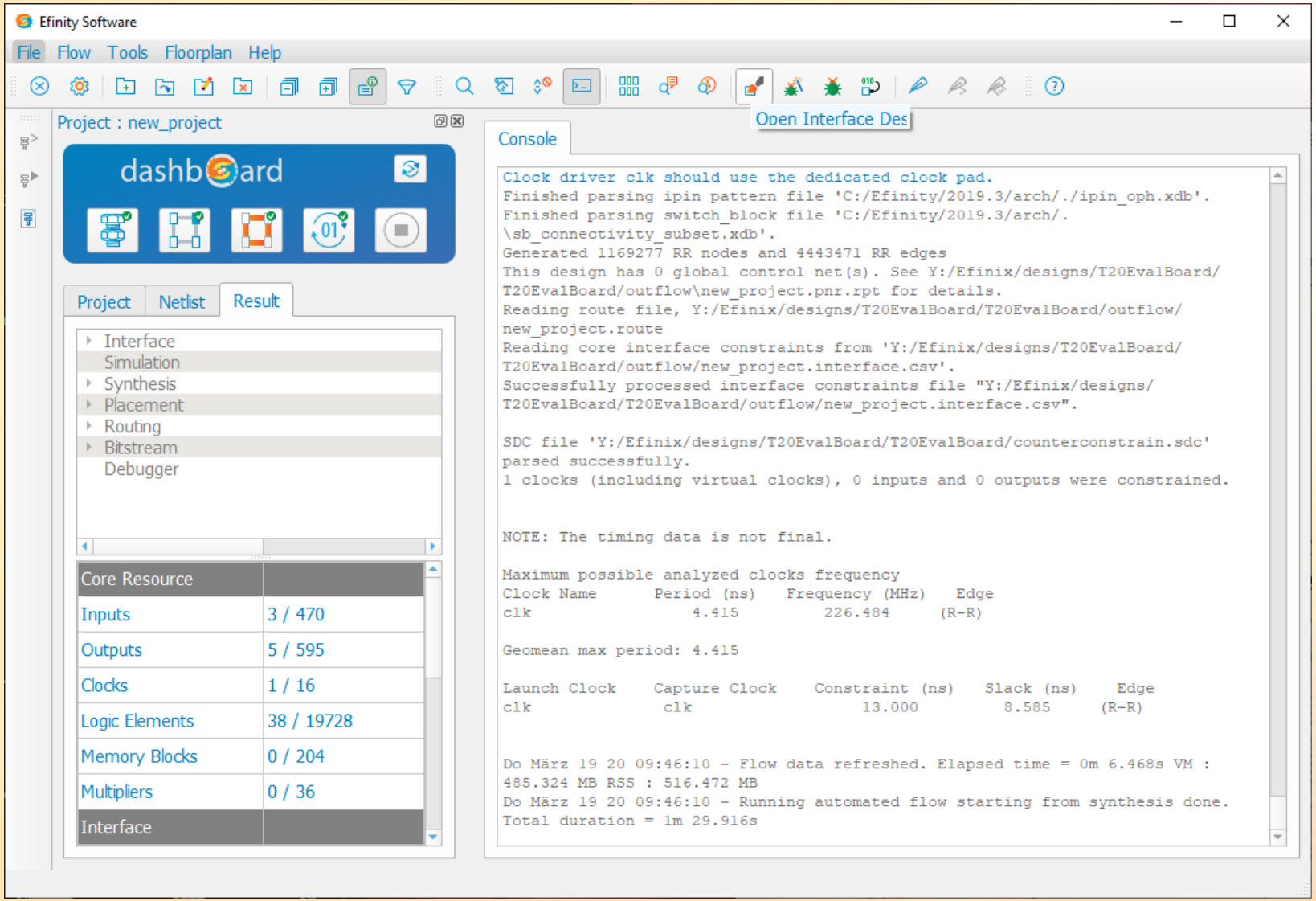

- If the Automated flow button is grayed out click on the button to activate the automated flow.

- Click on the synthesis icon and the flow will run automnatically

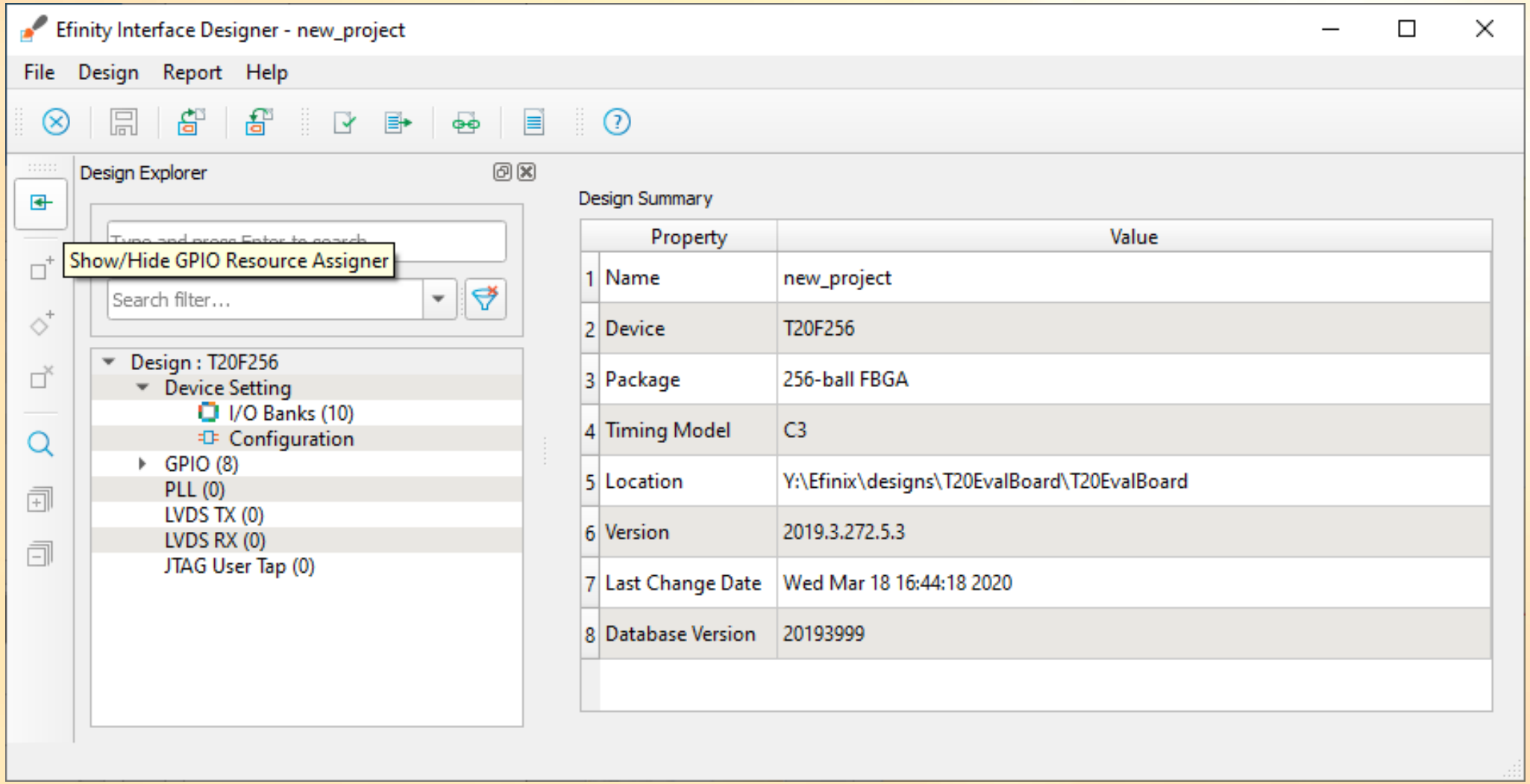


EFINIX®

# Assign the top level Signals to Pins.
# 1. Open Interface Designer



# Show GPIO Resource Assigner

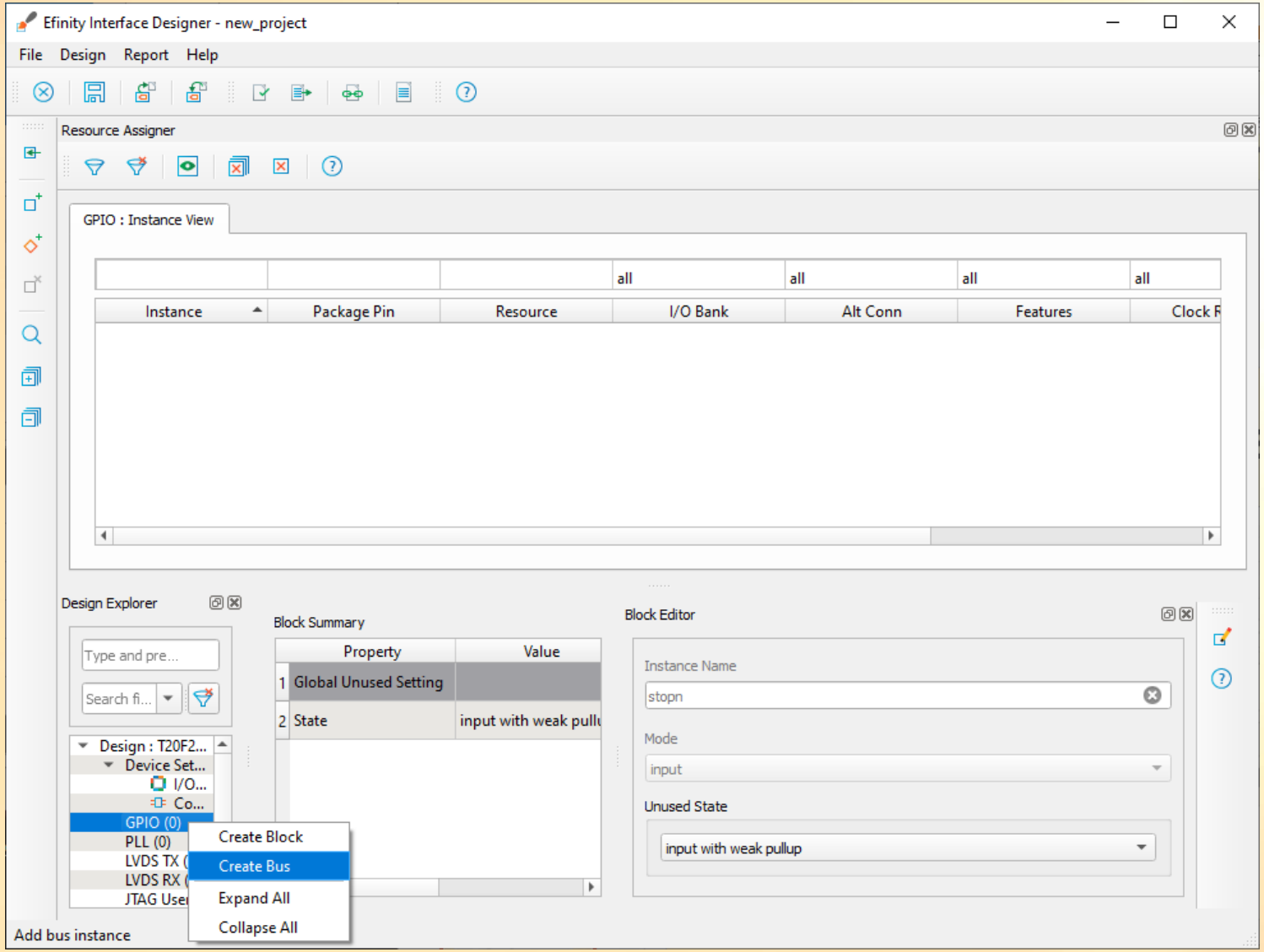



| | Property | Value |
|---|---|---|
| 1 | Name | new_project |
| 2 | Device | T20F256 |
| 3 | Package | 256-ball FBGA |
| 4 | Timing Model | C3 |
| 5 | Location | Y:\Efinix\designs\T20EvalBoard\T20EvalBoard |
| 6 | Version | 2019.3.272.5.3 |
| 7 | Last Change Date | Wed Mar 18 16:44:18 2020 |
| 8 | Database Version | 20193999 |

