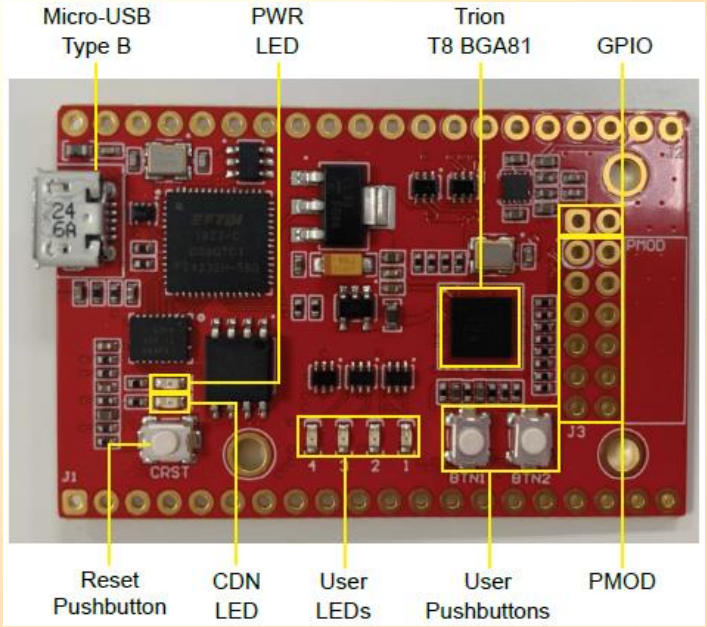
EFINIX®

Accelerating Your Innovation

Trion Xyloni T8 LAB

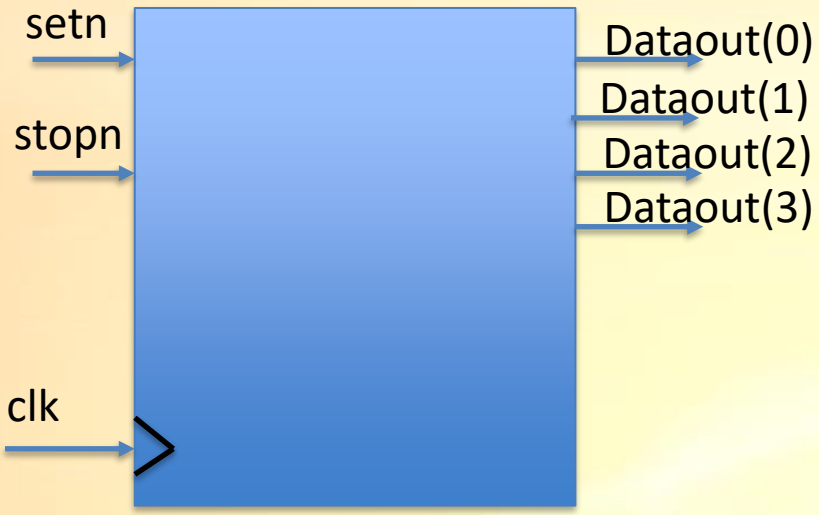
By Harald Werner

Version: 1.0.0



EFINIX®

Design (simple up counter with set and stop)



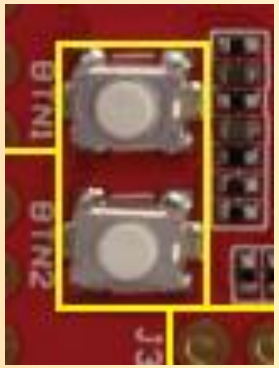
Code Editor

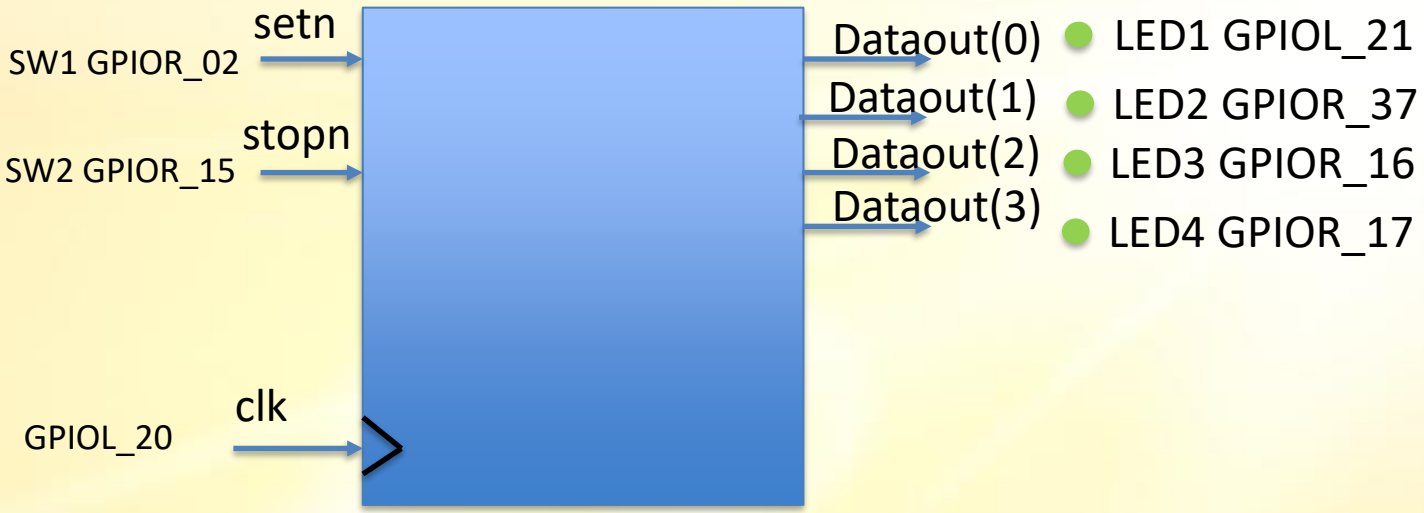
counter.vhd

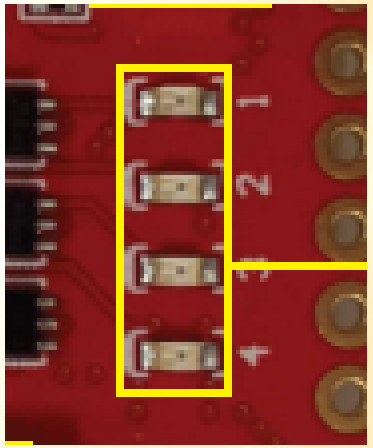
```
1  -- smal example design for the Xyloni Efinix Eval Board
2  -- By Harald Werner
3  -- 15.10.2020
4  library ieee;
5  use ieee.std_logic_1164.all;
6  use ieee.std_logic_unsigned.all;
7
8  entity counter is
9  port ( clk          : in std_logic;                          -- clock input. Could be from internal osc (T8) or from the external 33.3MHz clock use GPIOl_20
10          setn       : in std_logic;                          --  Set signal, low active; sett all outputs to '1' (LED are high active, means all LEDs msut be ON) GPIOR_02
11          stopn      : in std_logic;                          -- Stop signal, low active Stop counting GPIOR_15
12          Dataout    : out std_logic_vector(3 downto 0));     -- Output data connected to the LEDs (low high); GPIOR_17,GPIOR_16,GPIOLR37,GPIOL_21
13 end counter;
14
15 architecture vers1 of counter is
16 signal cnt: std_logic_vector ( 29 downto 0) := (others => '0');
17 Begin
18 cnt_proc : process(clk, setn)
19              Begin
20                  if setn = '0' then
21                      cnt     <= (others => '1');
22                      dataout <= (others => '1');
23                  elsif clk'event and clk = '1' then
24                      if stopn = '0' then
25                          cnt <= cnt;
26                      else
27                          cnt <= cnt +1;
28                      end if;
29 --          Dataout <= cnt(17 downto 14);               --17 downto 14 OK for the 10Khz internal oscillator. For the 33MHz external clock use 29 downto 26
30                      Dataout <= cnt(29 downto 26);   --17 downto 14 OK for the 10Khz internal oscillator. For the 33MHz external clock use 29 downto 26
31                  end if;
32              end process;
```


EFINIX®

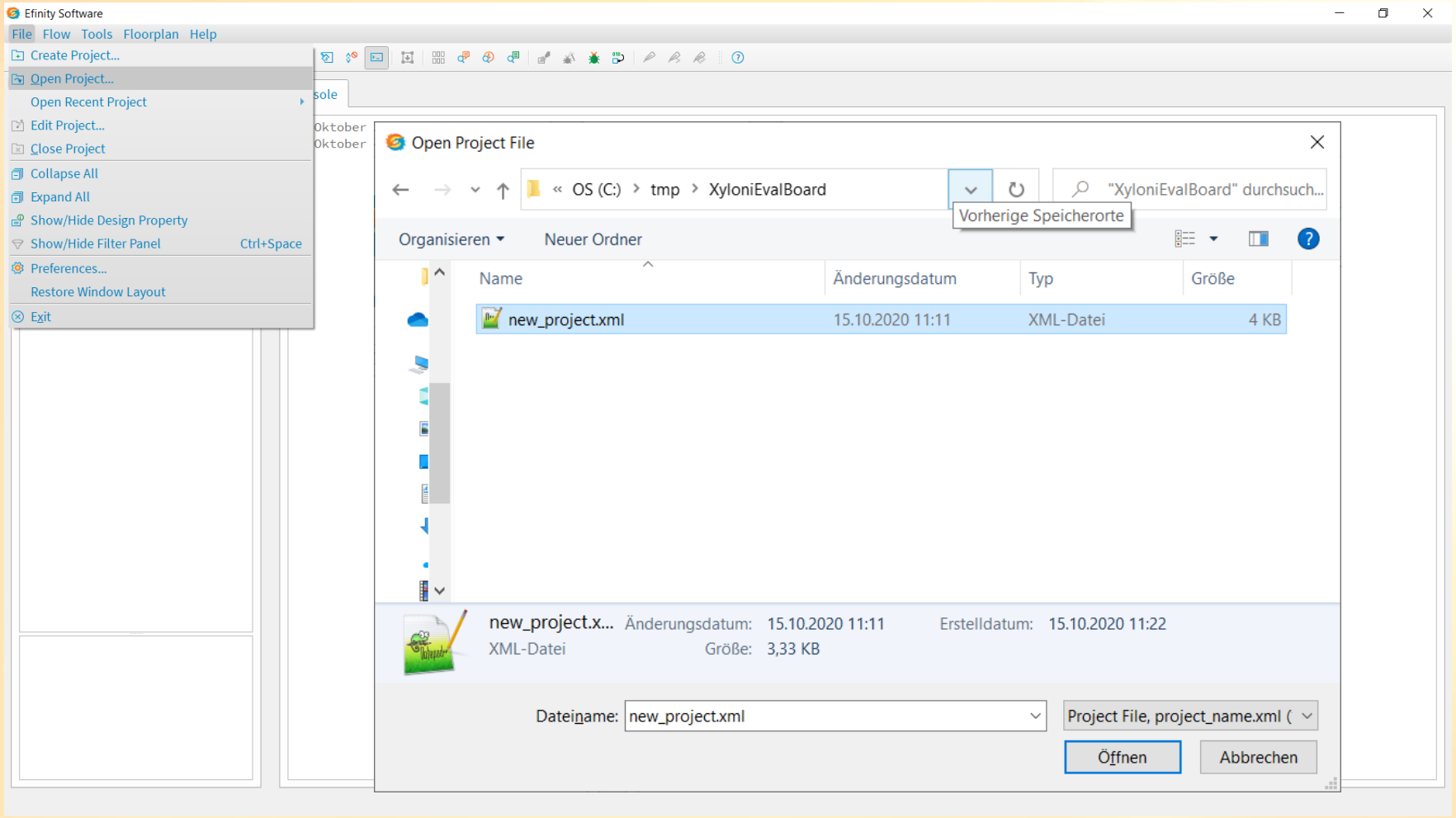
Design / Board connection



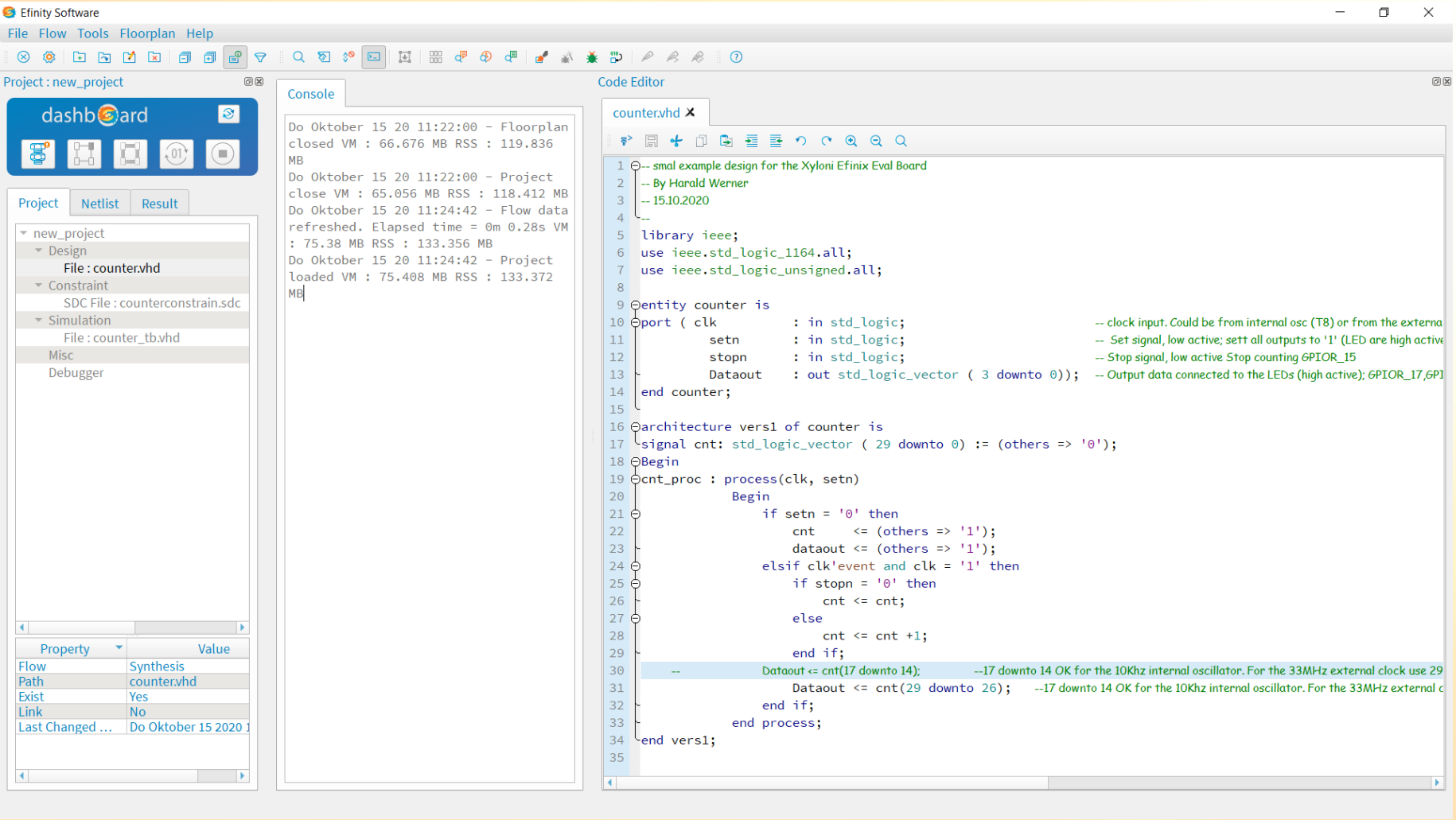




1. Open Efinity new_project.xml



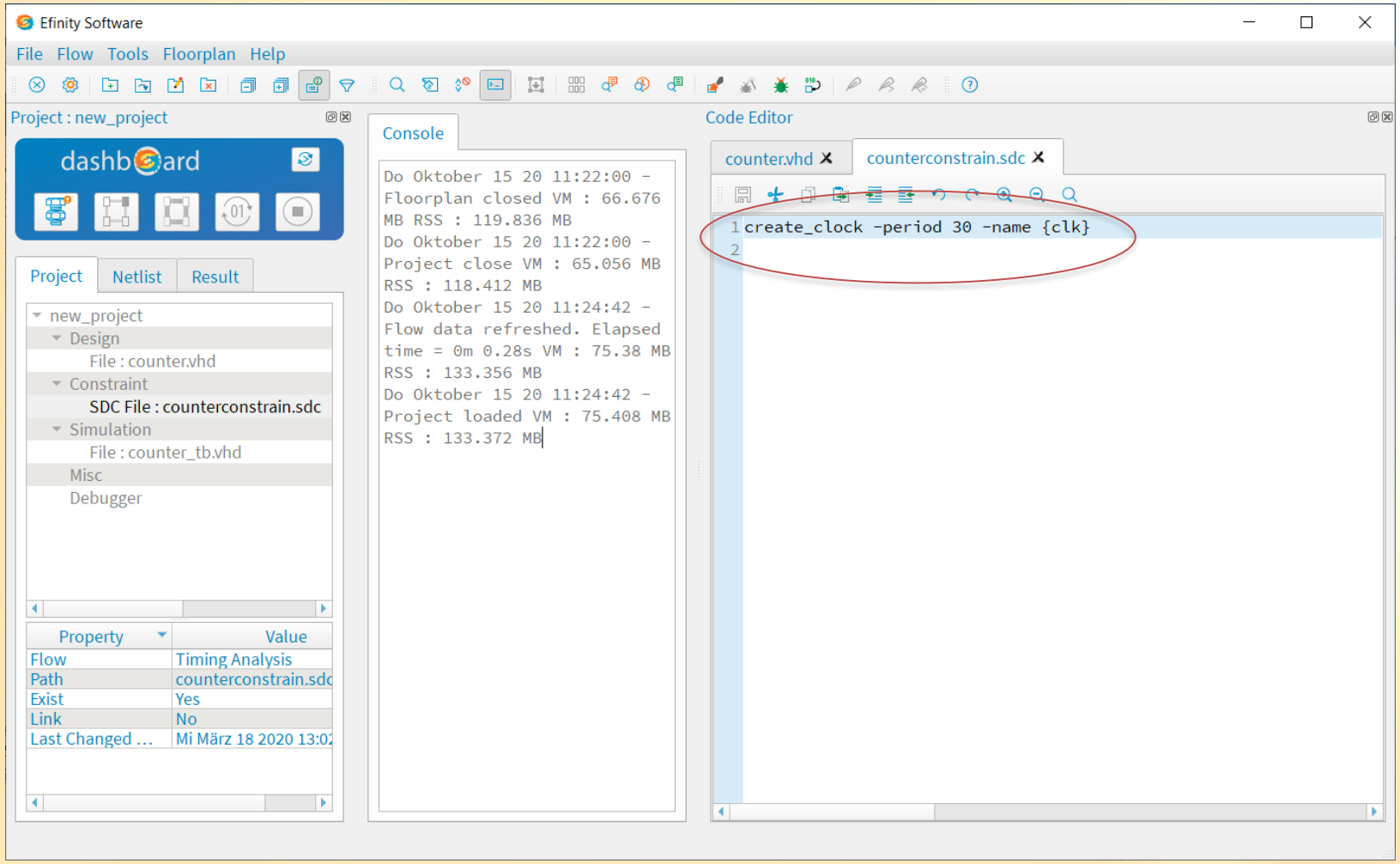
Double click on counter.vhd



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

EFINIX®

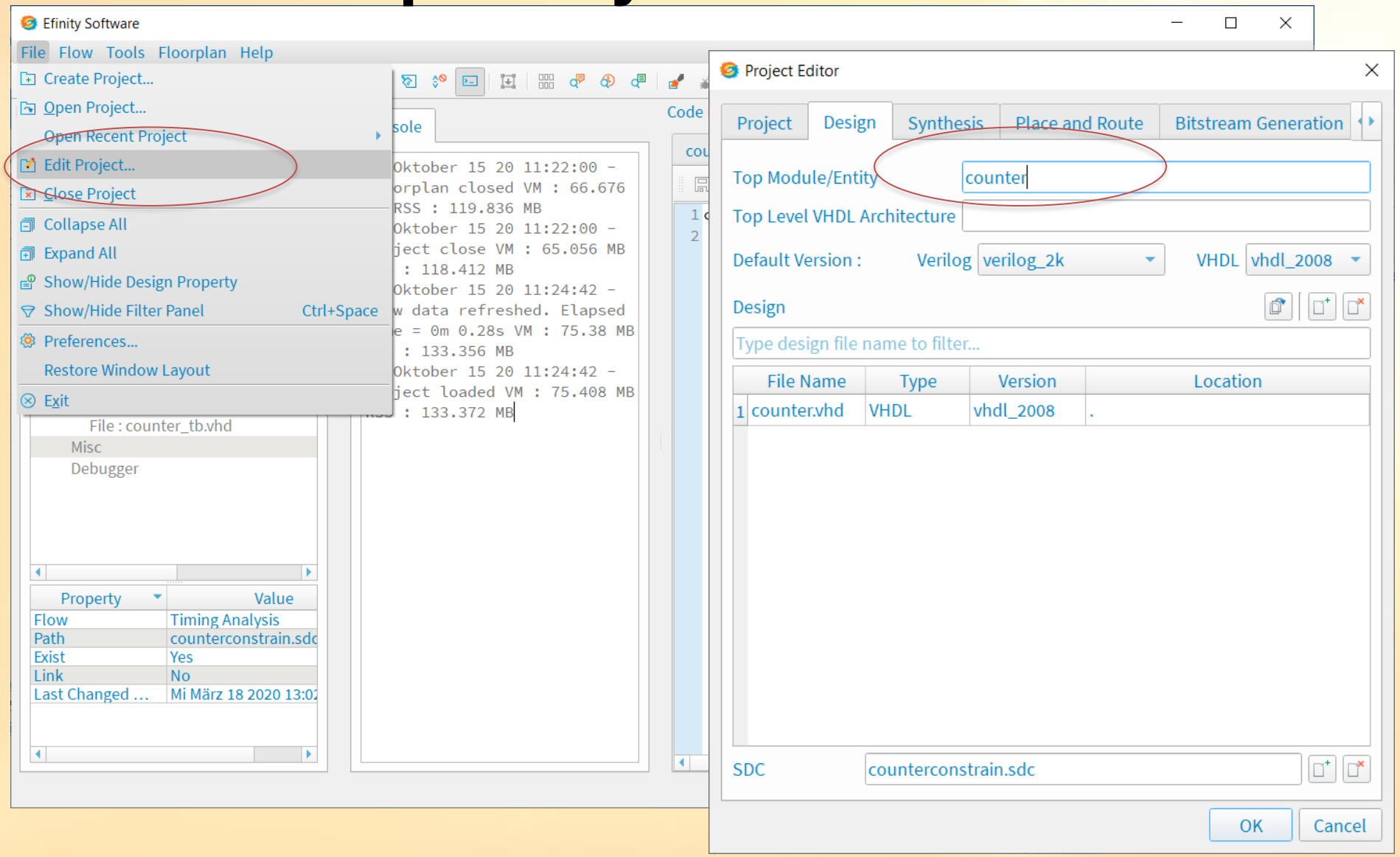
Double click on the timing constrain counterconstrain.sdc



