ΘΈΜΑ ΠΤΥΧΙΑΚΉΣ ΕΡΓΑΣΊΑΣ :

ΥΛΟΠΟΙΗΣΗ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ ΤΟΥ MICROBLAZE ΕΠΕΞΕΡΓΑΣΤΗ ΣΕ FPGA

ΑΡΙΘΜΌΣ ΜΗΤΡΏΟΥ :

2026202100134

ΟΝΟΜΑΤΕΠΏΝΥΜΟ :

ΑΝΑΣΤΆΣΙΟΣ ΠΟΥΤΑΧΊΔΗΣ

ΠΑΝΕΠΙΣΤΉΜΙΟ :

Πελοποννήσου.

ΤΜΉΜΑ :

Ψηφιακών Συστημάτων.

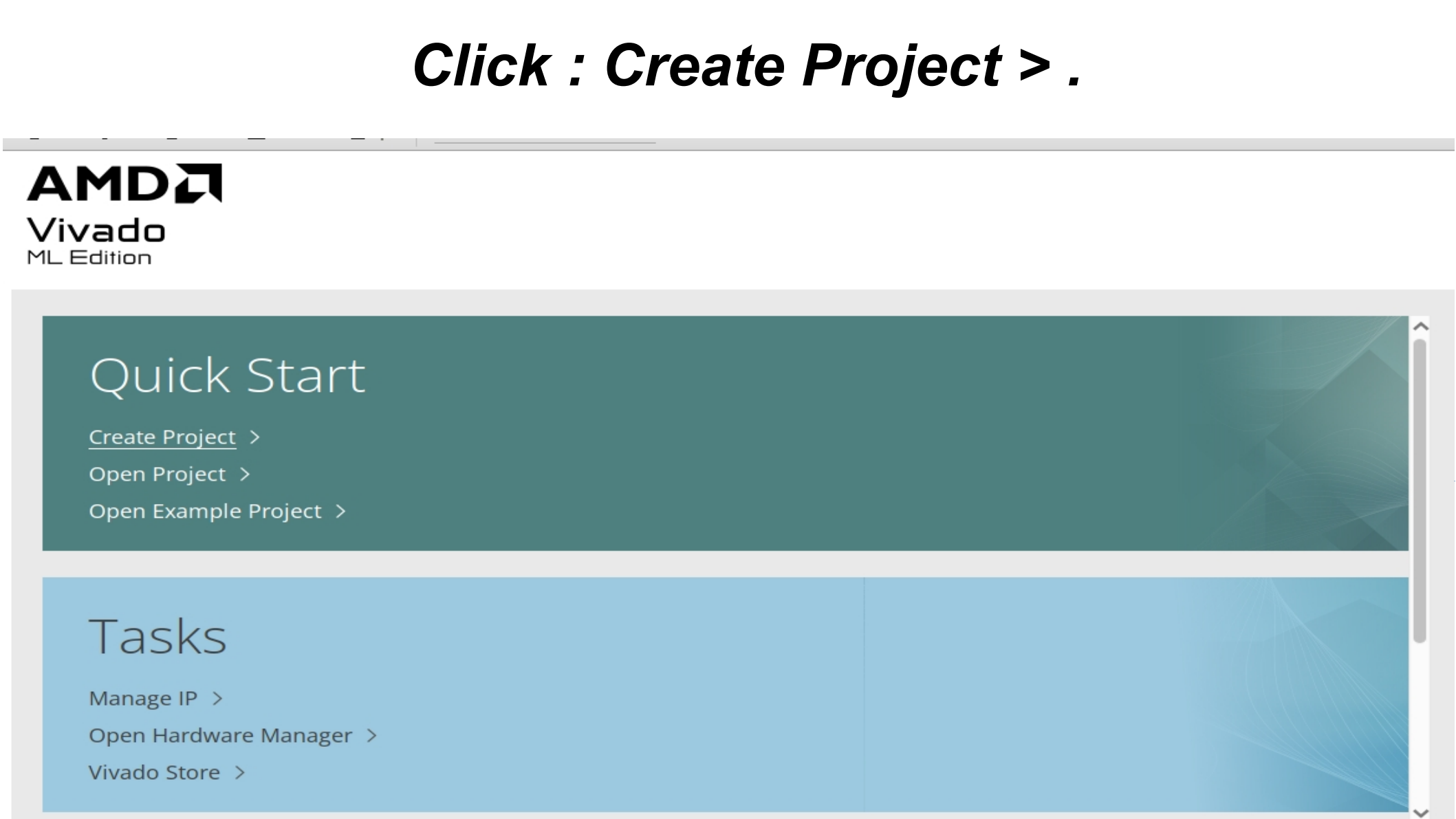
ΠΑΡΟΥΣΊΑΣΗ :

Πτυχιακής Εργασίας 20’.

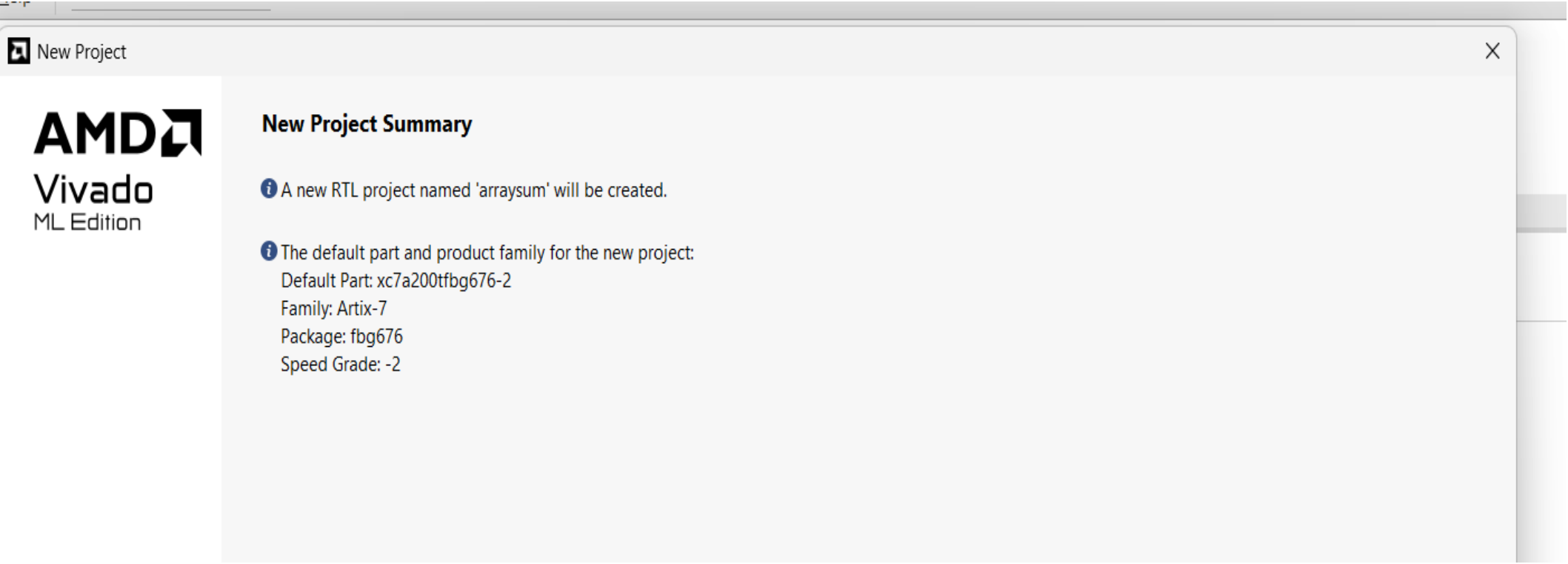
ΕΡΓΑΛΕΊΑ ΥΛΟΠΟΊΗΣΗΣ :

Vitis(SW DEVELOPER)2023.2

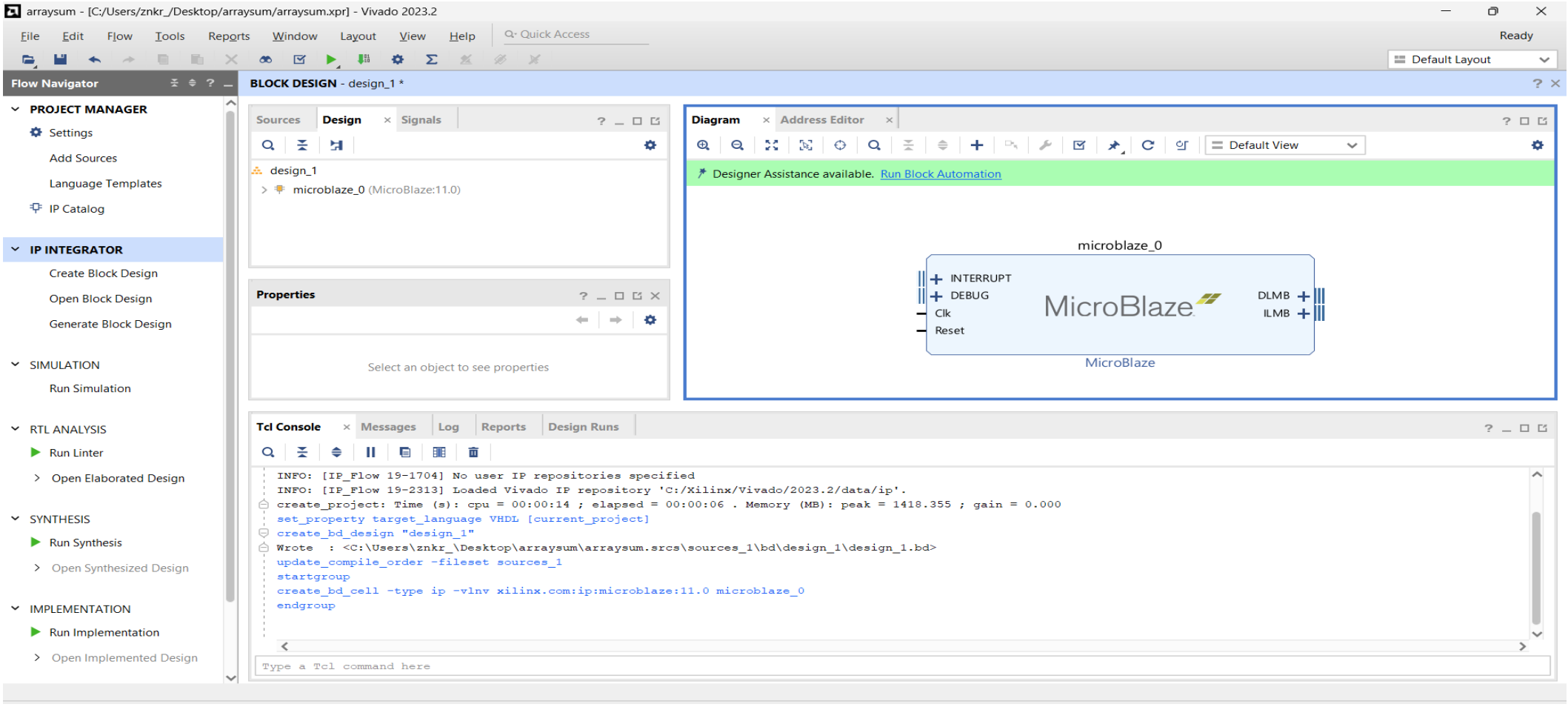
Click : Create Project > .