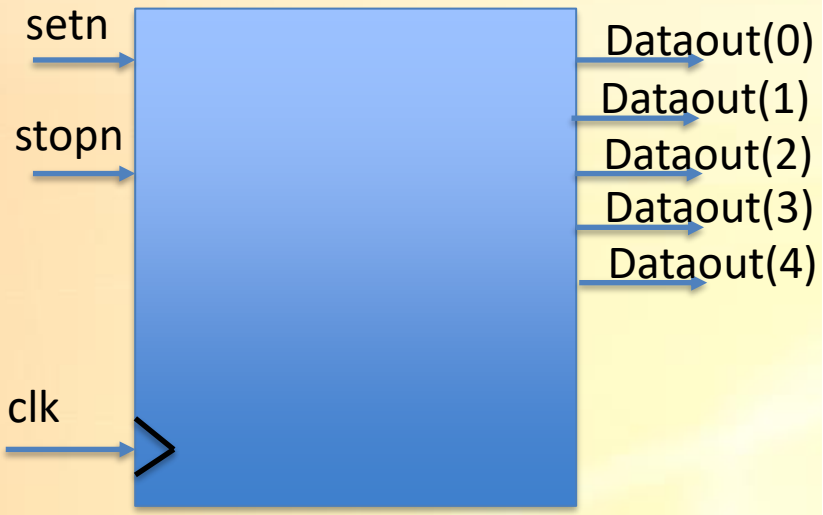


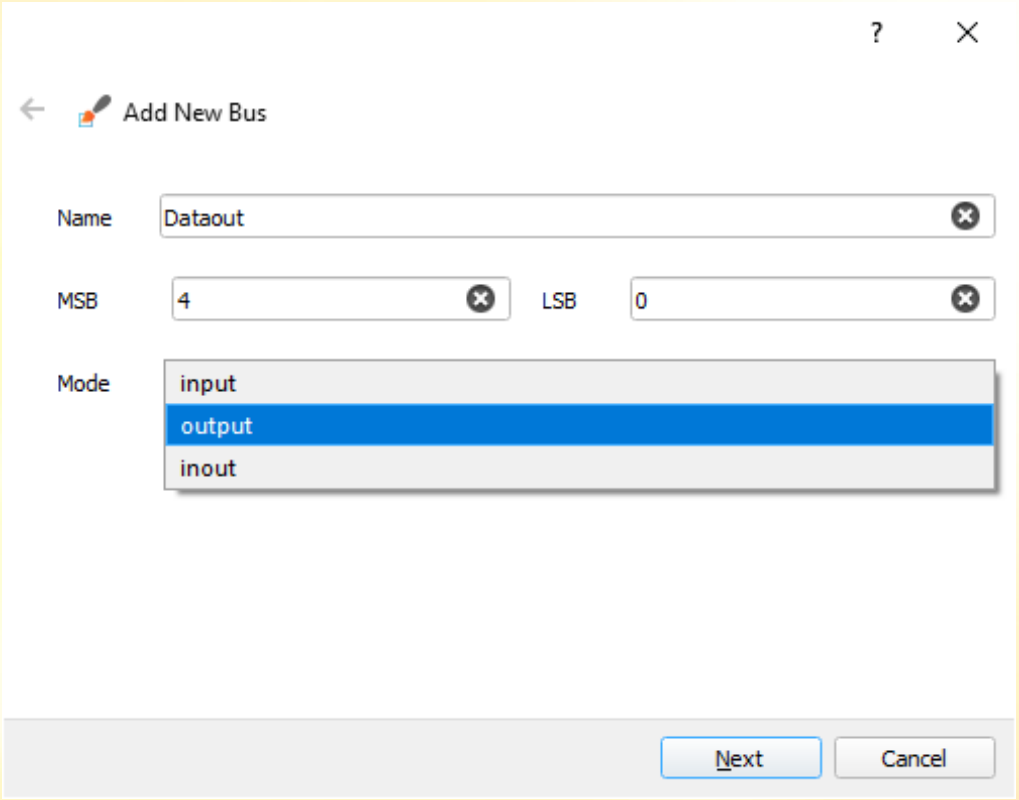
EFINIX®

# Select GPIO, RMB, Create Bus



# Create a Output BUS Dataout(4 downto 0)




Name: Dataout

MSB: 4

LSB: 0

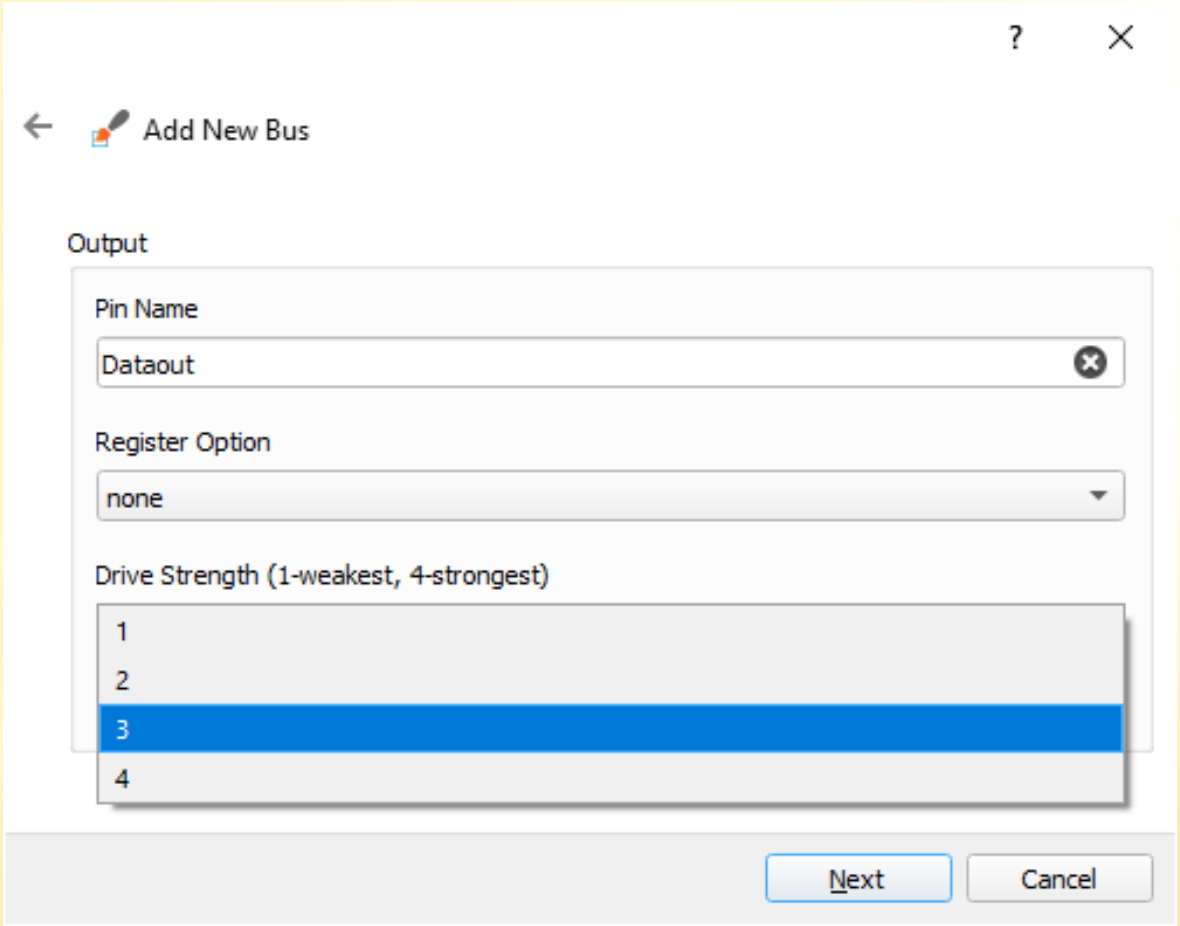
Mode: output

Click Next

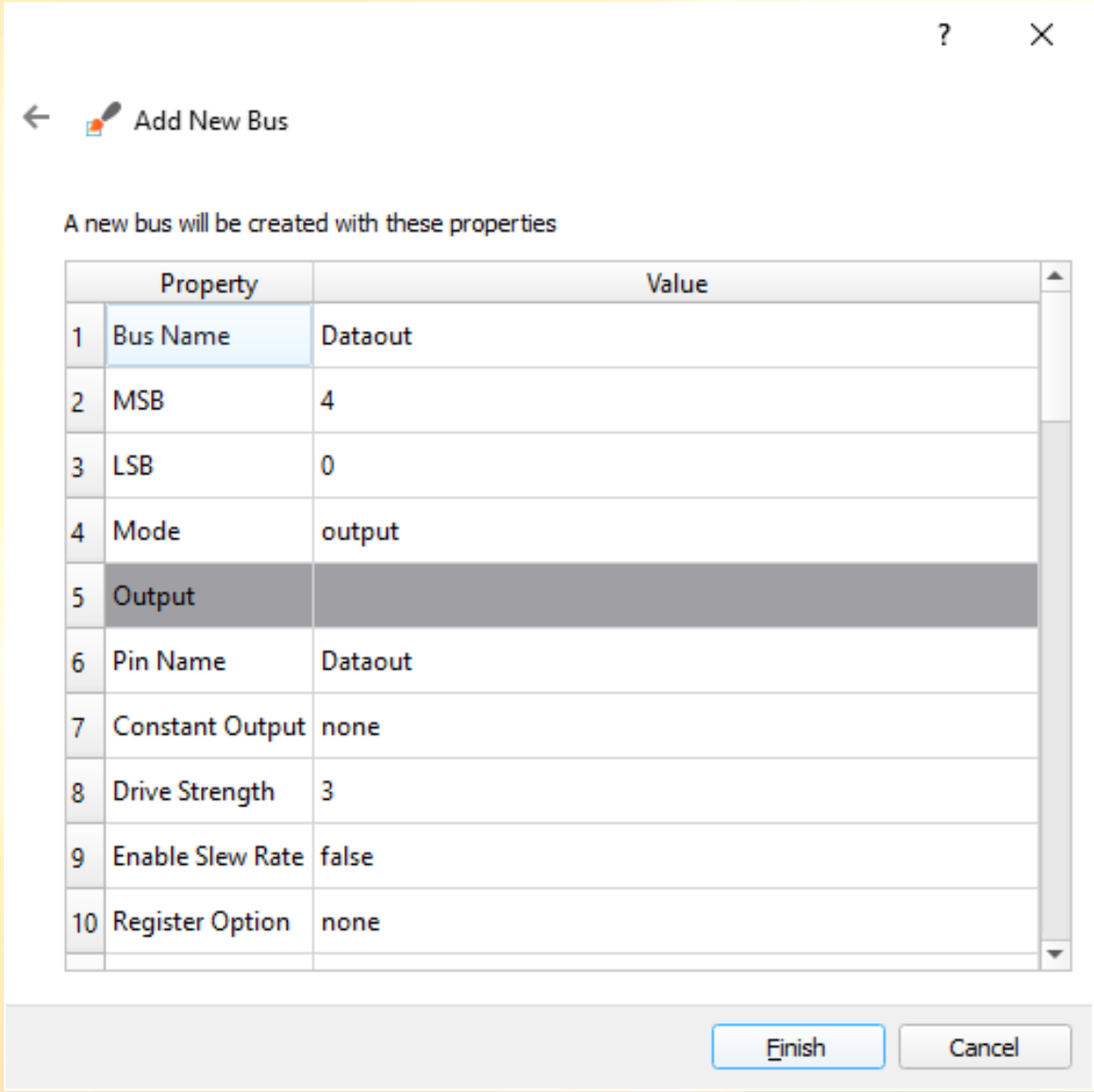


EFINIX®

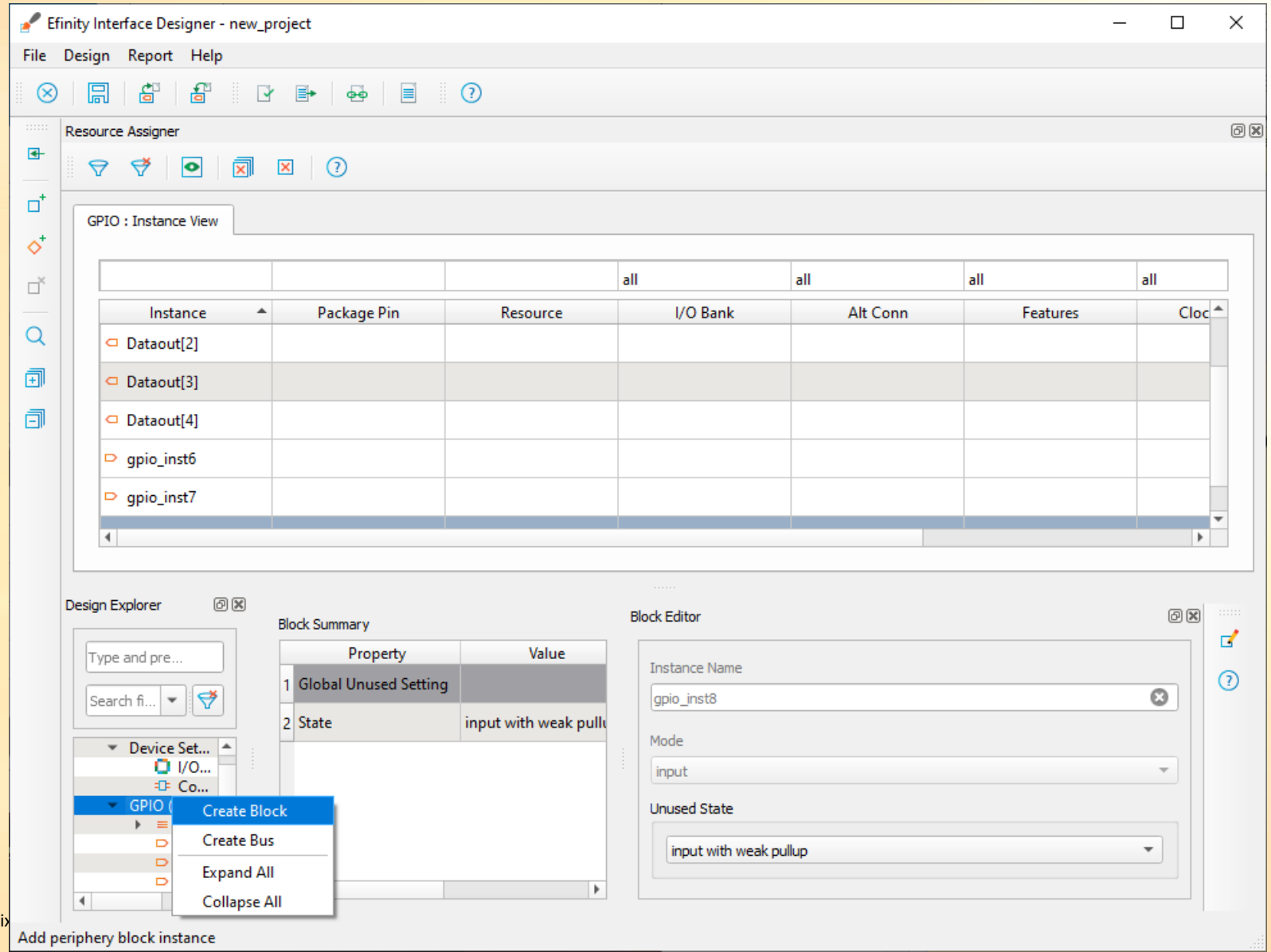
# Set drive strength to 3; click Next



# Click Finish



# Adding the additional Pins clk, stopn, setn




Select GPIO

Create Block

Select GPIO

Create Block

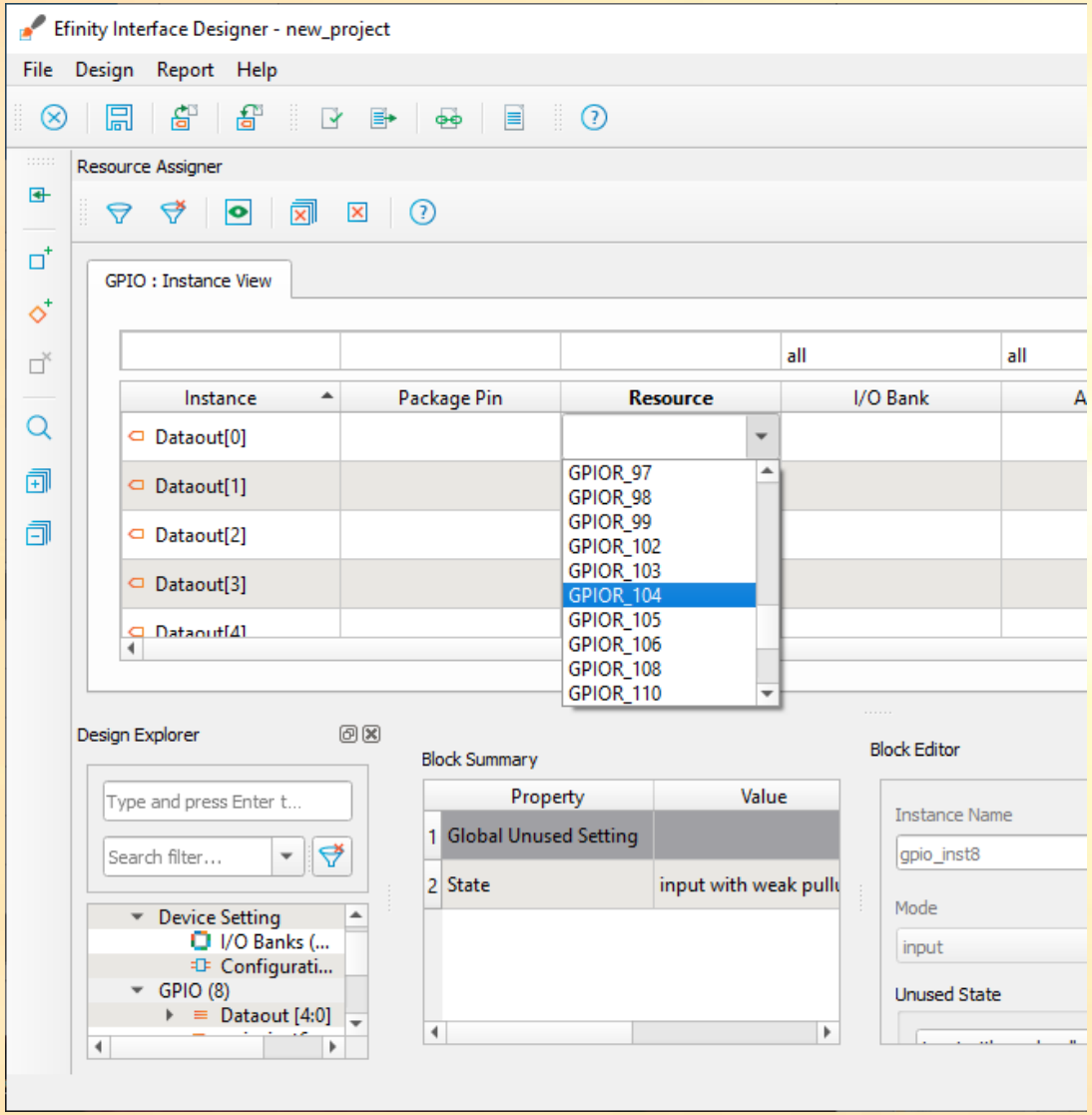
Select GPIO

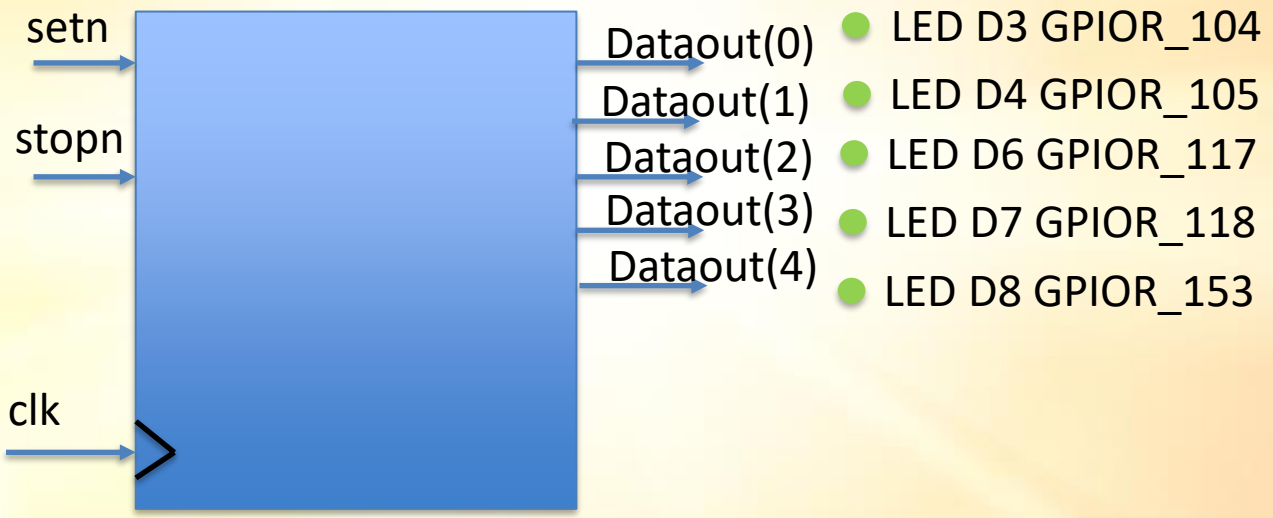
Create Block



EFINIX®

# Select GPIO: Instance View Dataout[0]->Resource



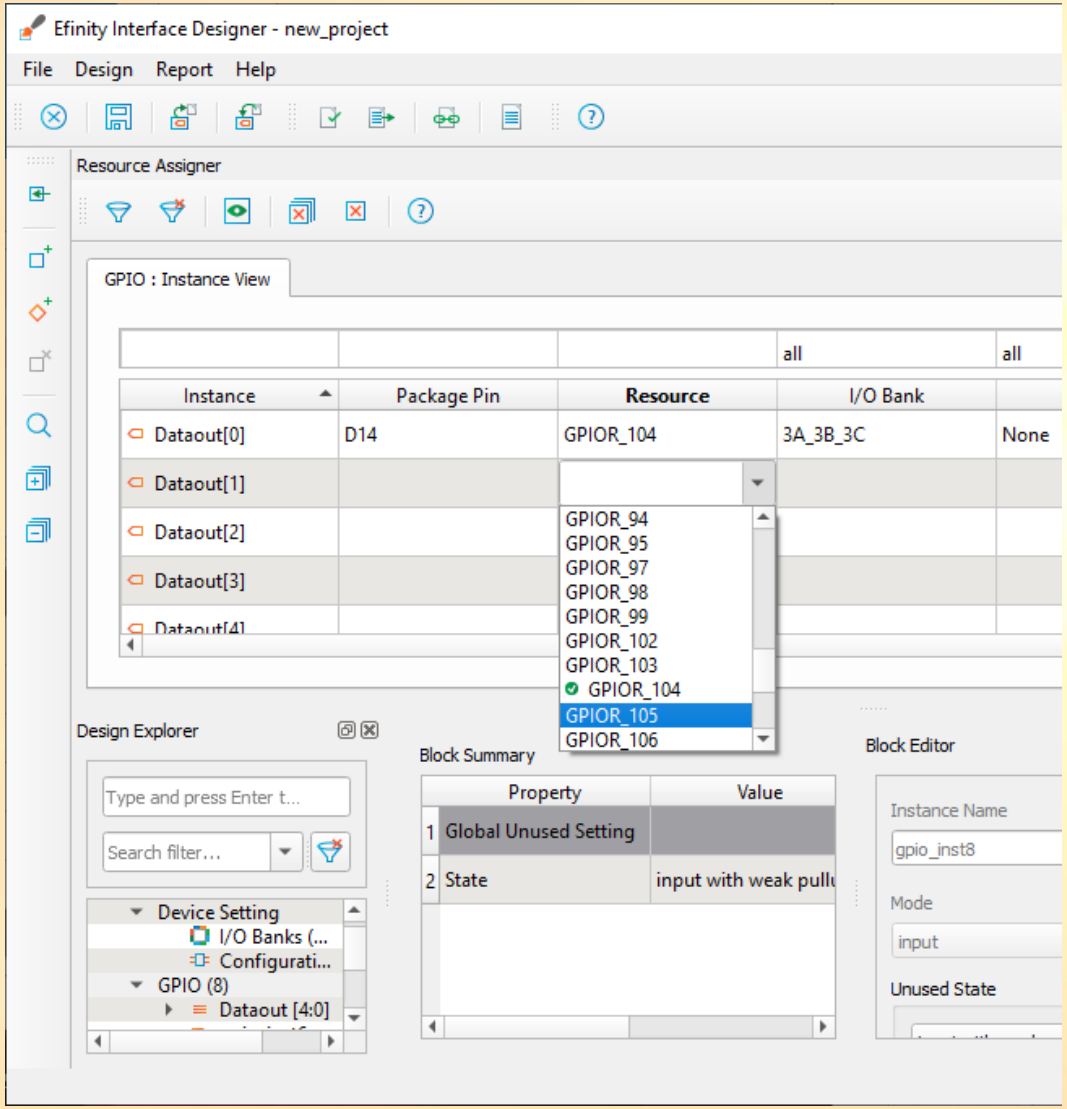



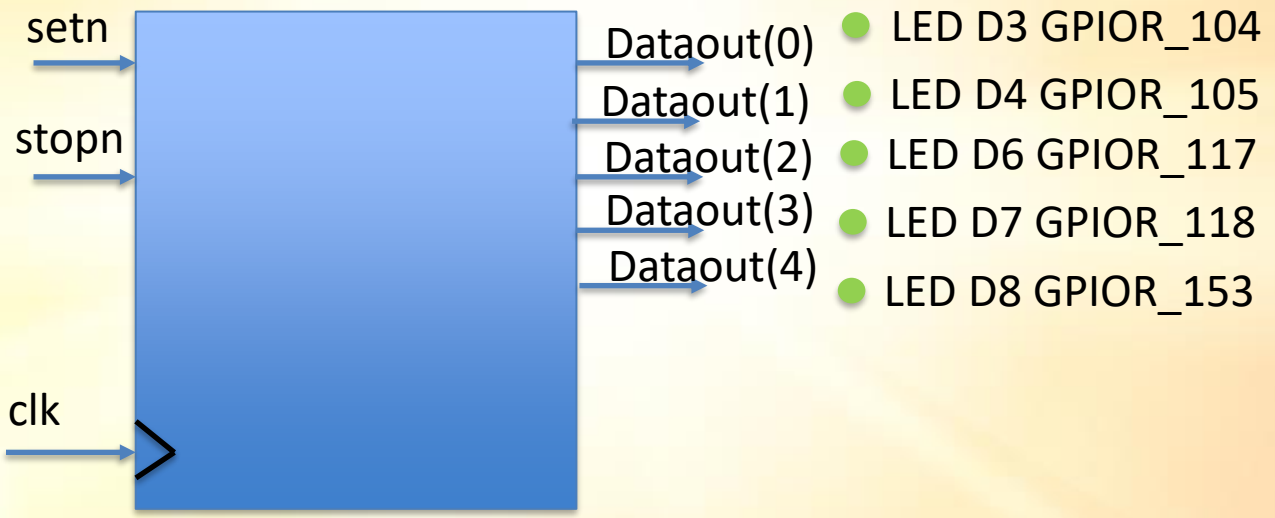

Dataout[0] ->GPIOR_104



EFINIX®

# Select GPIO: Instance View Dataout[1]->Resource



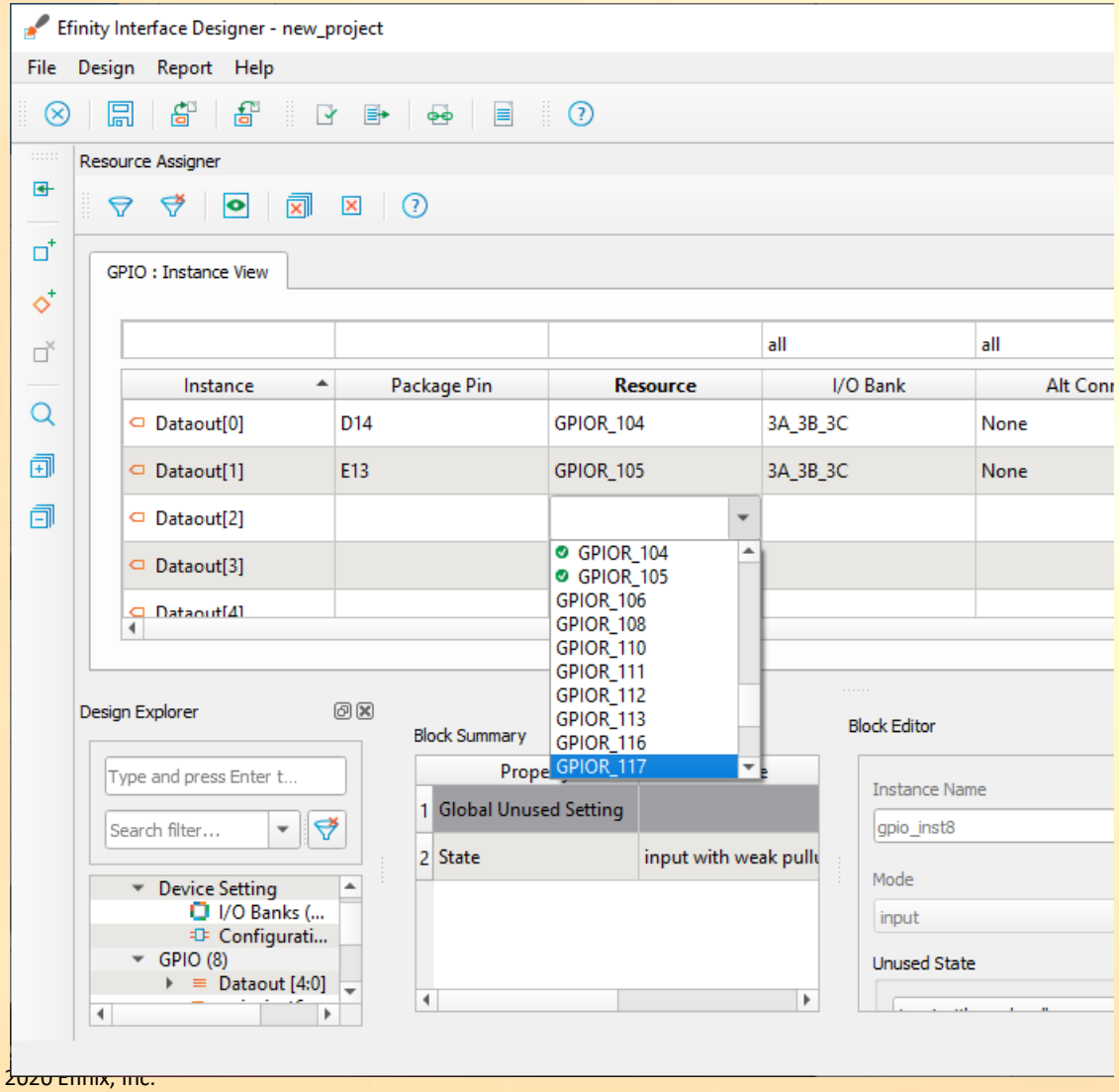



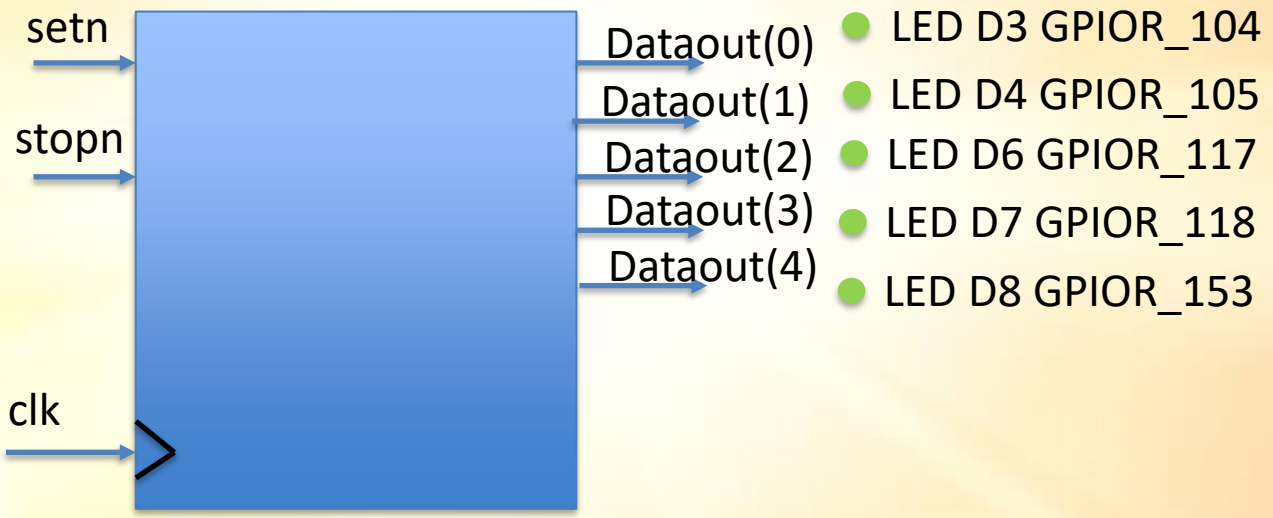

Dataout[1] ->GPIOR_105



EFINIX®

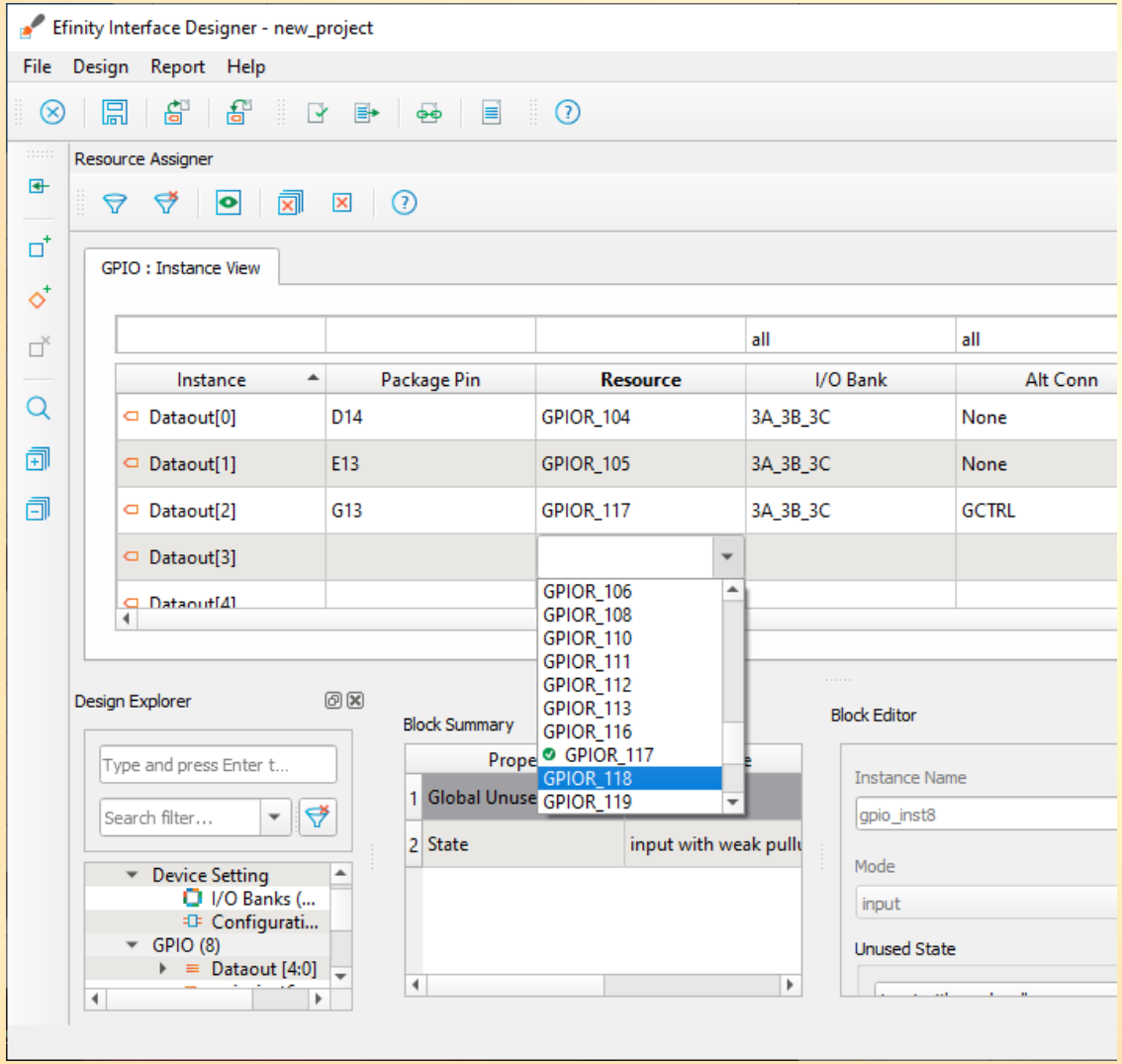
# Select GPIO: Instance View Dataout[2]->Resource