The counterconstrain.sdc will show up and you can see the 30ns period constrain for the 33MHz clock

EFINIX®

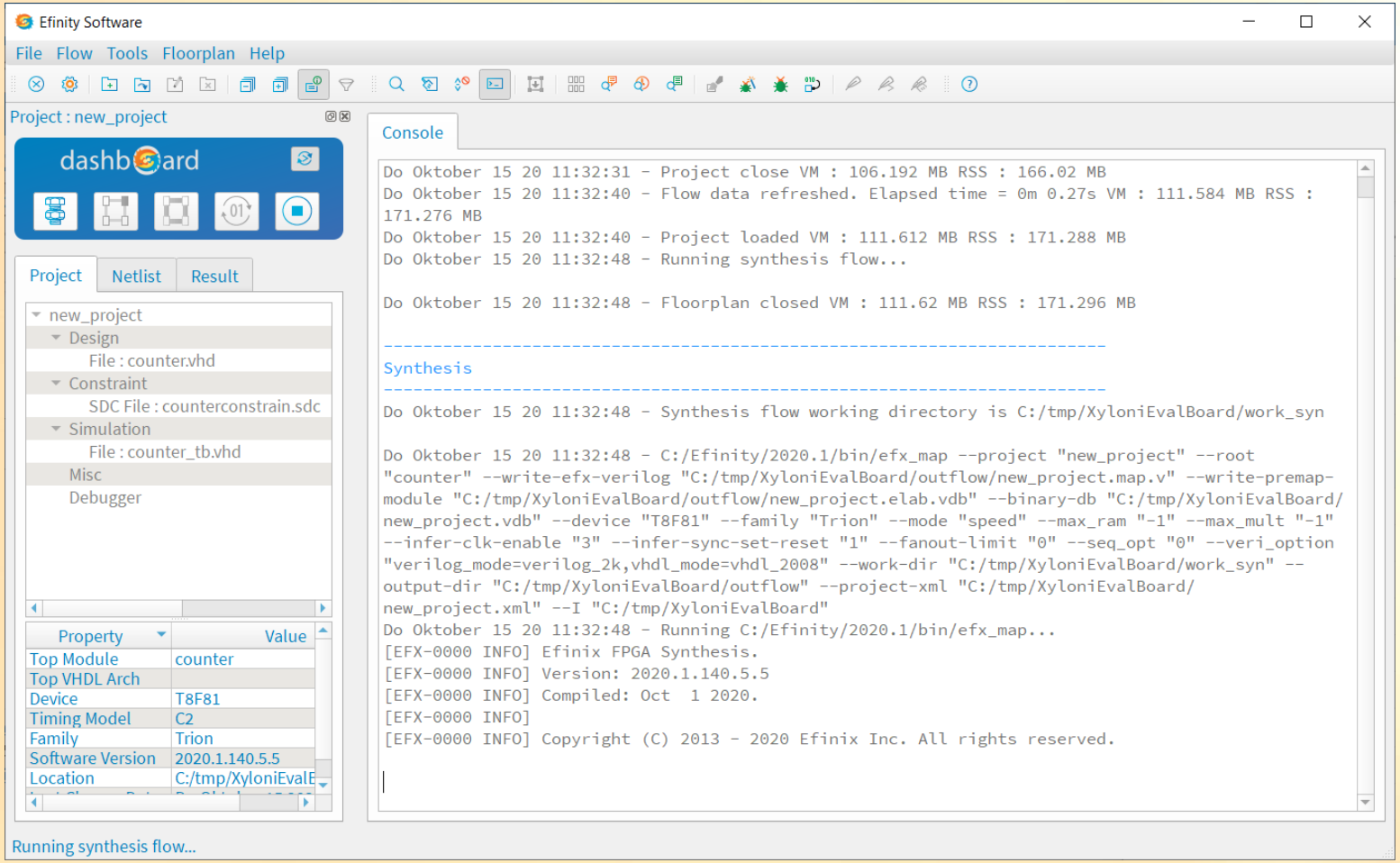
Set the top entity name



Enter counter to top Module/Entity and click OK

EFINIX®

Run the whole flow or just Synthesis

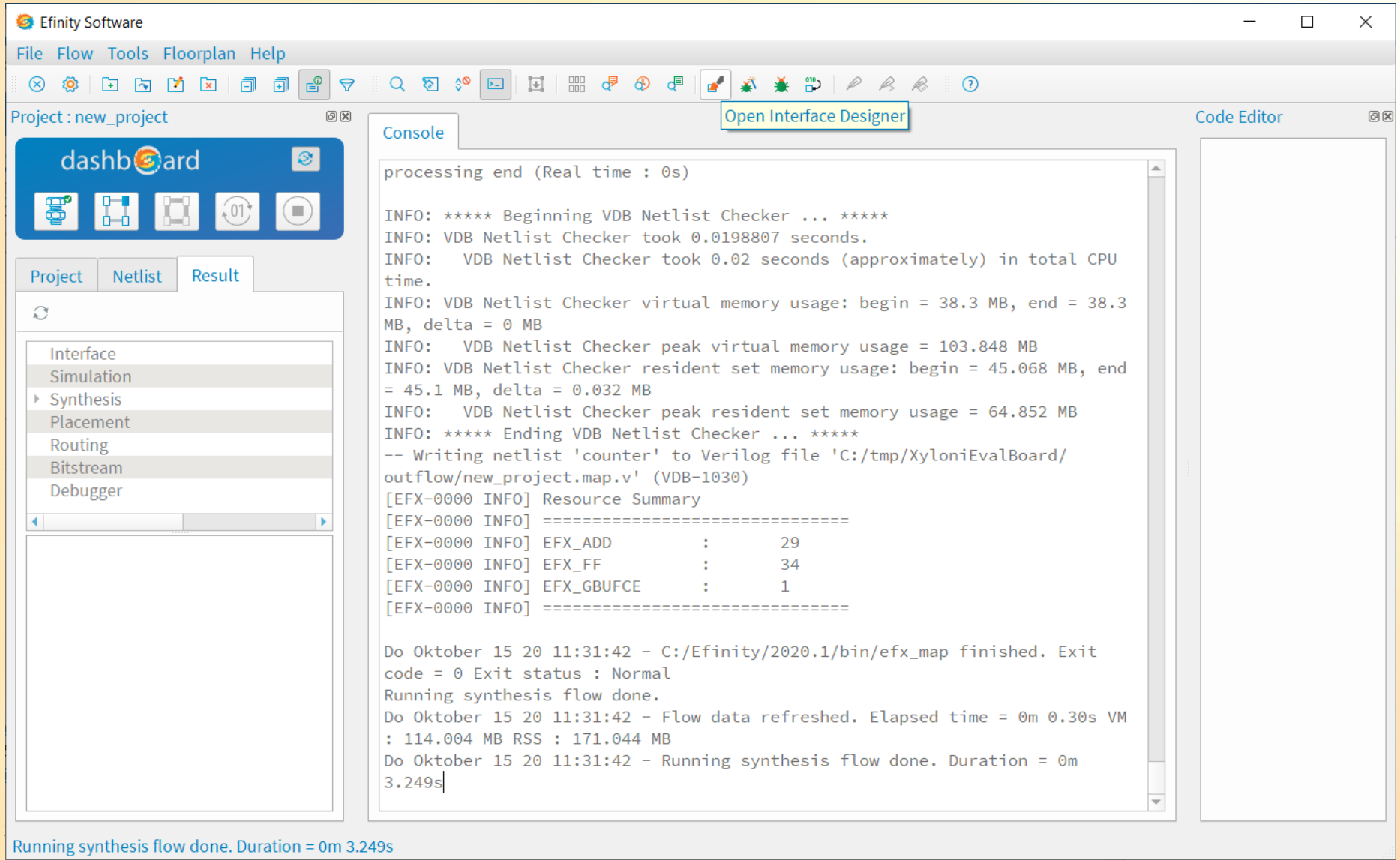


- If the Automated flow button is grayed out click on the button to activate the automated flow.

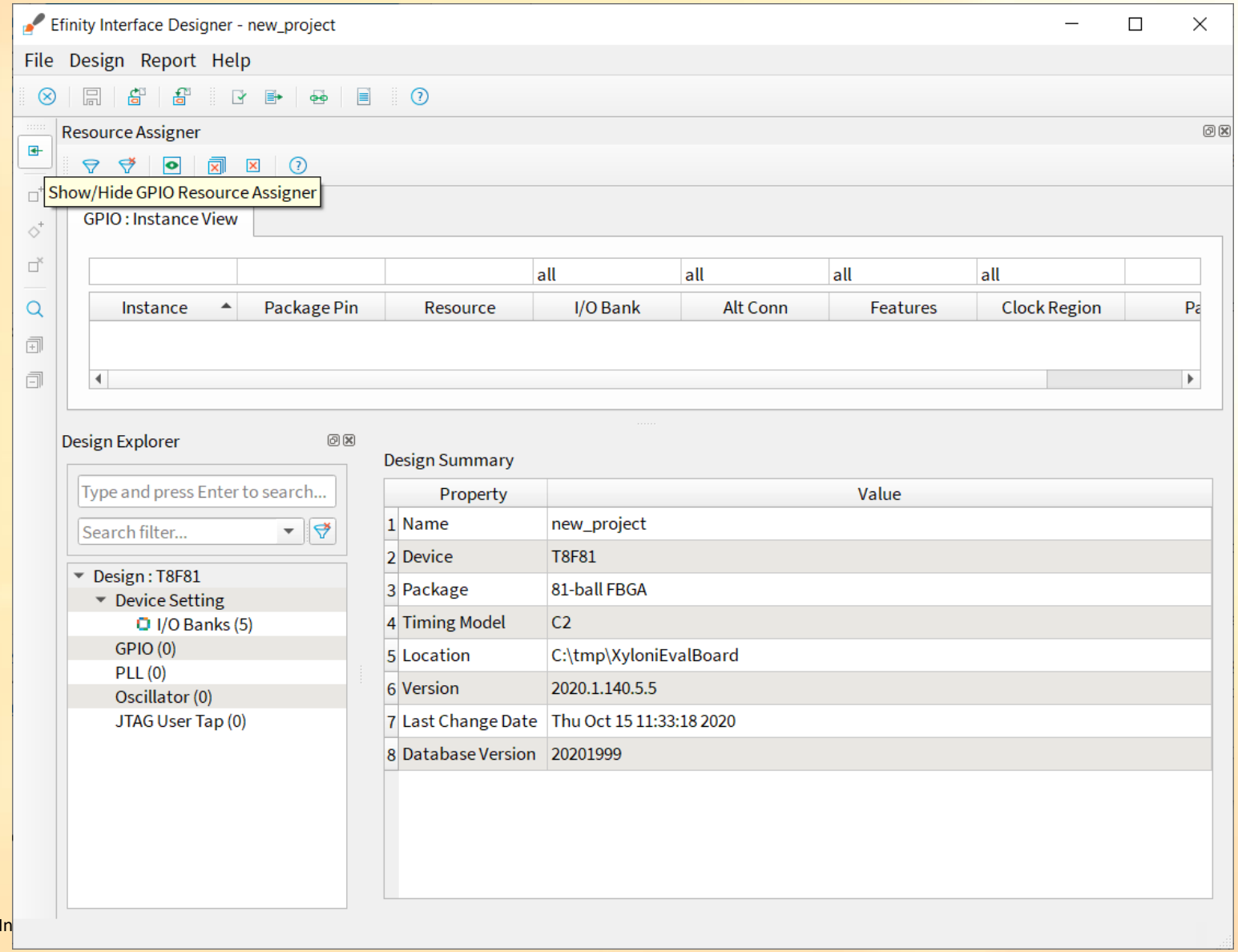
- Click on the synthesis icon and the flow will run automnatically

EFINIX®

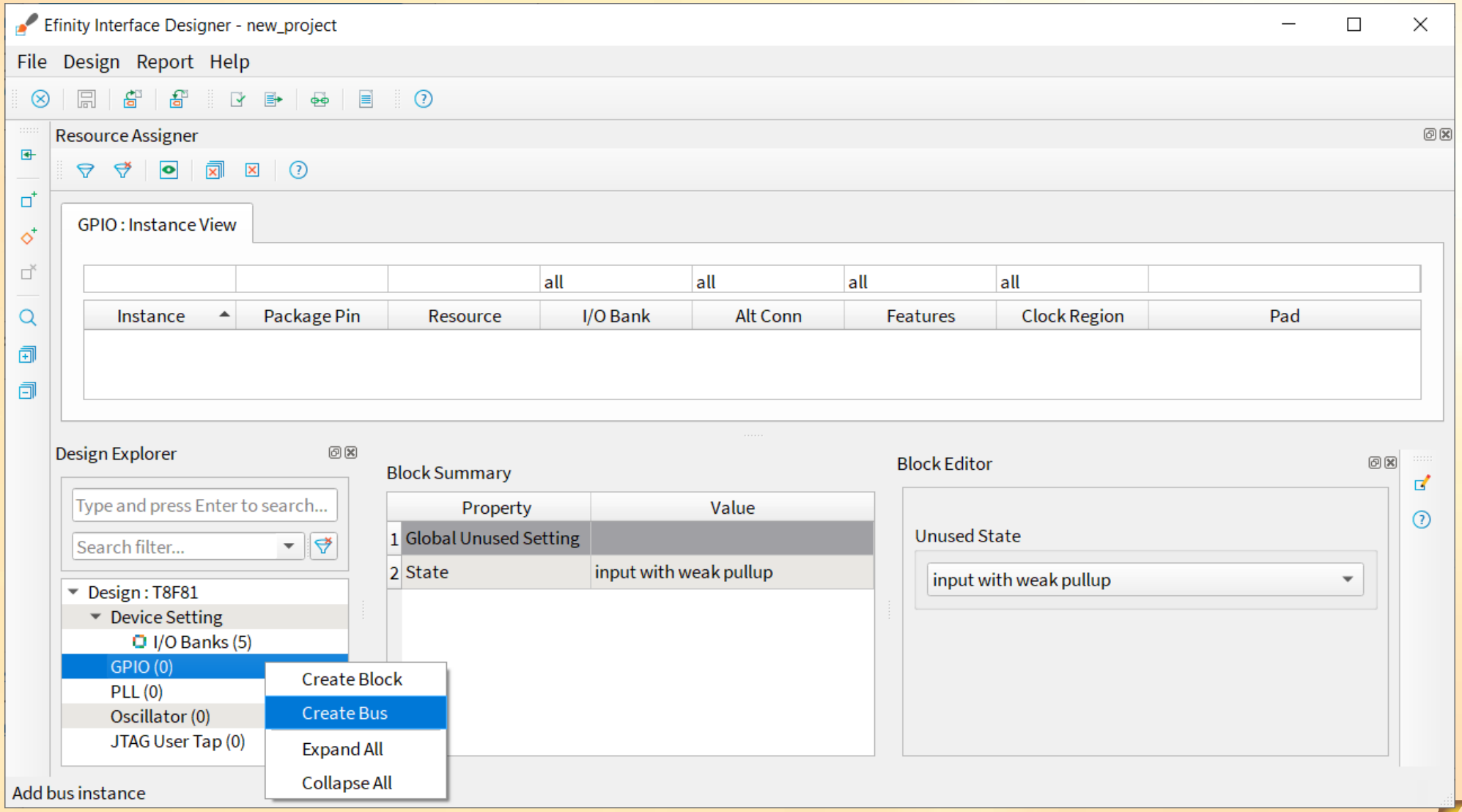
Assign the top level Signals to Pins.
1. Open Interface Designer



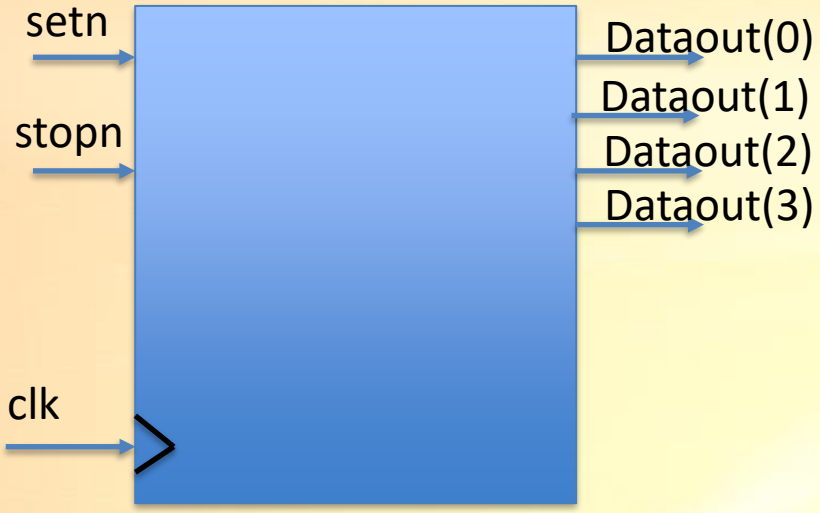
Show GPIO Resource Assigner

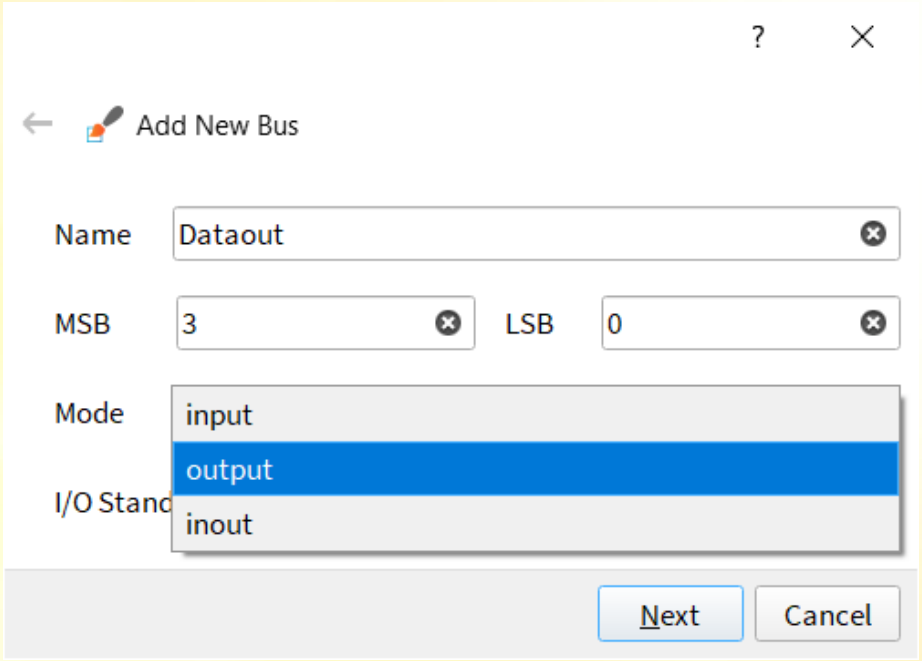


Select GPIO, RMB, Create Bus



Create a Output BUS Dataout(3 downto 0)