Project Name : arraysum
Xilinx Board Part : xc7a200tfbg676-2

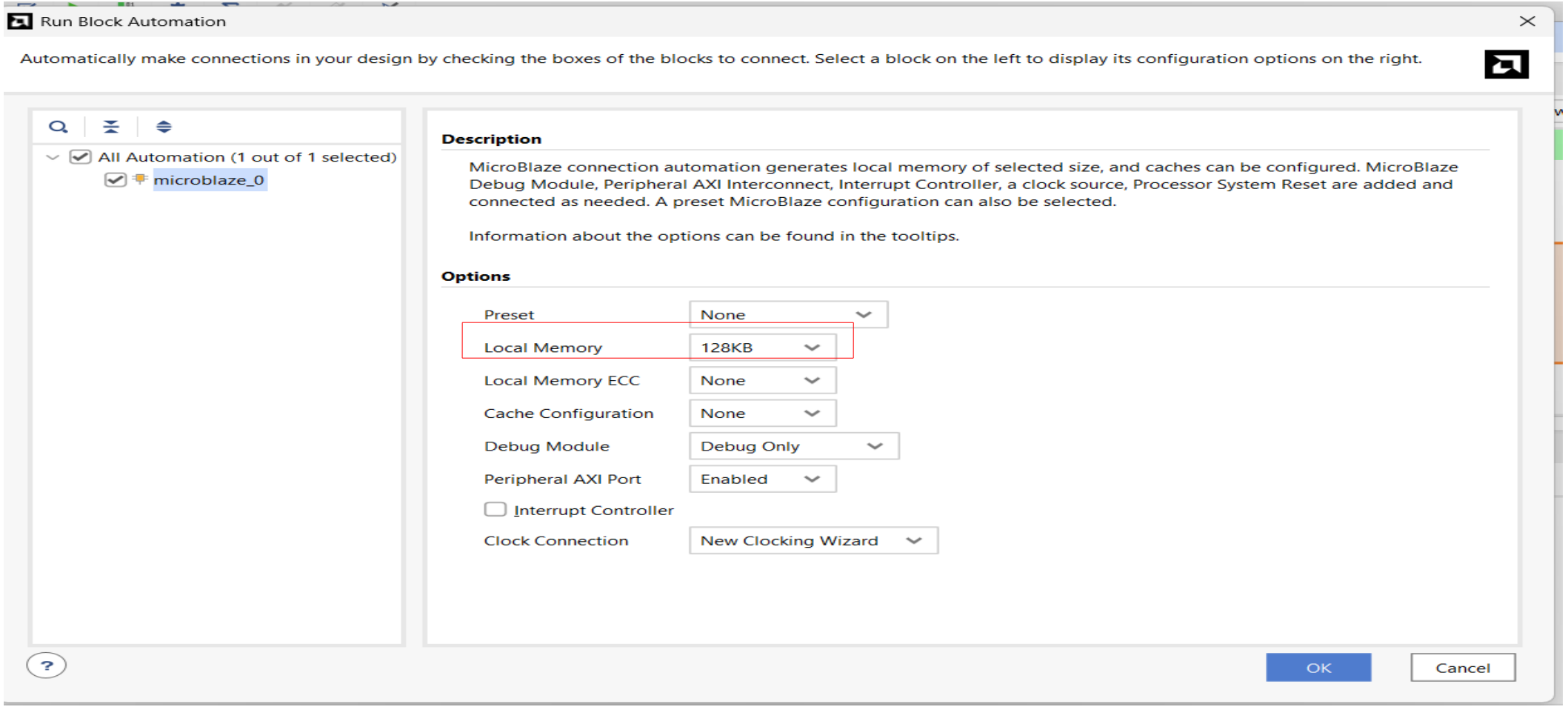


Create Block Design :
Add MicroBlaze and Run Block Automation.

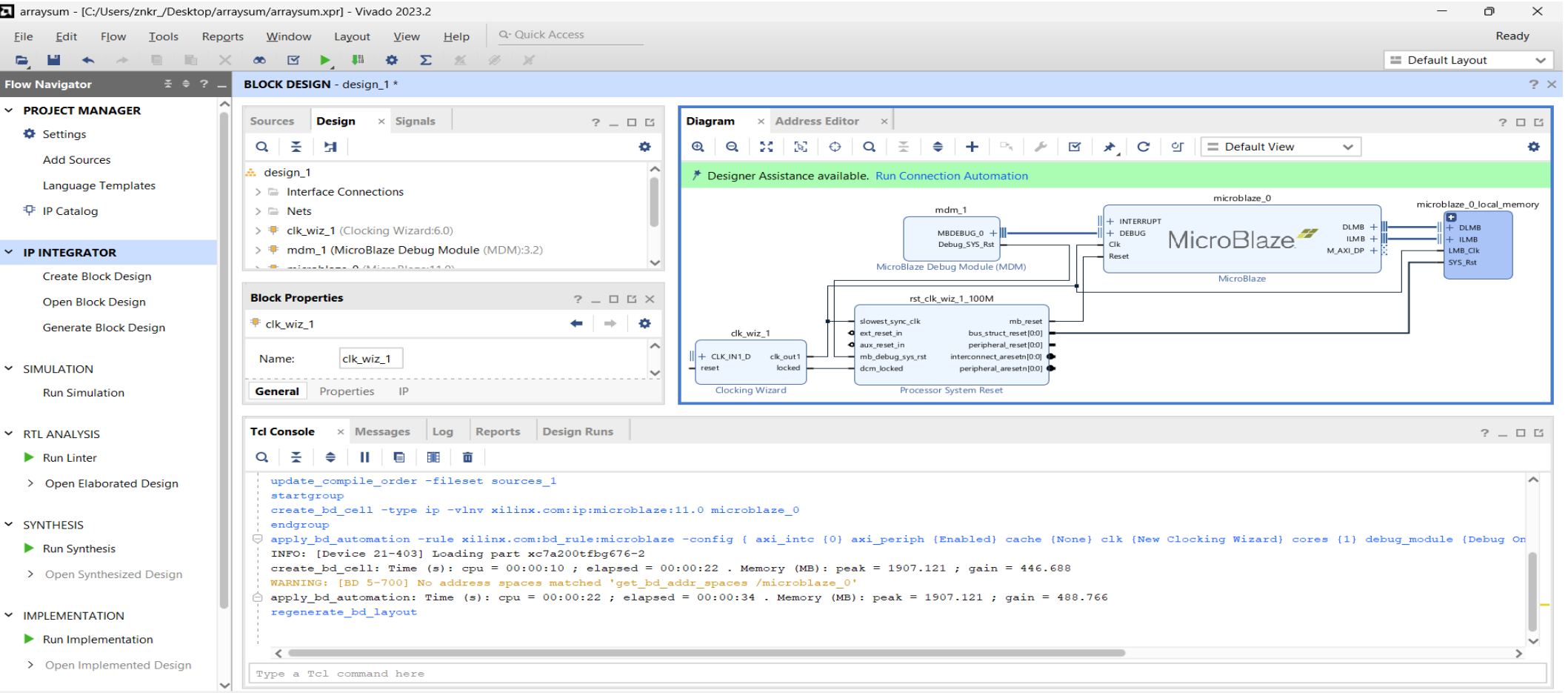


Set : Local Memory 128KB and Click OK.

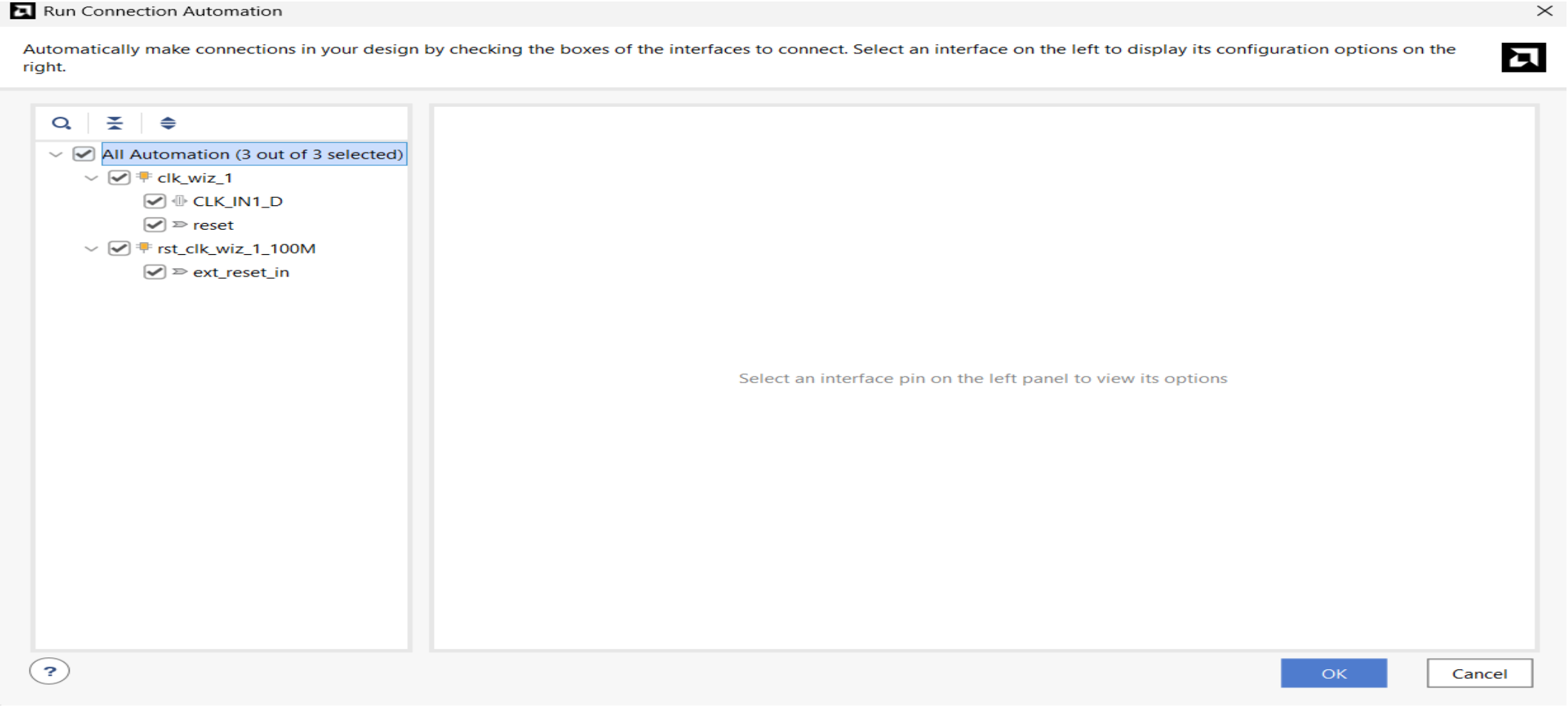
(Improves Performance).