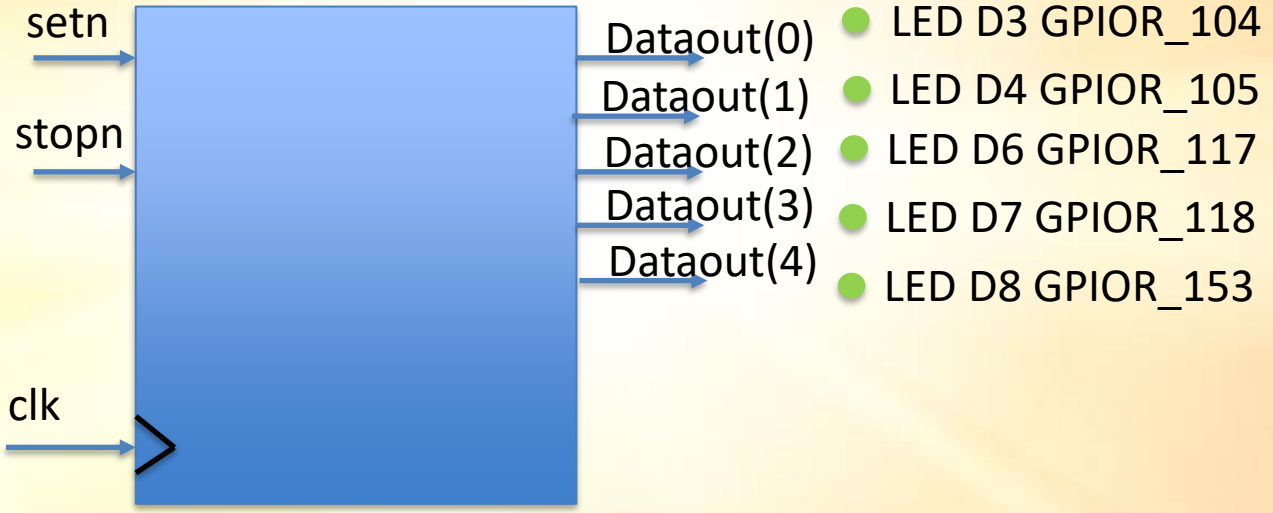



Dataout[2] ->GPIOR_117

# Select GPIO: Instance View Dataout[3]->Resource



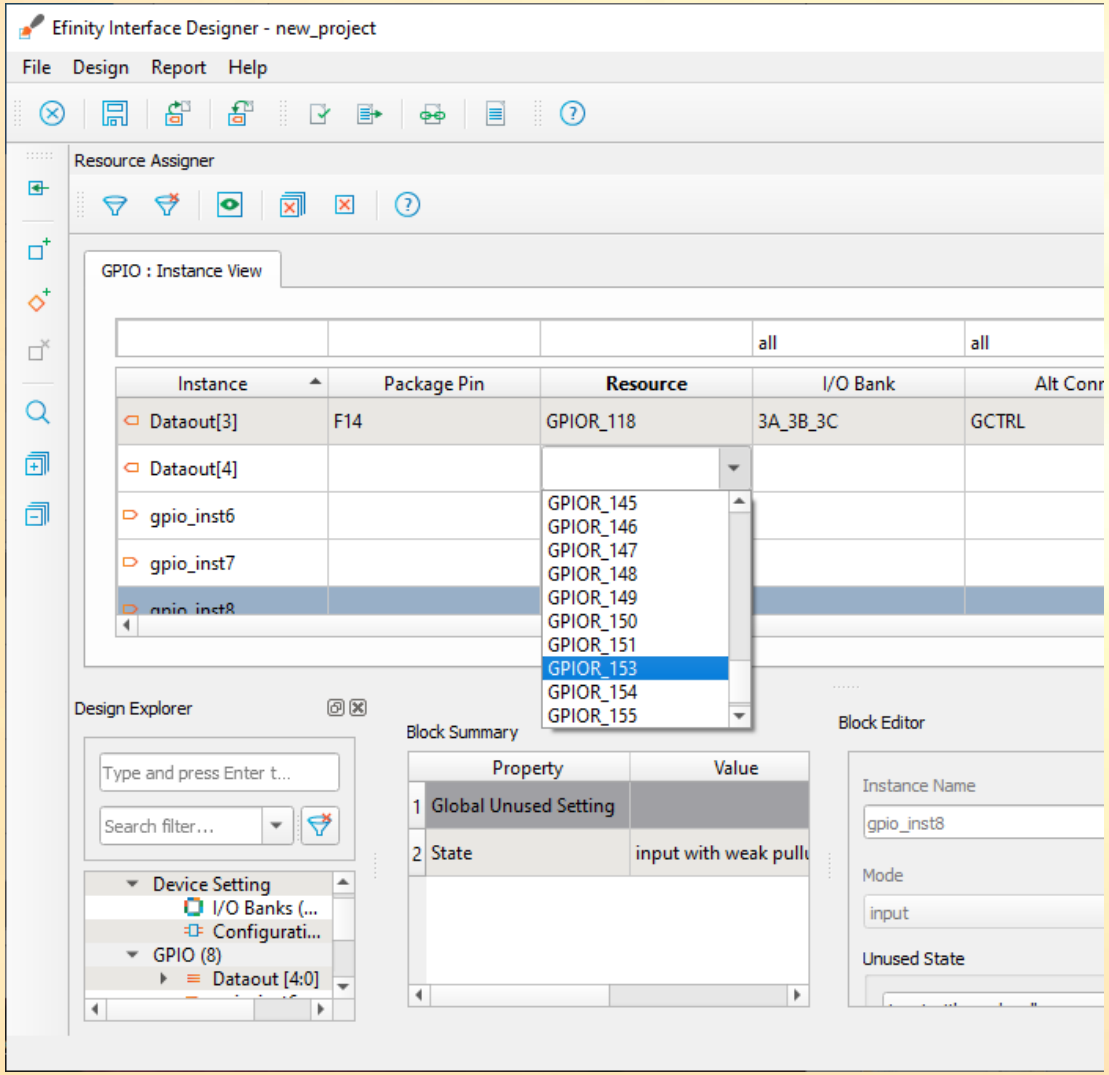



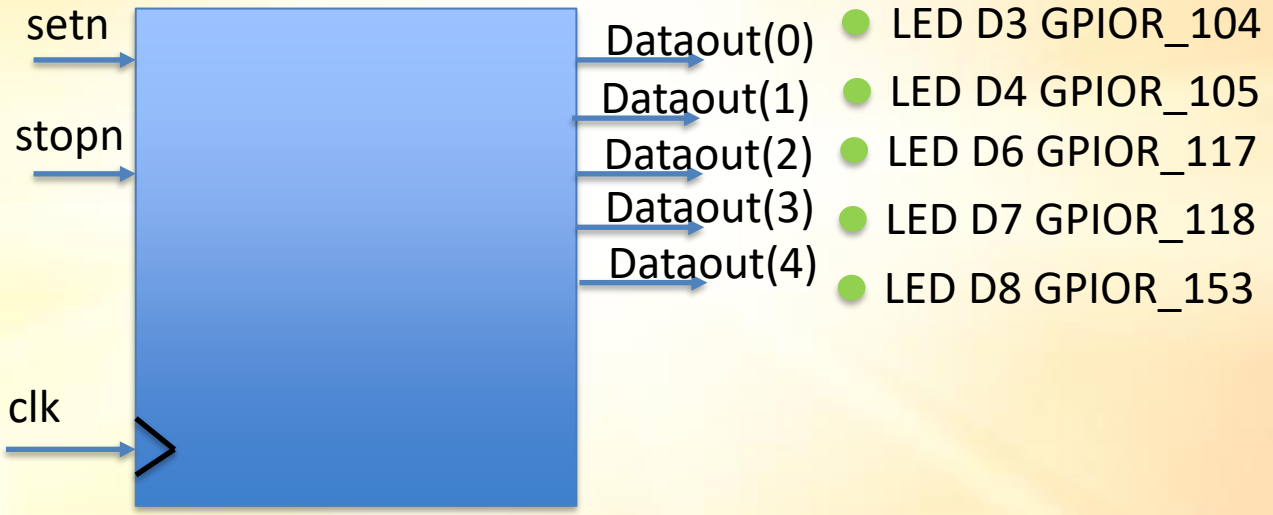

Dataout[3] ->GPIOR_118



EFINIX®

# Select GPIO: Instance View Dataout[4]->Resource



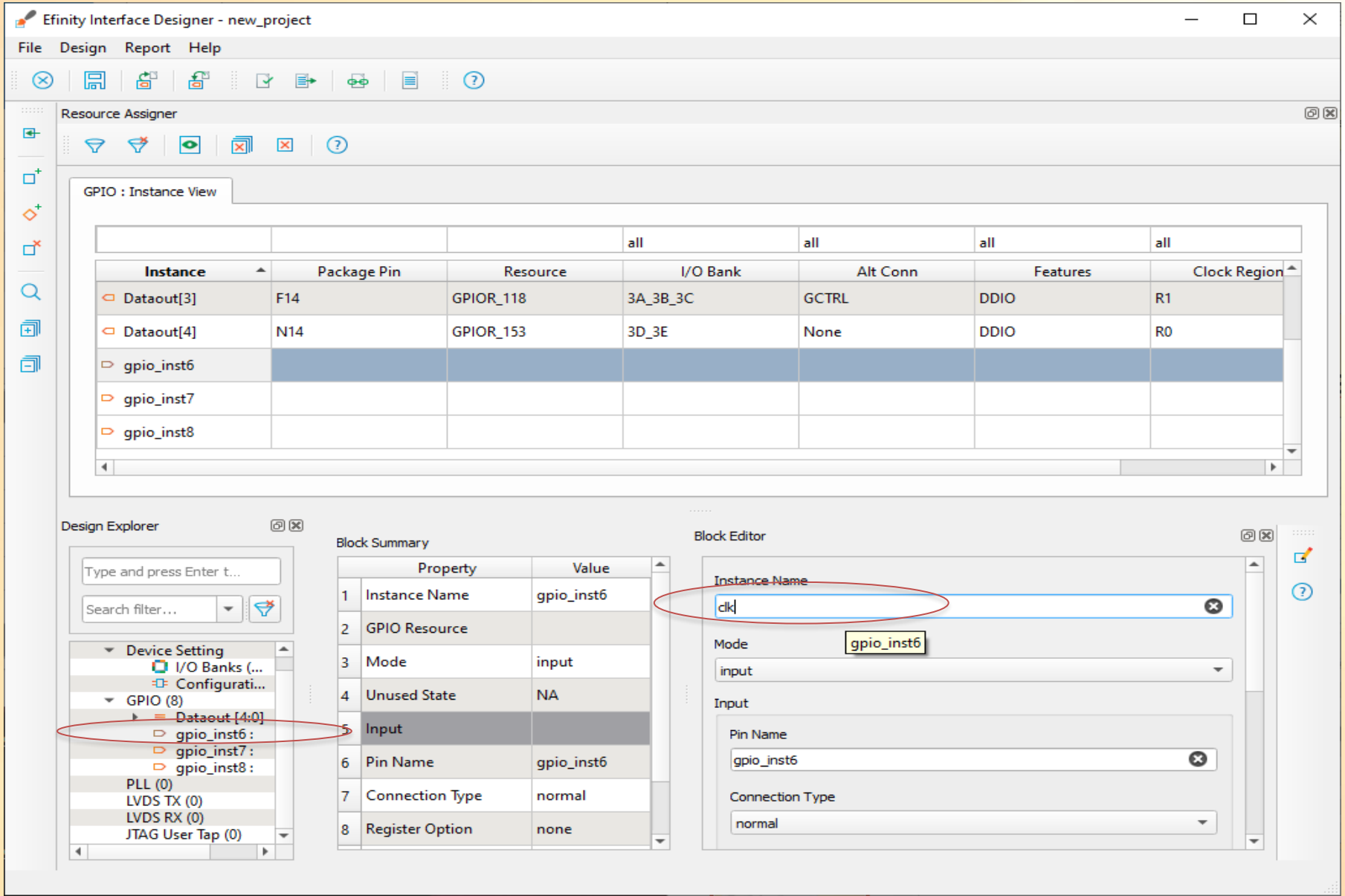




Dataout[4] ->GPIOR_153



EFINIX®

# Change the name from the remaining Inputs: clk

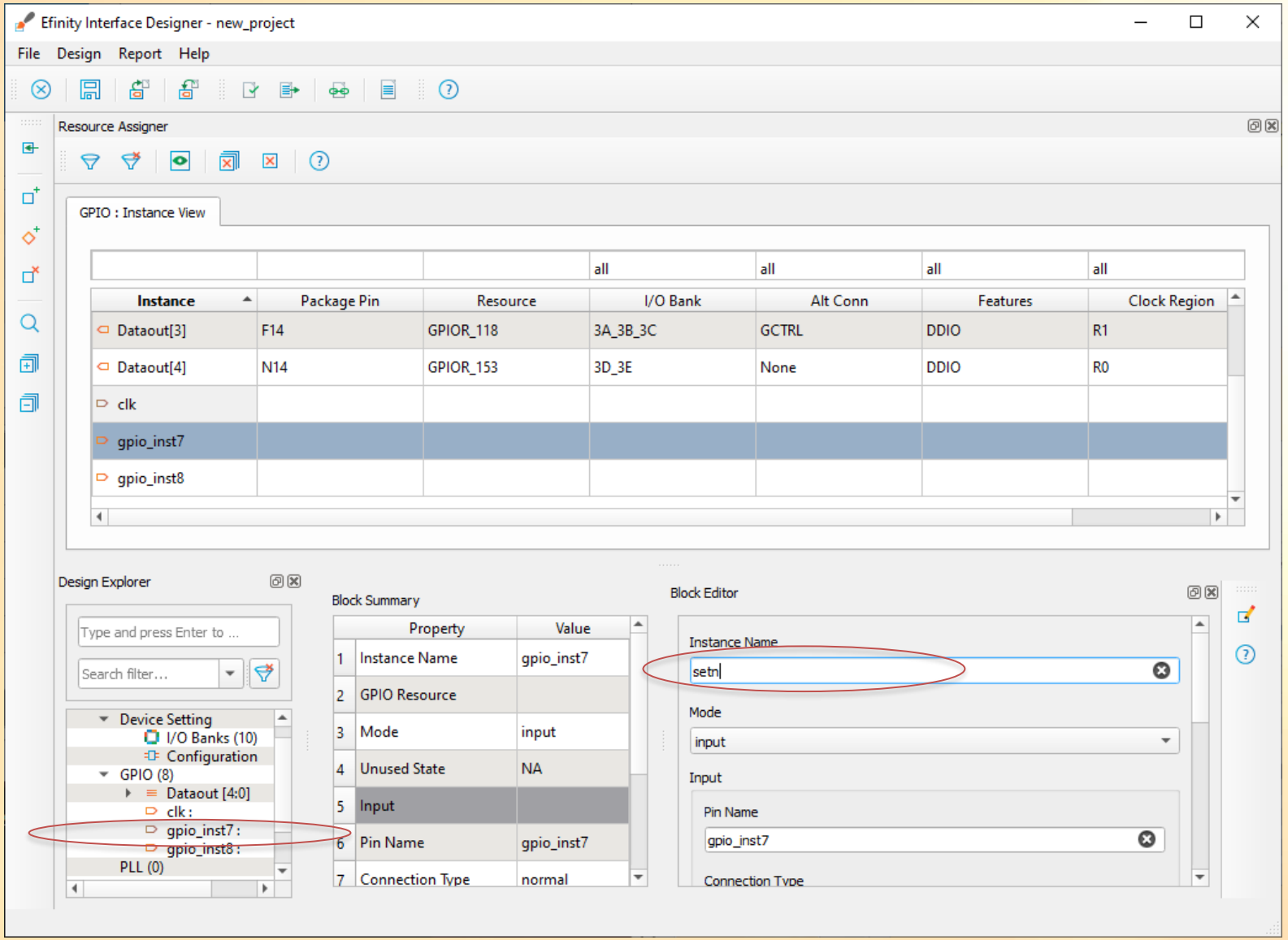



Select gpio_inst6

Enter name in the Block Editor clk **and press enter !**

EFINIX®

# setn



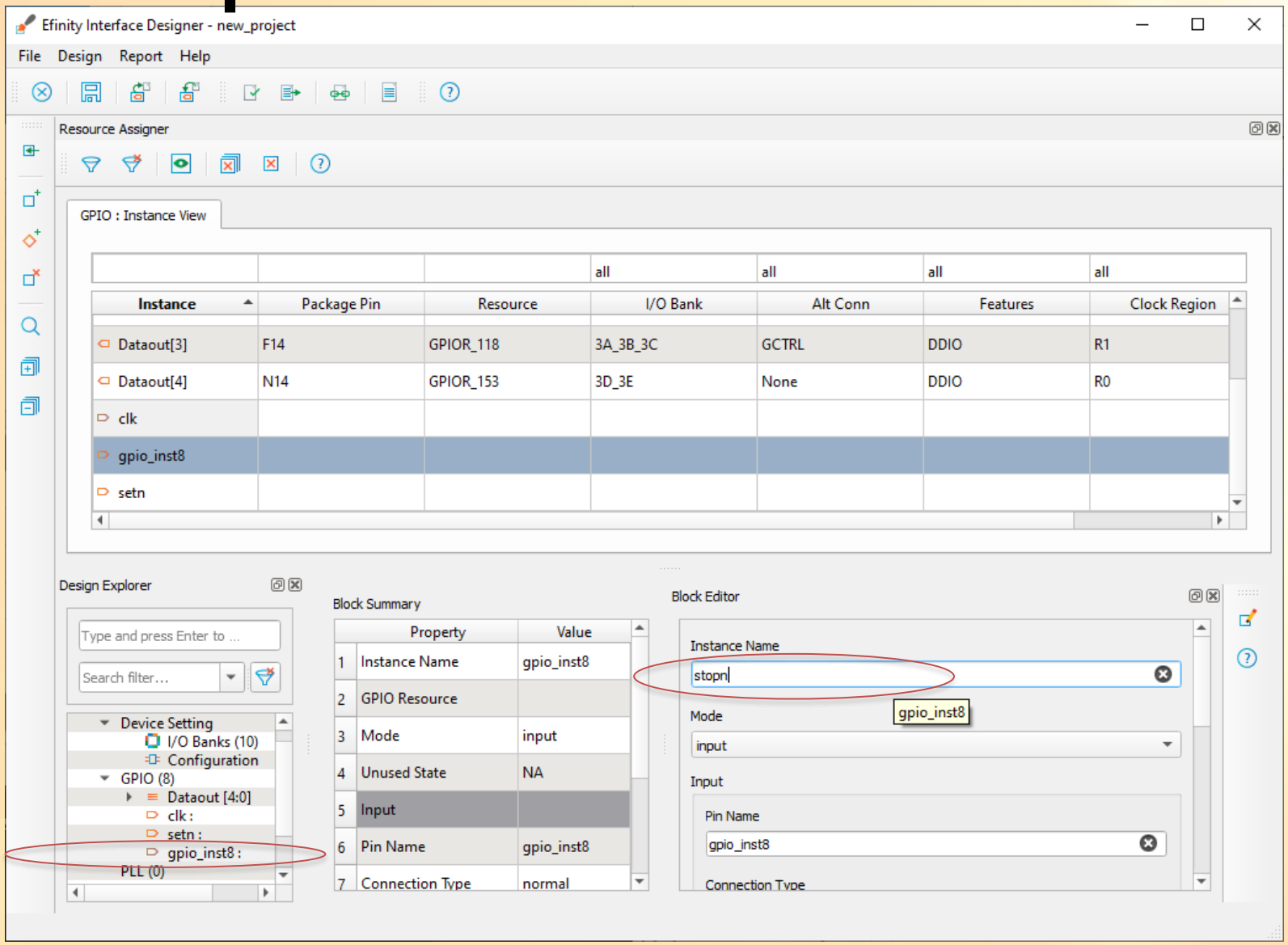

Select gpio_inst7

Enter name in the Block Editor setn **and press enter !**



EFINIX®

# stopn



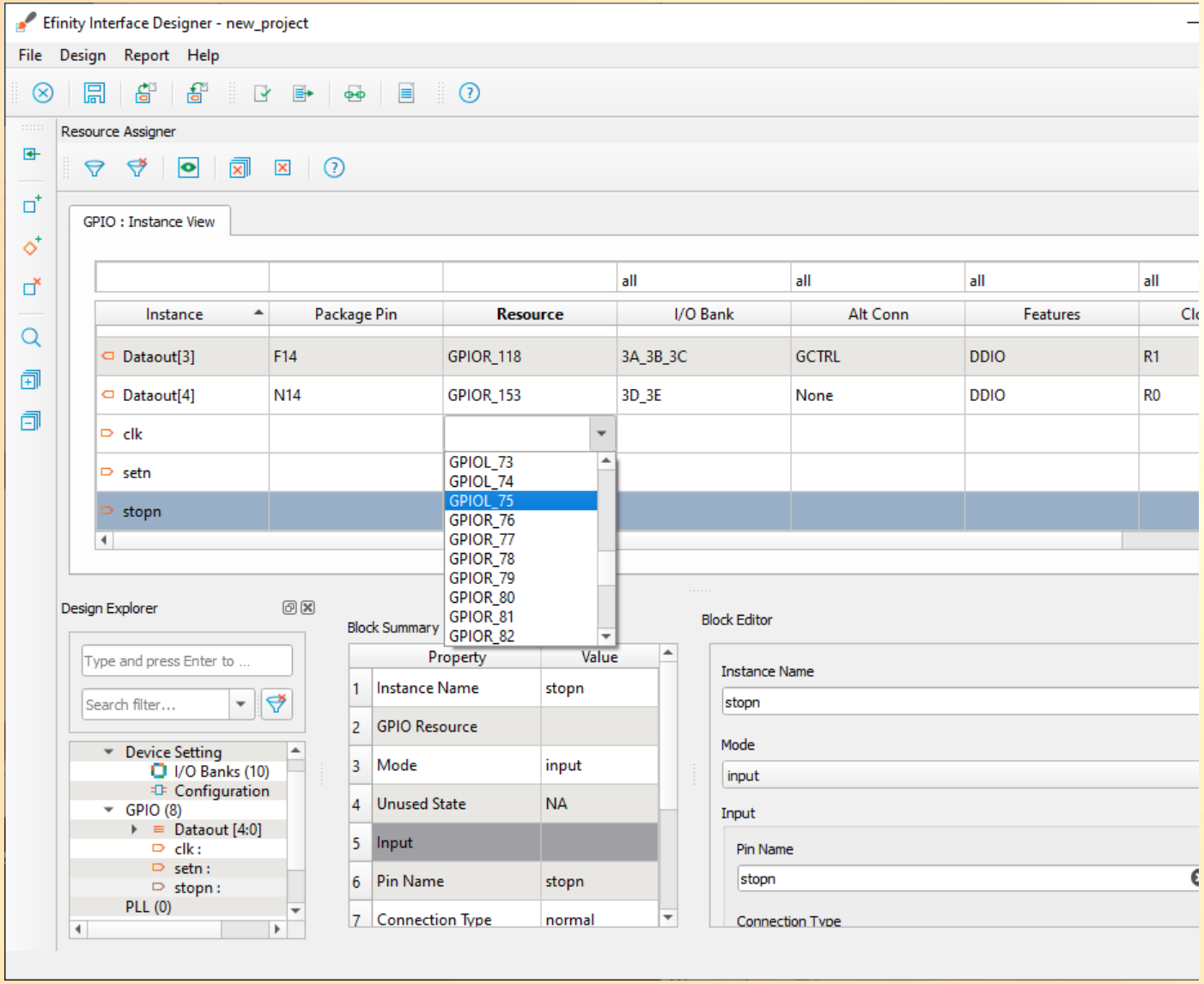

Select gpio_inst8

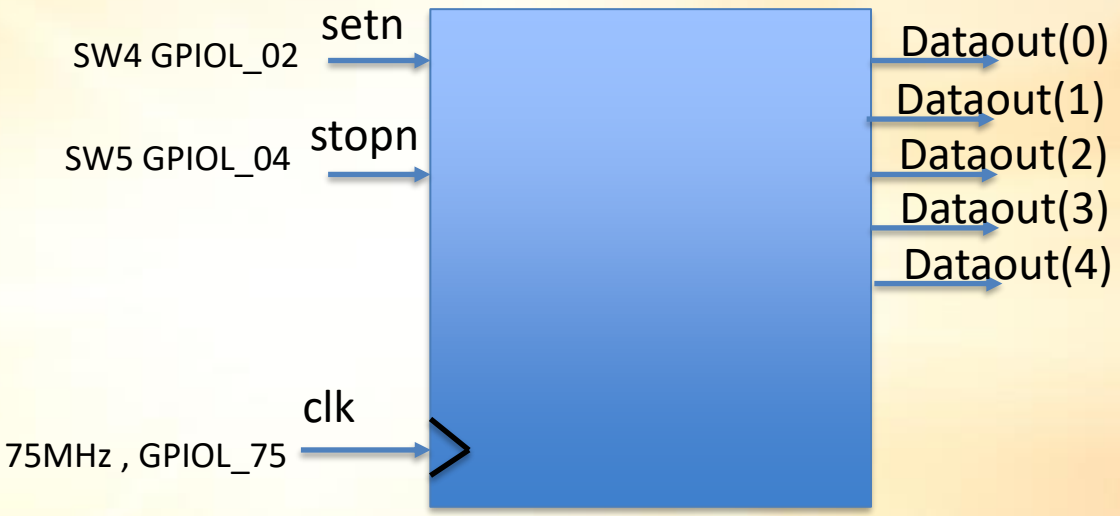
Enter name in the Block Editor stopn **and press enter !**