Name: Dataout

MSB: 3

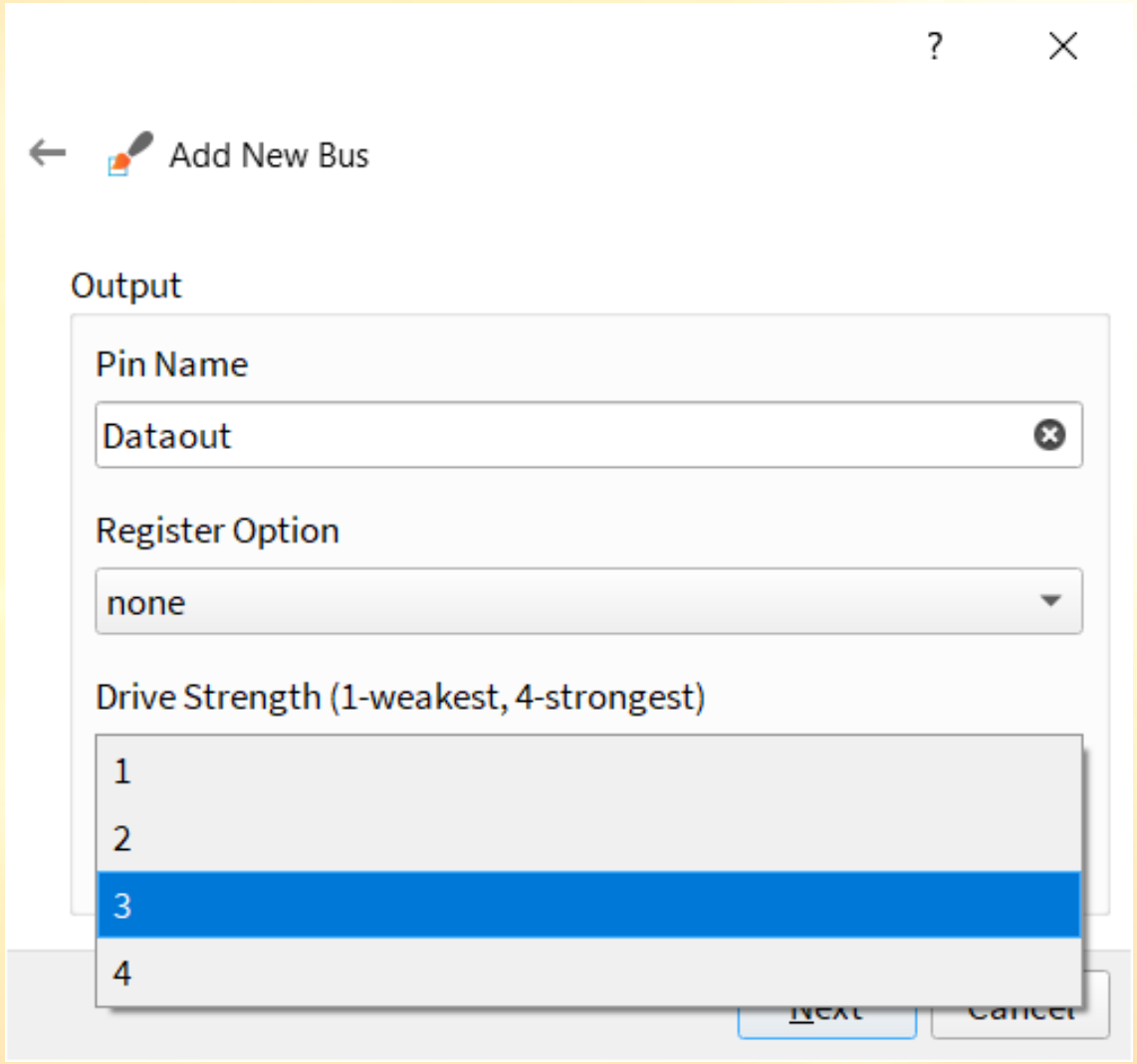
LSB: 0

Mode: output

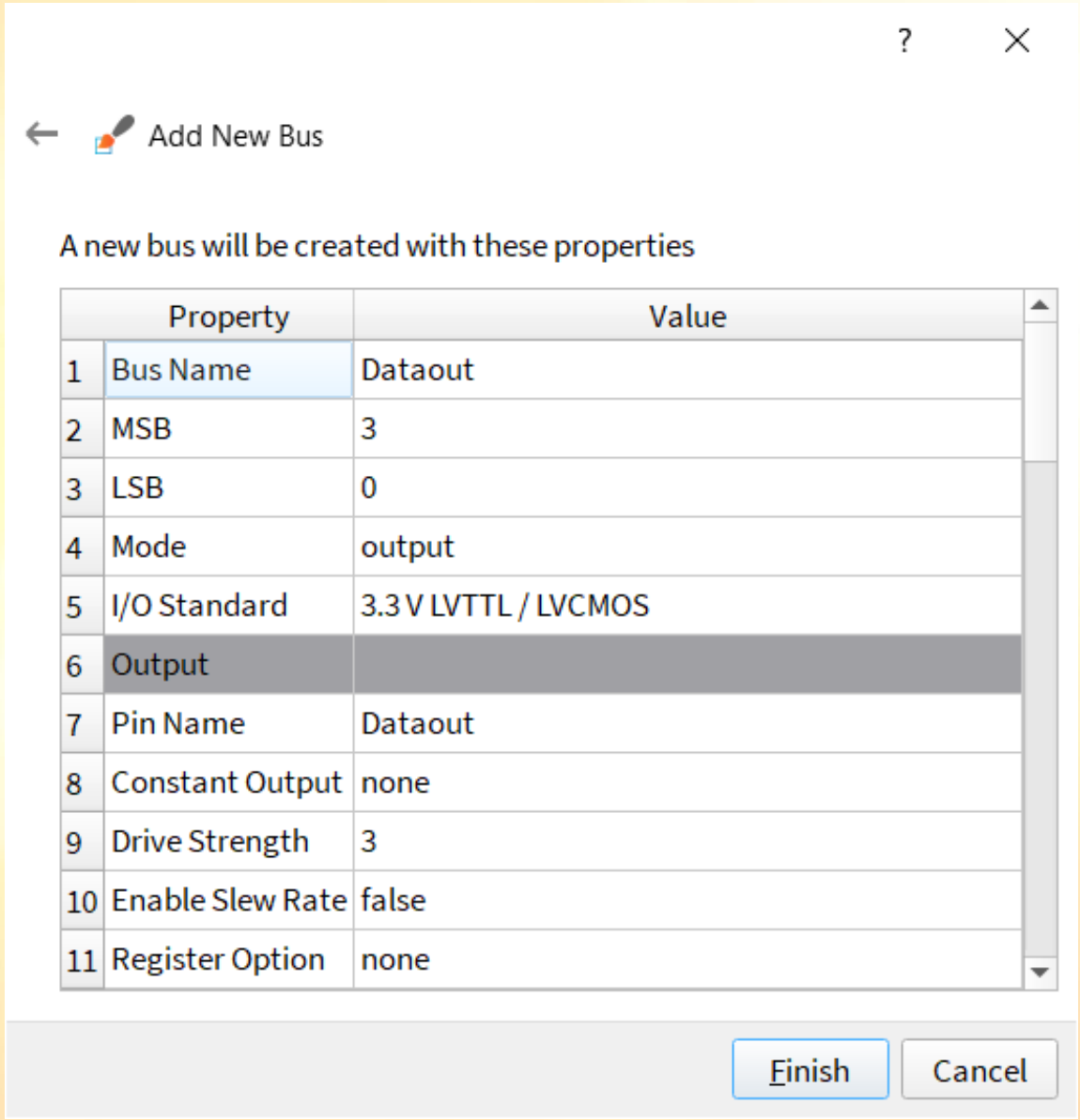
Click Next

EFINIX®

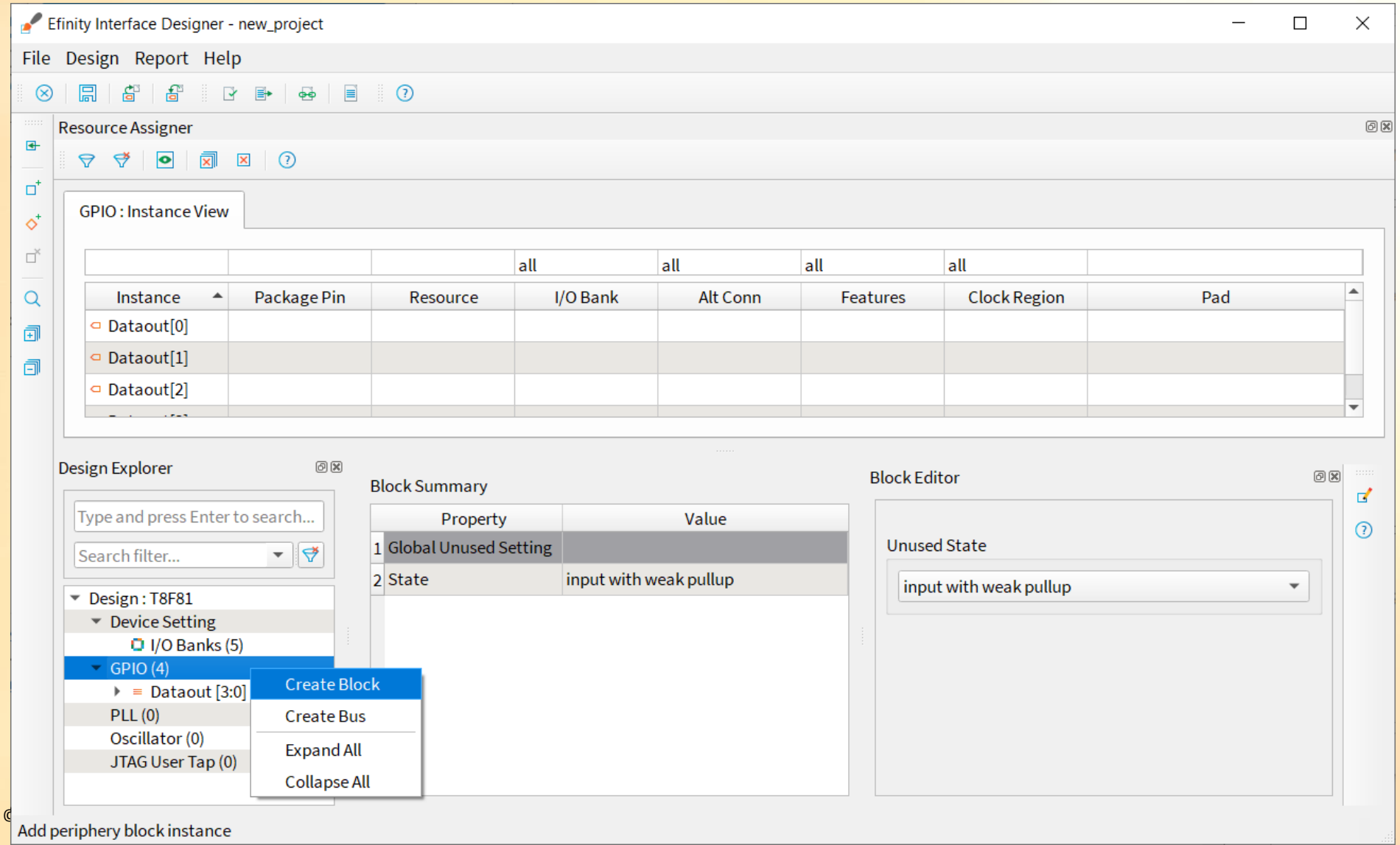
Set drive strength to 3; click Next



Click Finish



Adding the additional Pins clk, stopn, setn



Select GPIO

Create Block

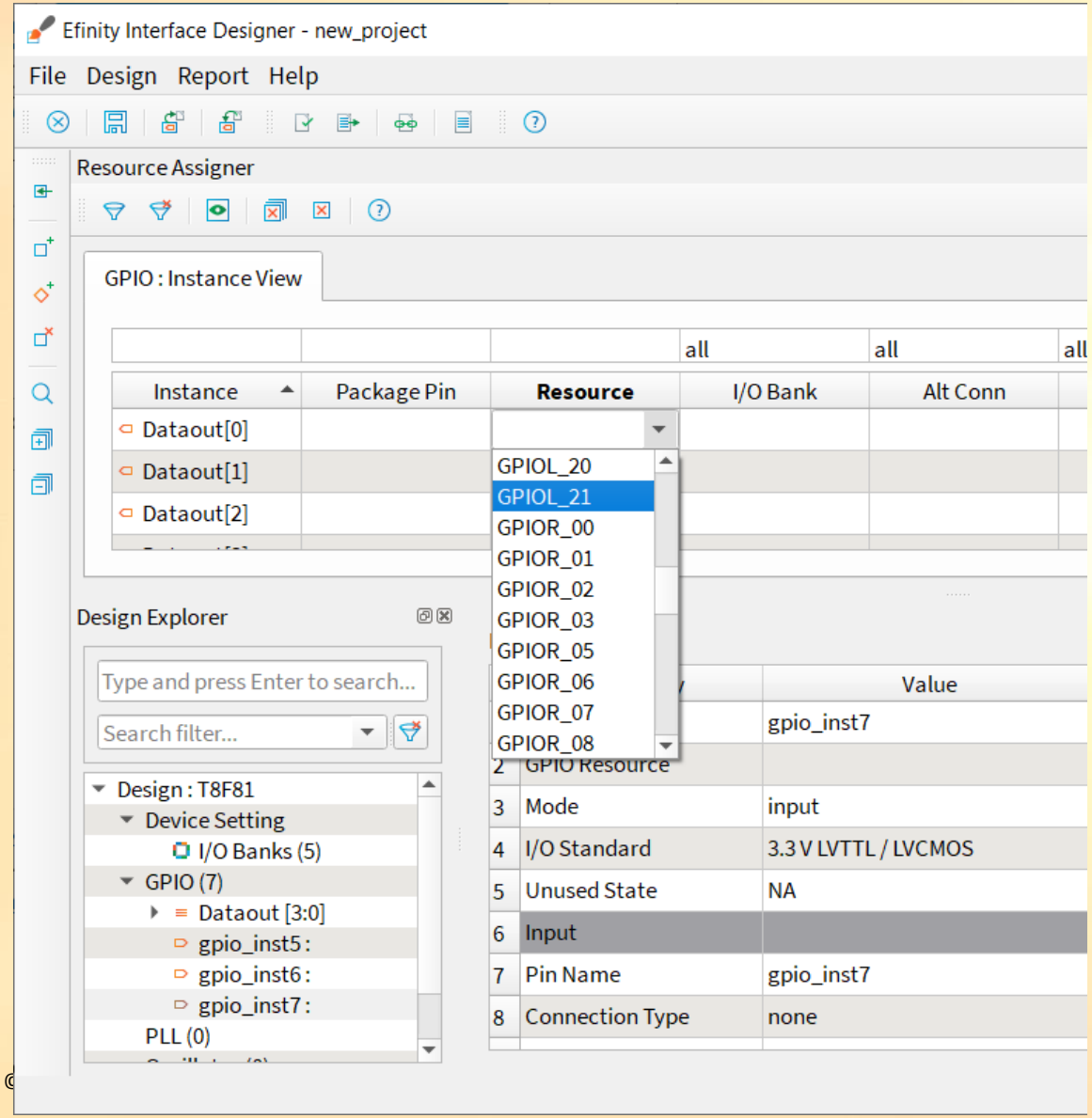
Select GPIO

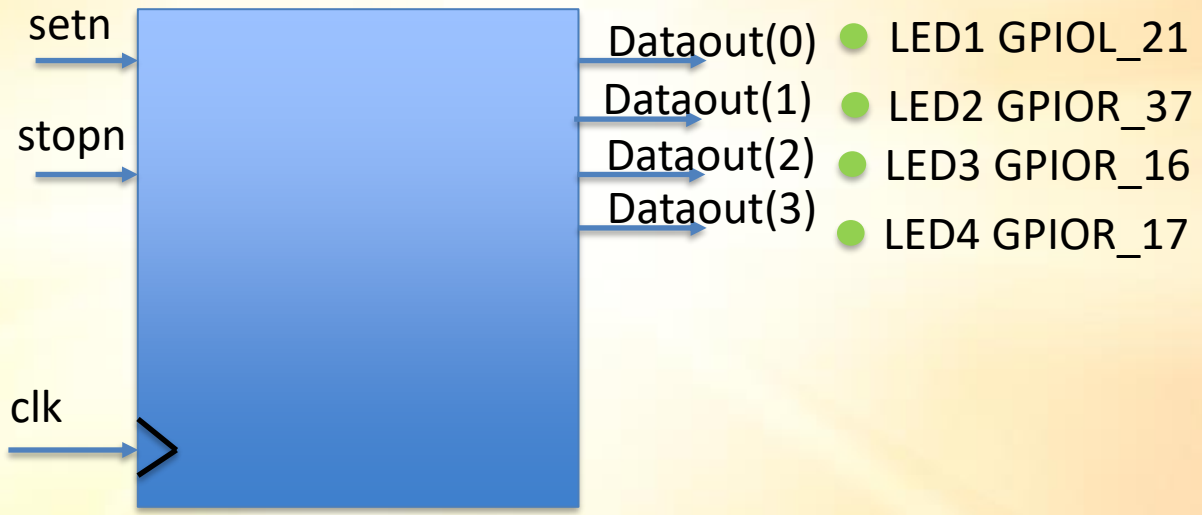
Create Block

Select GPIO

Create Block

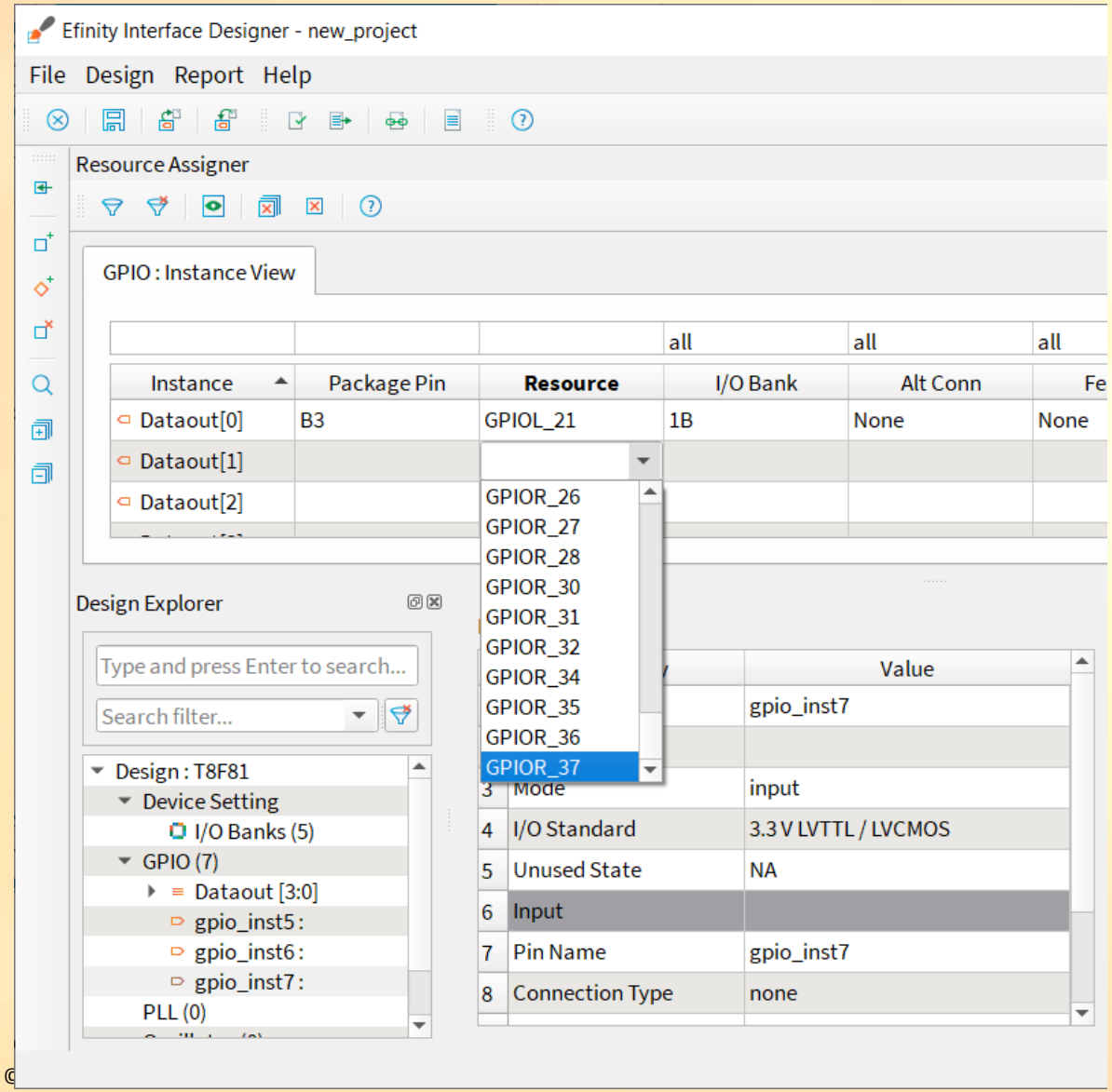
Select GPIO: Instance View Dataout[0]->Resource