Click : Run Connection Automation.

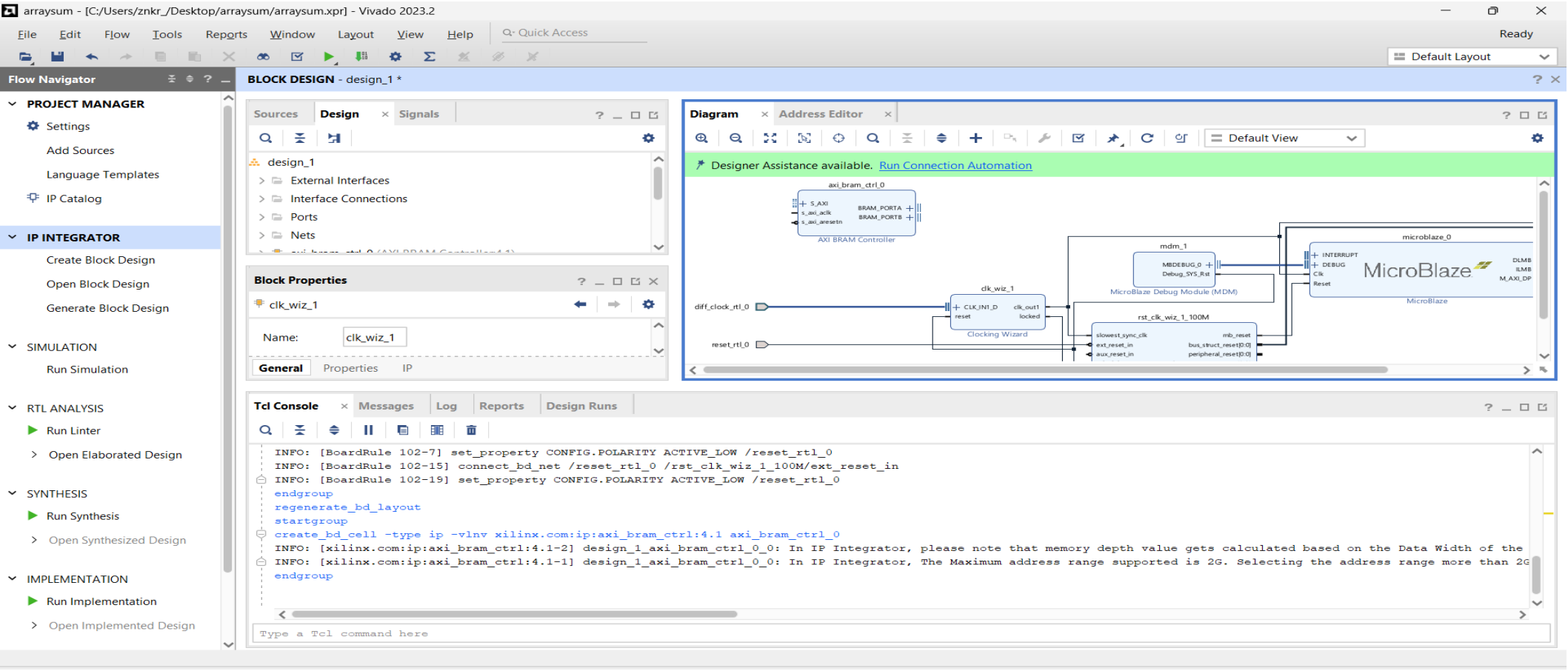


Select : All(3 out of 3 selected) and Click OK.

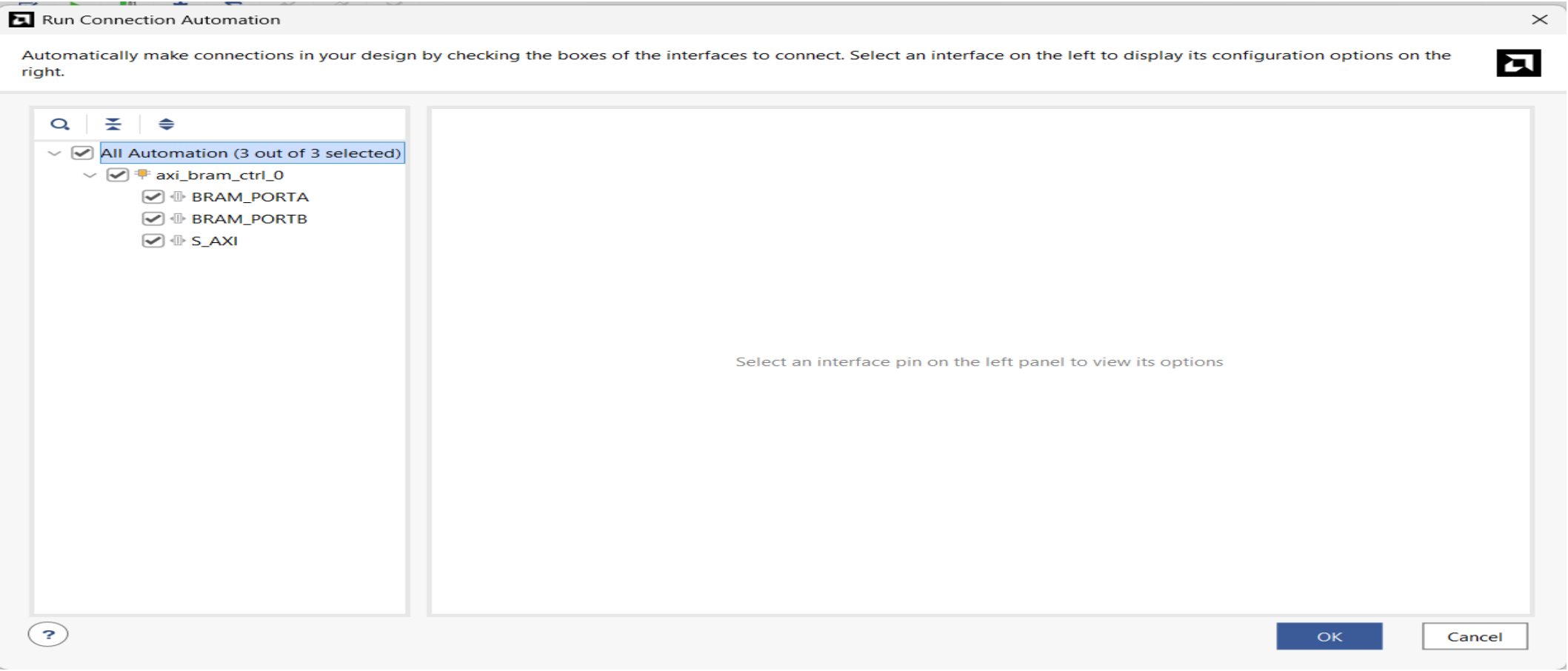


Add BRAM and Run Connection Automation.

(BRAM : boosts system performance by minimizing the delay in data access).



Select : All(3 out of 3 selected) and Click OK.

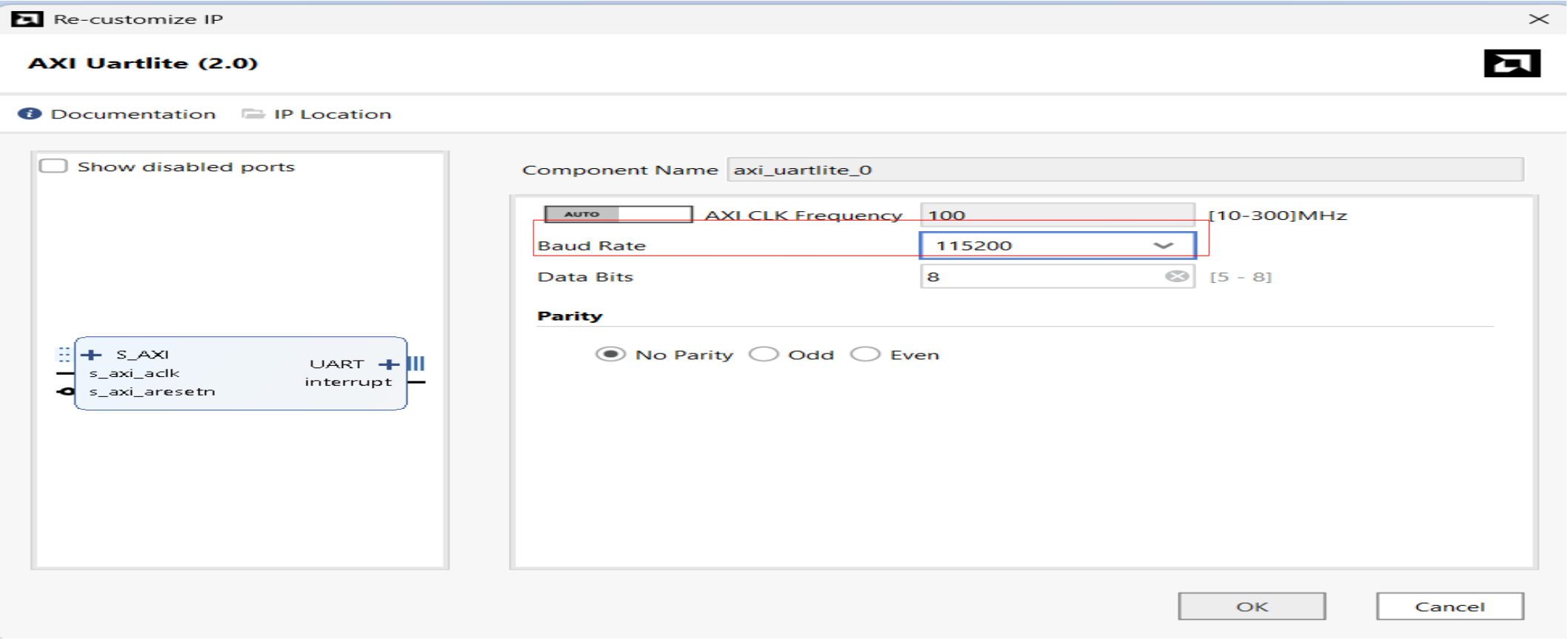


Add UART and Set : Baud Rate 115200.