EFINIX®

# Assign the Resource CLK=>GPIOL_75

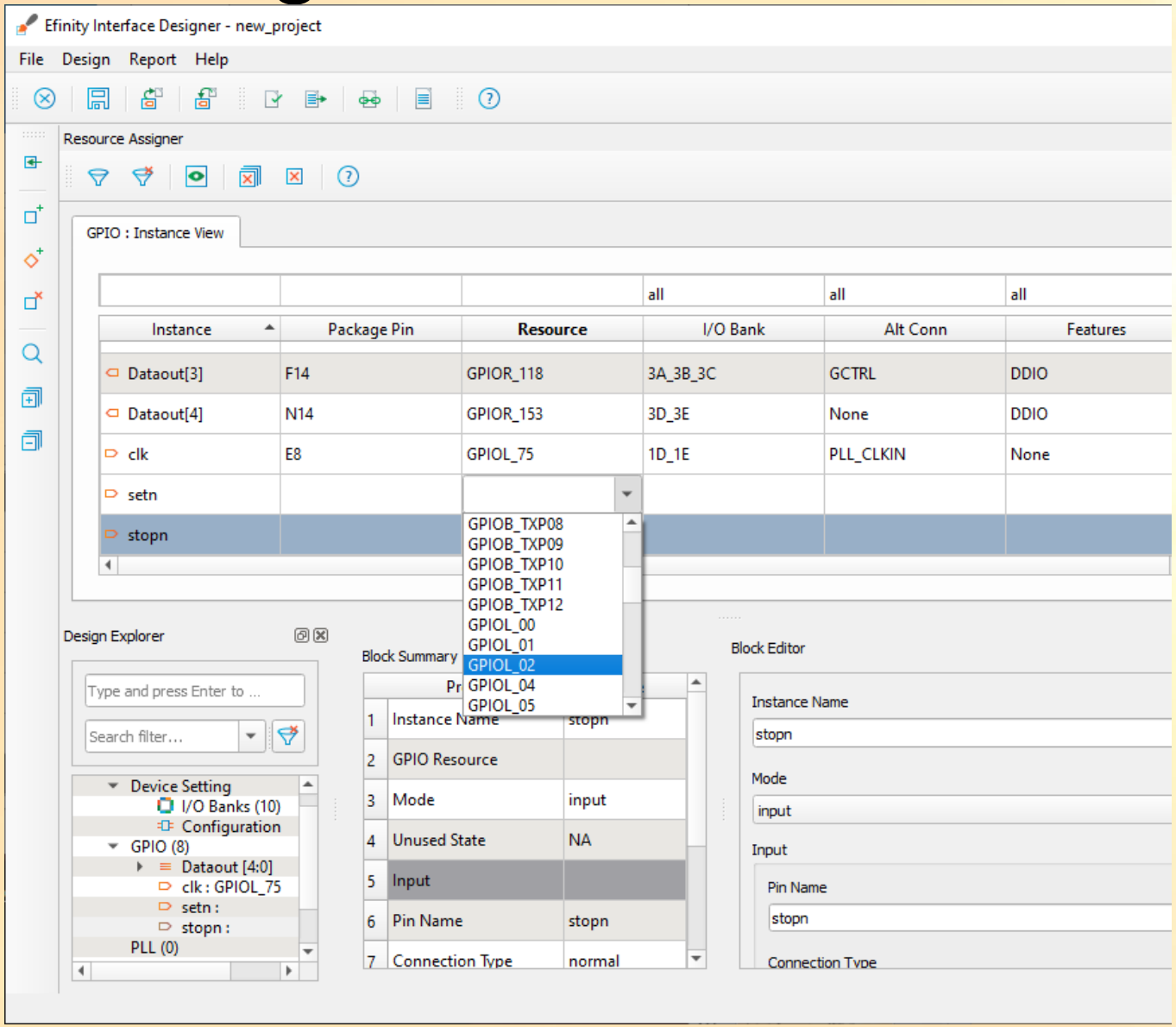


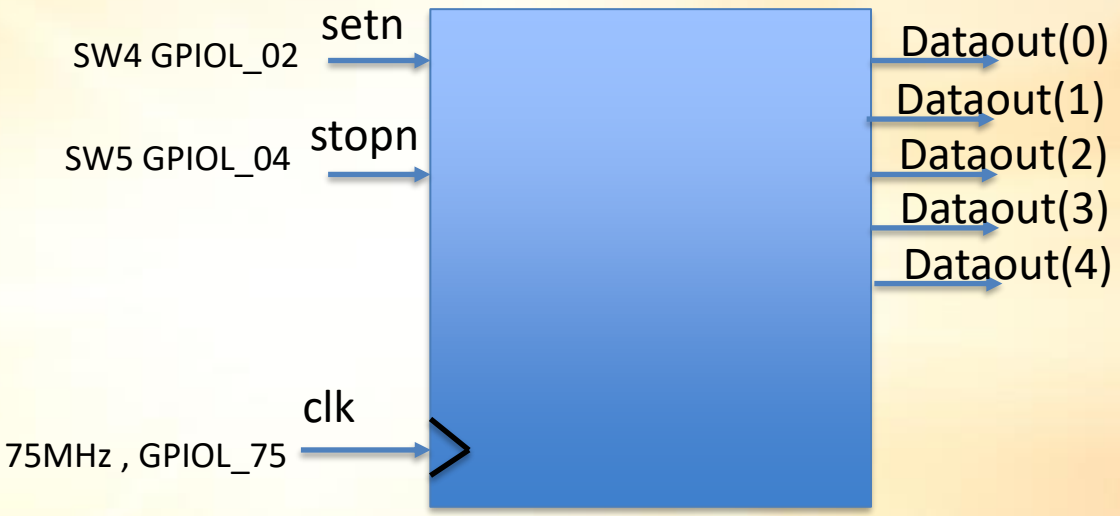






EFINIX®

# Assign the Resource setn=>GPIOL_02

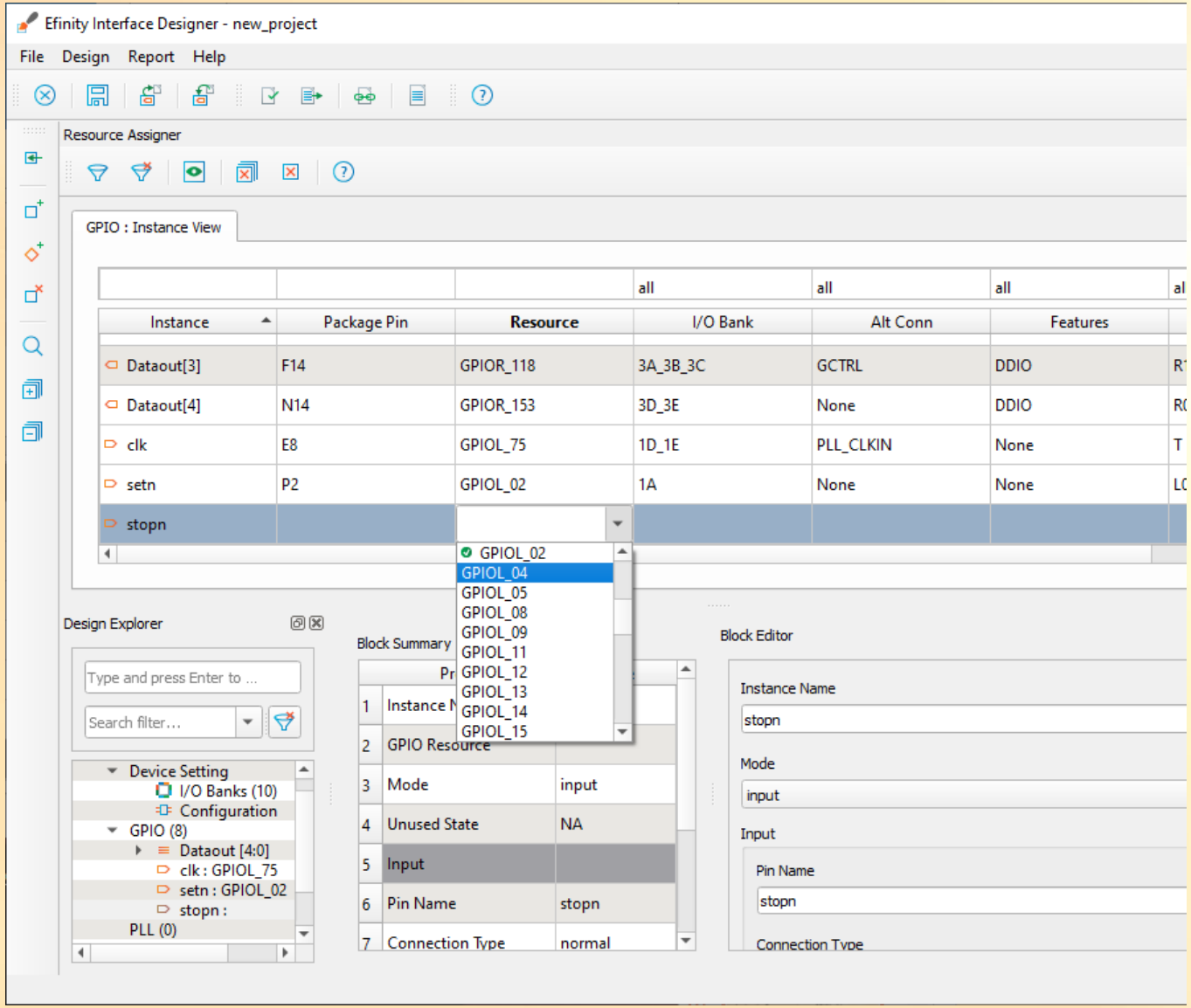


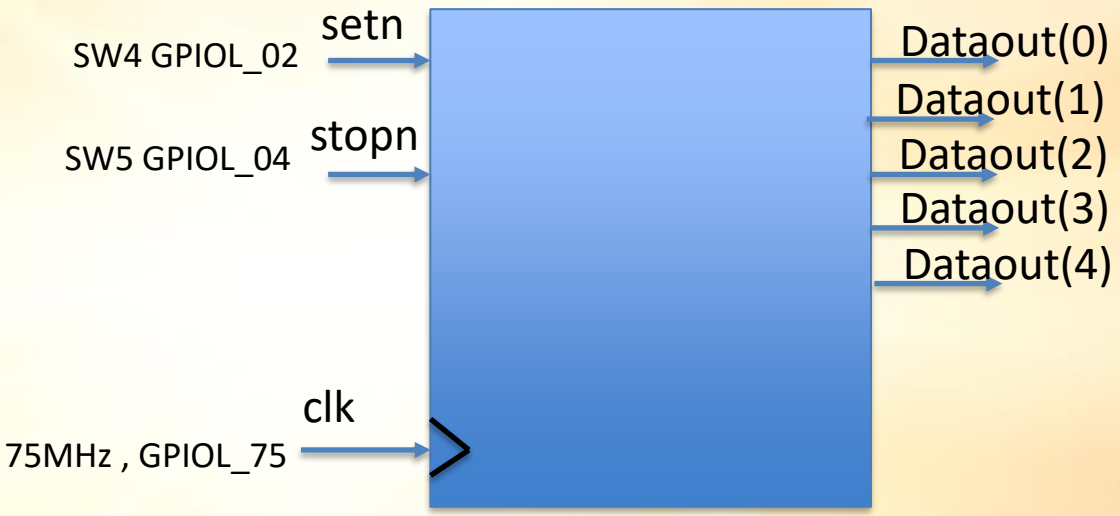






EFINIX®

# Assign the Resource stopn=>GPIOL_04

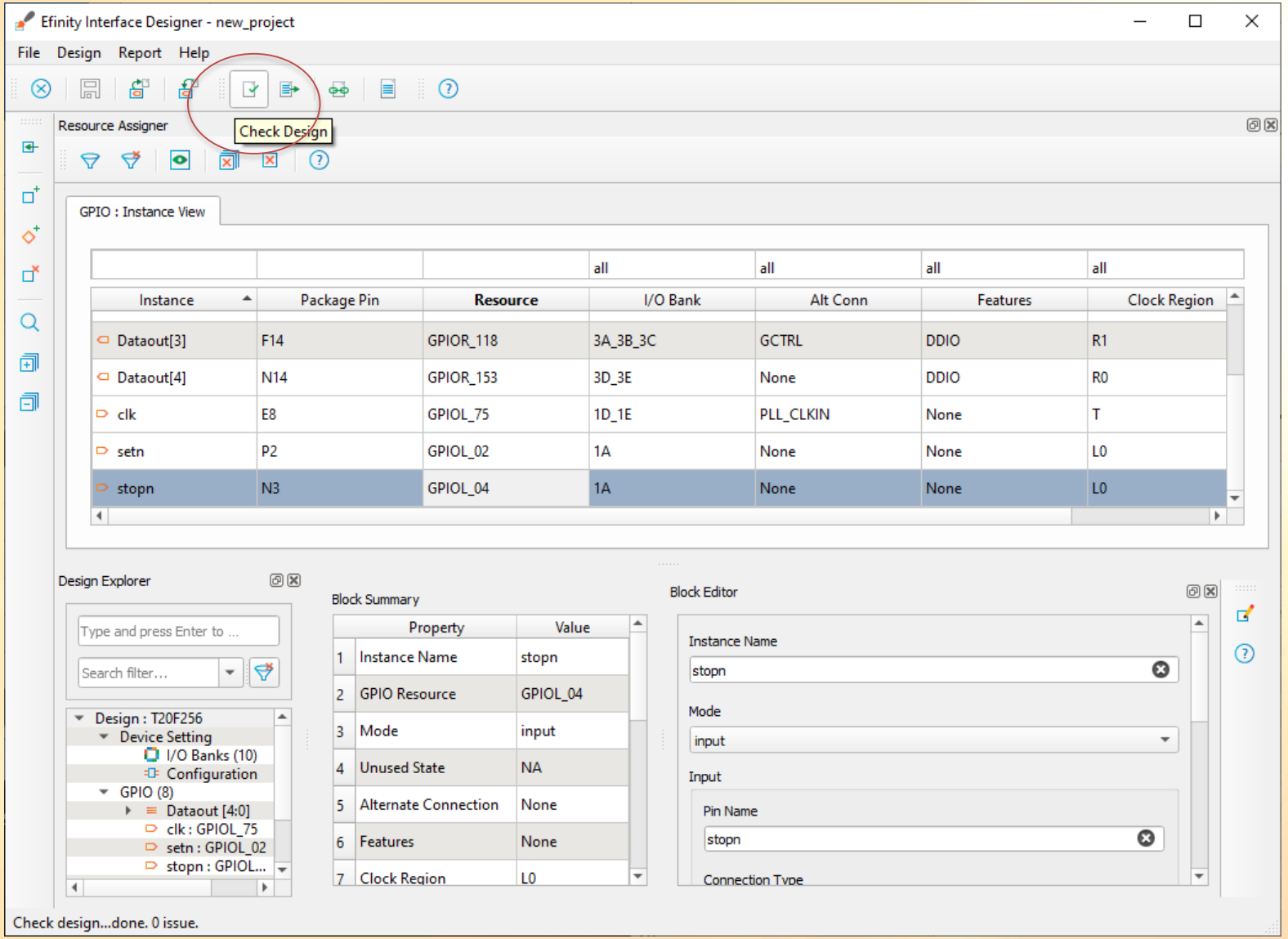


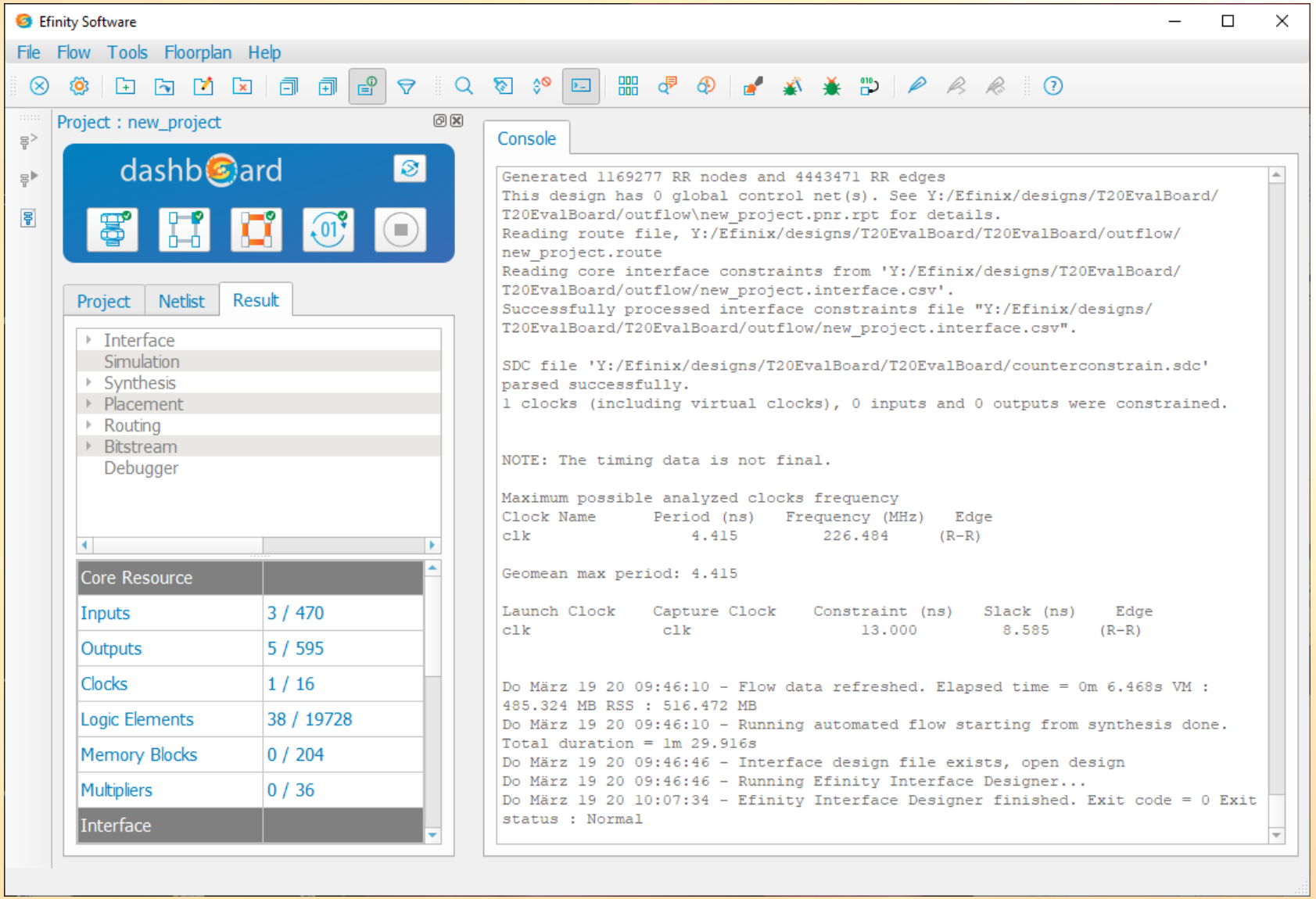



EFINIX®

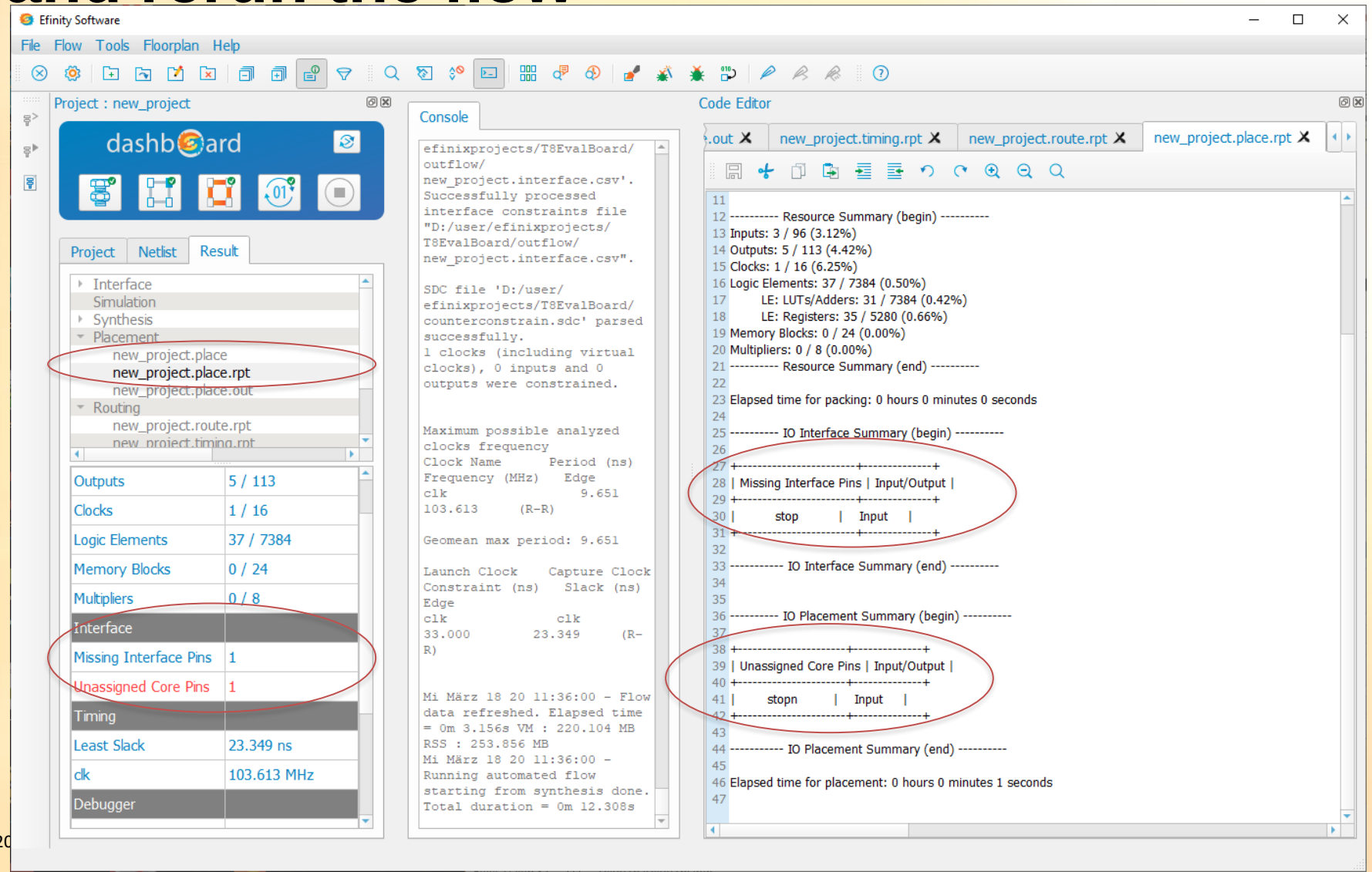
# Check the design and close the interface designer



# Run the flow with clicking on placement



# If you have an unassigned pin, check the Placement report, go back to interface designer and fix the name and rerun the flow

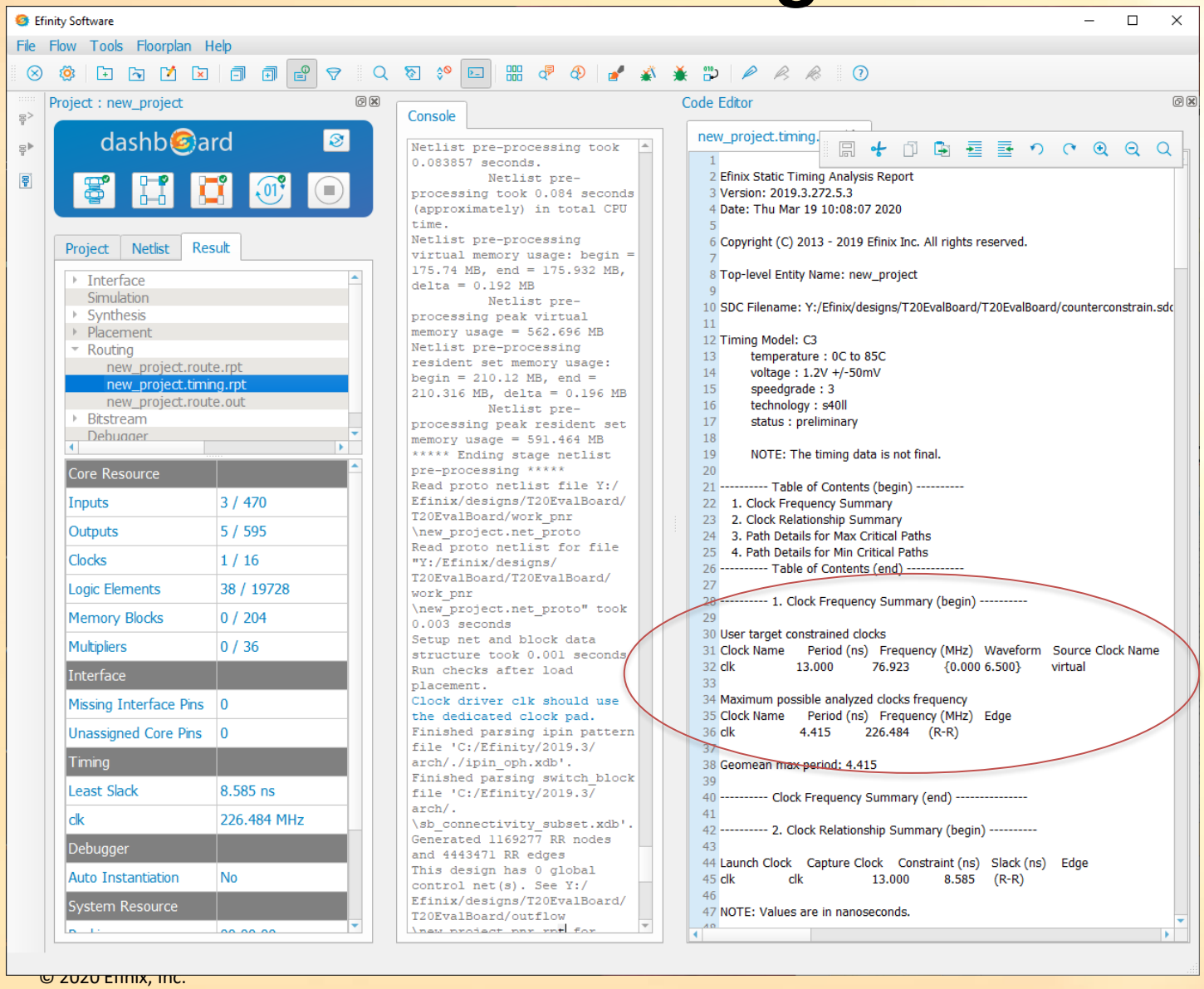



Typo in Interface designer: **stop** instead of **stopn**



EFINIX®

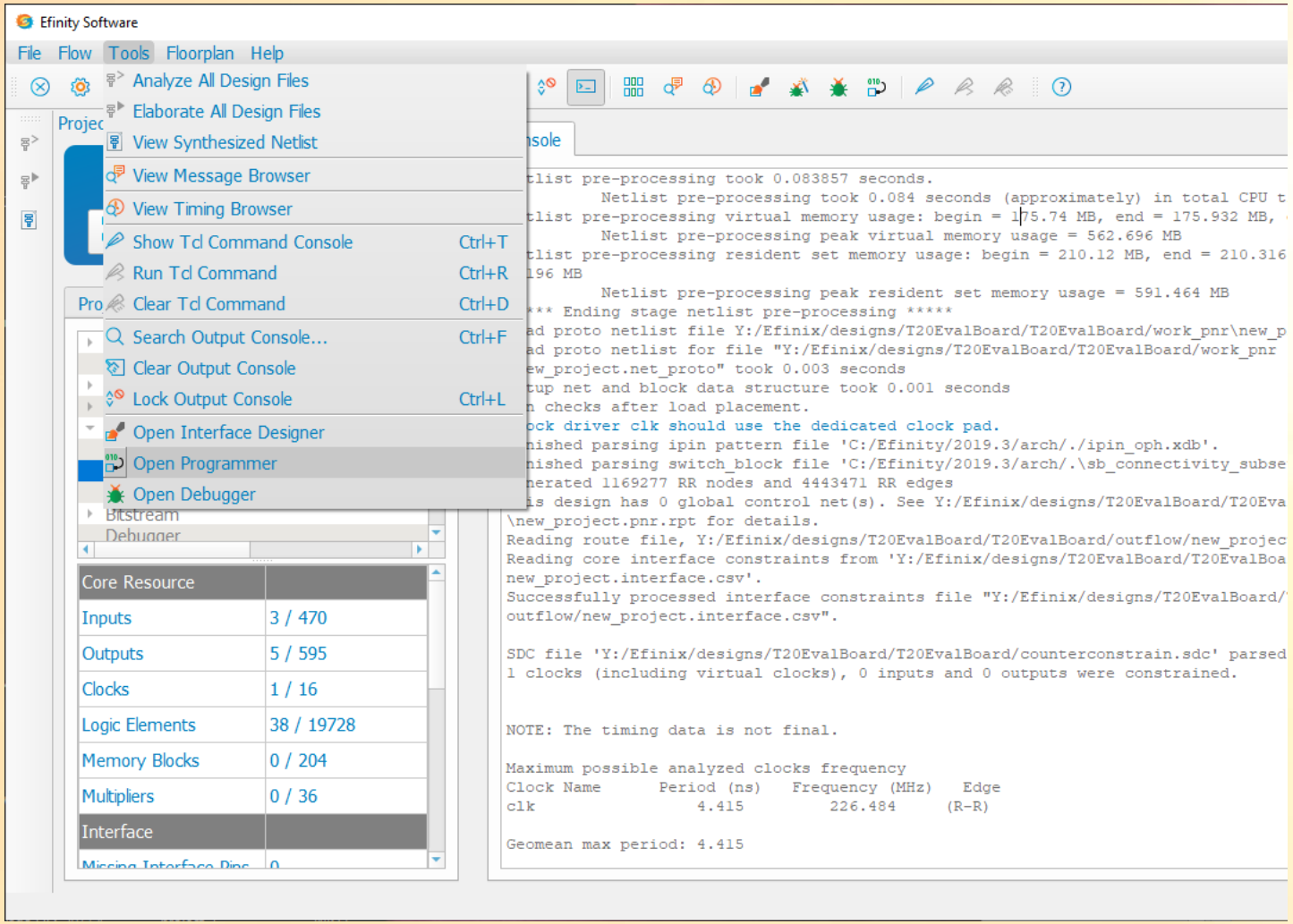