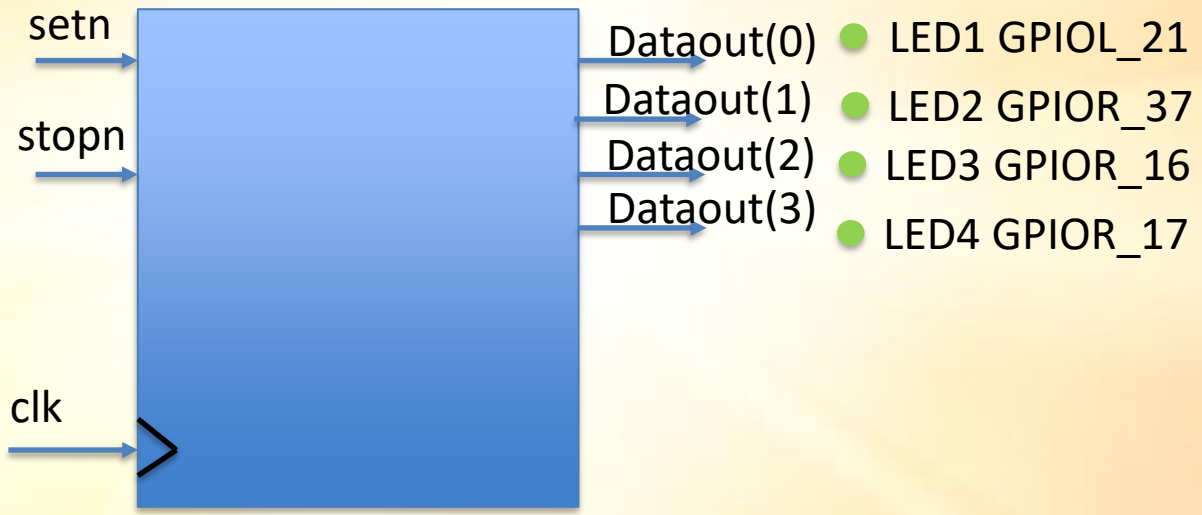




Dataout[0] ->GPIOL_21

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

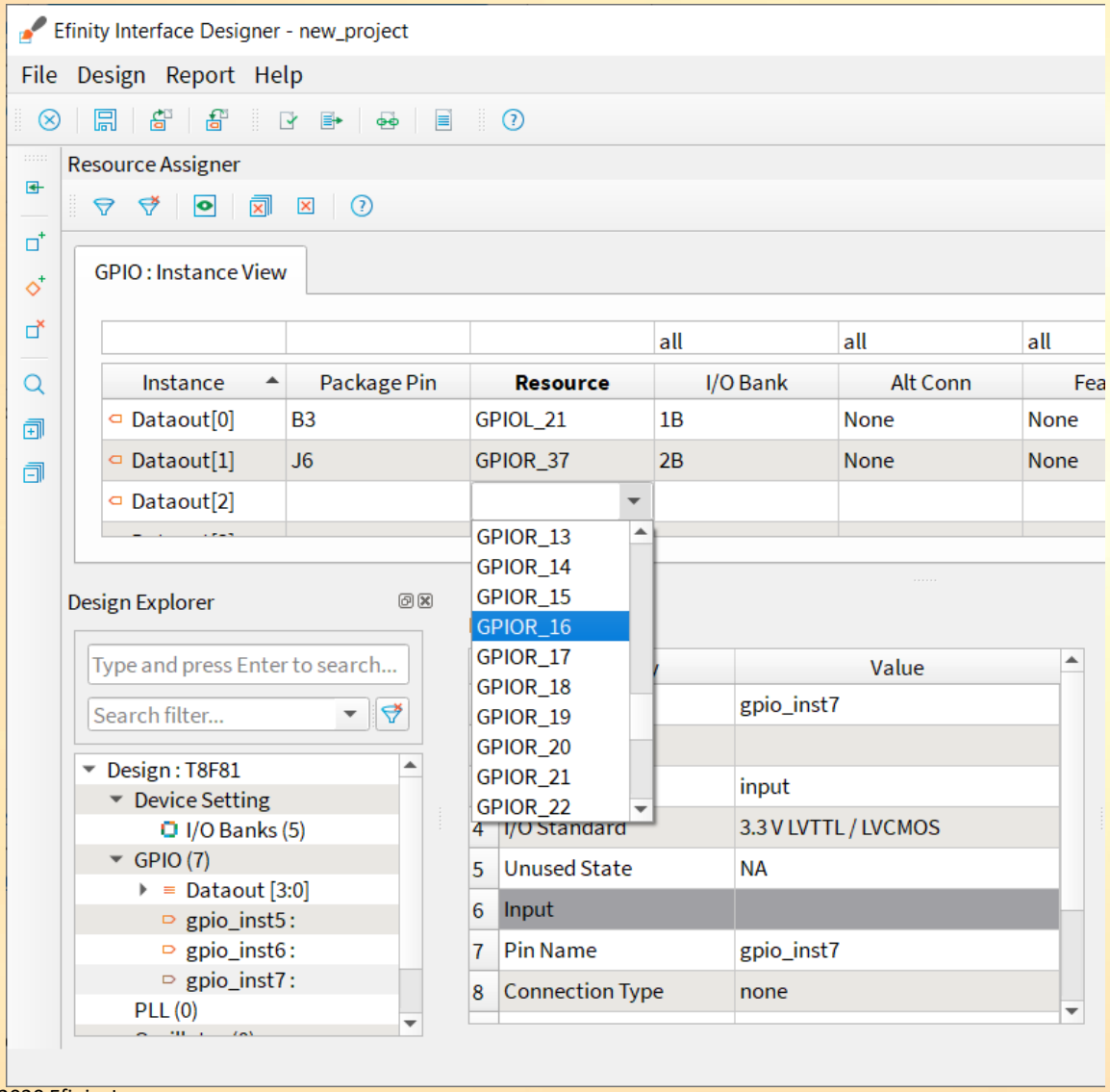


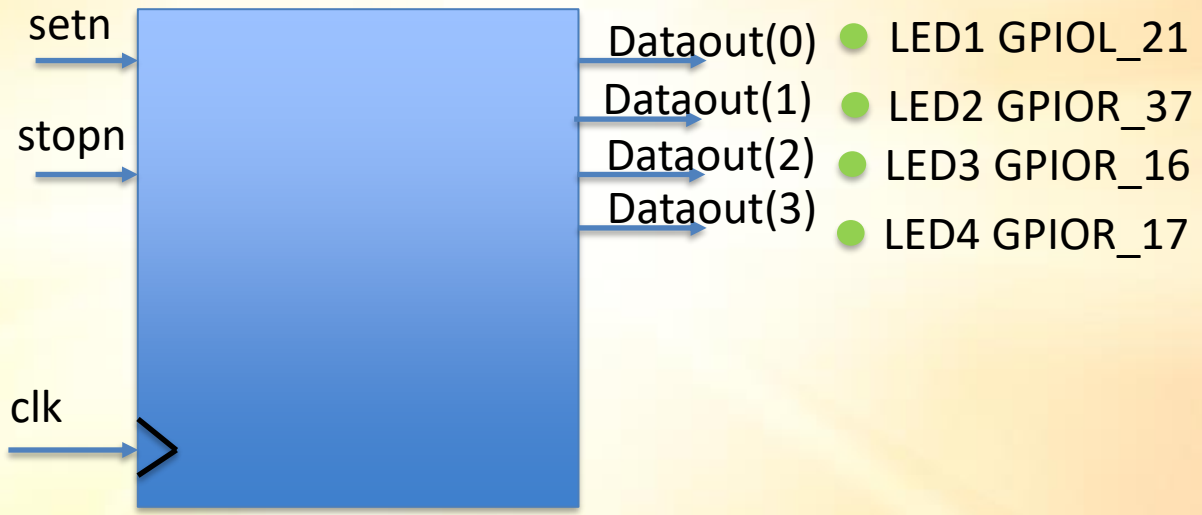


Dataout[1] ->GPIOR_37

EFINIX®

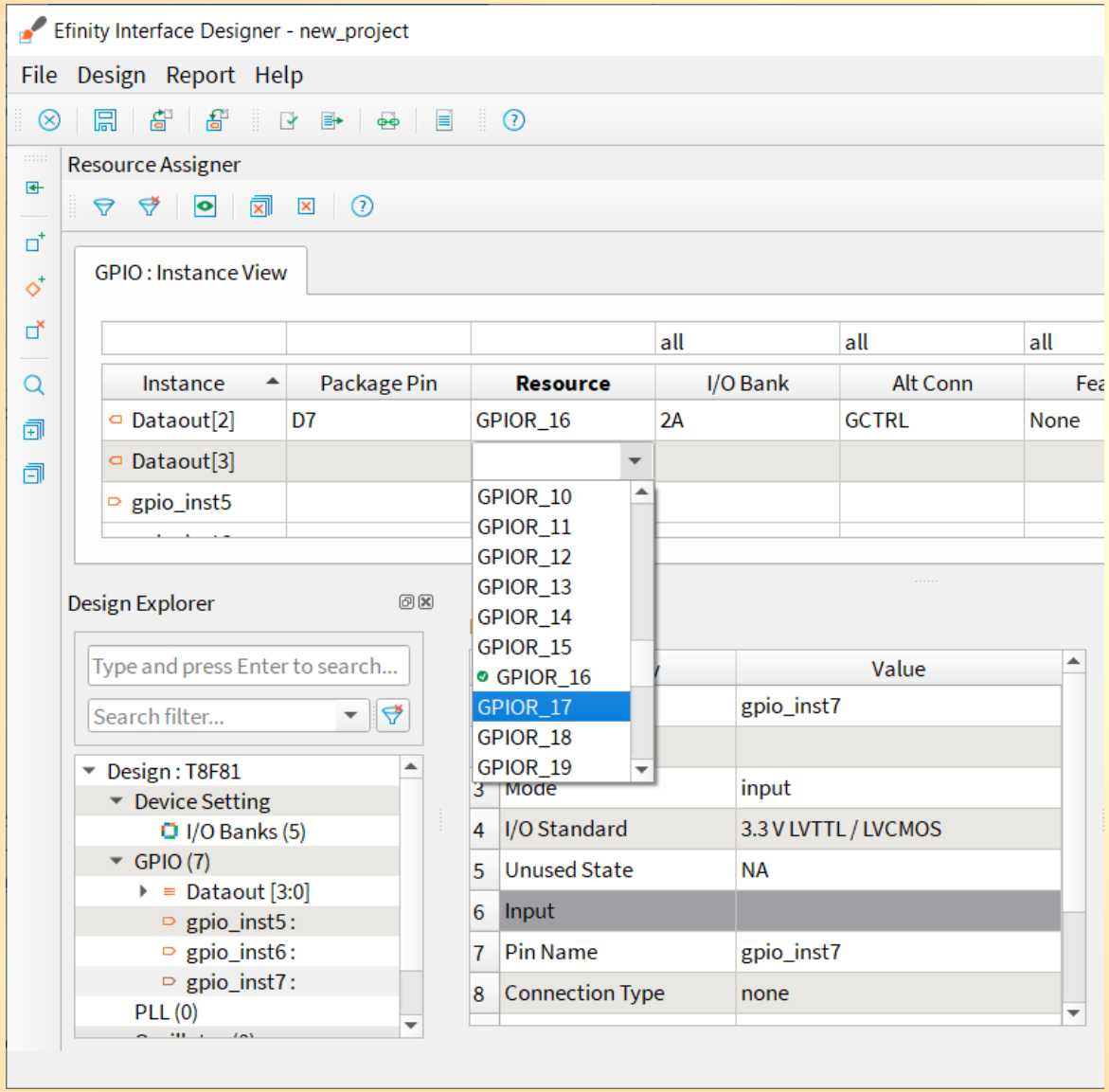
Select GPIO: Instance View Dataout[2]->Resource

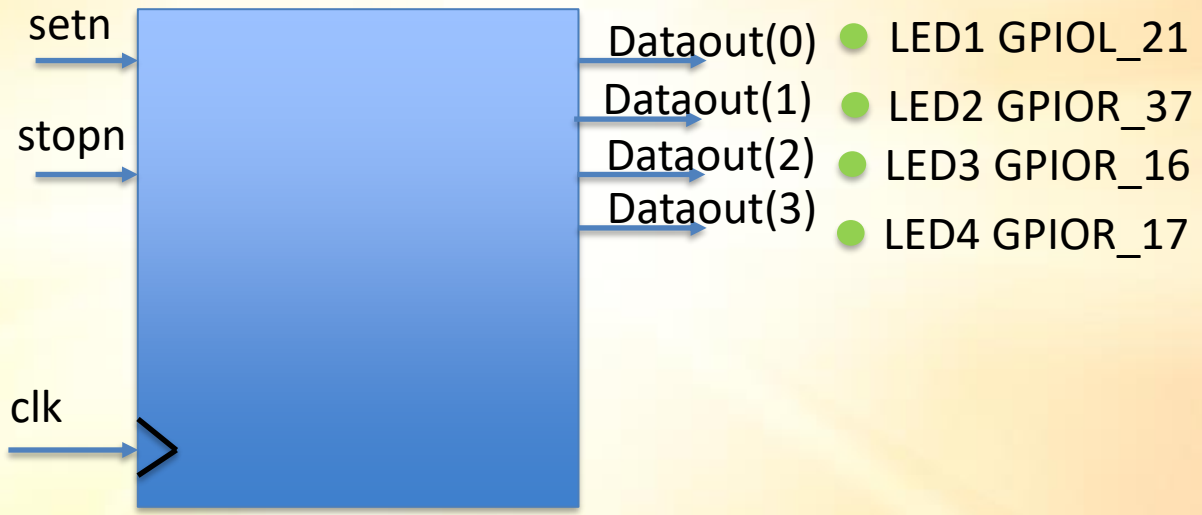




Dataout[2] ->GPIOR_16

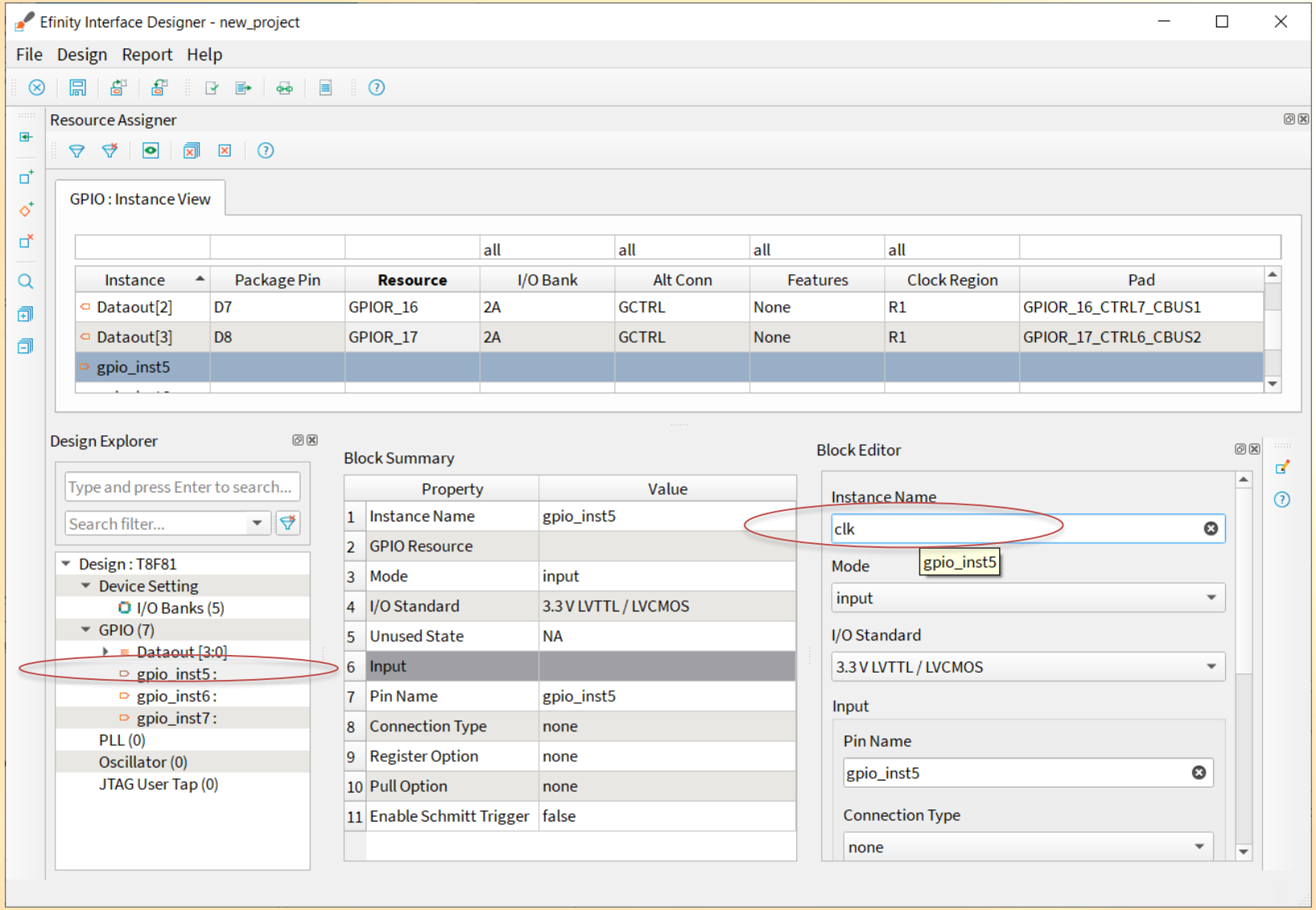
Select GPIO: Instance View Dataout[3]->Resource





Dataout[3] ->GPIOR_17

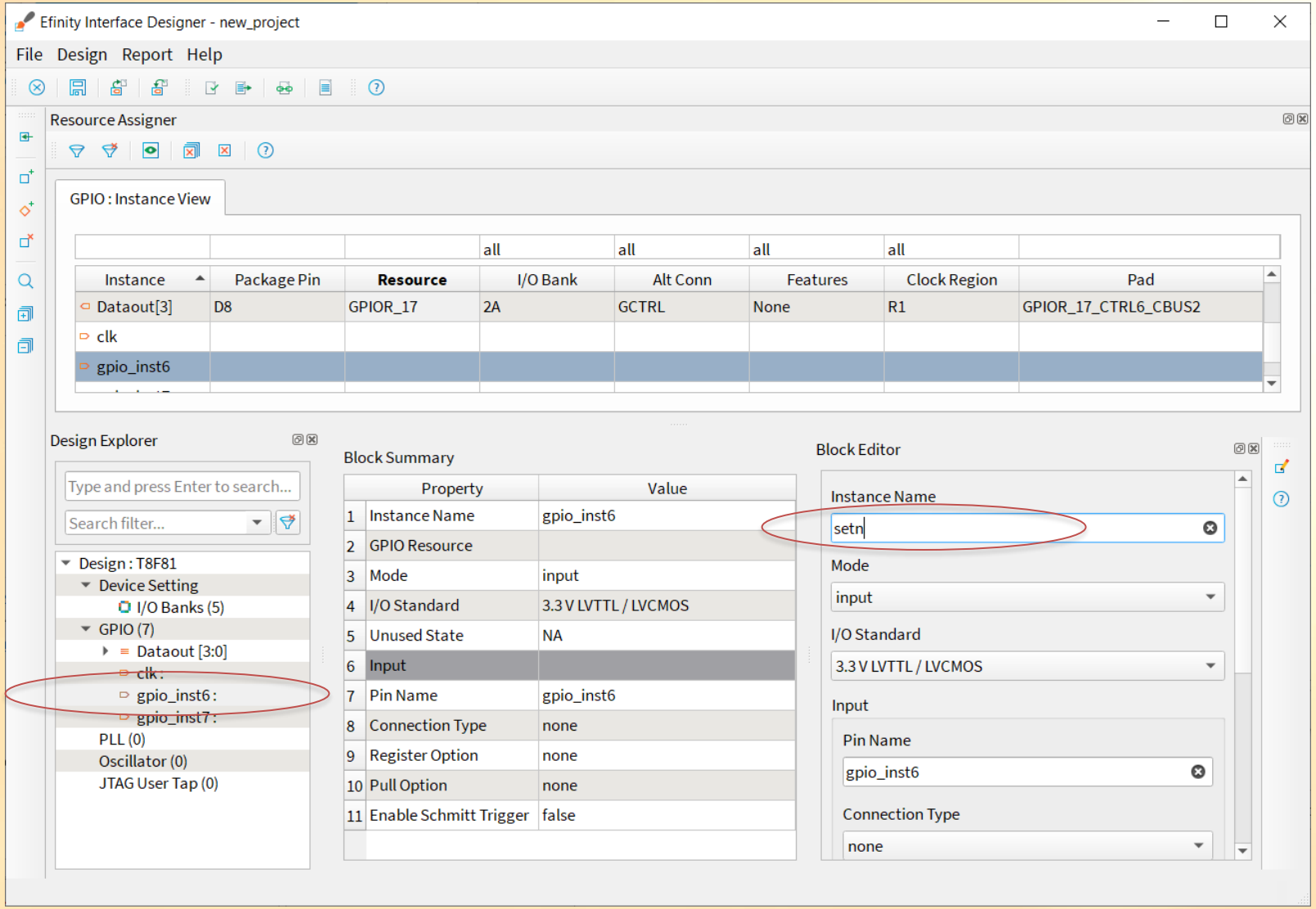
Change the name from the remaining Inputs: clk