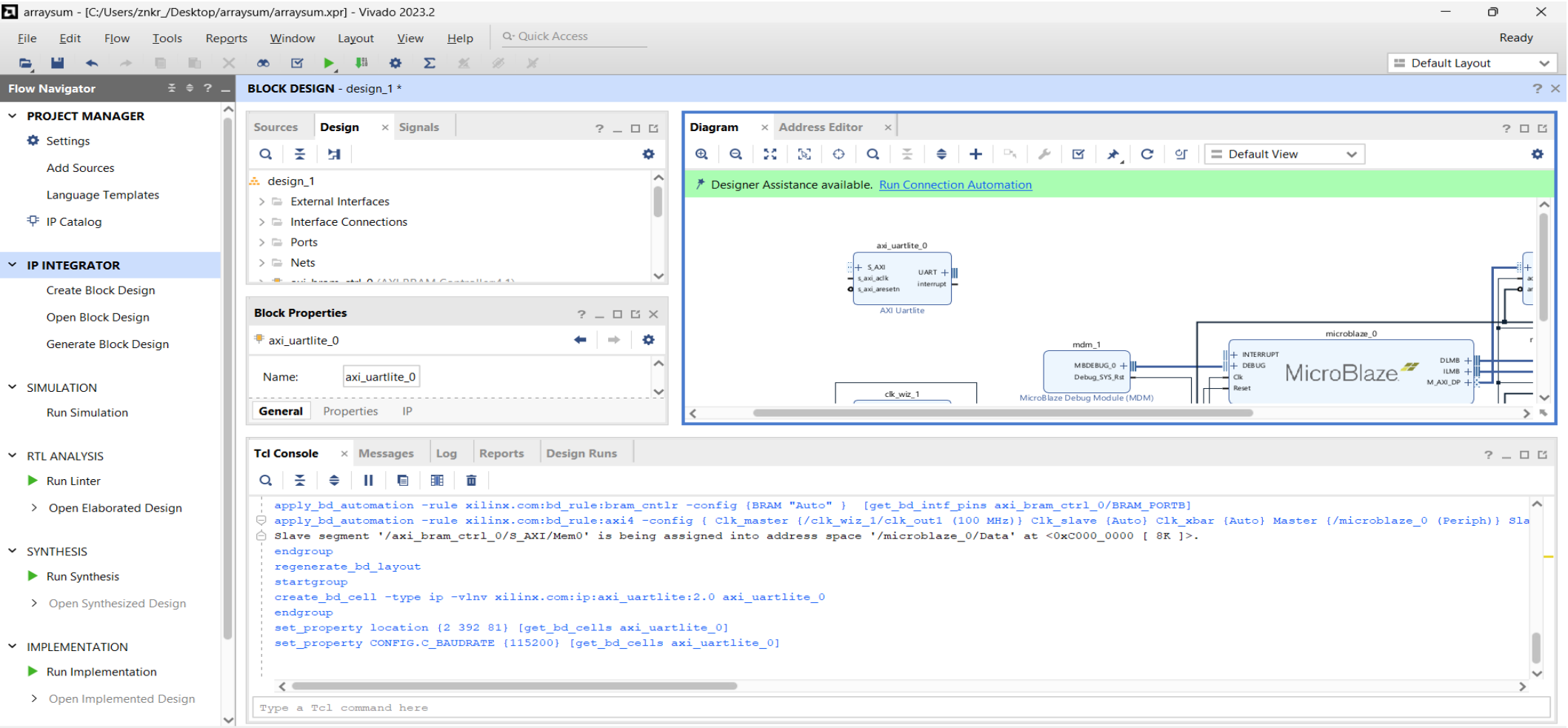
***Baud Rate:**
Symbols/Second

(My testbench is written for this baud rate).

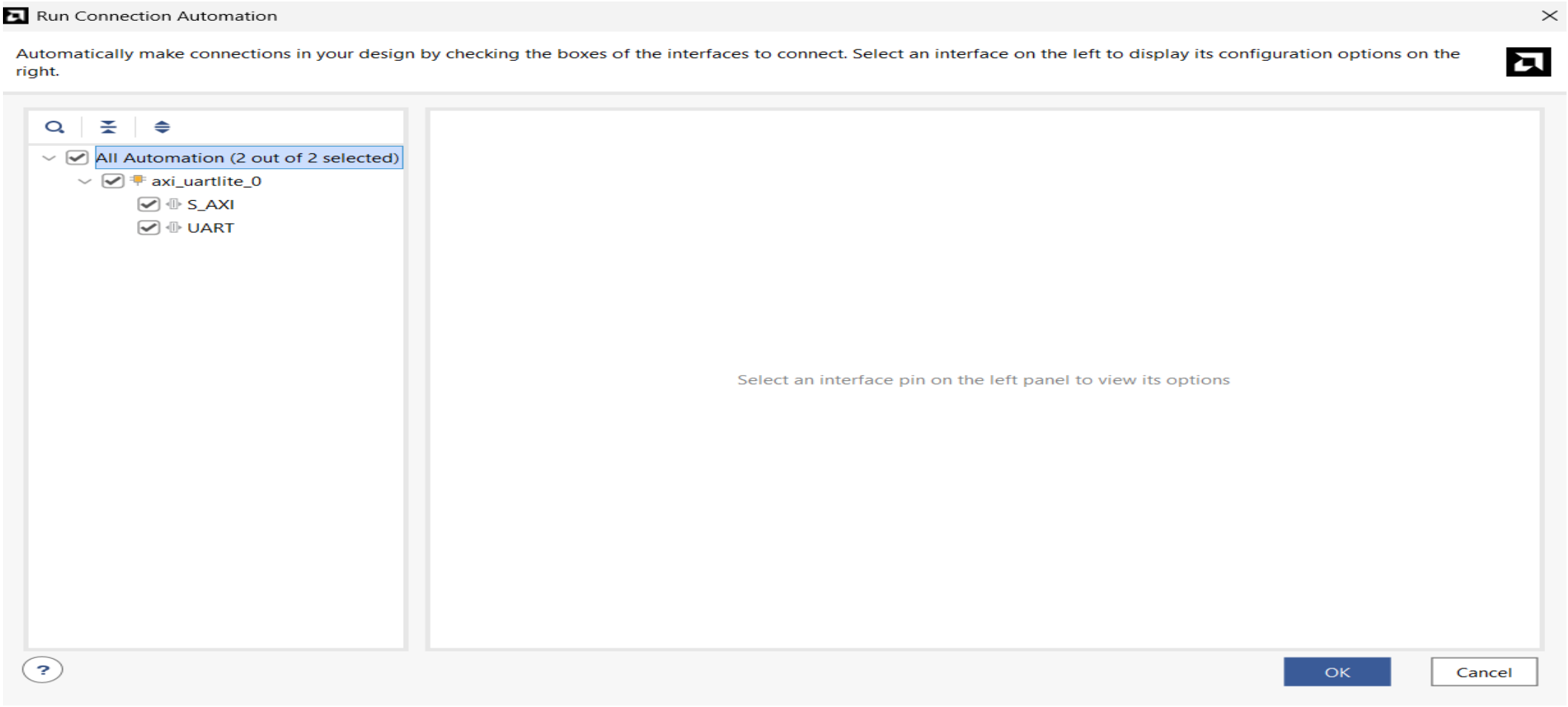
(UART : MicroBlaze can send and receive serial data with external devices).



Click : Run Connection Automation.

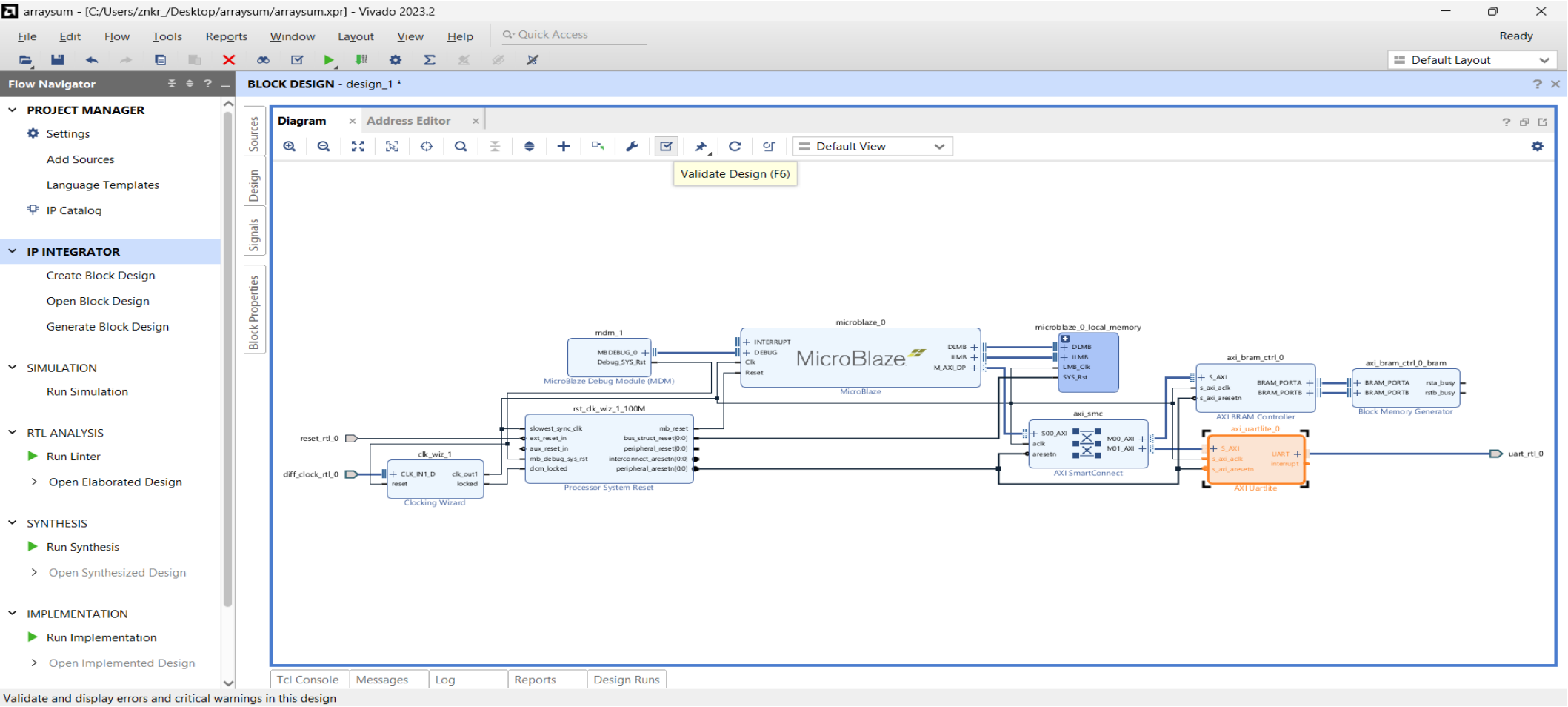


Select : All(3 out of 3 selected) and Click OK.