# Check static timing




To check the static timing select routing_new_project.timing.rpt.

Here you will find the constrains from the constrain file (clk :13ns) and the result

# Program the Device



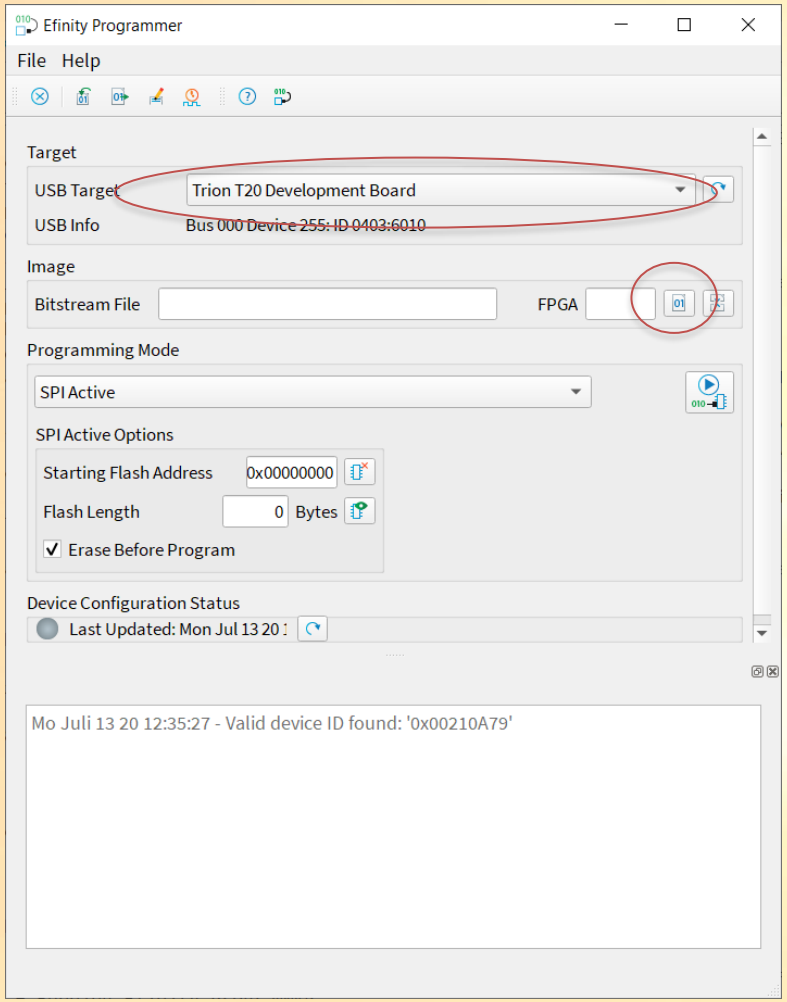

Open Programmer

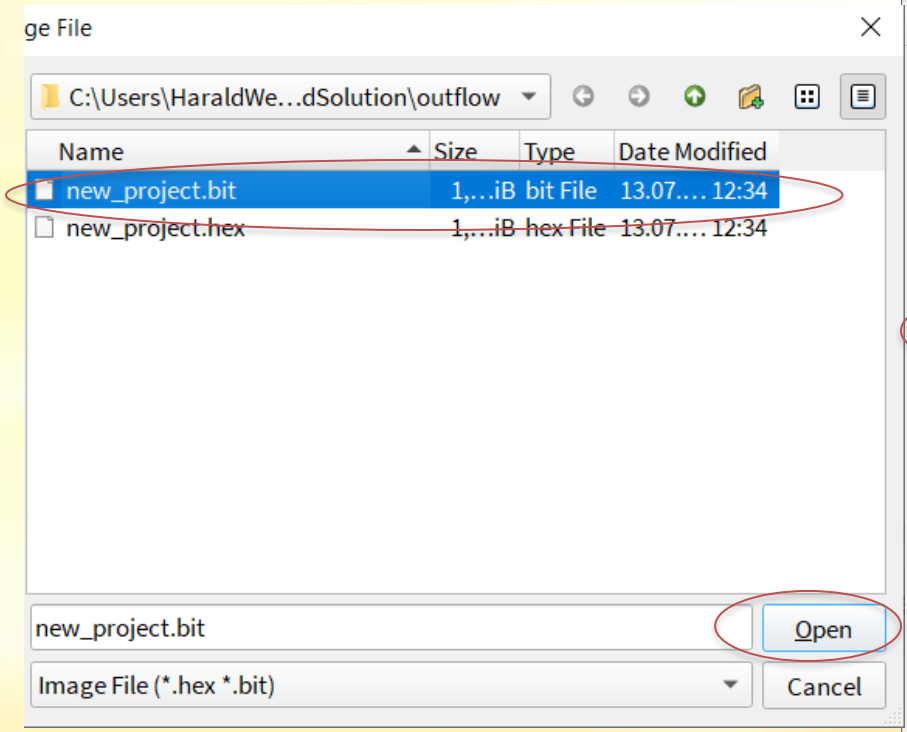
Tools->OpenProgrammer

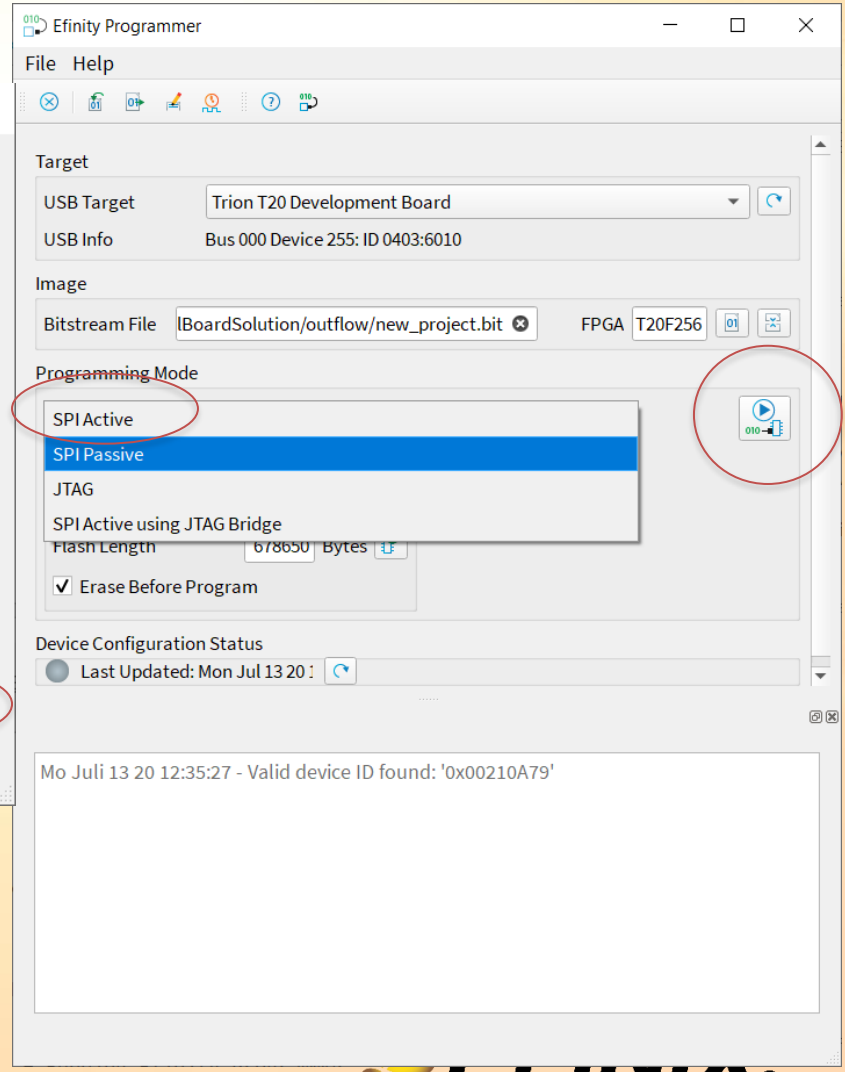


EFINIX®

# Select Image File, Start Program to the SPI Flash (Check USB Target Trion T20 Development Board)



# Problem solving

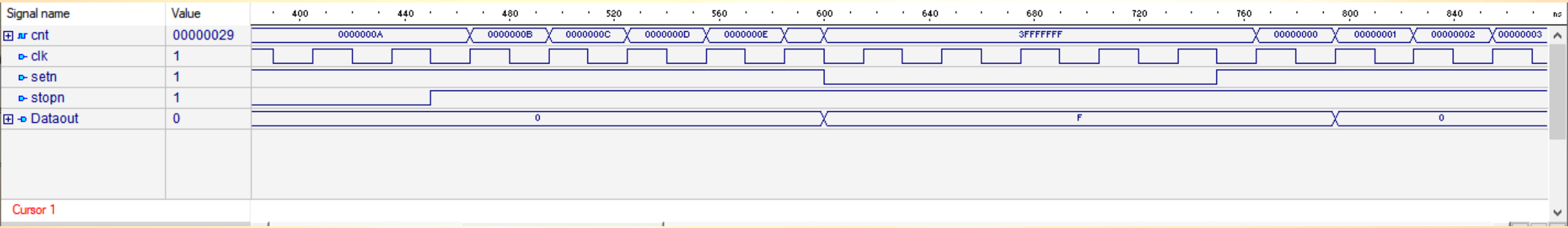
- If you can not see the Trion T20 Development Board, check if you installed the corect USB driver with zadig



EFINIX®

# Simulation

- For HDL simulation you can use any VHDL simulator. The testbench is included in the design.
- Here a wave form the design with testbench with an ALDEC HDL simulator.