Select gpio_inst5

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

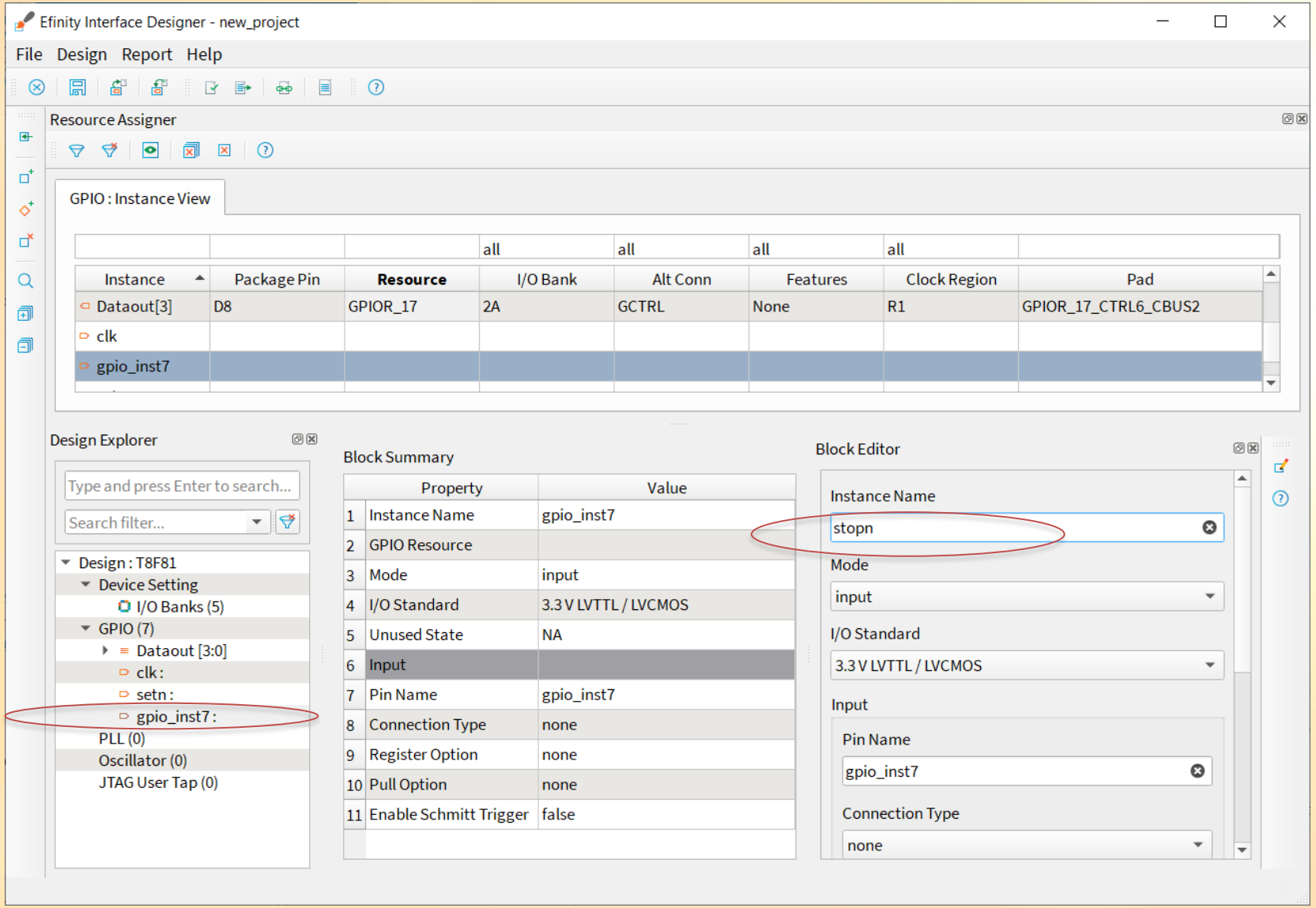
setn



Select gpio_inst6

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

stopn

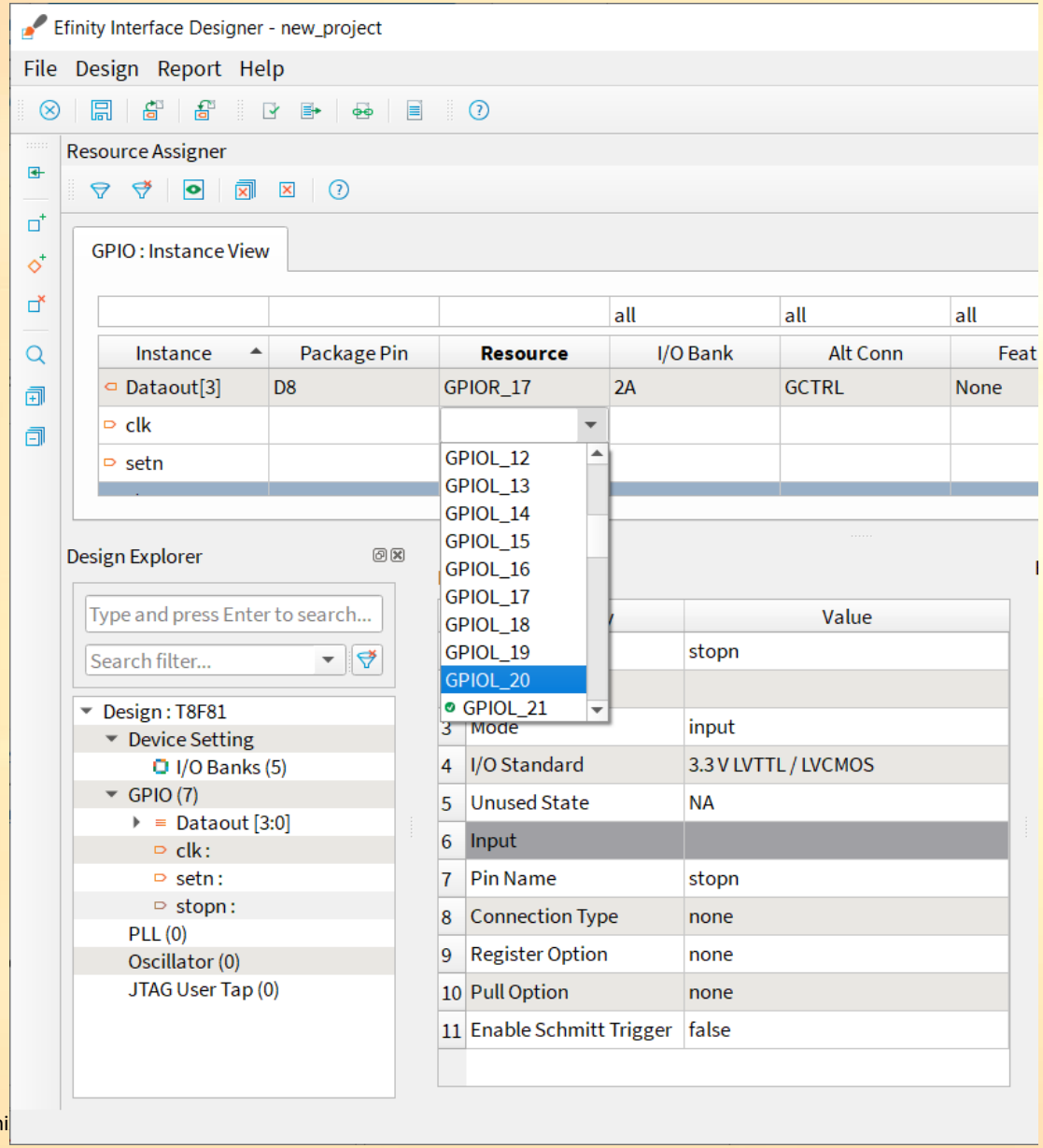


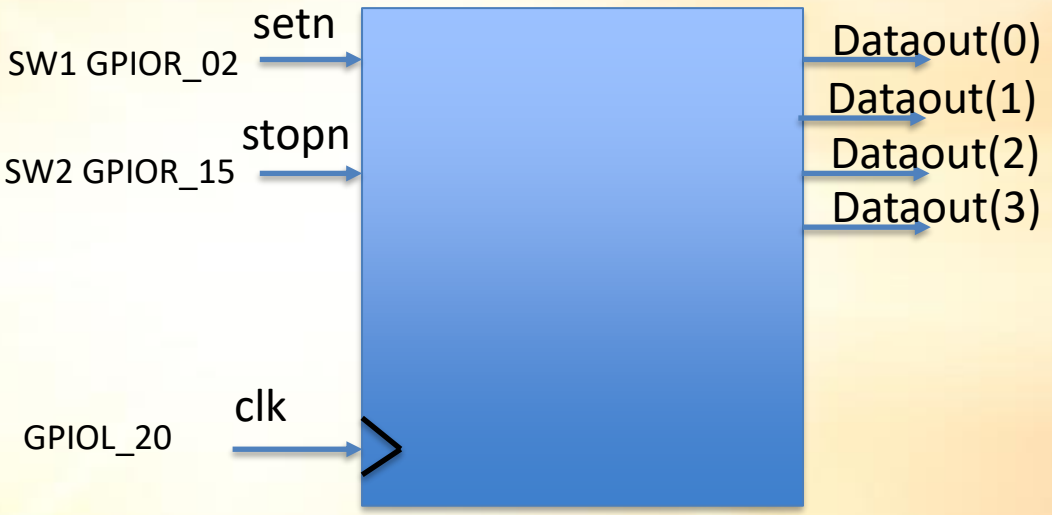
Select gpio_inst8

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

EFINIX®

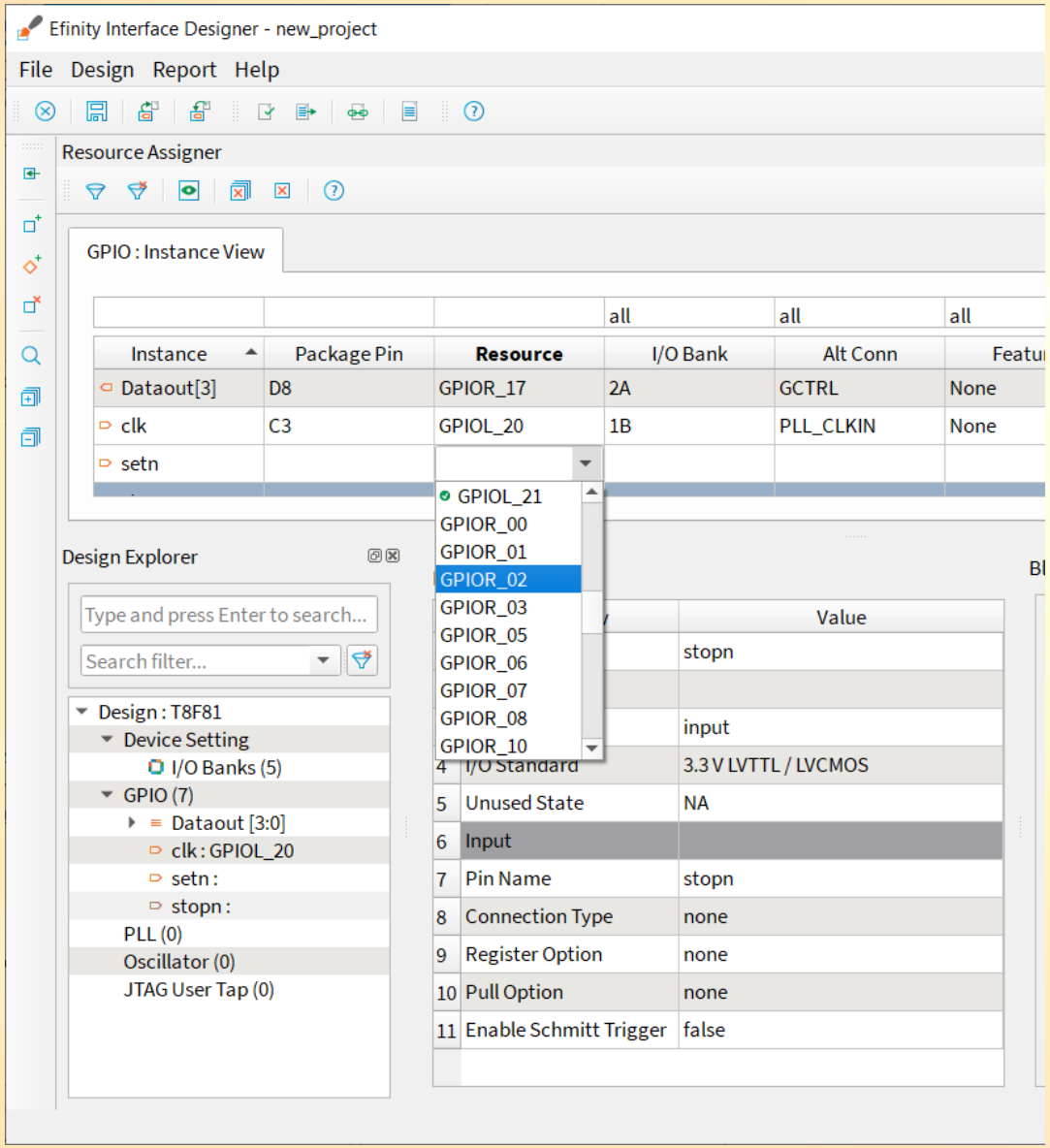
Assign the Resource CLK=>GPIOL_20

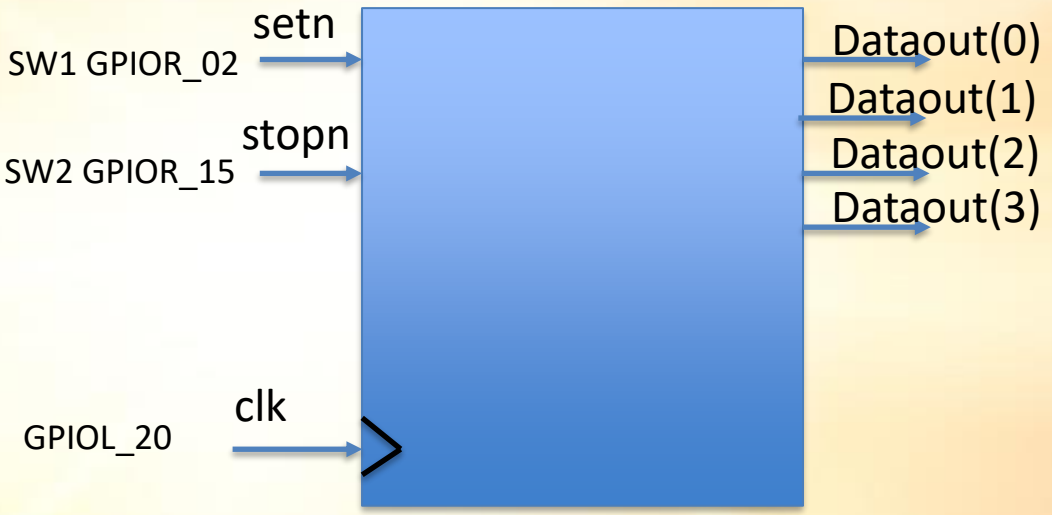




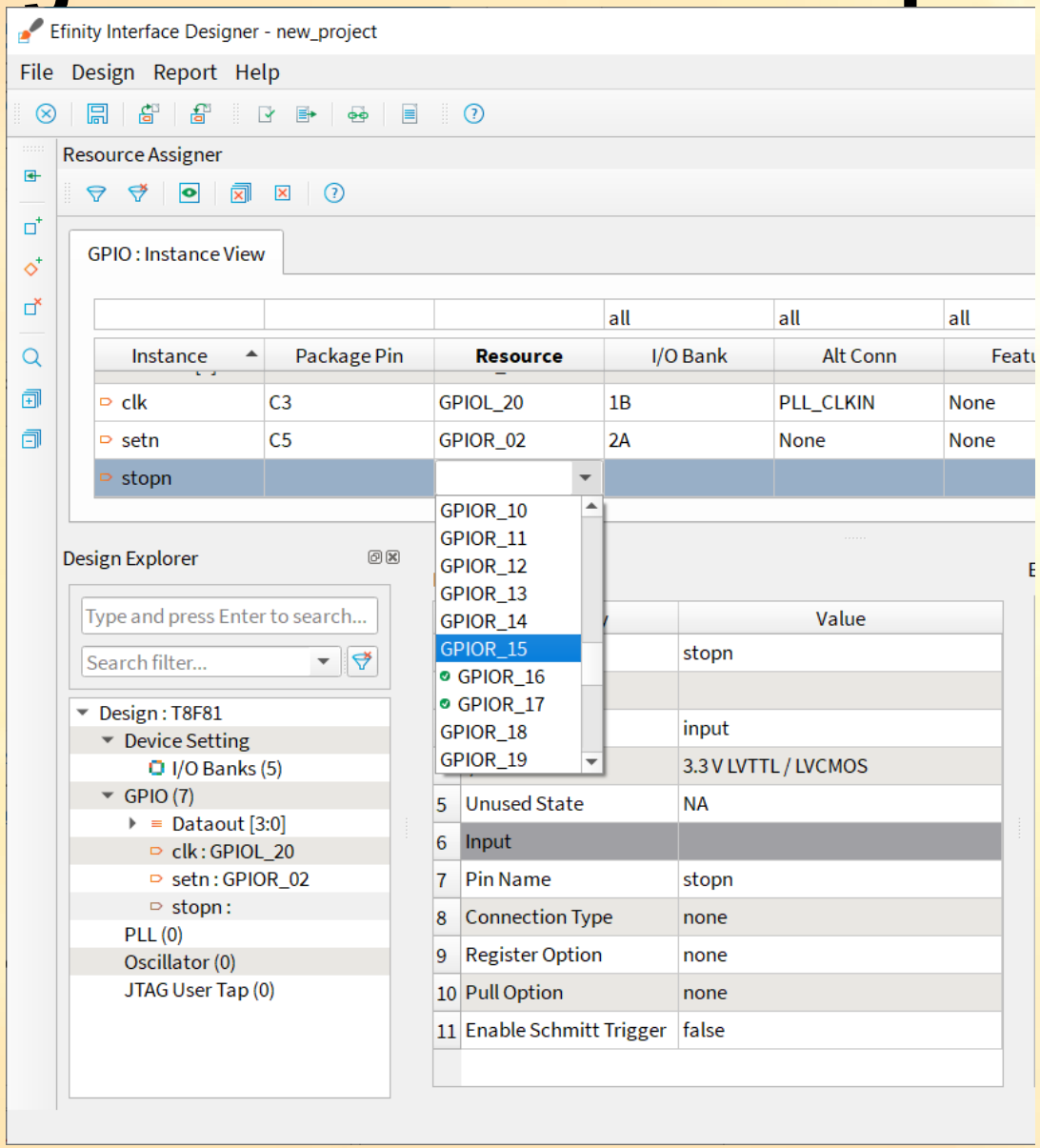
EFINIX®

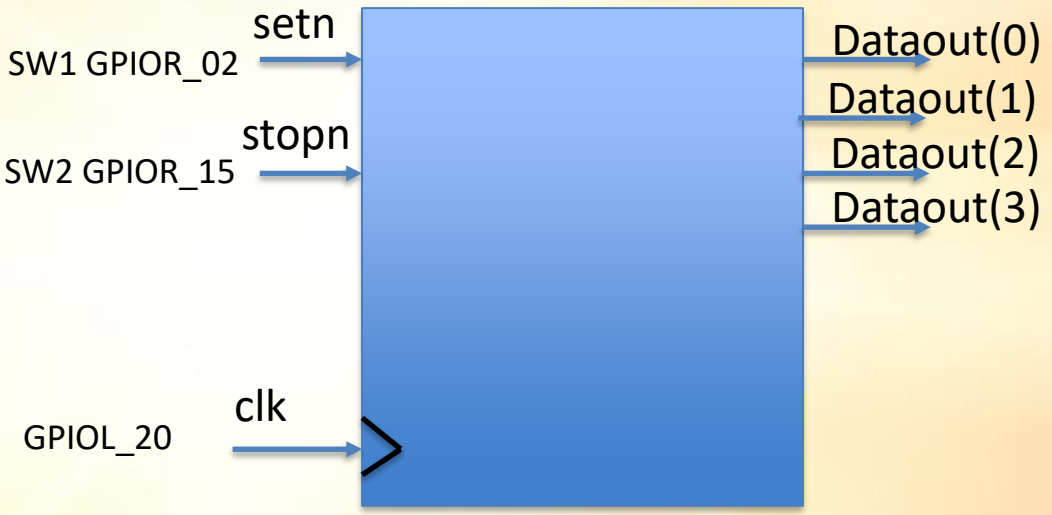
Assign the Resource setn=>GPIOR_02



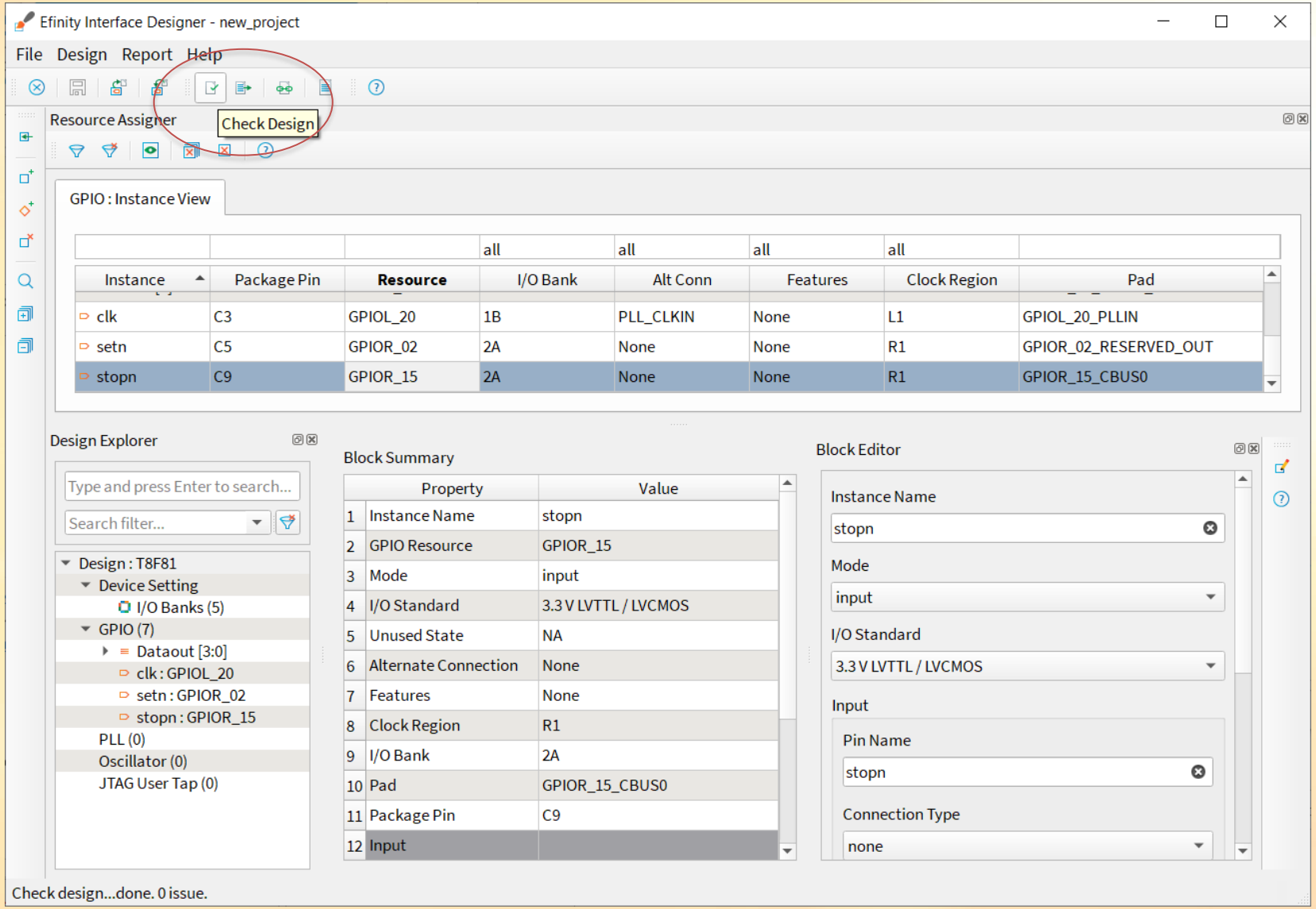


Assign the Resource stopn=>GPIOR_15

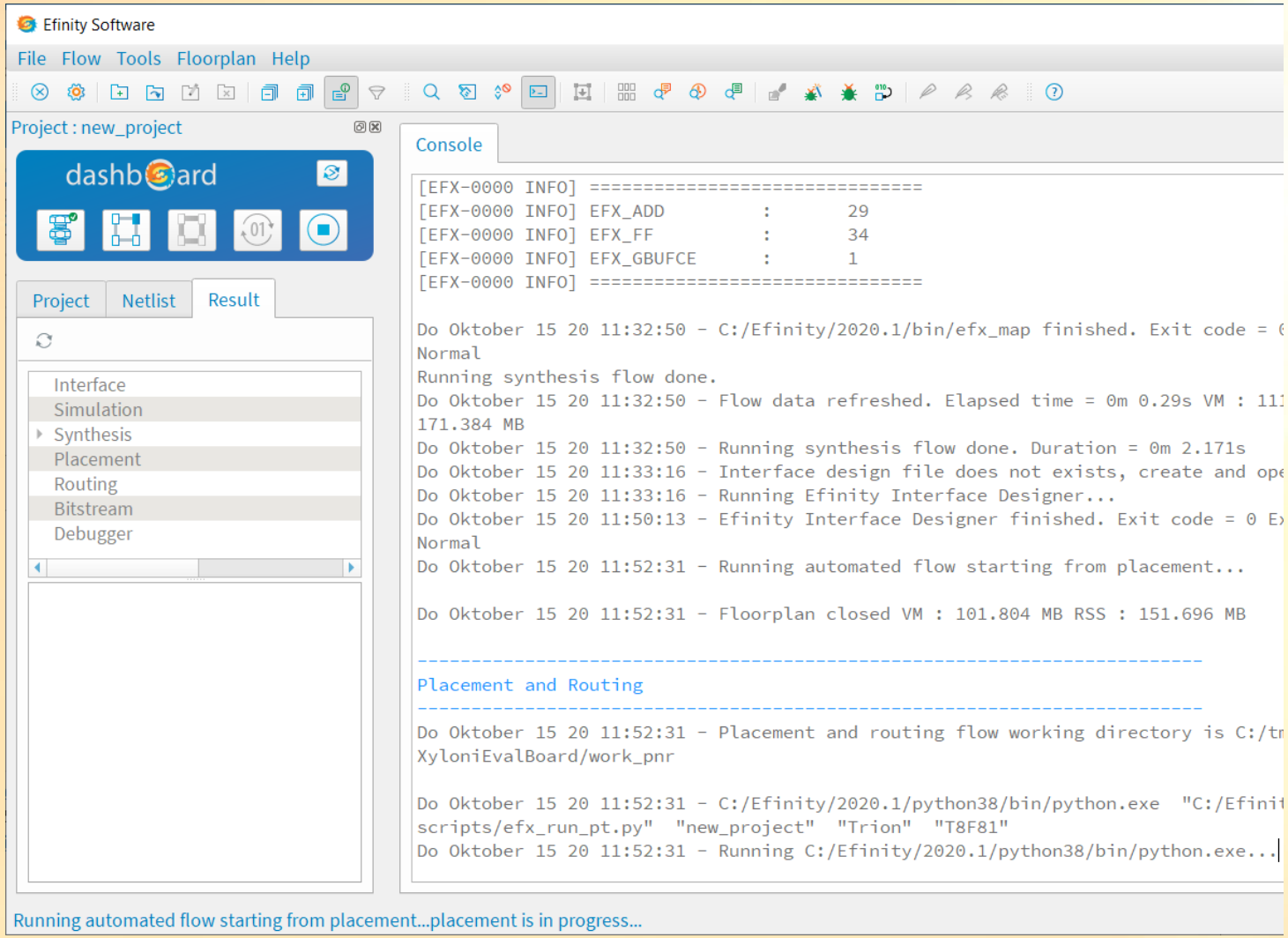




Check the design and close the interface designer



Run the whole flow with clicking on placement

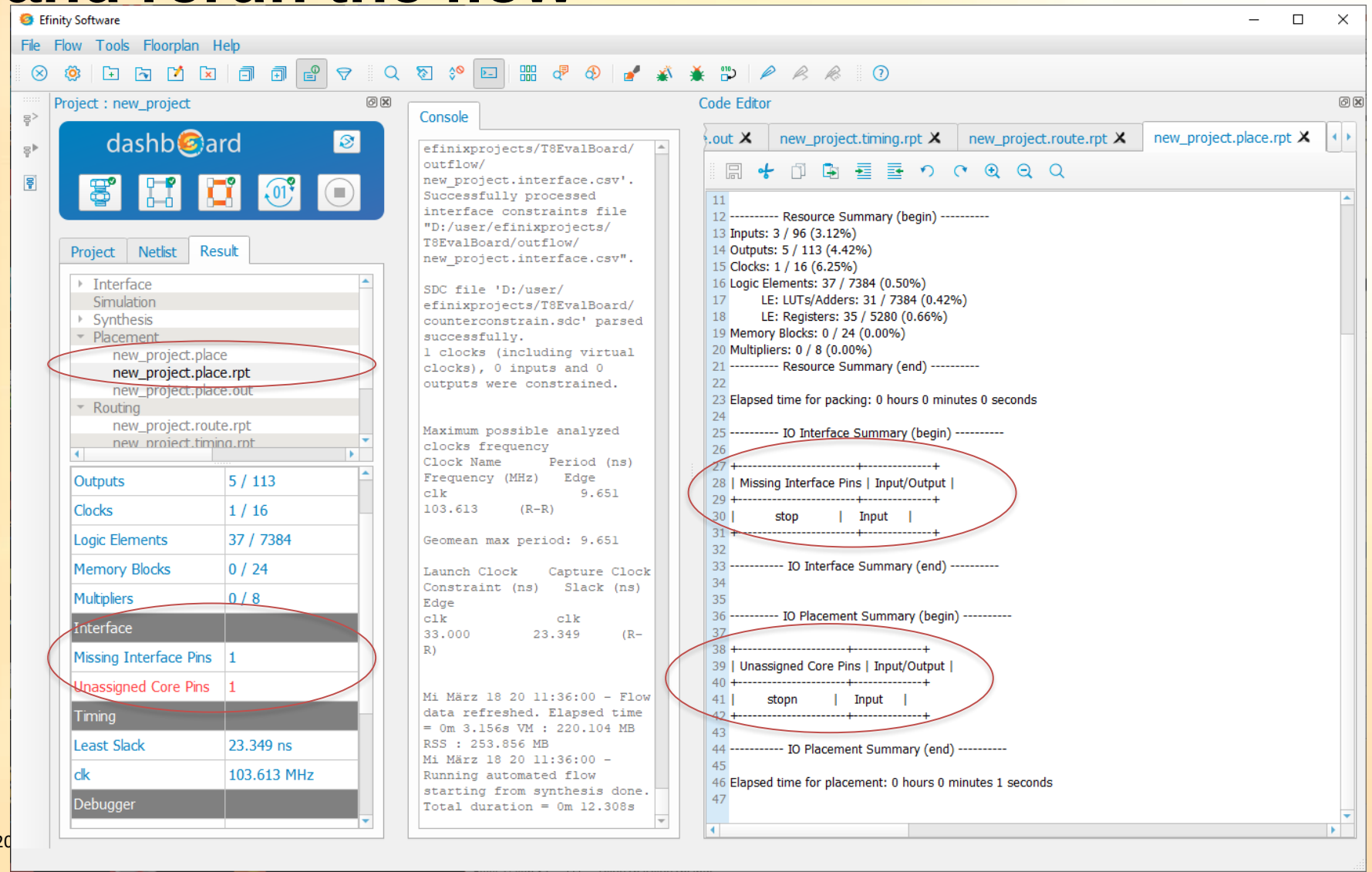


- If the Automated flow button is grayed out click on the button to activate the automated flow.

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