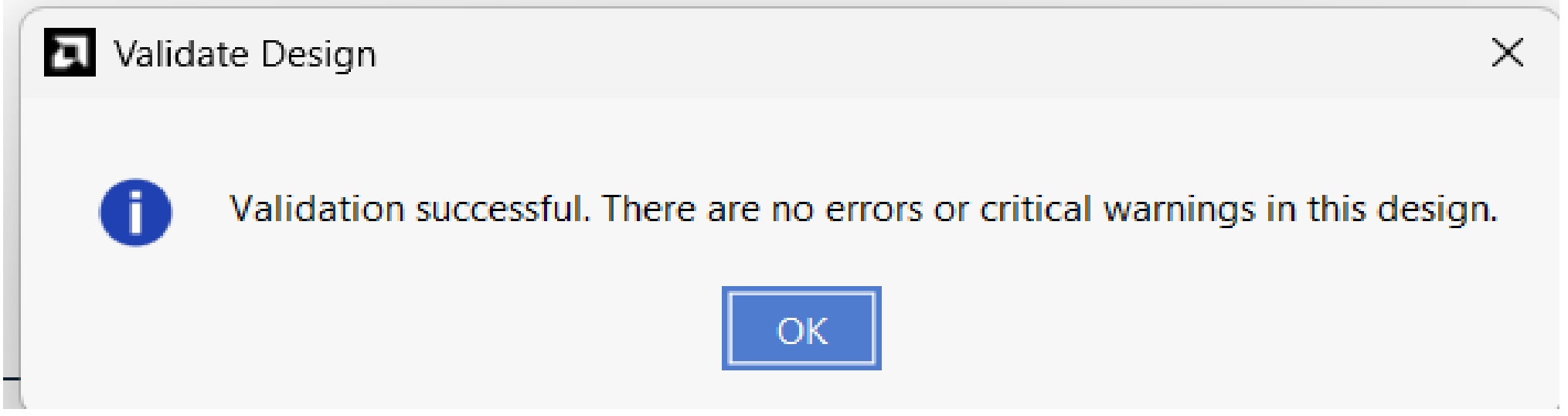
Click : Validate Design (F6).

(Given : The Form Of The Block Design).



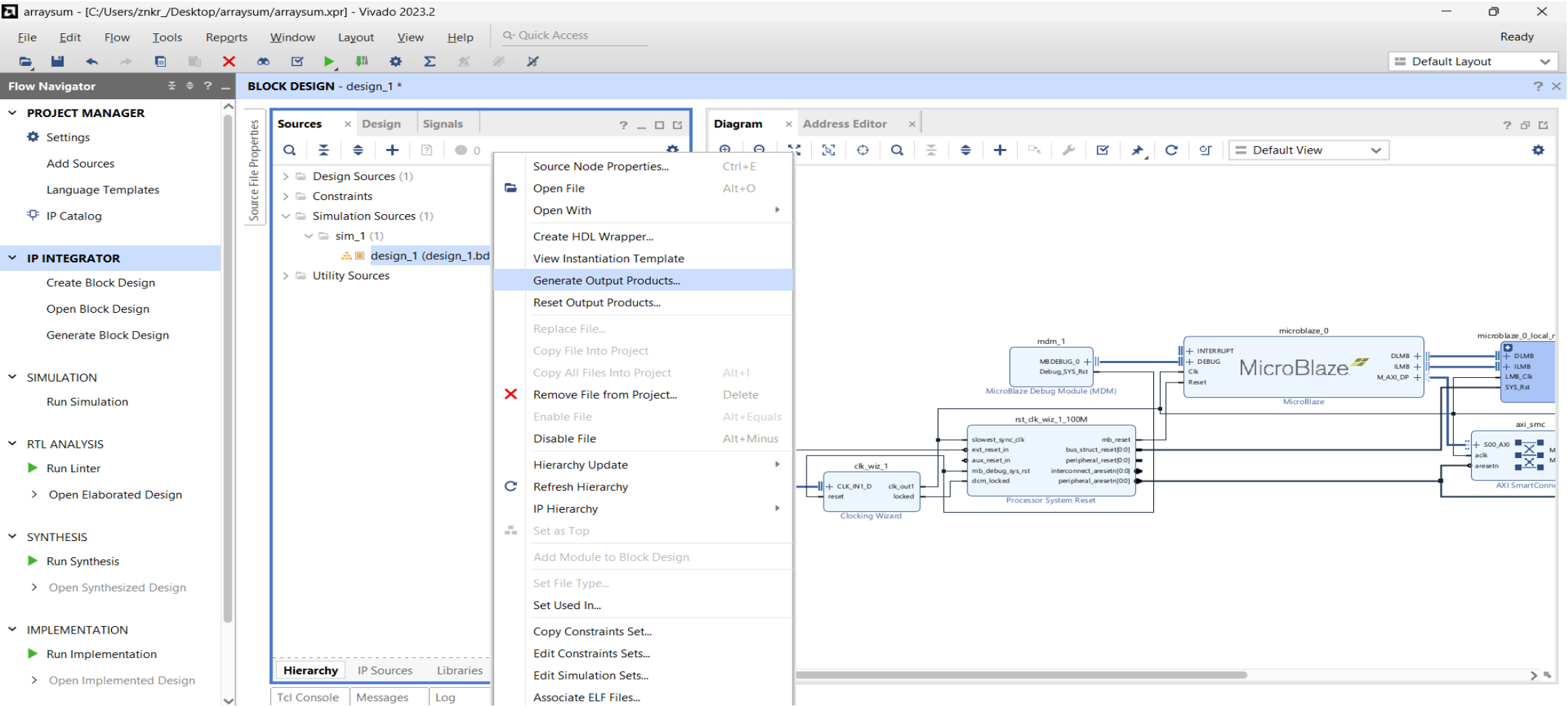
Click : OK. If The Validation Is Successful.

(This window will open).



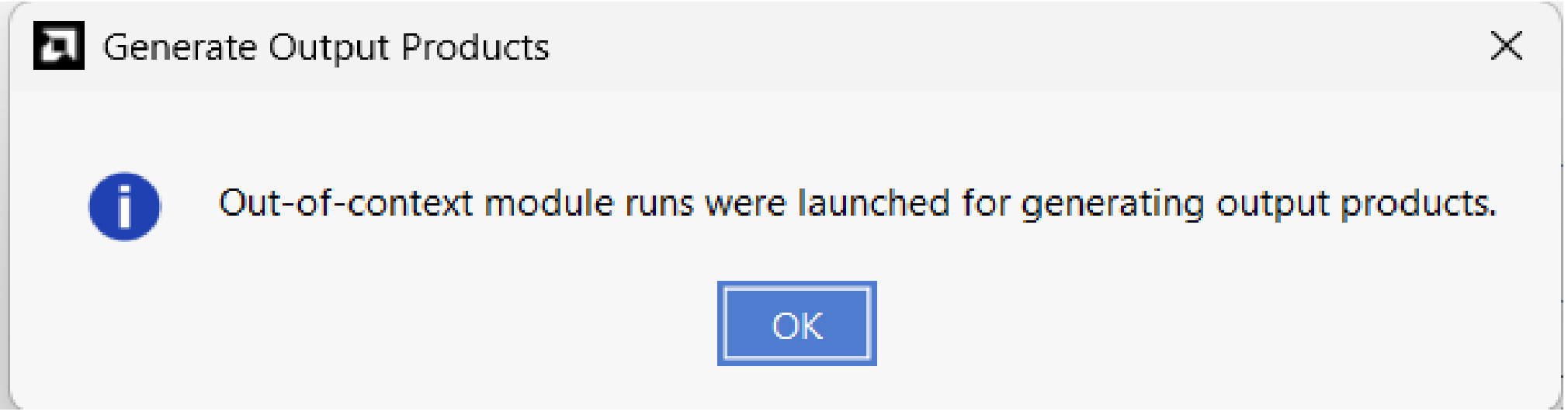
Click : Generate Output Products…

(For synthesizing the design and implementing it to test the system).



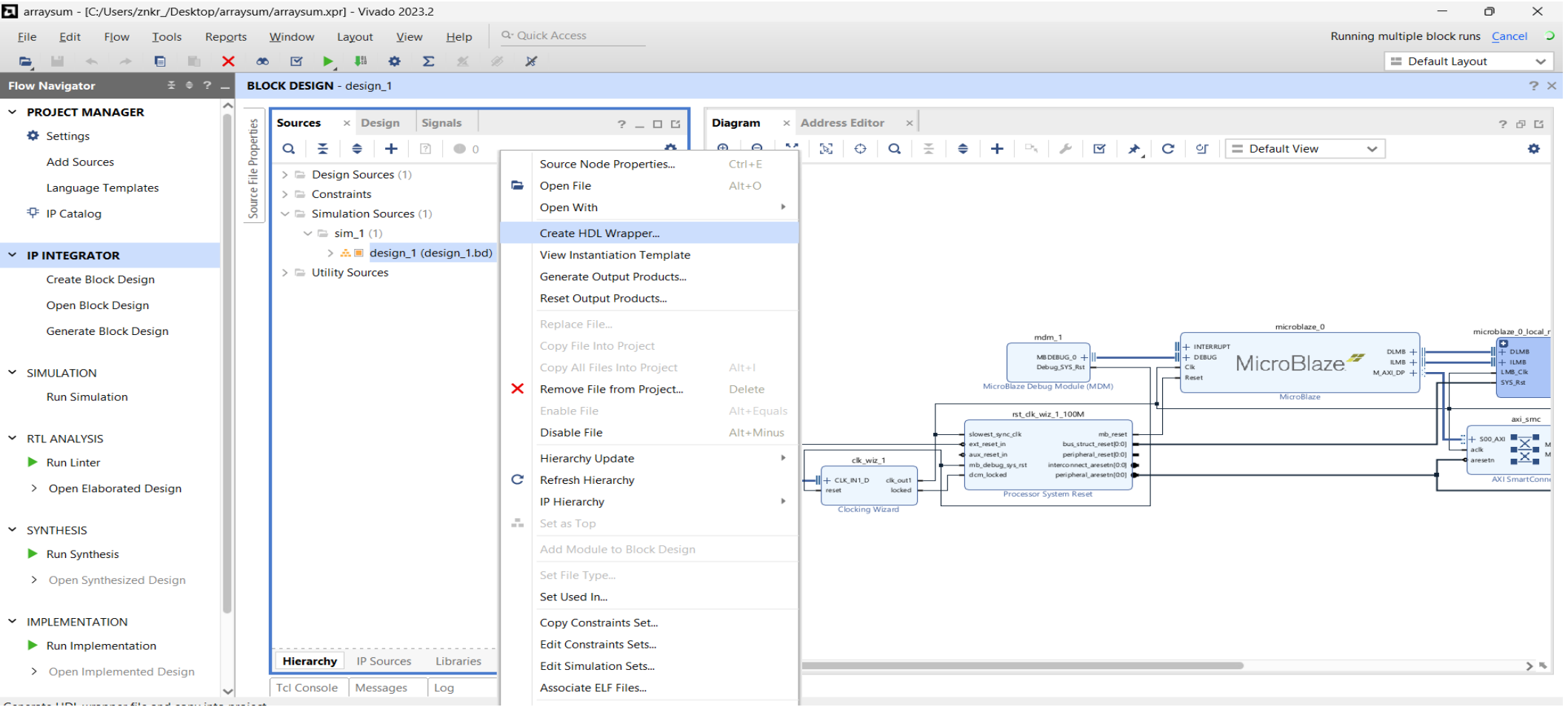
Click : OK. ***<u>If the</u> synthesis <u>and</u> implementation <u>are</u> successful.***