- If you would like to simulate the synthesized netlist, include the new_project.dbg.map.v below the outflow folder and include the simulation libraries <installation path>\Efinity\2020.1\sim_models\verilog folder instead of the RTL Design

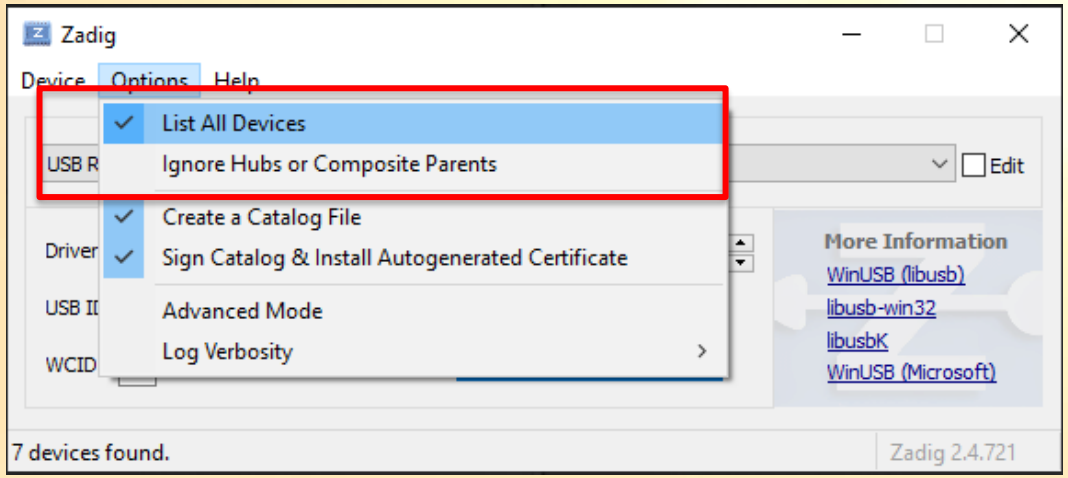


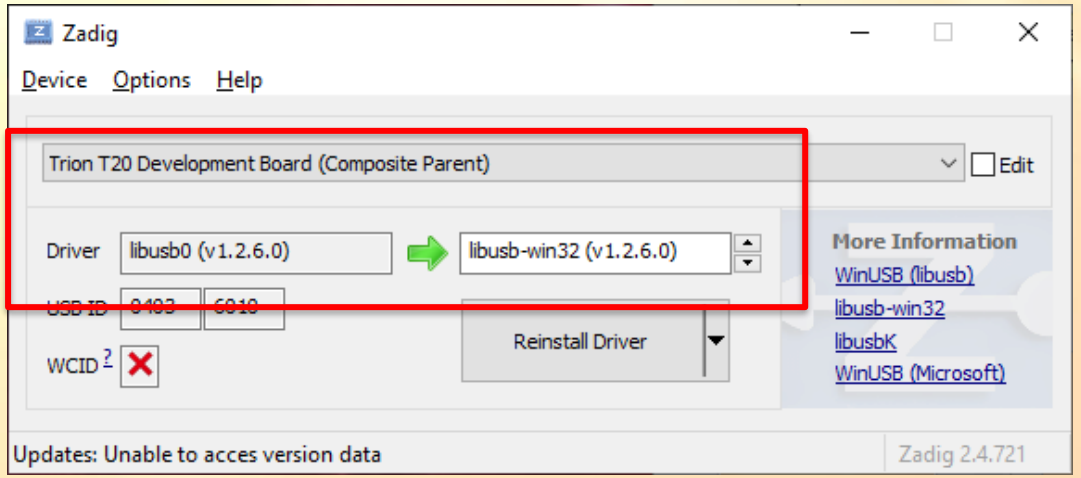
EFINIX®

# Design Debugging

# Driver Installation with zadig

- Check “List All Devices”
- Uncheck “Ignore Hubs or Composite Parents”
- Select “Trion T20 Development Board (Composite Parent)”
- Replace with **libusb-win32** driver



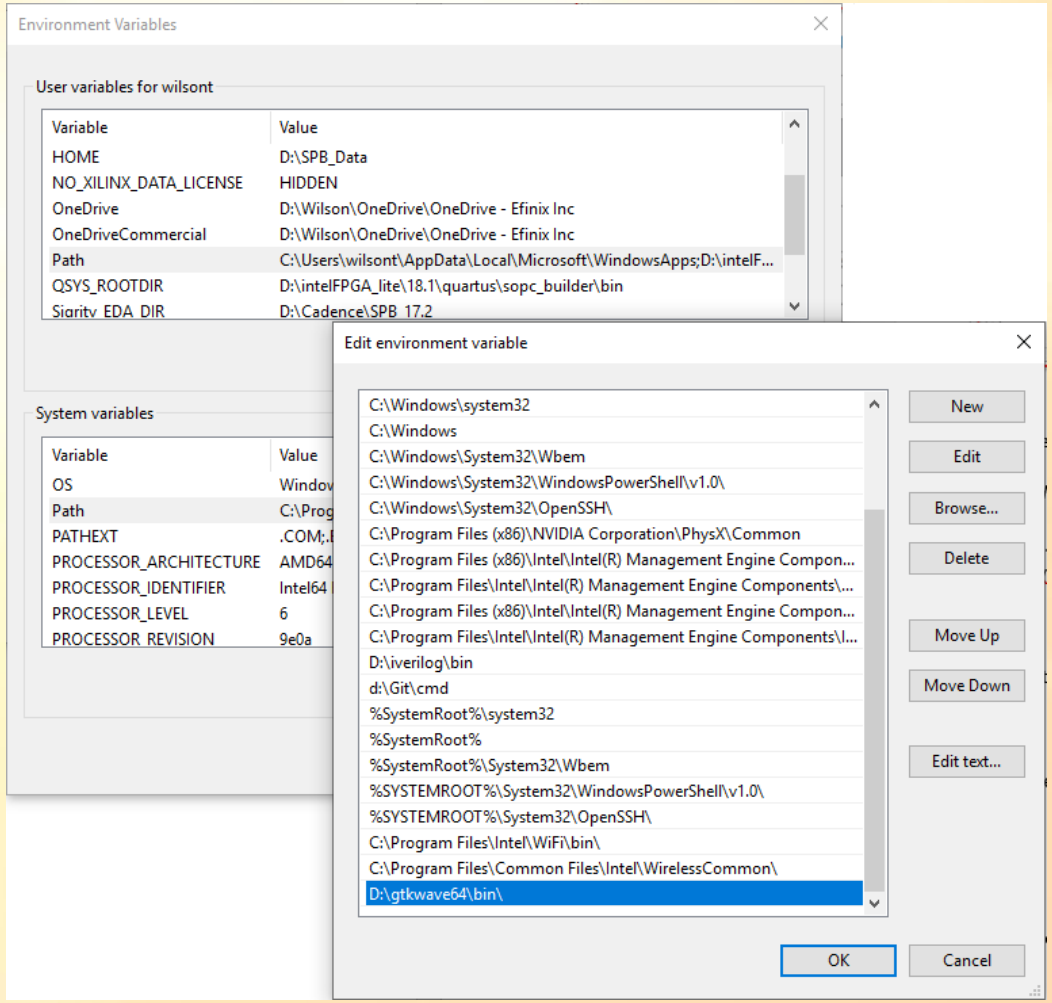






EFINIX®

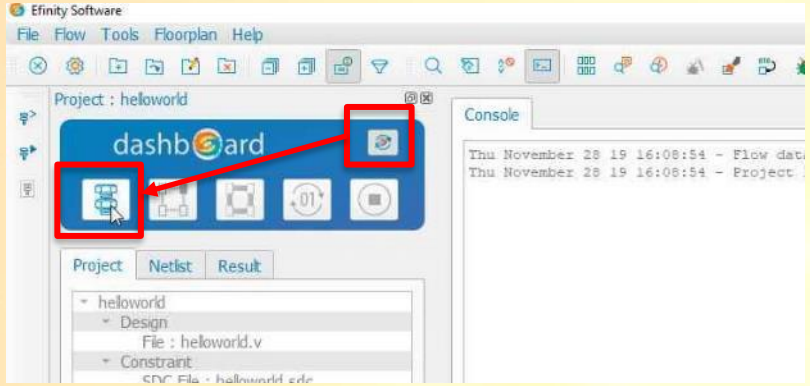
# GTKWave

- Install GTKWave
- To launch GTKWave automatically after data capture from Logic Analyzer, please add **“$GTKWave_folder$\bin\”** into Path of System Variables