EFINIX®

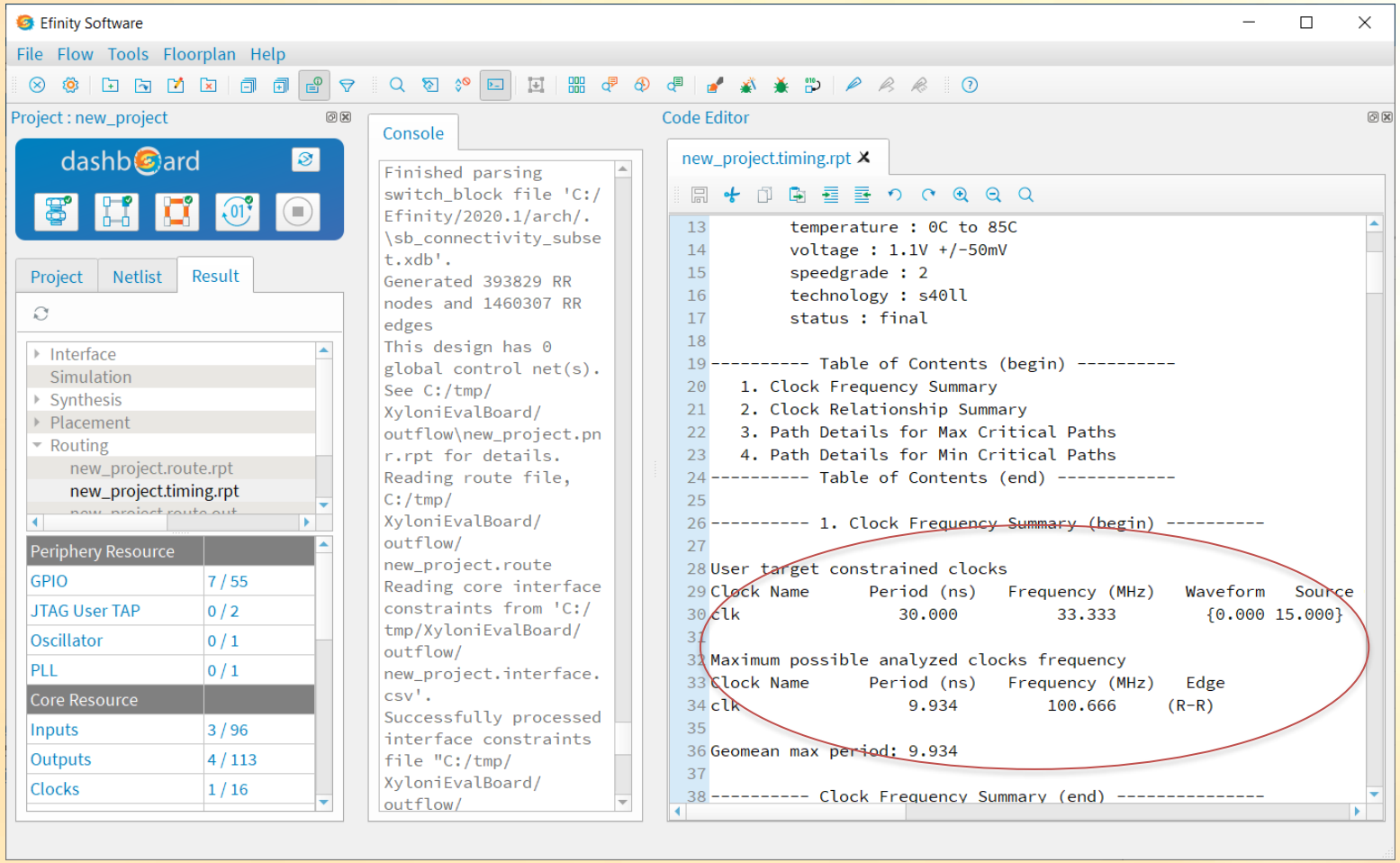
If you have an unassigned pin, check the Placement report, go back to interface designer and fix the name and rerun the flow



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

EFINIX®

Check static timing



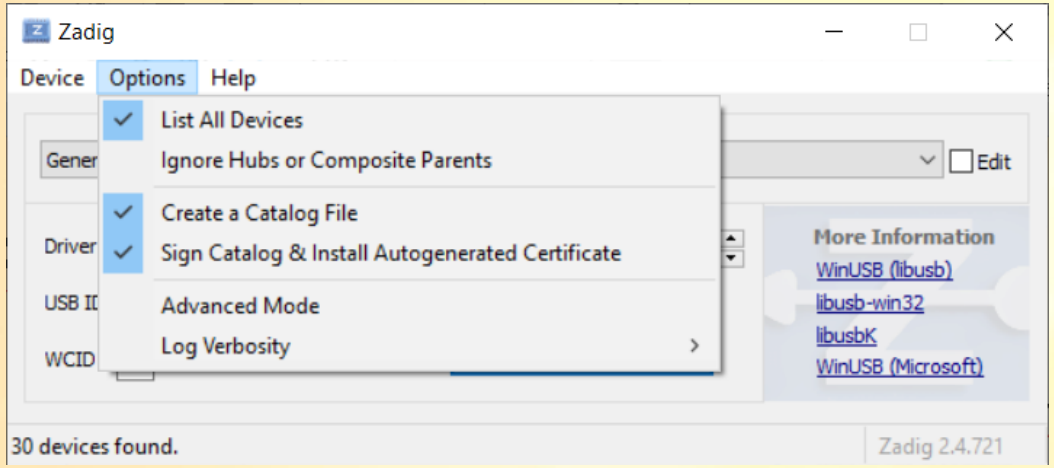
To check the static timing select routing_new_project.timing.rpt.

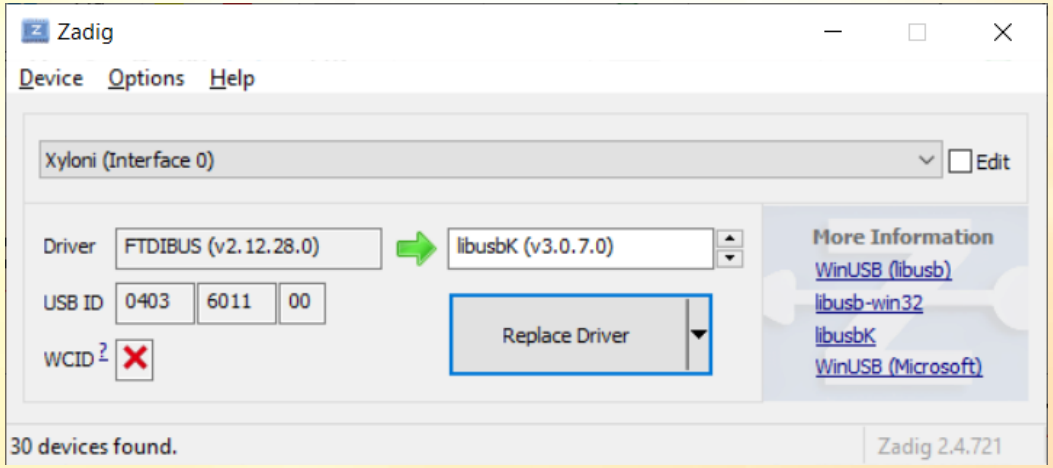
Here you will find the constrains from the constrain file (clk :30ns) and the result

EFINIX®

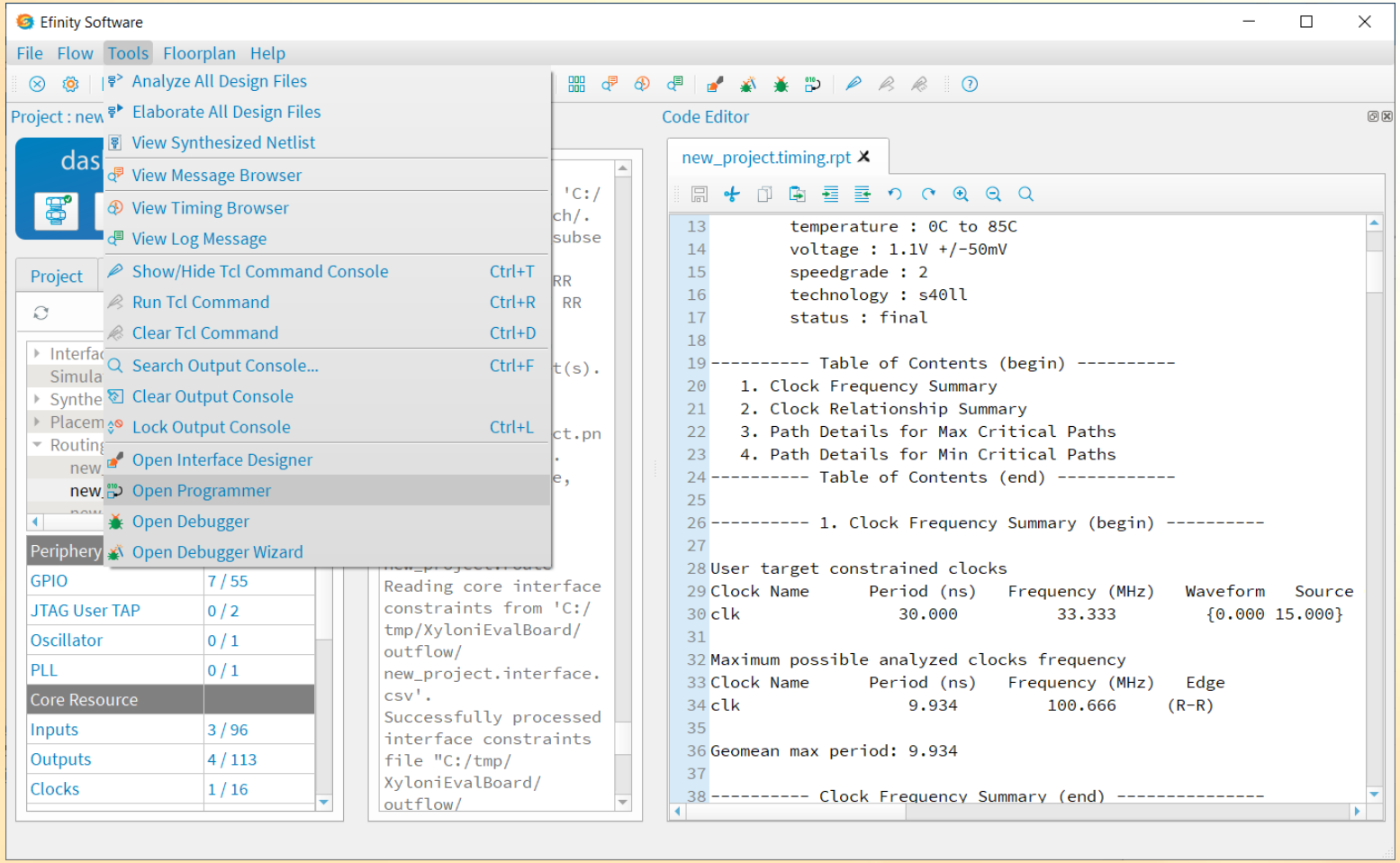
Program the Device

- To programm the Device you have to install the USB driver first !
- Select Xyloni (Interface 0) for SPI programming.
- This is for Software Efinity 2020.1. Efinity 2020.2 will be different !





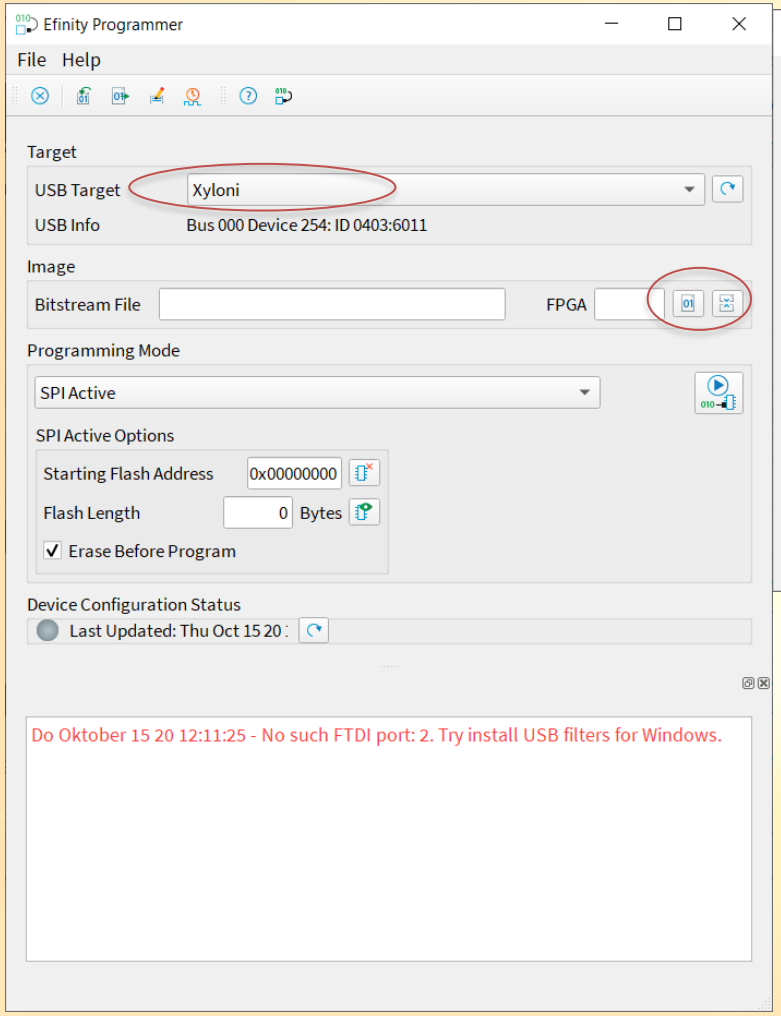
Program the Device

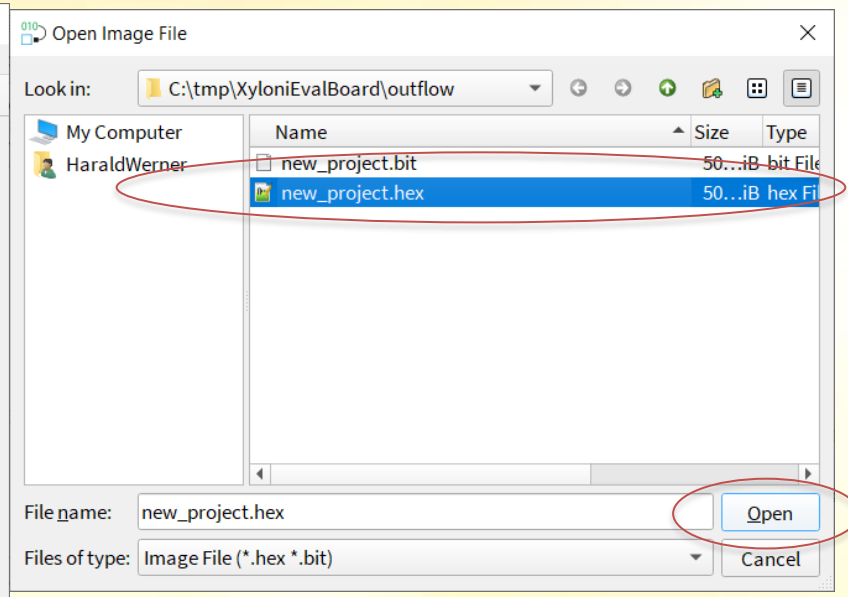


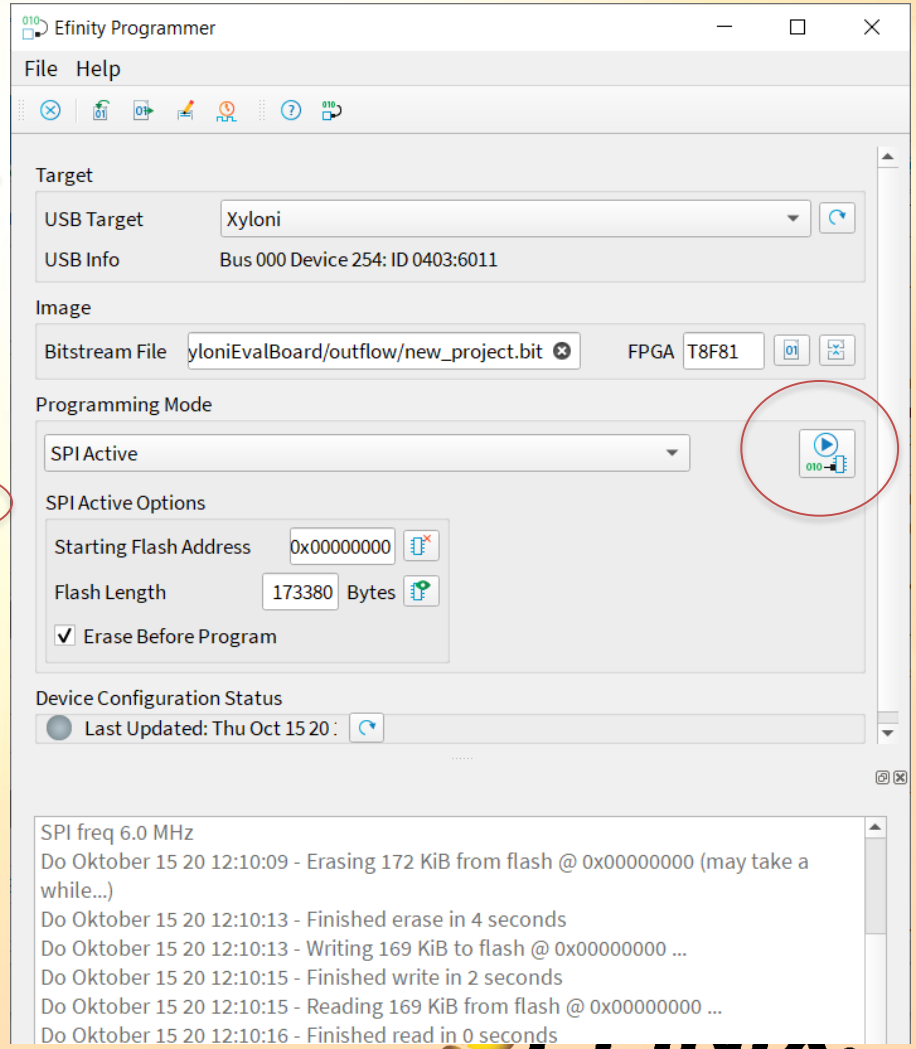
Open Programmer

Tools->OpenProgrammer

Select Image File, Start Program (Check USB Target: Xyloni)