<u>(This window will open).</u>



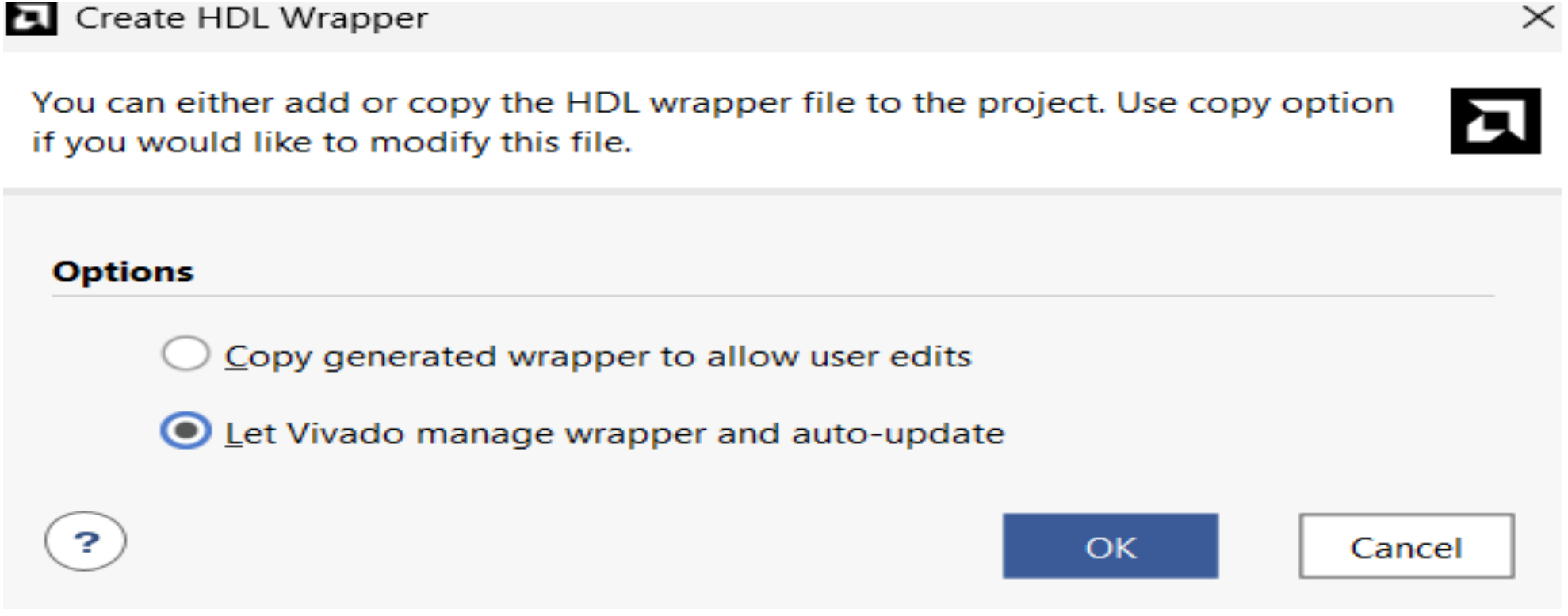
Click : Create HDL Wrapper…

(Manages the design within the FPGA).



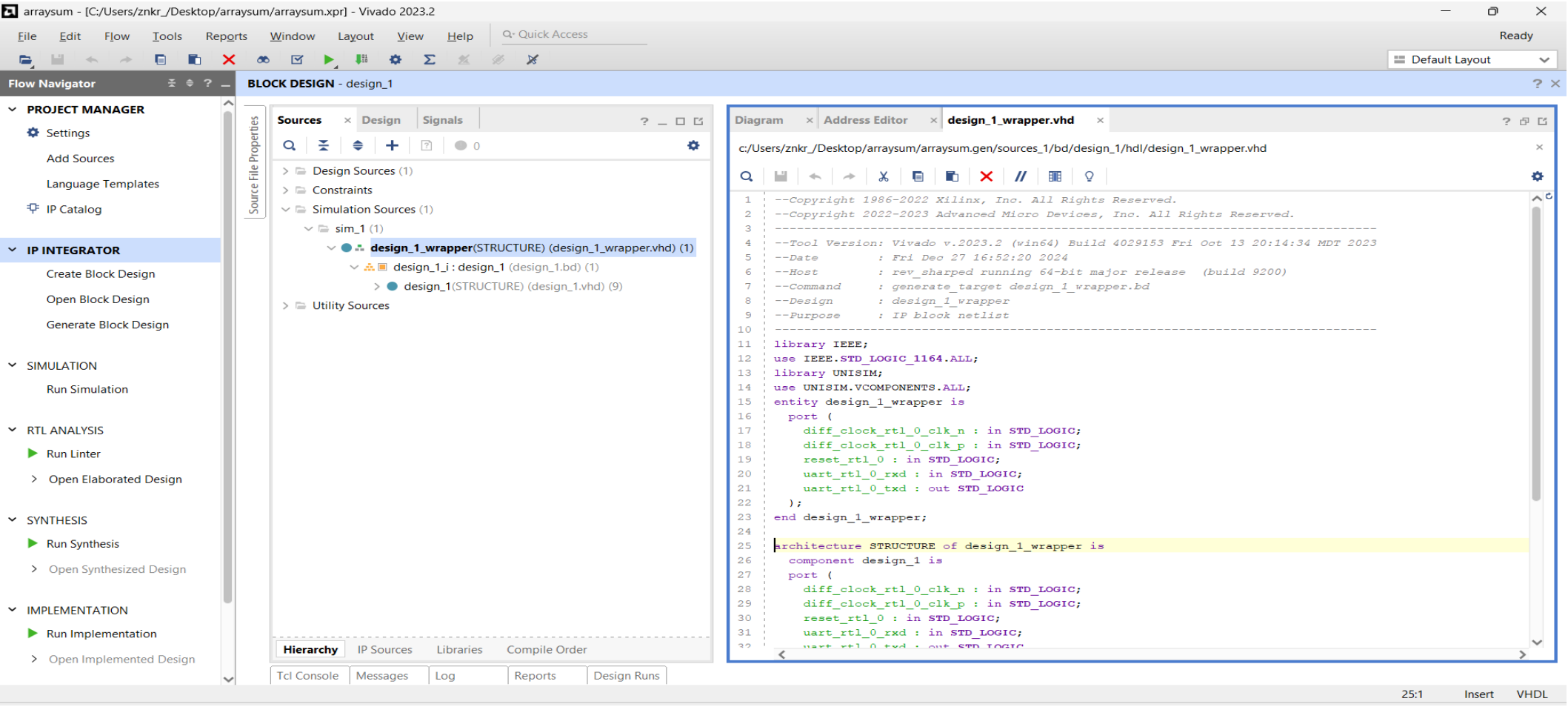
Click : OK.

(Let Vivado manage wrapper and auto-update).



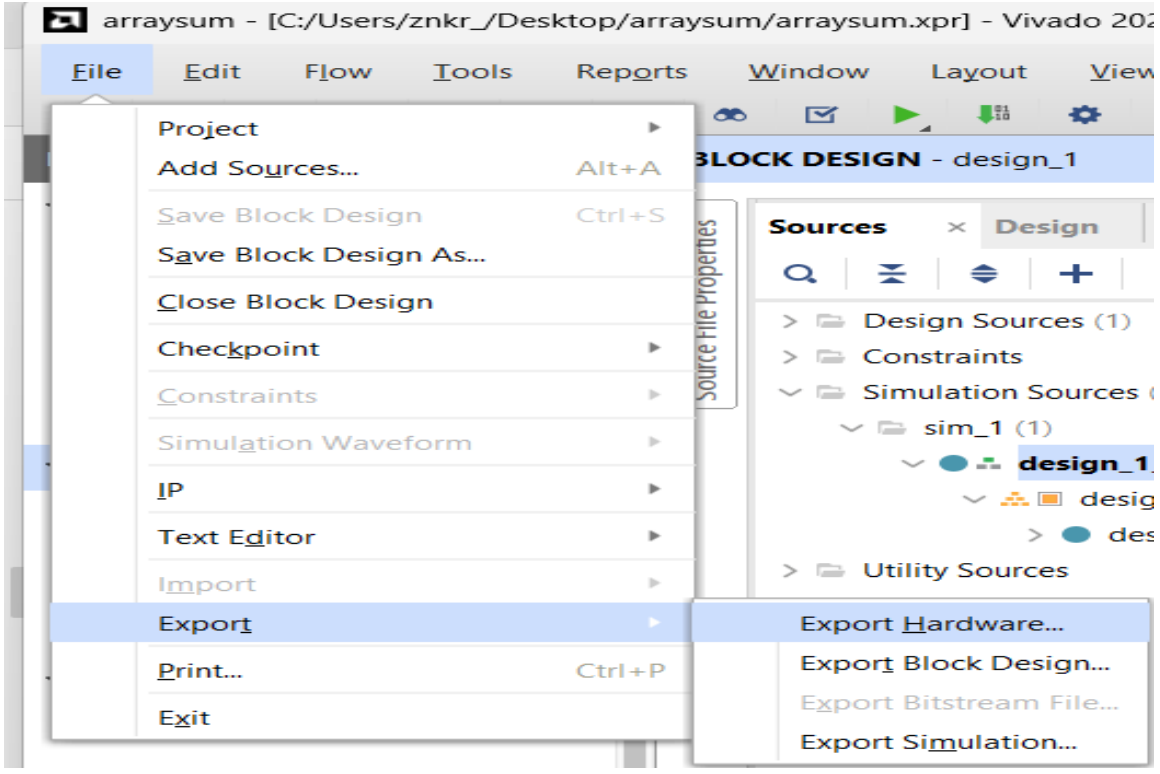
HDL Wrapper Generated.