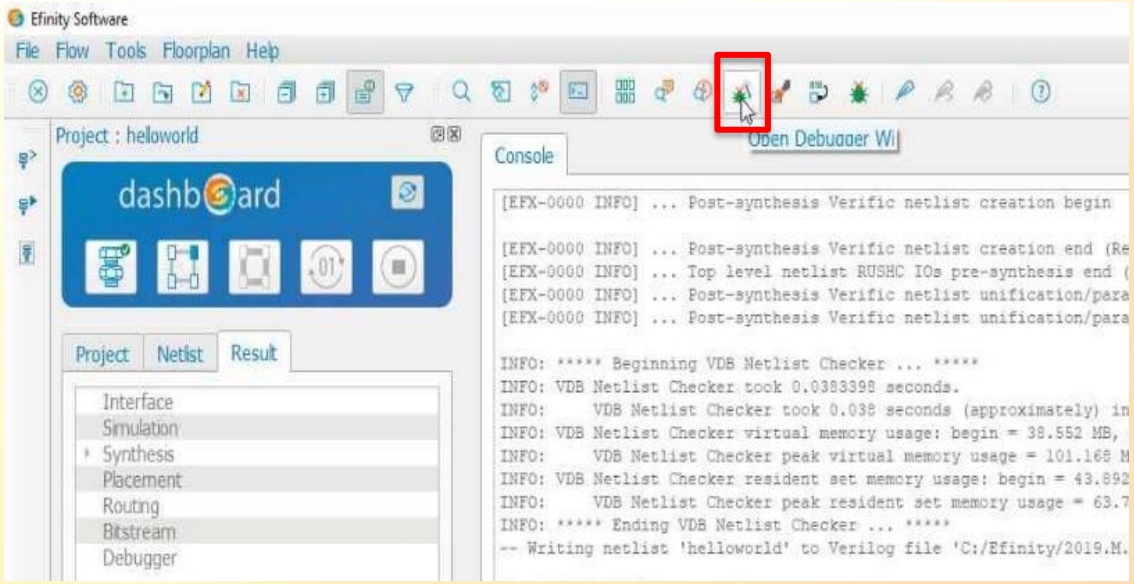


# Auto Instantiation of the Debug core

1. Allow step-by-step project flow, and run synthesis



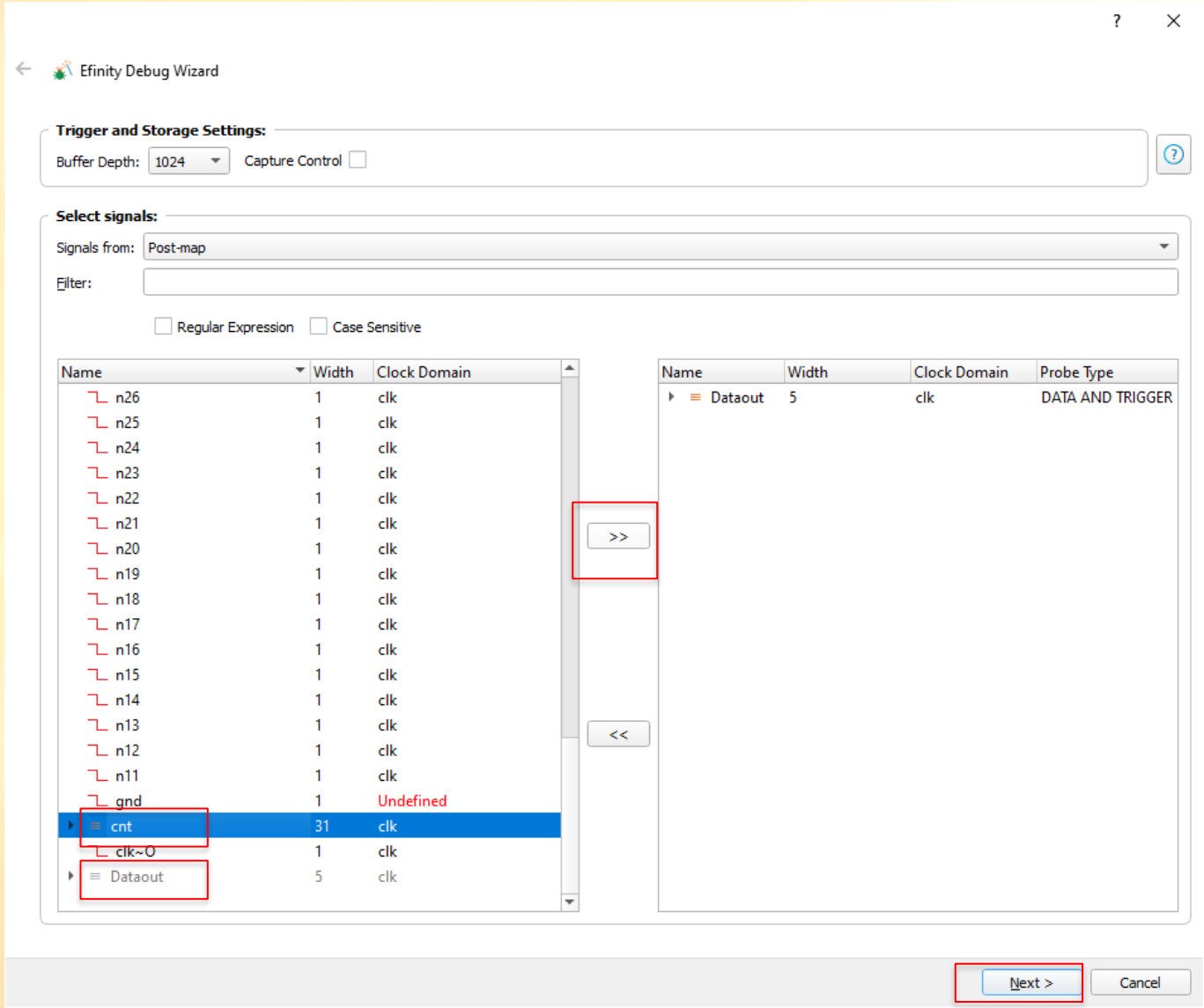

2. Start Debugger Wizard



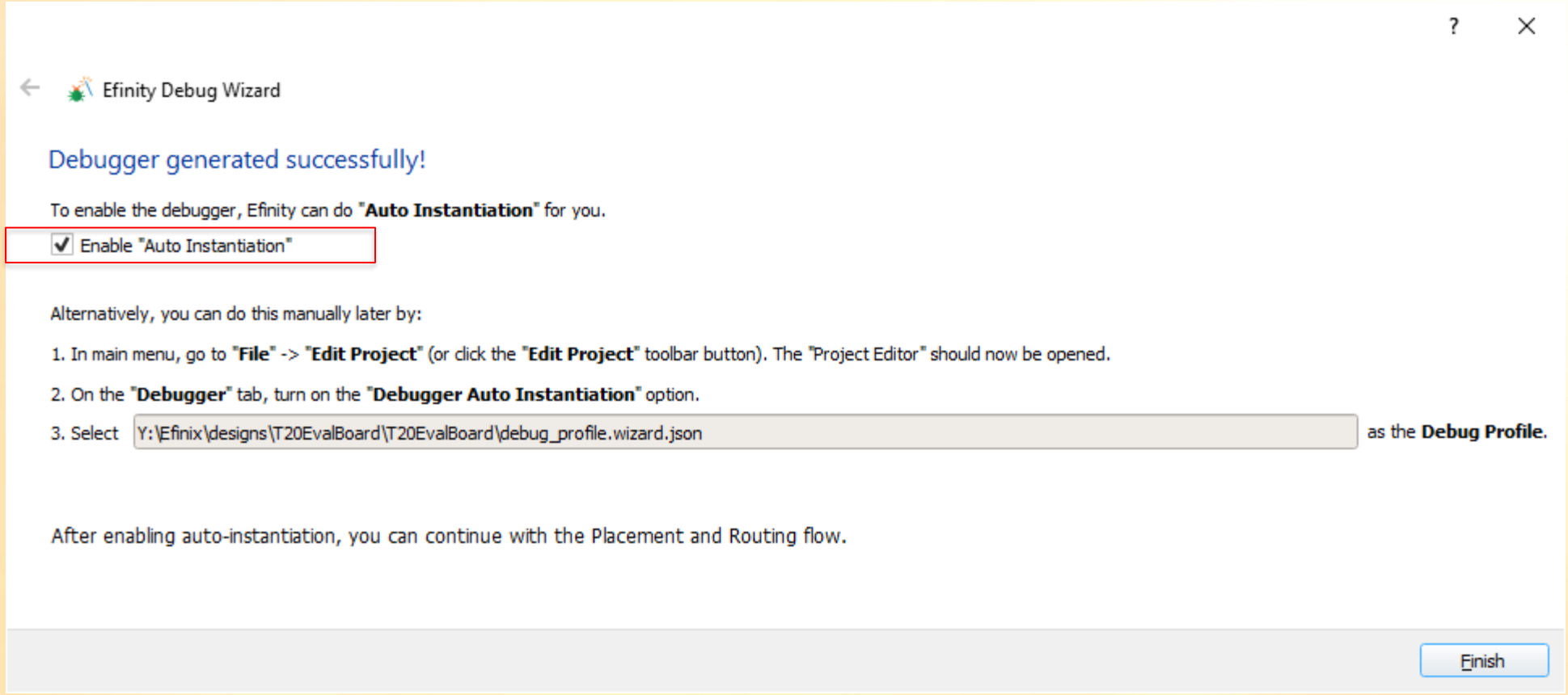



EFINIX®

# Select cnt and Dataout and klick >>

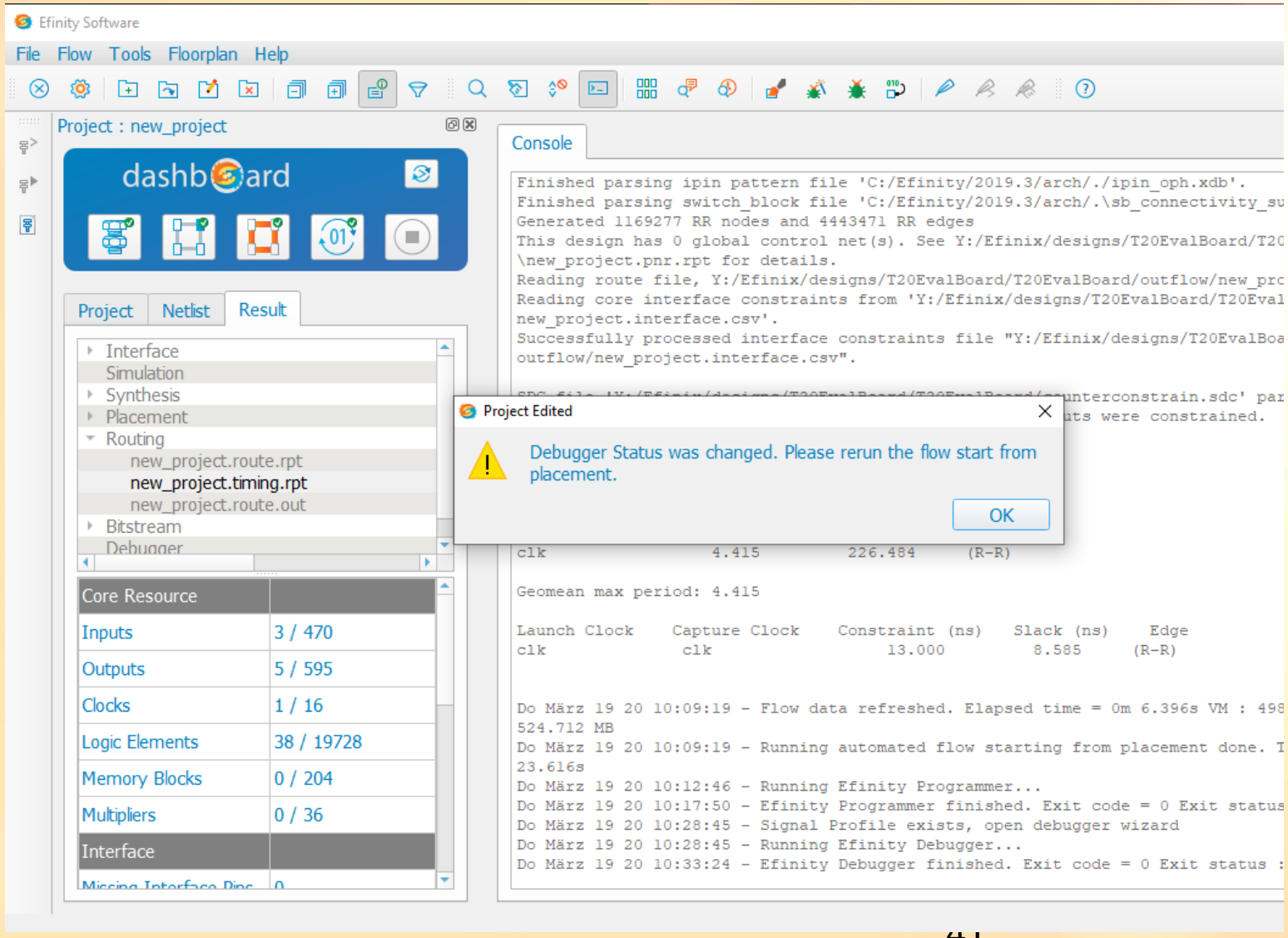


# Click Finish



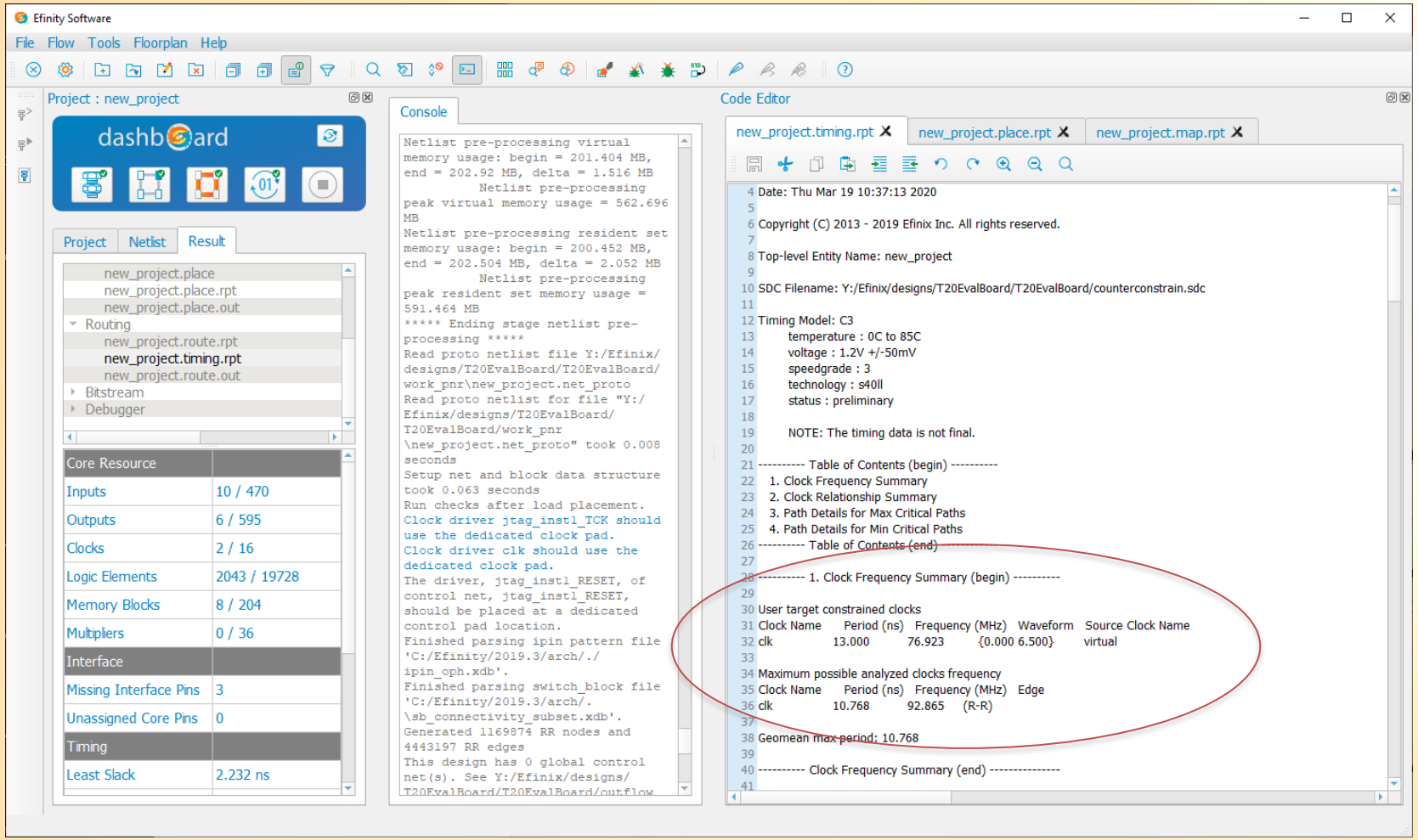

Enable Auto instantiation

# Click OK and run the whole flow with clicking 




- If the Automated flow button is grayed out click on the button to activate the automated flow.  
- Click on the synthesis icon and the flow will run automnatically 

# Check static timing




To check the static timing select routing_new_project.timing.rpt.

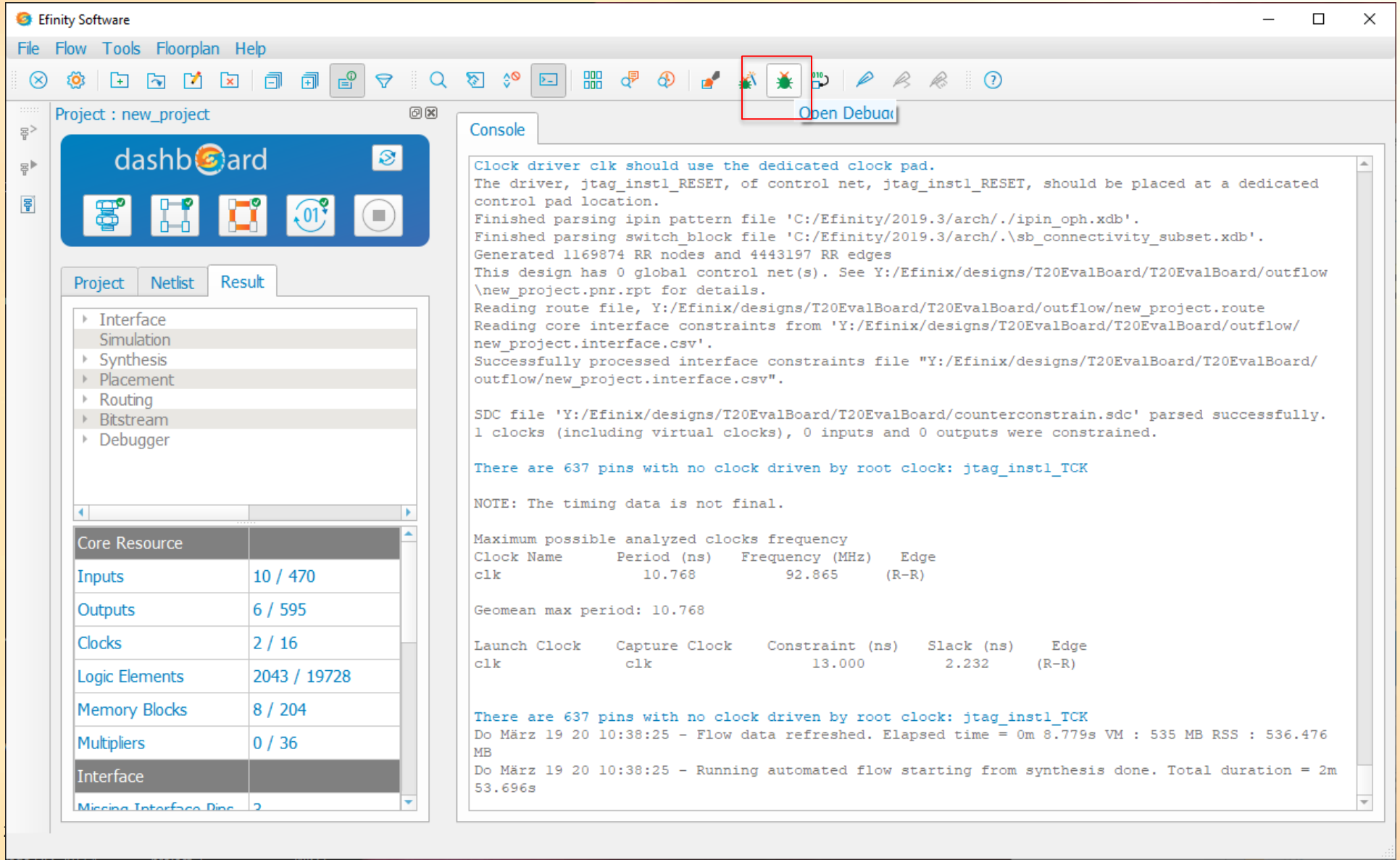
Here you will find the constrains from the constrain file (clk :13ns) and the result.

With the Debug core the design must meet the timing constrain to ensure correct behavior !