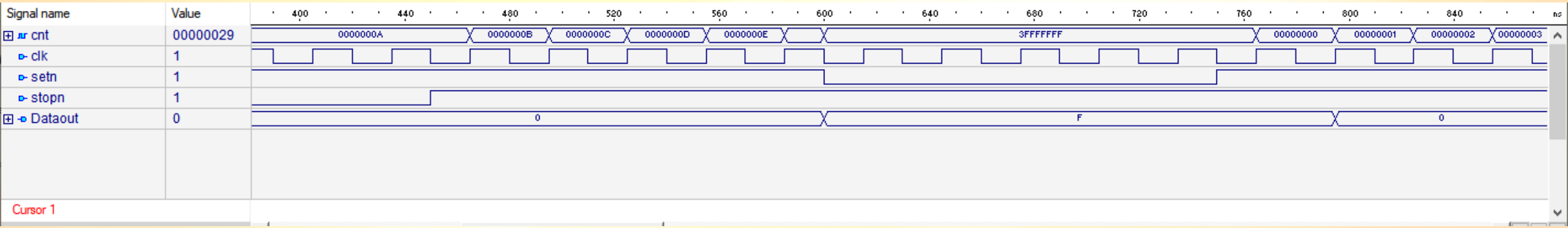


Problem solving

- All LED ON: Check VHDL if you have the correct setting for the frequency ! (29 downto 26) vs (17 donwto 14)
- If you can not see the Xyloni, check if you installed the correct USB driver with zadig

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 with OSC

1. change VHDL file

Code Editor

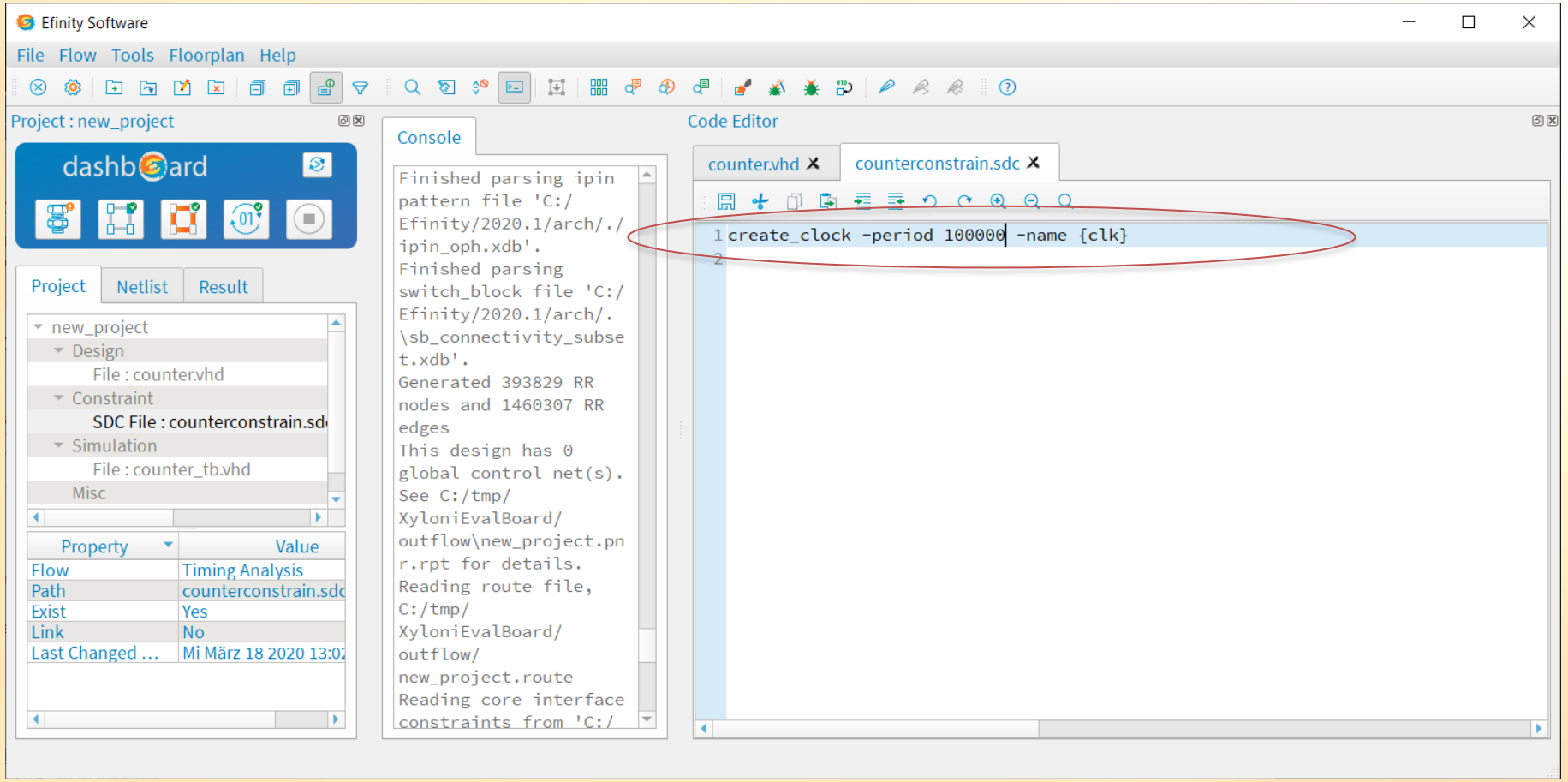
counter.vhd

```
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. Could be from internal osc (T8) or from the external 33.3MHz clock use GPIOl_20
11        setn         : in std_logic;                          --  Set signal, low active; sett all outputs to '1' (LED are high active, means all LEDs msut be ON) GPIOR_02
12        stopn        : in std_logic;                          -- Stop signal, low active Stop counting GPIOR_15
13        Dataout      : out std_logic_vector ( 3 downto 0));   -- Output data connected to the LEDs (high active); GPIOR_17,GPIOR_16,GPIOR_37,GPIOL_21
14 end counter;
15
16 architecture vers1 of counter is
17 signal cnt: std_logic_vector ( 29 downto 0) := (others => '0');
18 Begin
19 cnt_proc : process(clk, setn)
20            Begin
21                if setn = '0' then
22                    cnt     <= (others => '1');
23                    dataout <= (others => '1');
24                elsif clk'event and clk = '1' then
25                    if stopn = '0' then
26                        cnt <= cnt;
27                    else
28                        cnt <= cnt +1;
29                    end if;
30                    Dataout <= cnt(17 downto 14);          --17 downto 14 OK for the 10Khz internal oscillator. For the 33MHz external clock use 29 downto 26
31 --                 Dataout <= cnt(29 downto 26);      --17 downto 14 OK for the 10Khz internal oscillator. For the 33MHz external clock use 29 downto 26
32                end if;
33            end process;
34 end vers1;
35
```