(With name : design_1_wrapper.vhd).



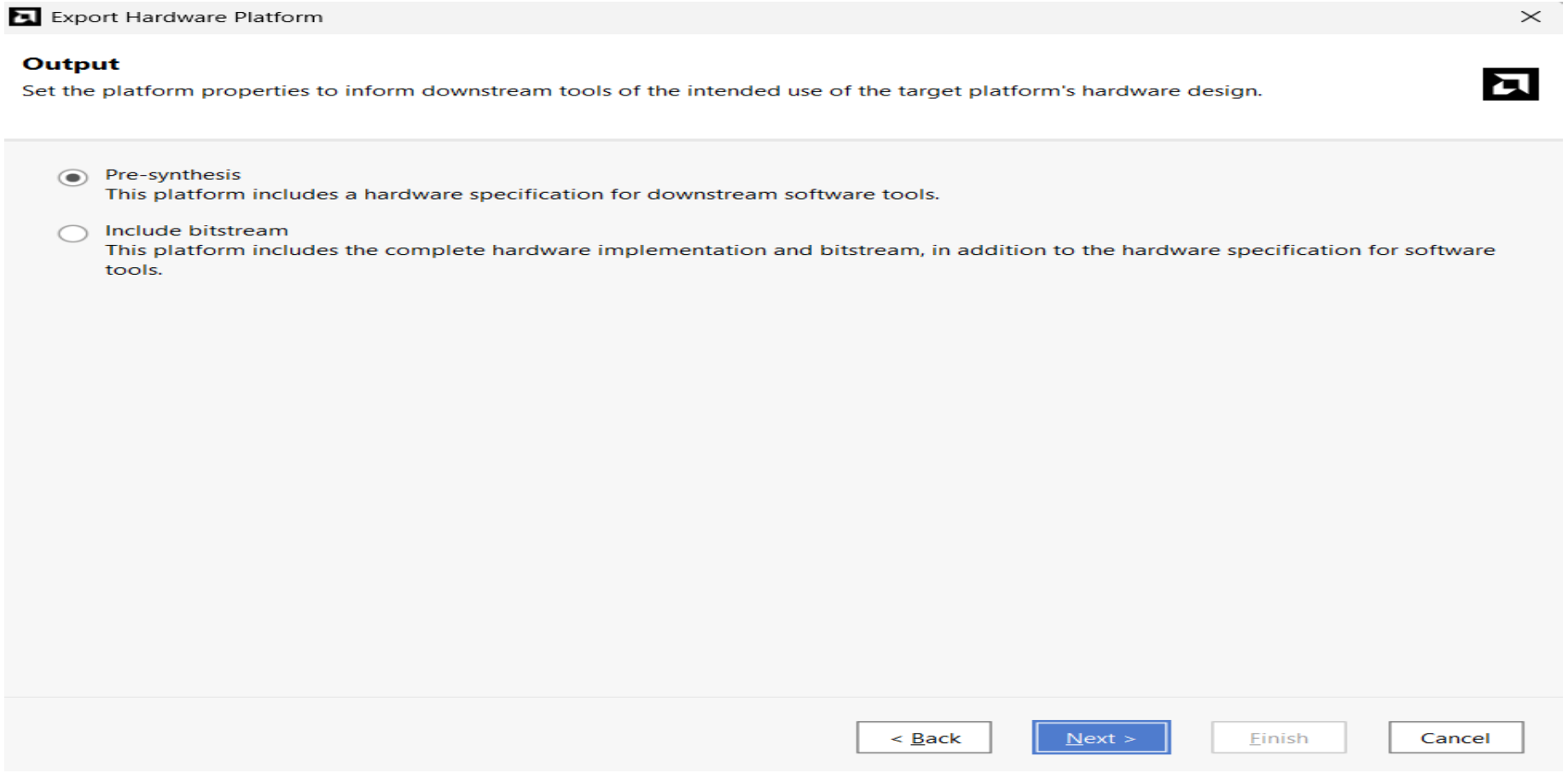
Click : Export Hardware…

(implementing the design on an FPGA).



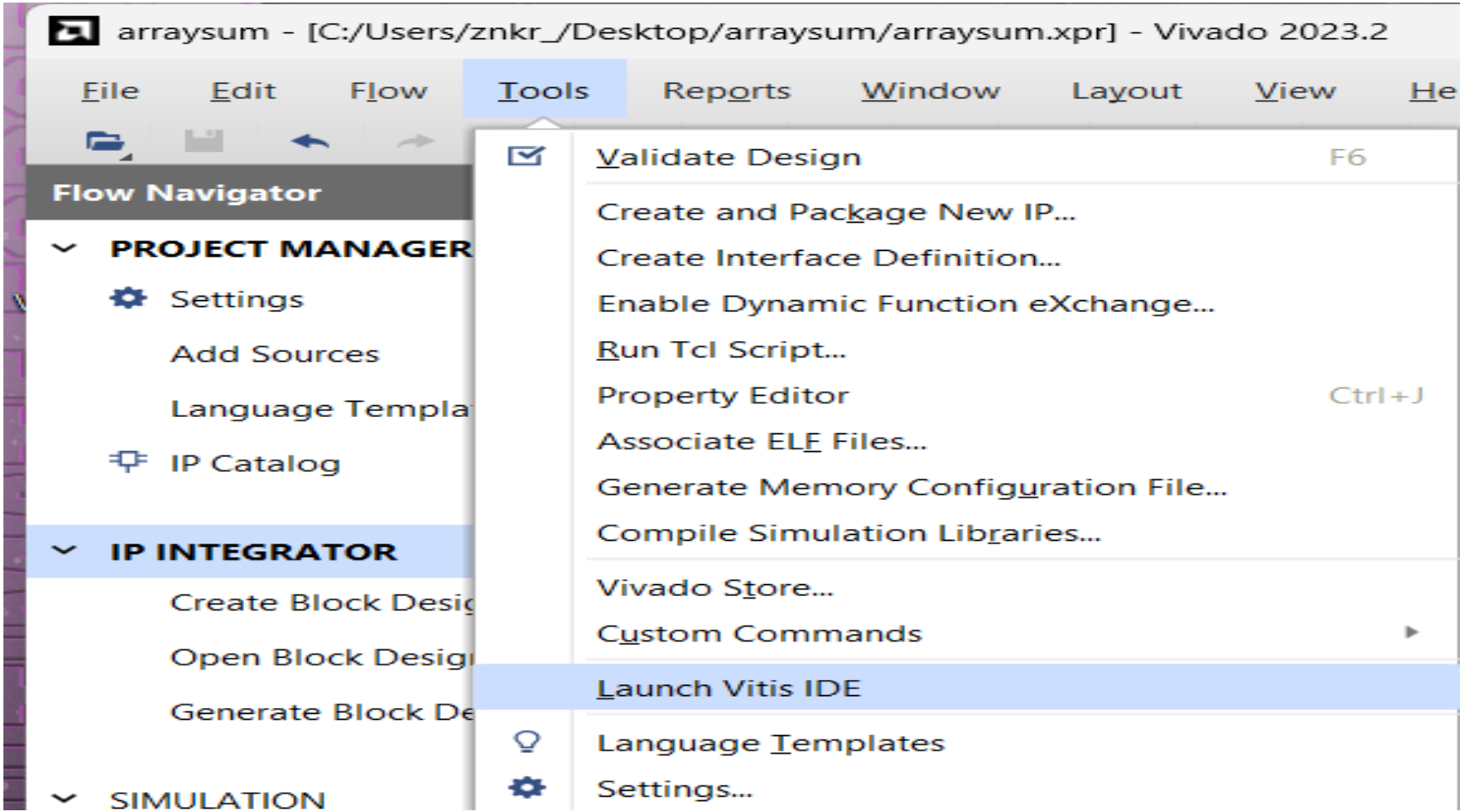
Click : Next > .

(pre-synthesis ensures that the hardware is ready for software development).



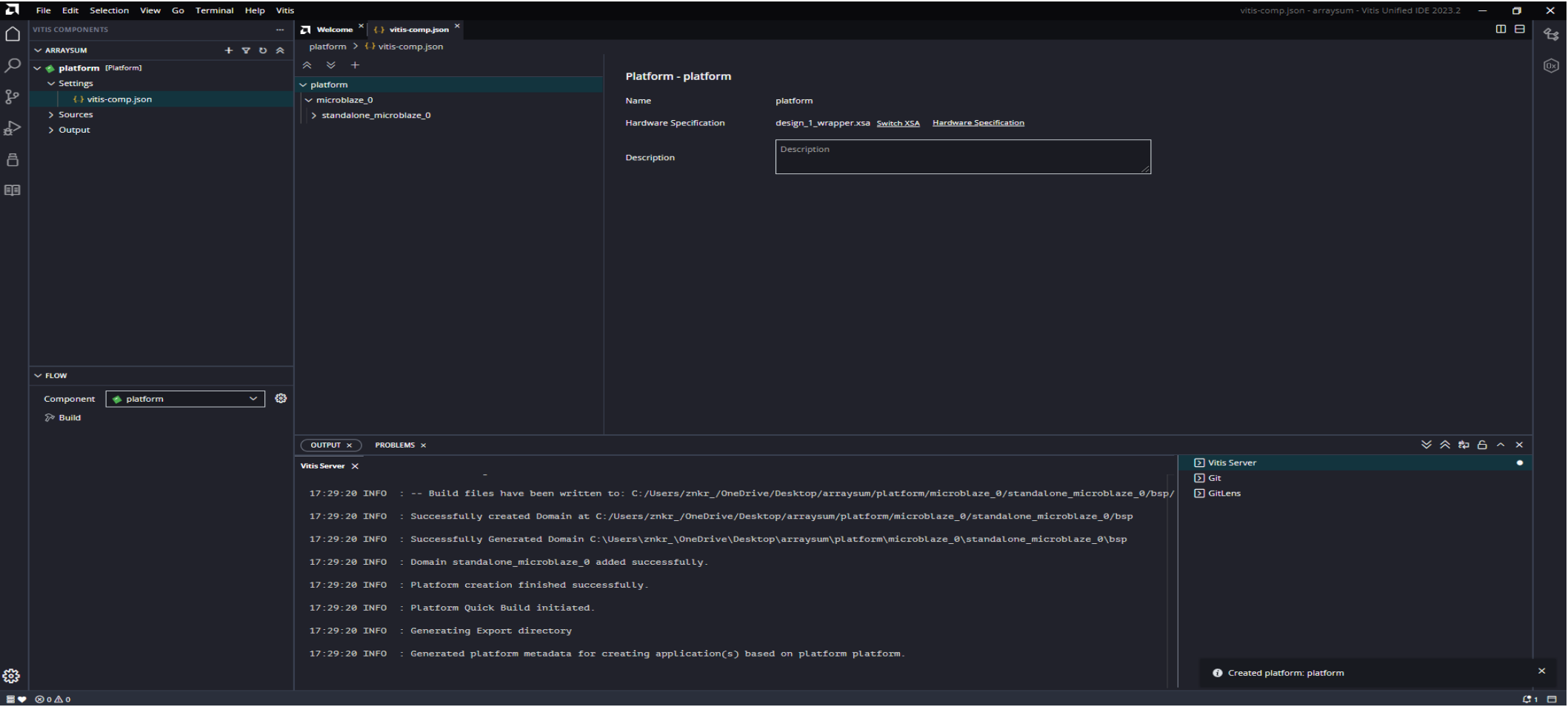
Click : Launch Vitis IDE .