EFINIX®

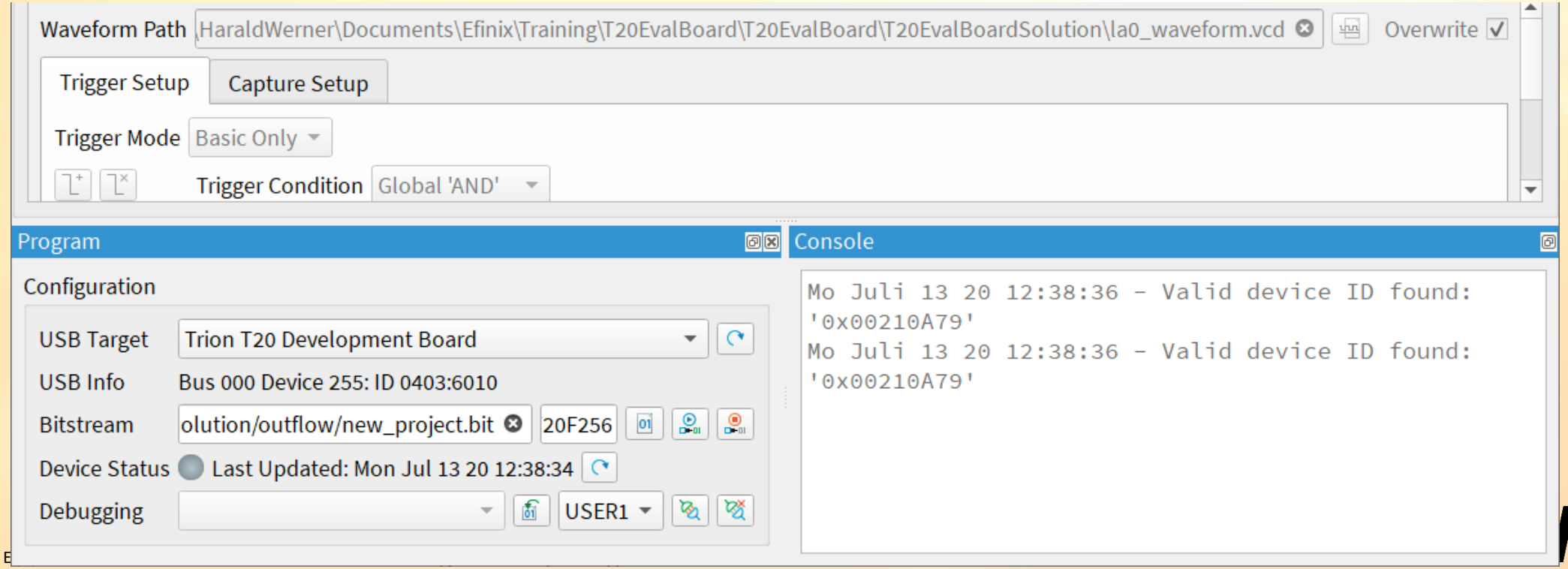
# Open Debugger



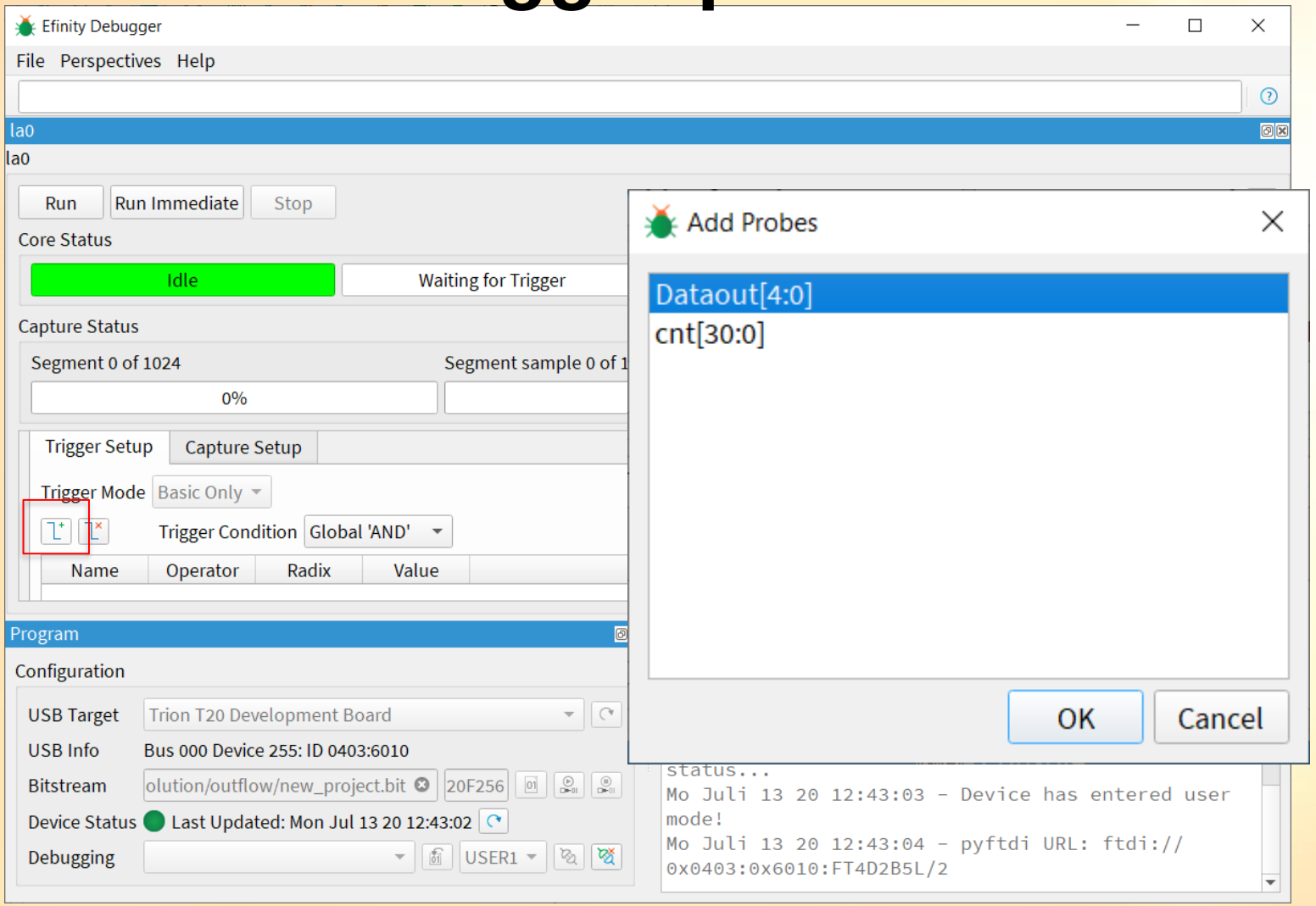

If Debugger is not starting, please double check the driver setting with ZADIG

# Program the Device via JTAG

- Select hex file from you project
- Start programming
- Connect Debugger



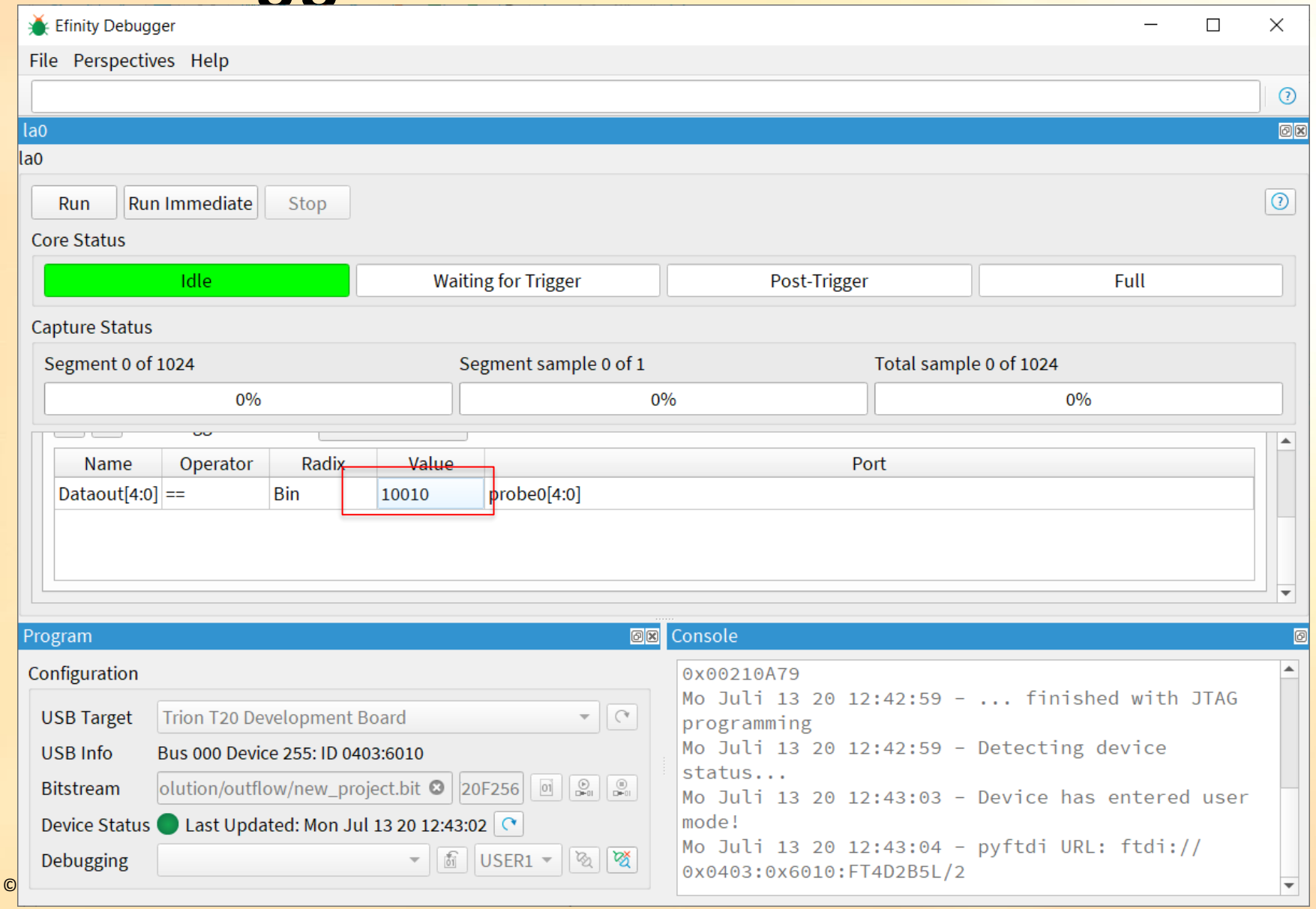
# Define a trigger point




Select Trigger

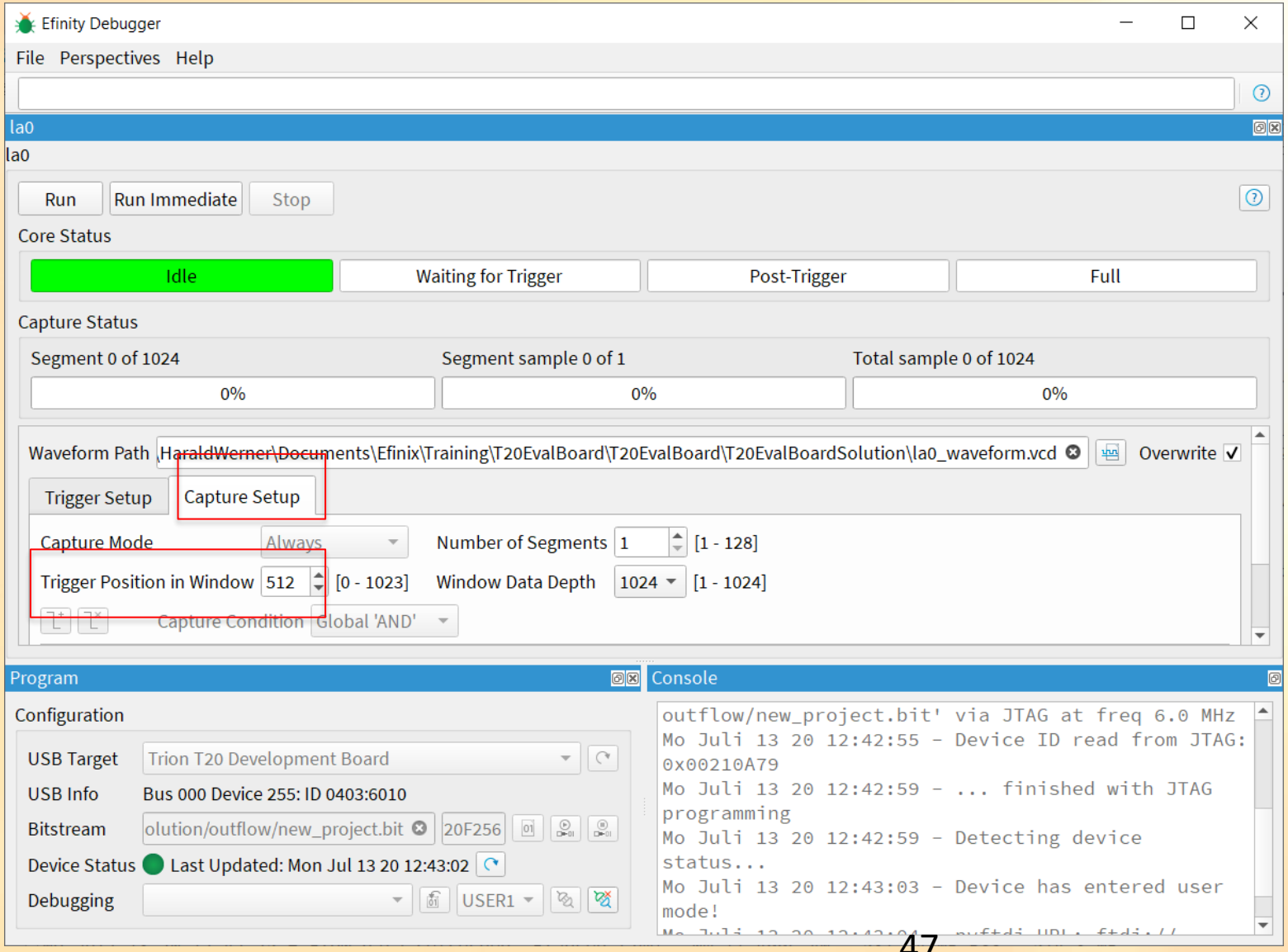
Select Dataout

# Set trigger value



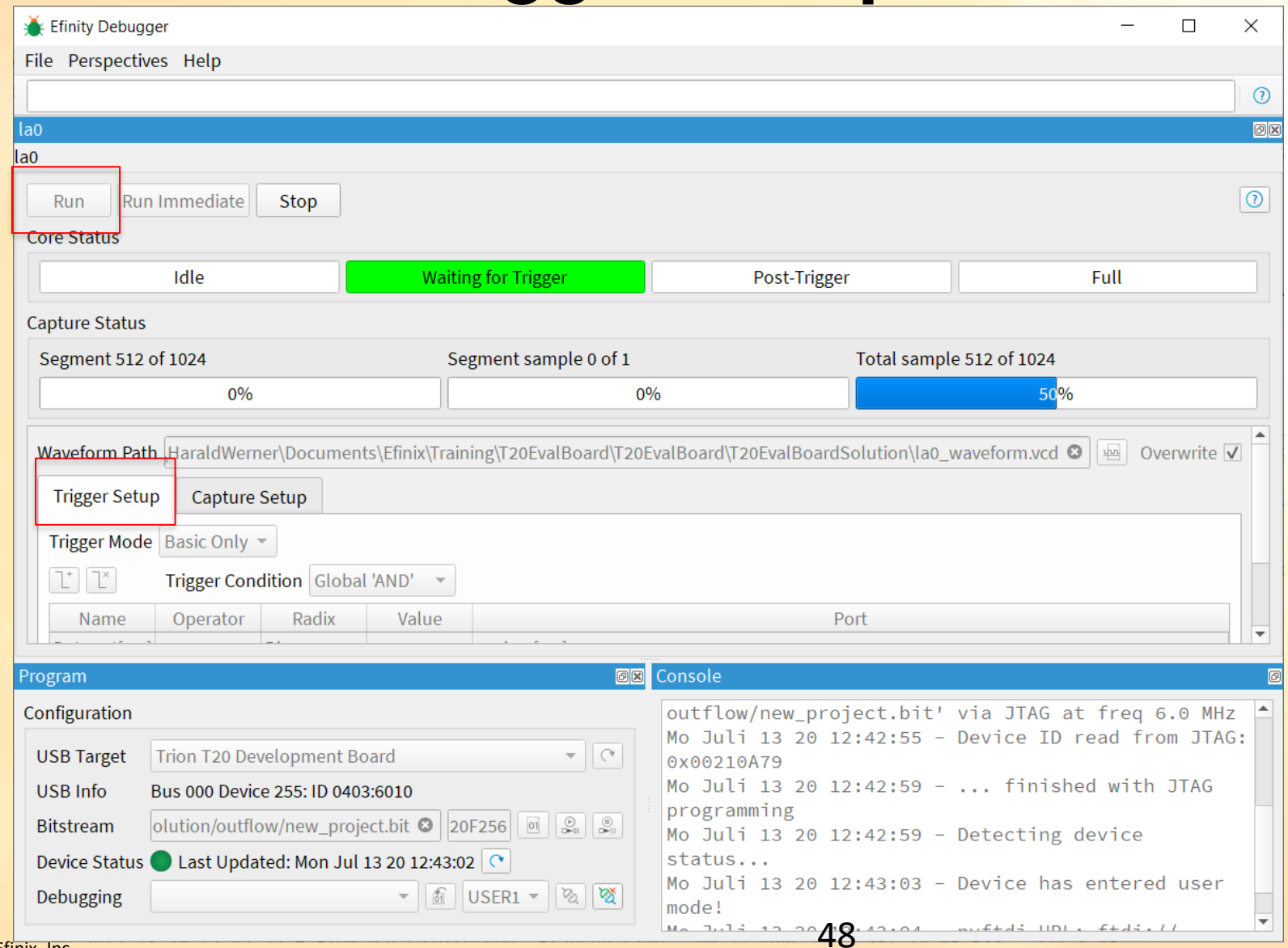

- Select Value
- Enter e.g.: 10010

# Set the trigger position with clicking on Capture SETUP

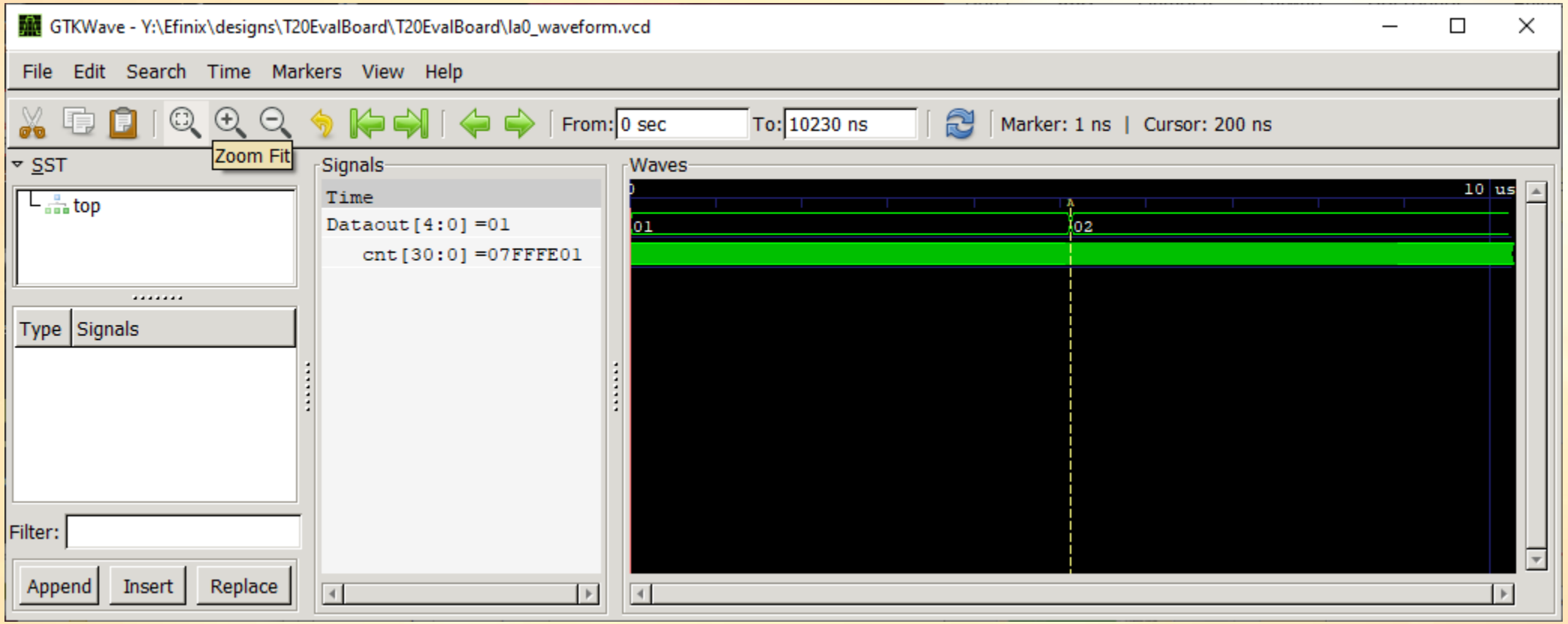



- Trigger position is default in the middel of the memory. Here @512 (of 1024)

# Go back to Trigger Setup and run the Core

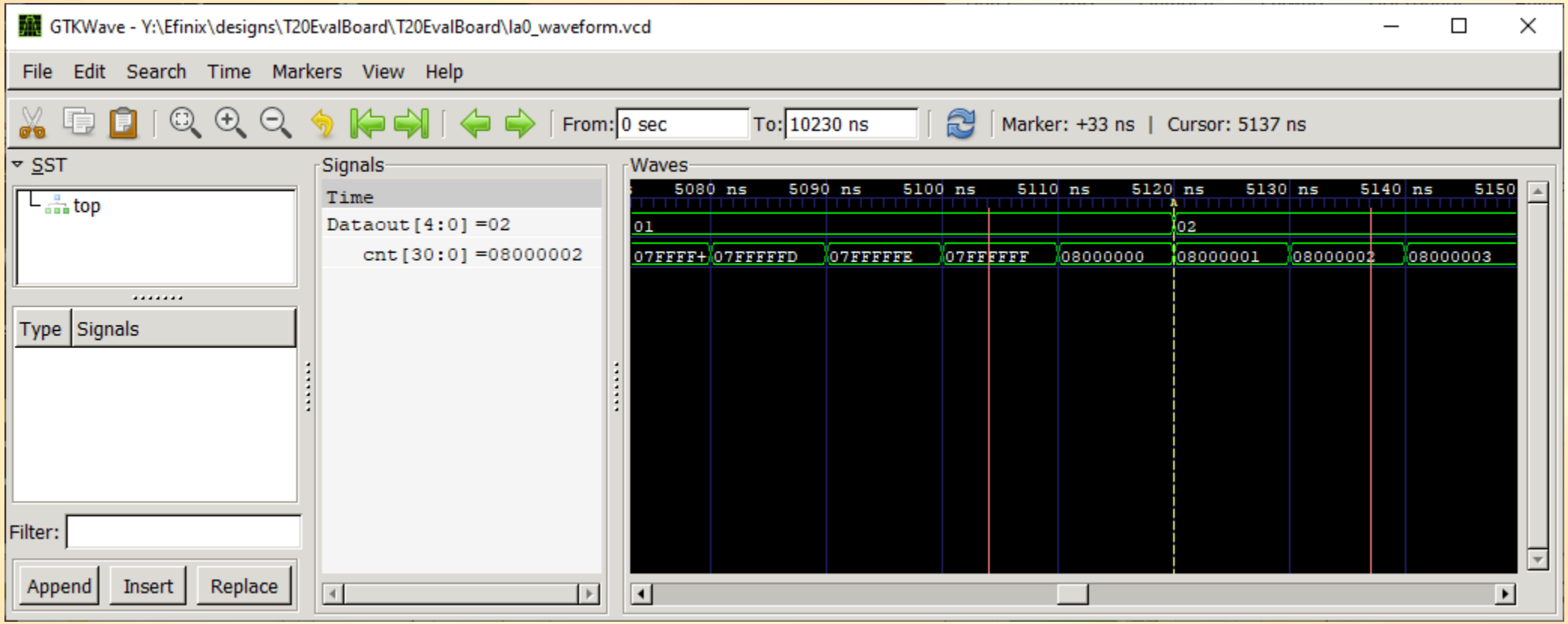


# GTKWave will open automatically



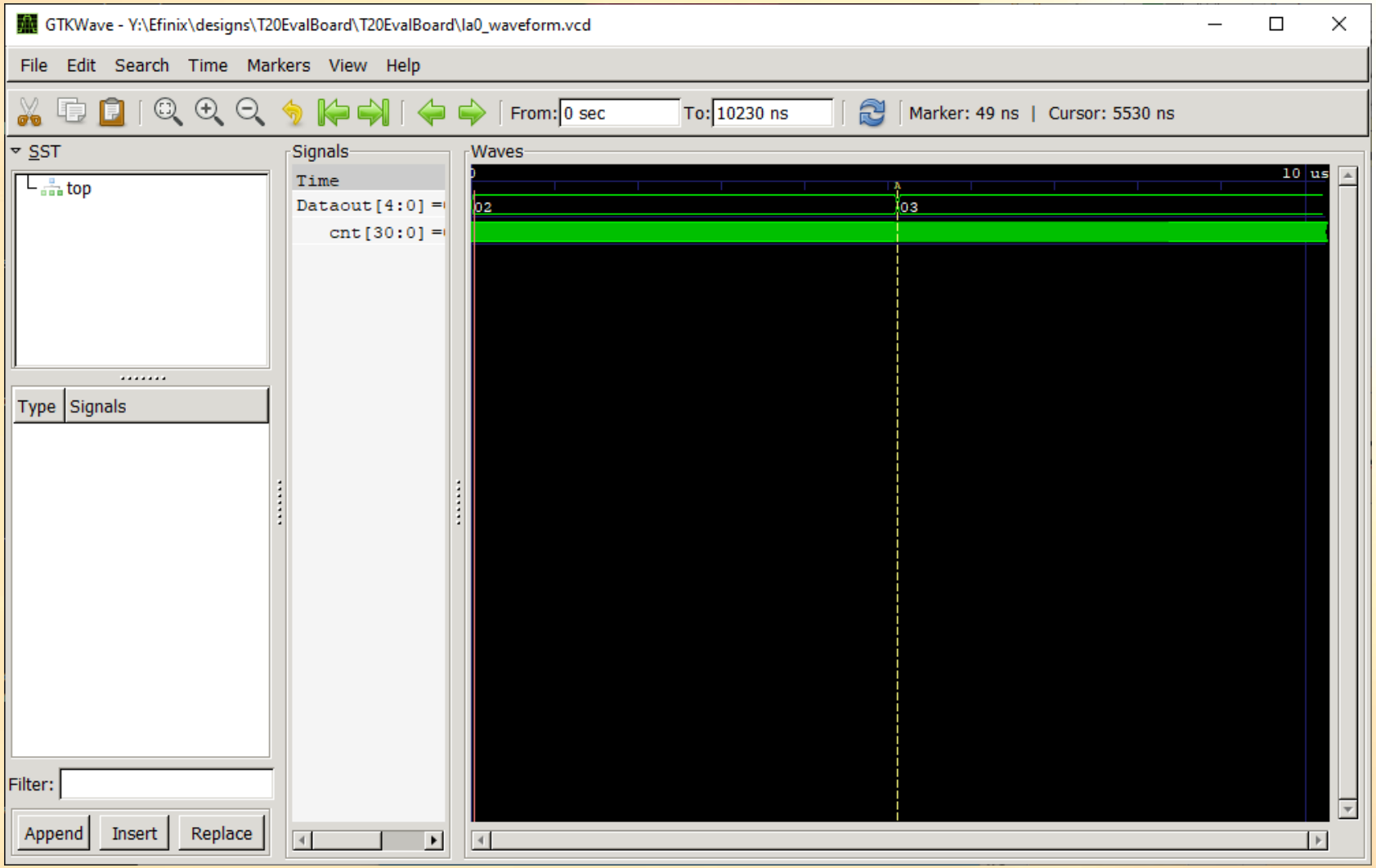

- Click on Zoom fit to see the whole content

# With the RMB you can zoom in




- Place the cursor of the area of interest, press RMB and move the cursor, release RMB

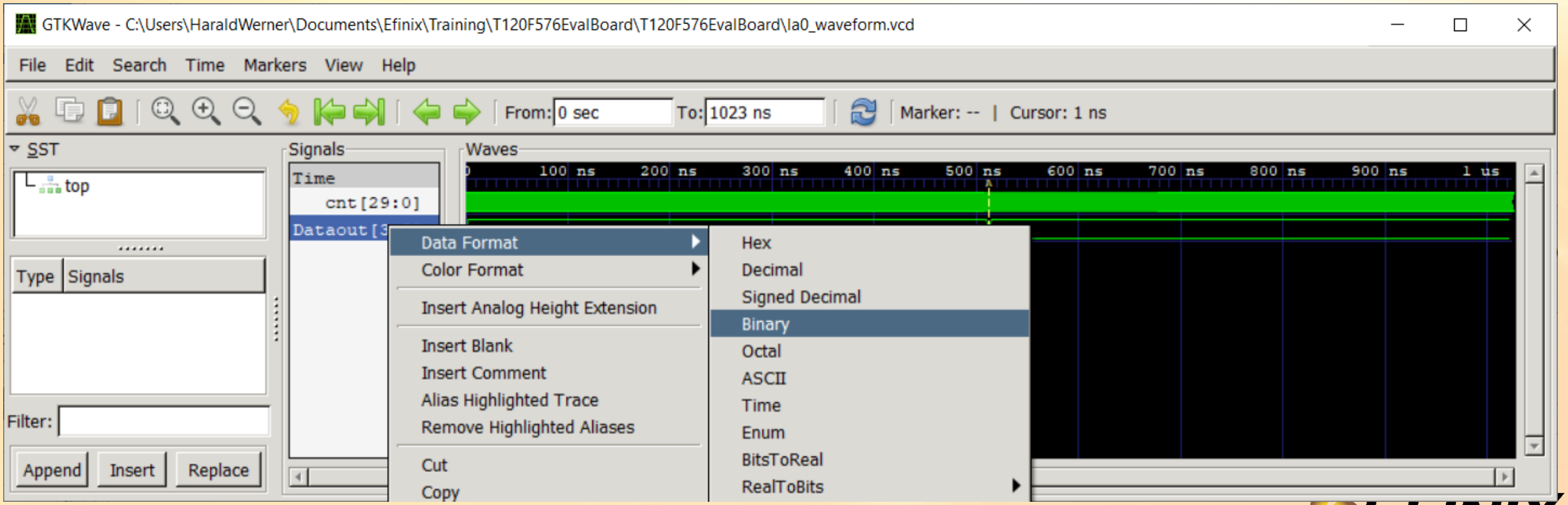
# You can change the trigger value and run again



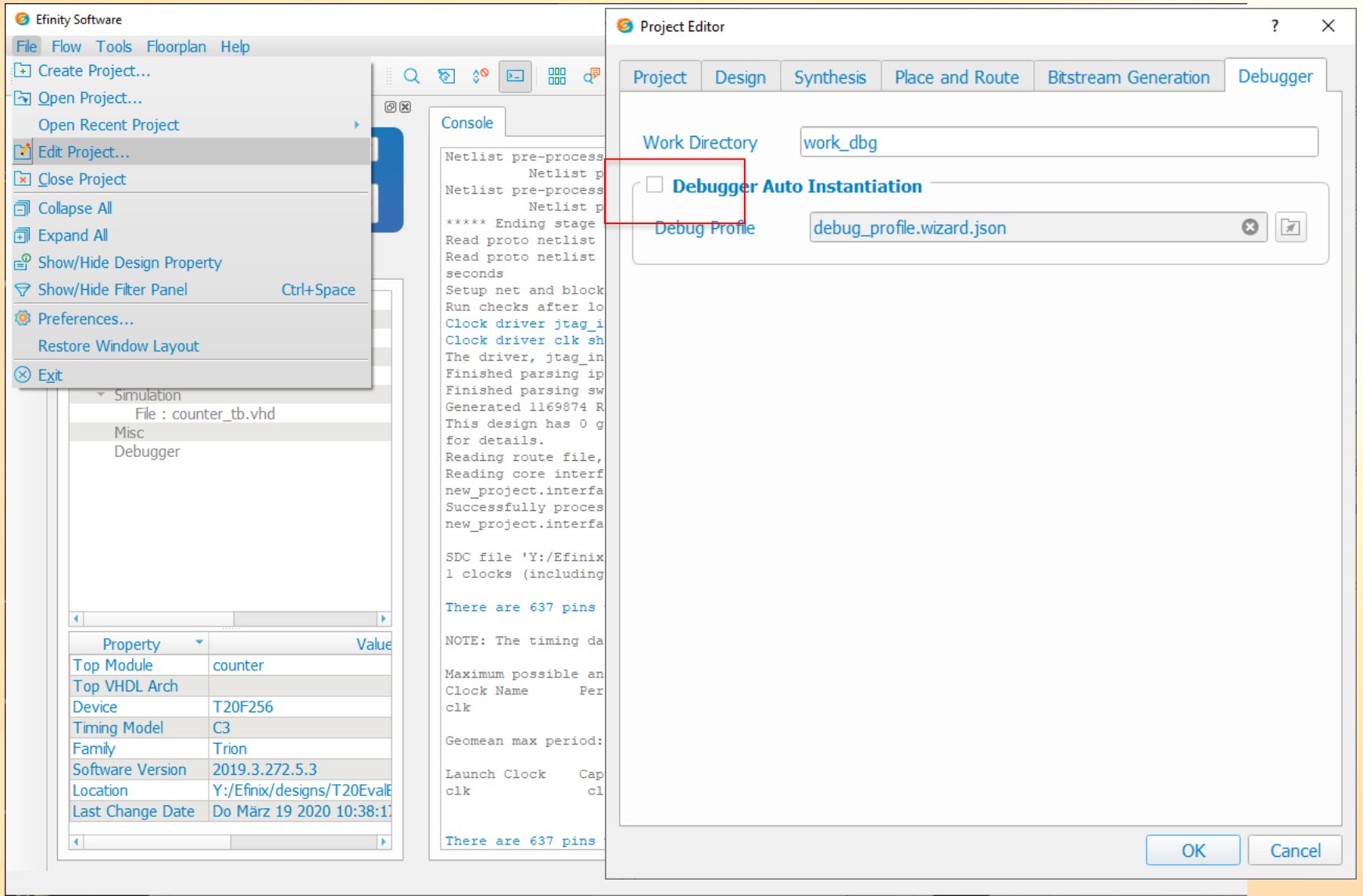

- E.g: change to 00011
- Press Run

# Change Radix

- To change the radix of the signal, select the signal, RMB->Data Format->Binary



# Remove the debug core




Deselect the Debugger Auto Instantiation.

Click OK

Run the whole Design flow

EFINIX®

# Information

- If you need the complete Project, extract the T20EvalBoardSolution.zip file. Here you will find the project with the I/O pin assignments.