EFINIX®

Design with OSC

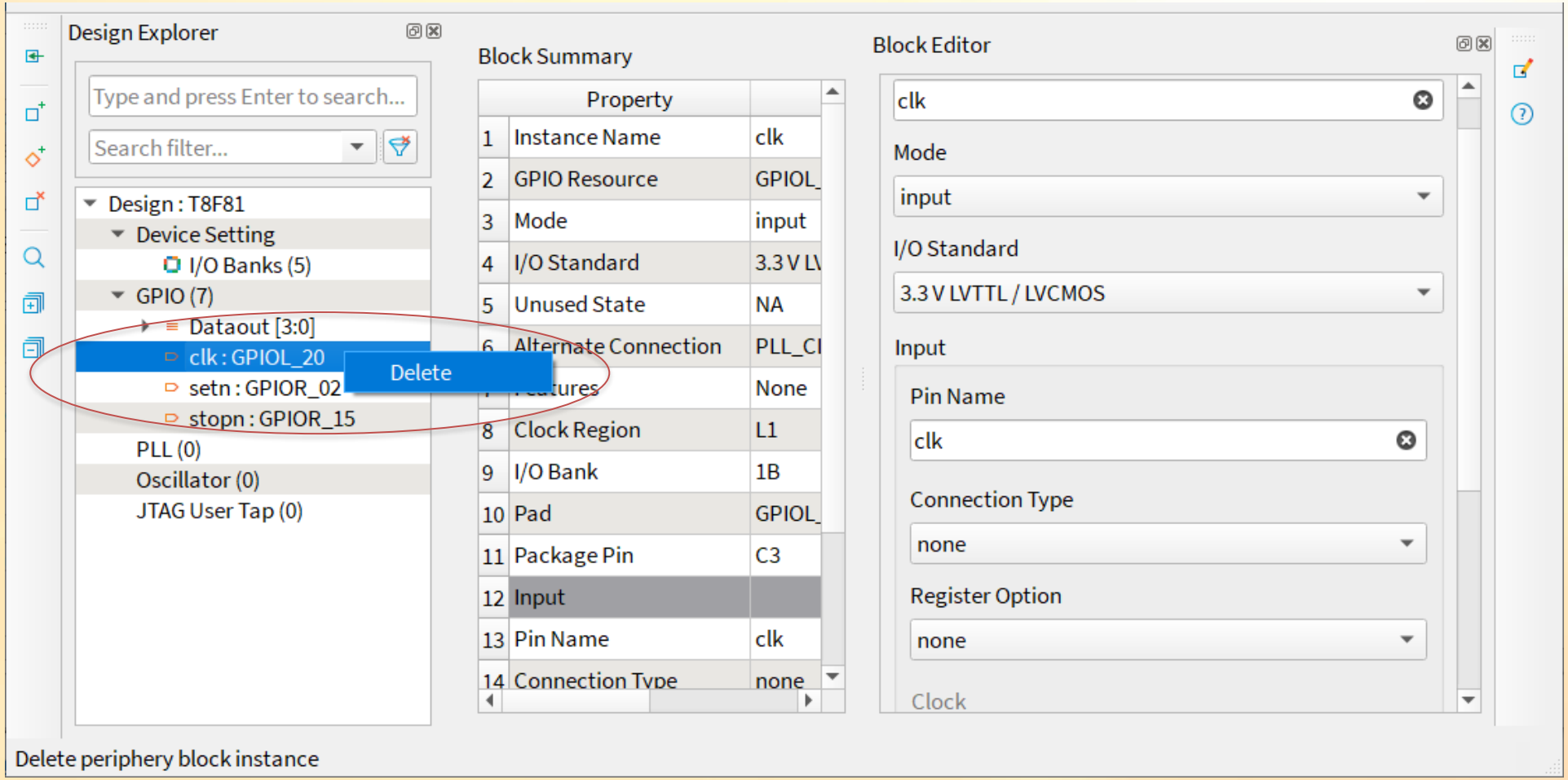
2. change constrain file



EFINIX®

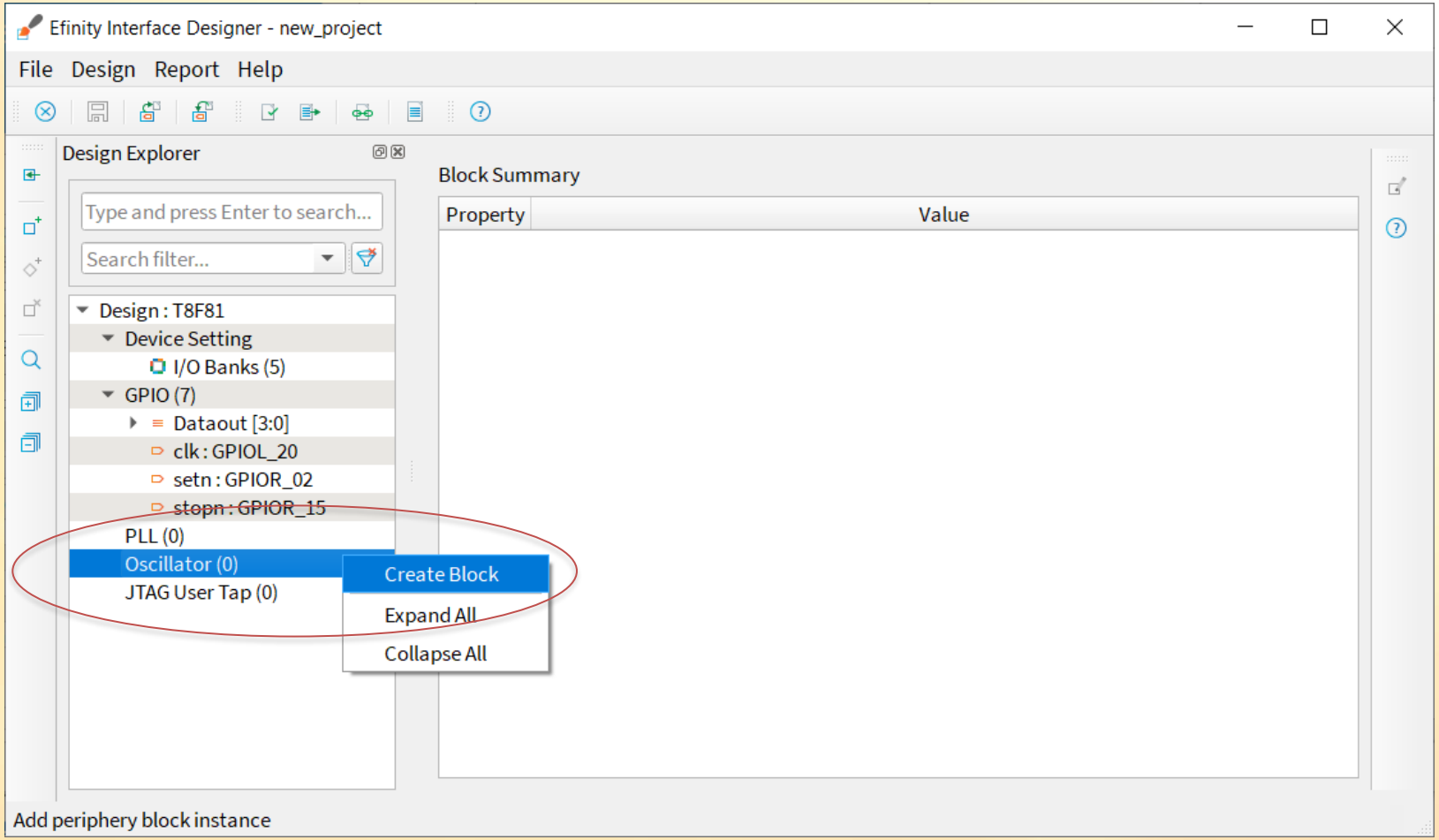
Change Interface designer

- 3. Delete clk GPIO; Seelct clk->RMB->Delete



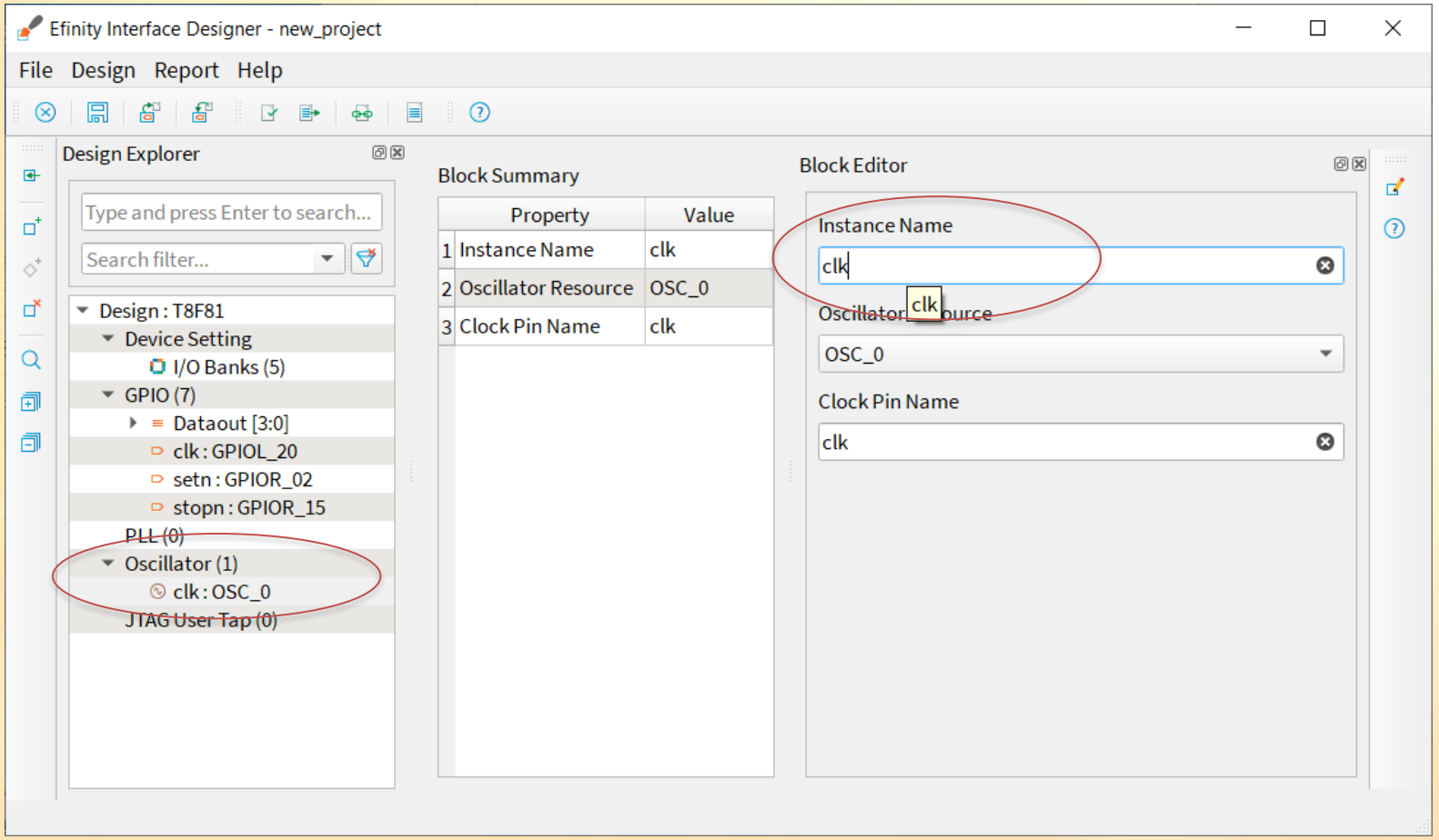
Change Interface designer

4. Add OSC; select Oscillator; RMB; Create Block

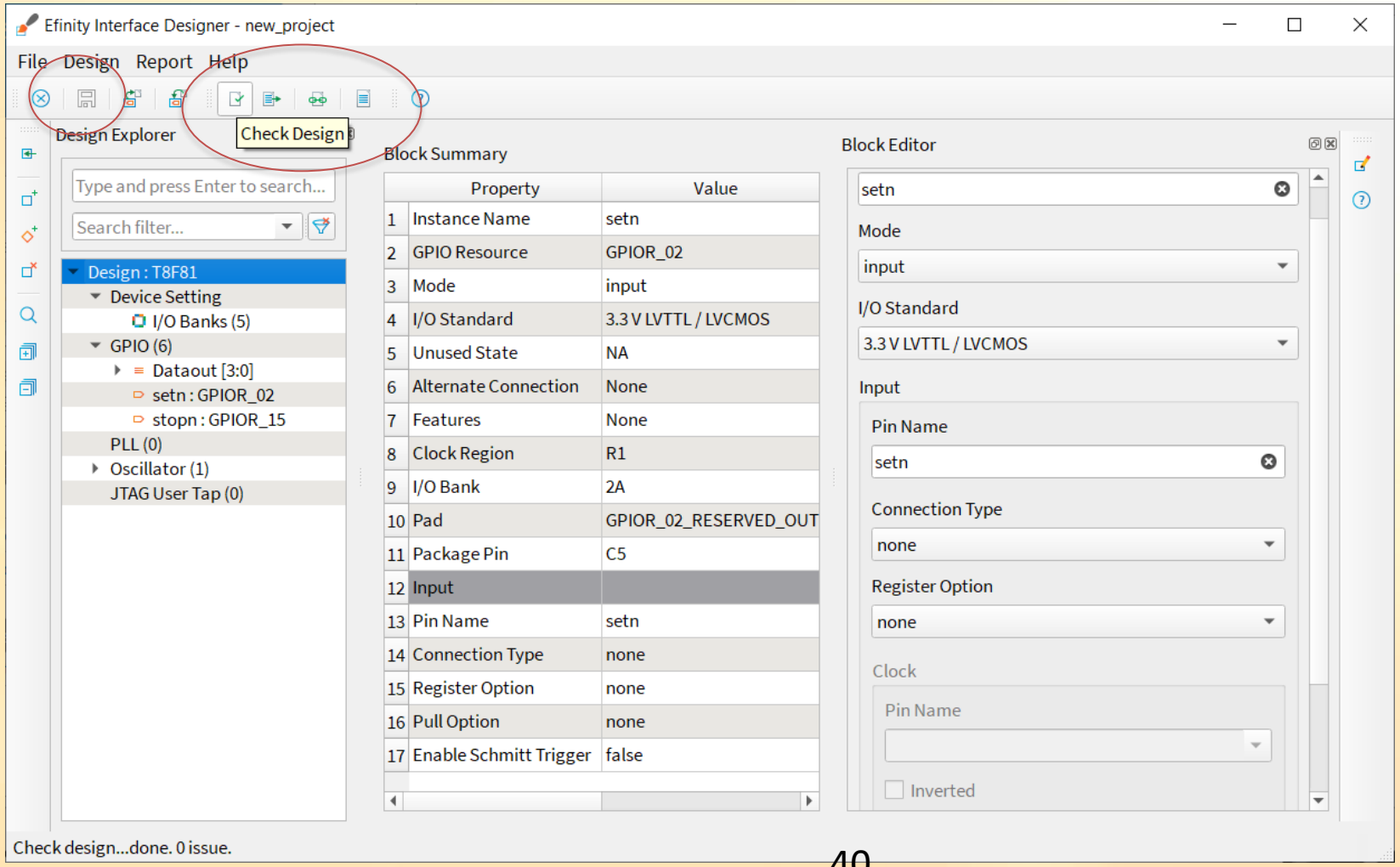


Change Interface designer

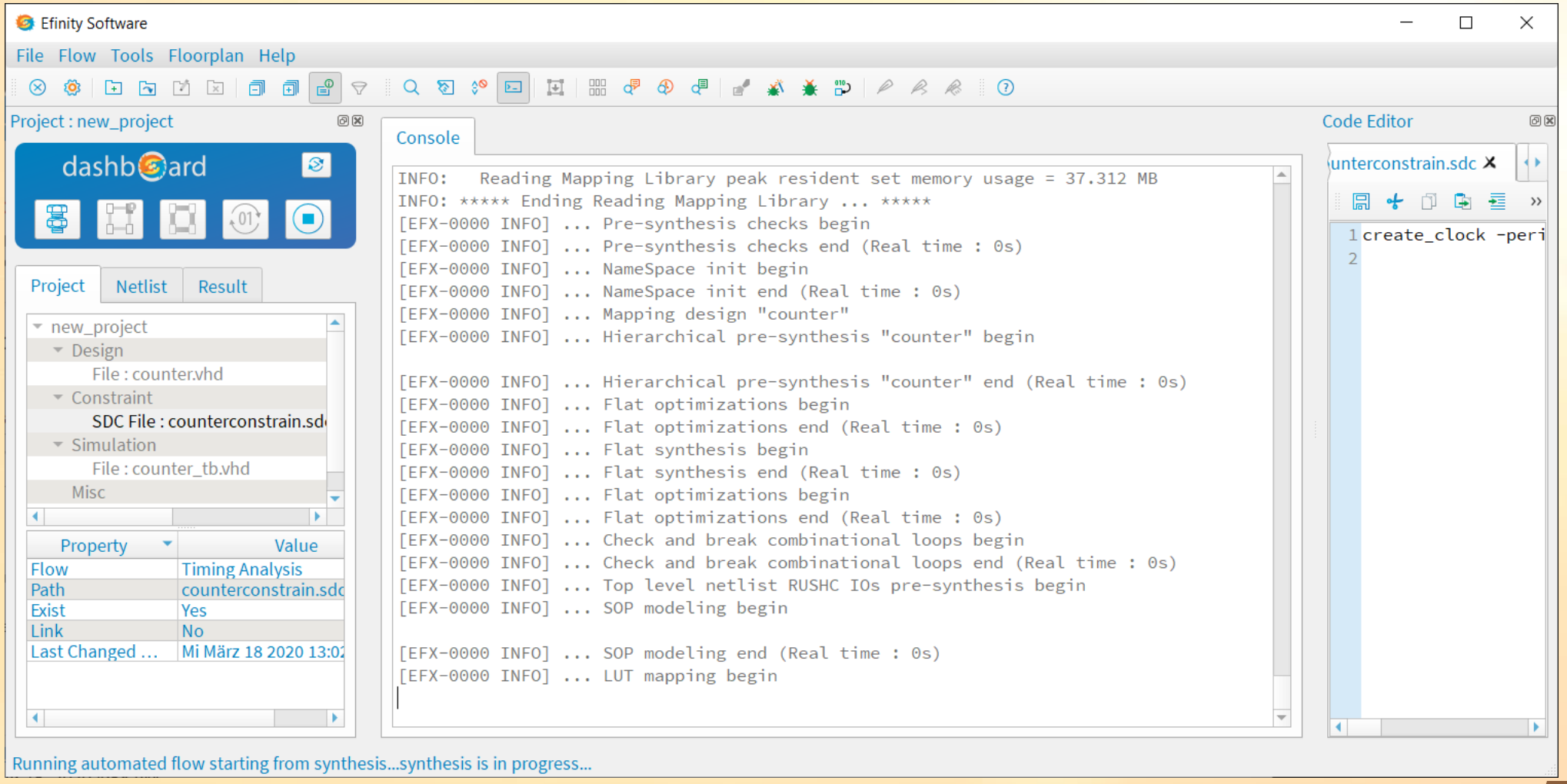
5. Assign clk to OSC Instance Name, **press ENTER !**



Check Design and save

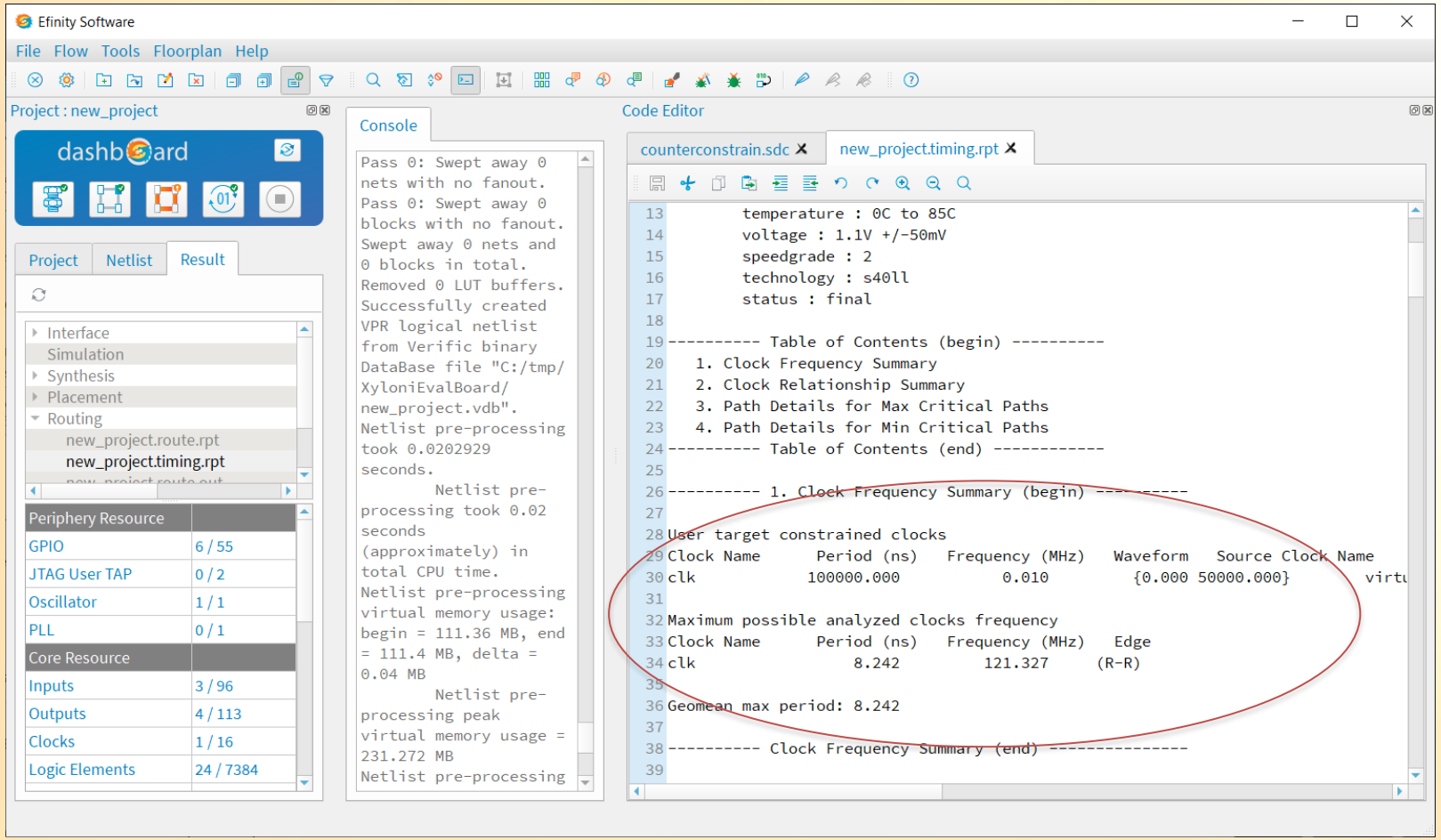


Run the whole flow with pressing Synthesis



EFINIX®

Check static timing

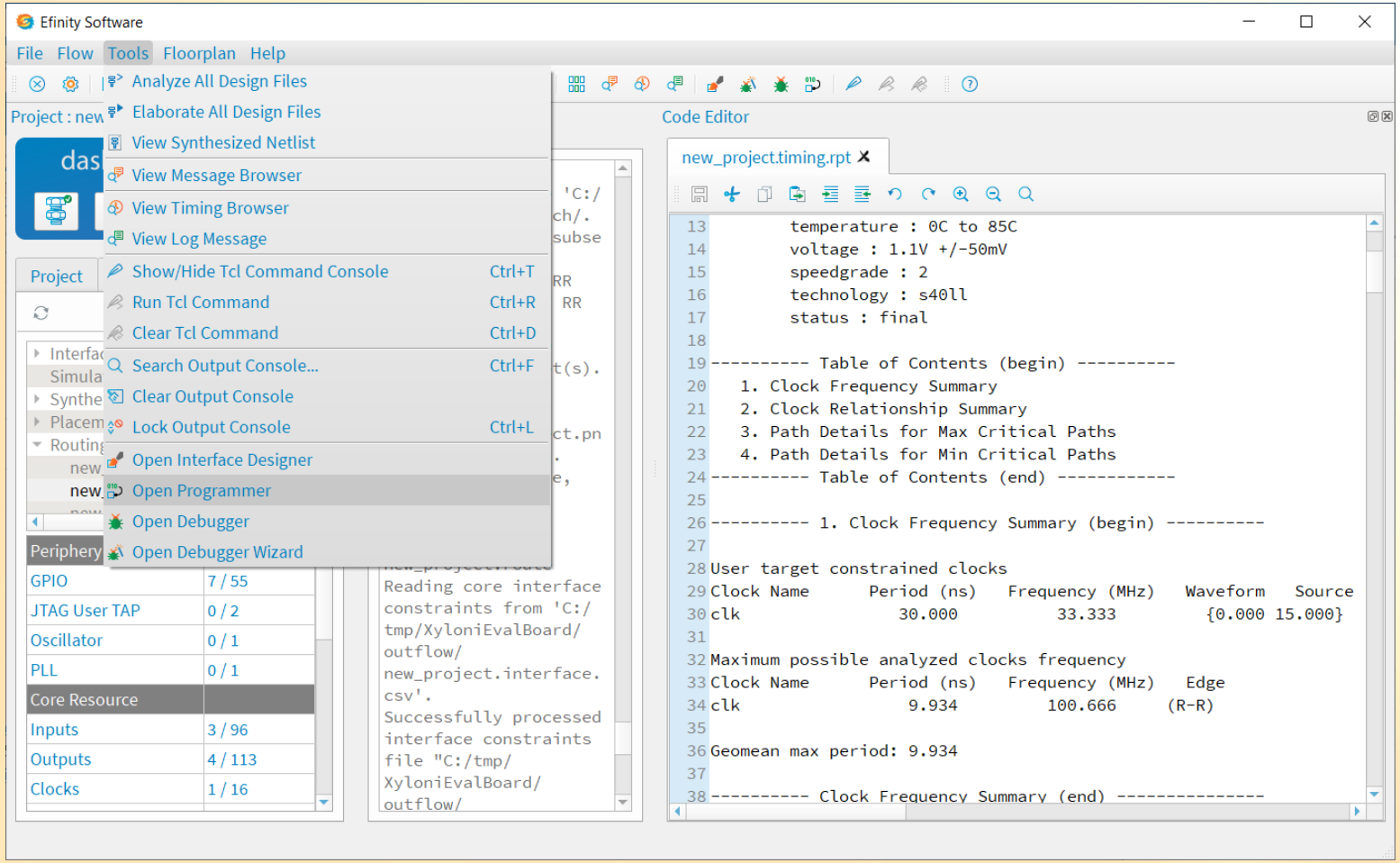


To check the static timing select routing_new_project.timing.rpt.

Here you will find the constrains from the constrain file (clk :100000ns) and the result

EFINIX®

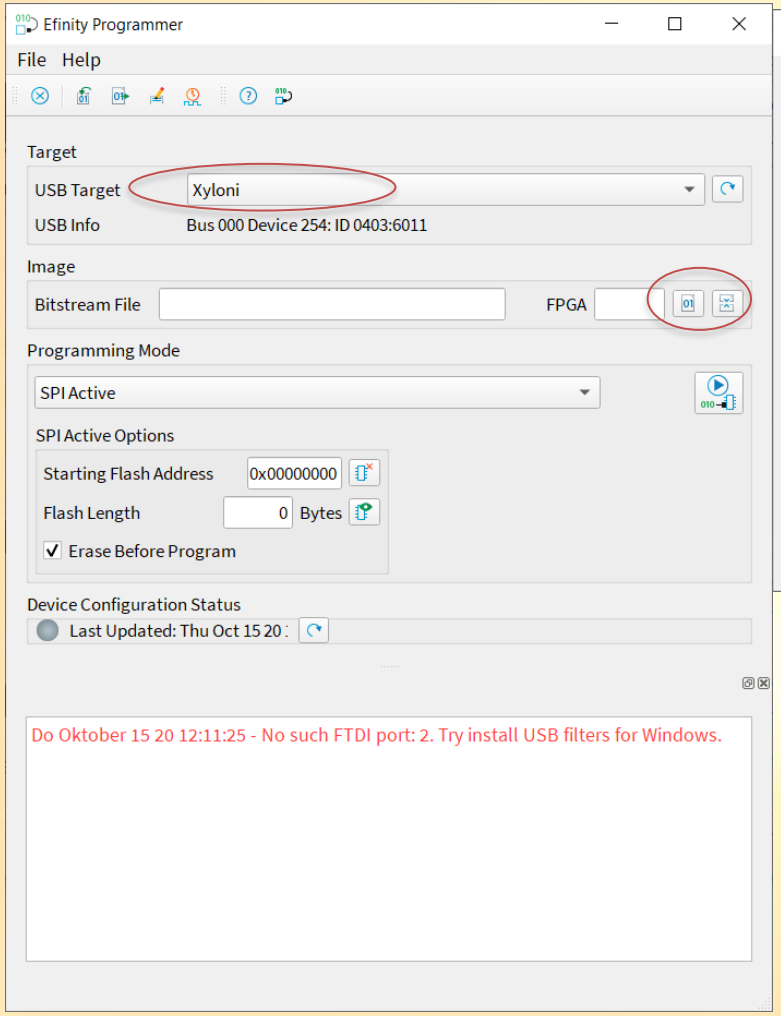
Program the Device

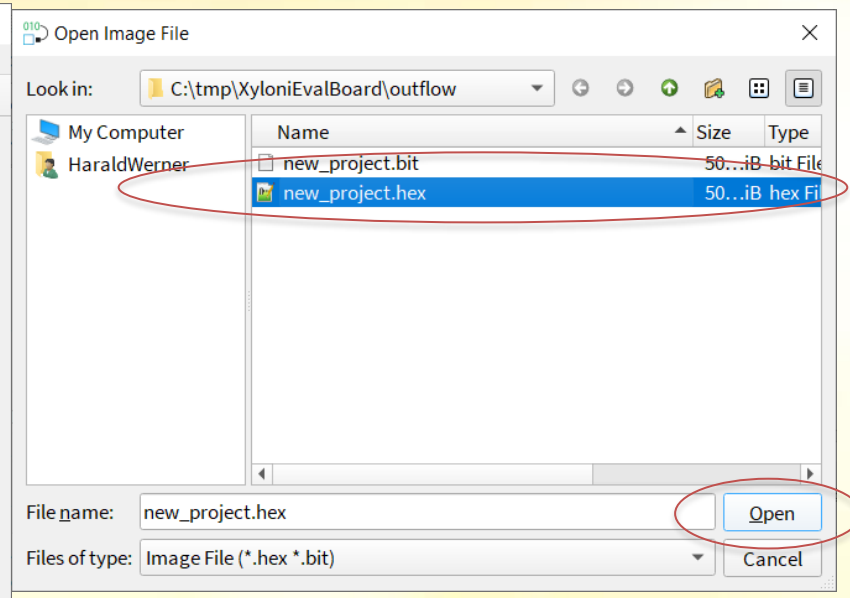


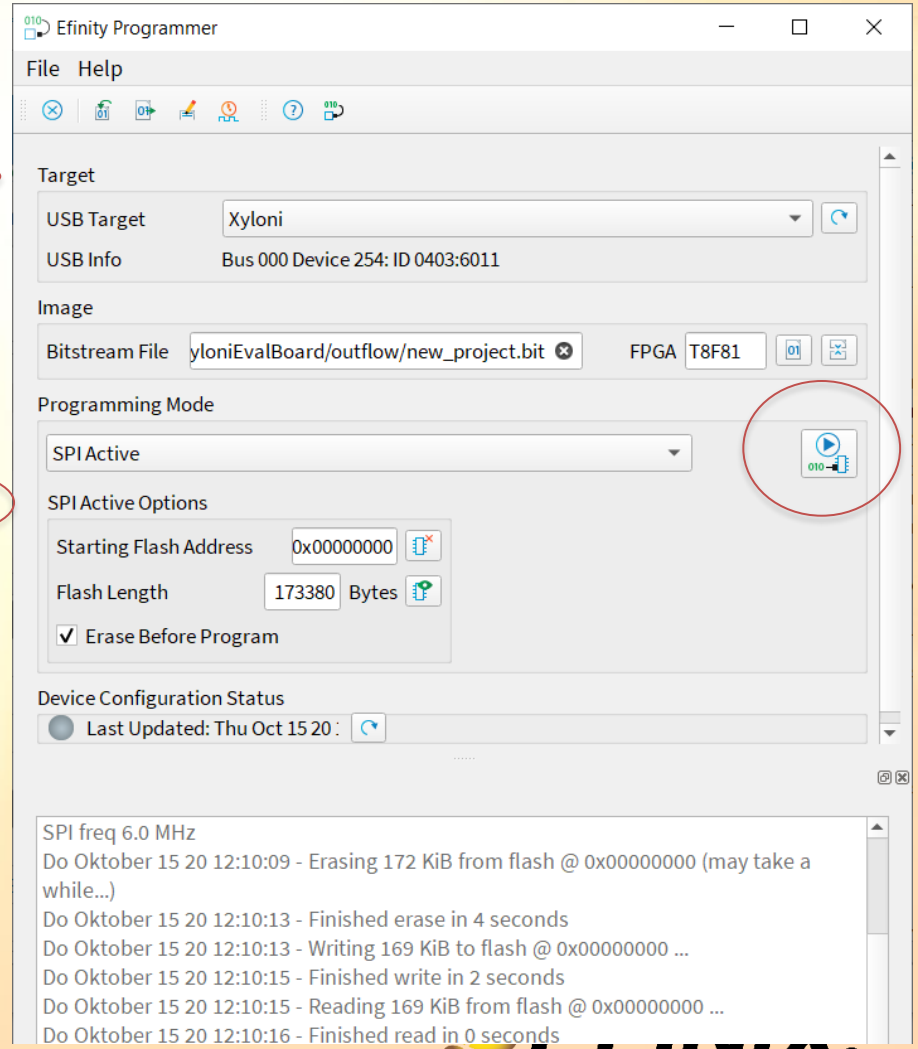
Open Programmer

Tools->OpenProgrammer

Select Image File, Start Program (Check USB Target: Xyloni)







Information

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