(Make sure .xsa file is exported successfully).



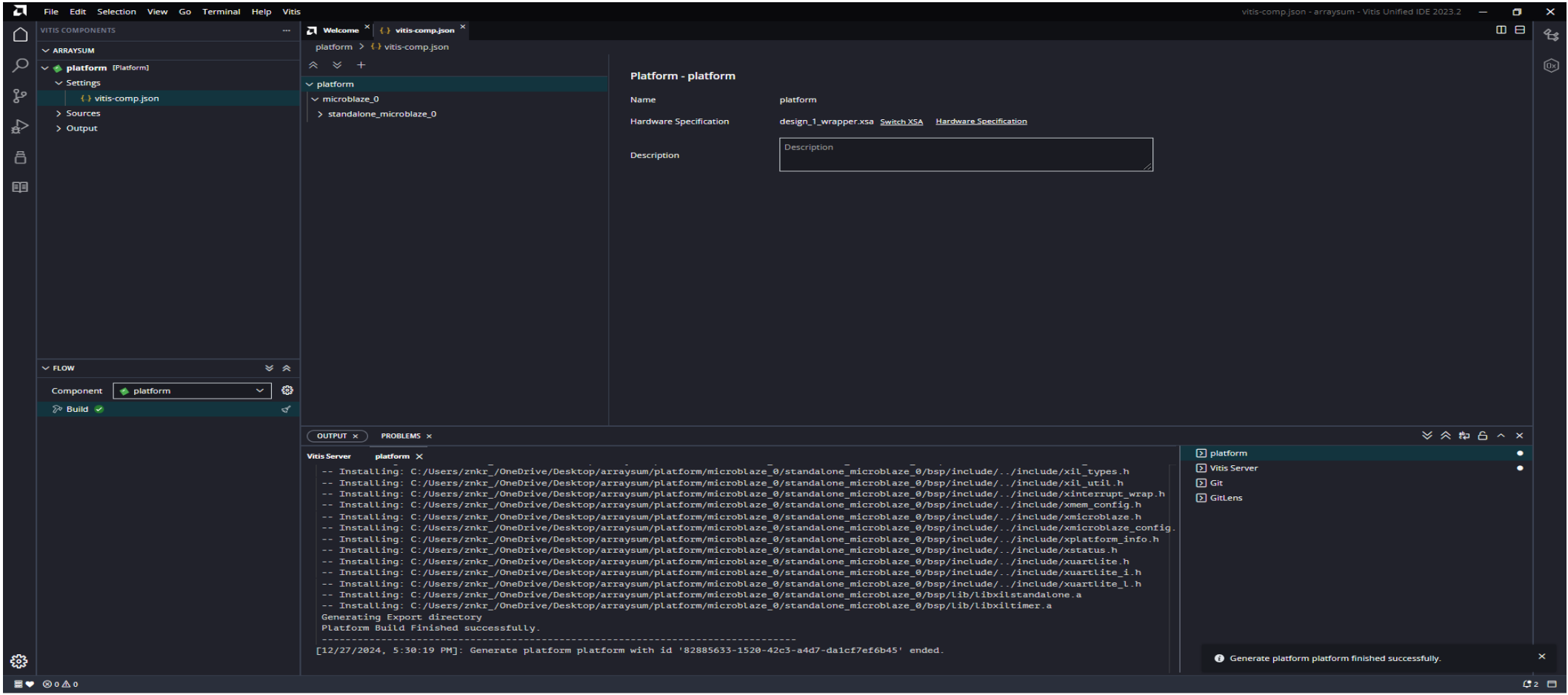
Open the project workspace *and* create a platform.

(Platform : based on the .xsa file).



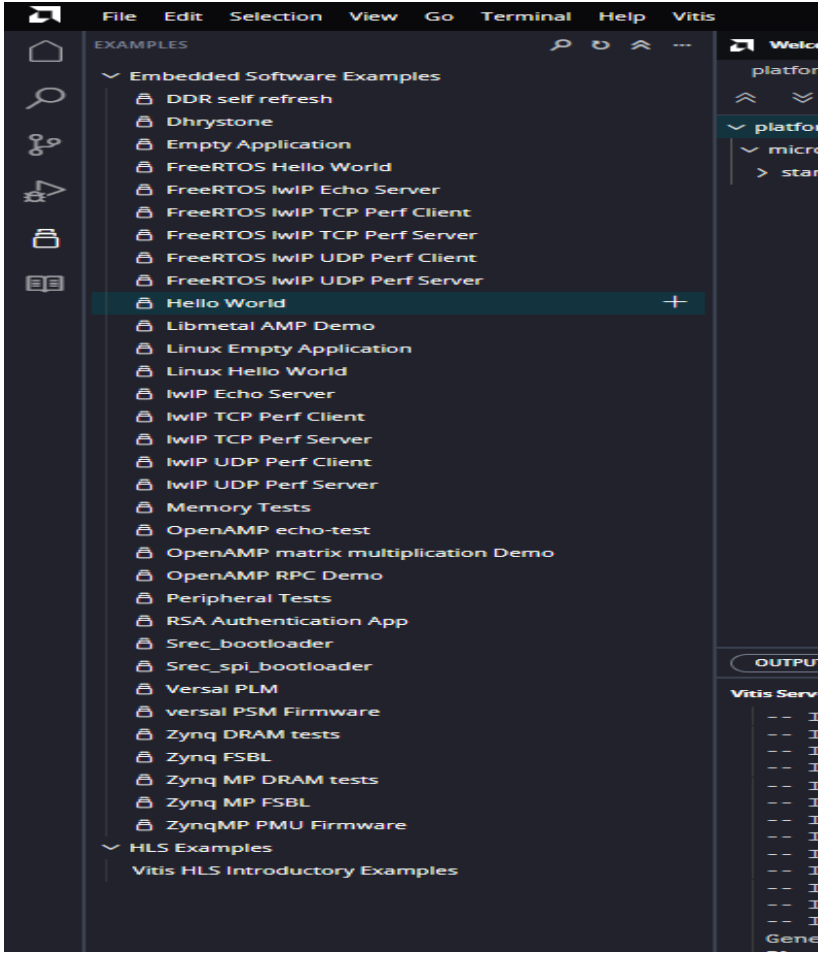
Click : Build.

(Build : To build the platform).



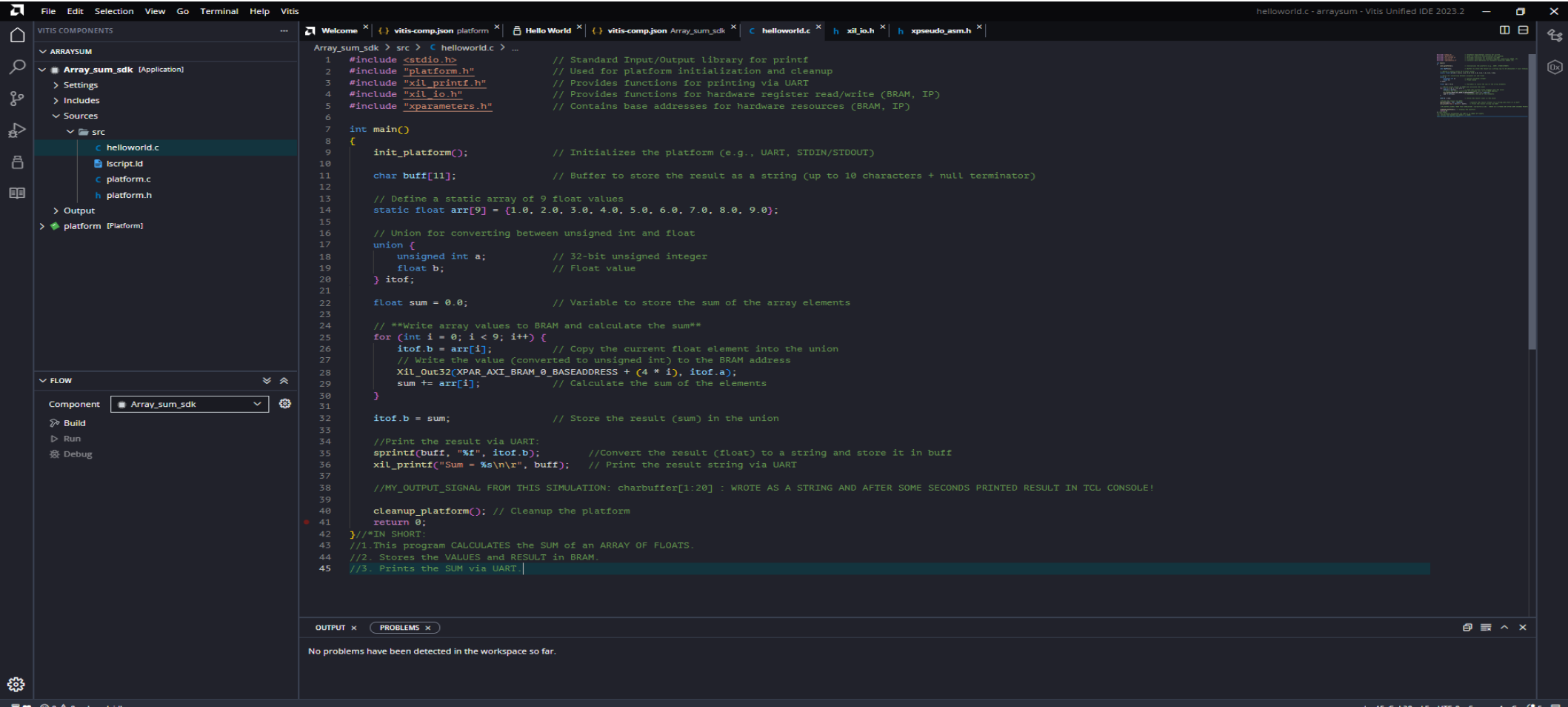
Select : Hello World Application From Templates.

(Renamed: Array_sum_sdk).



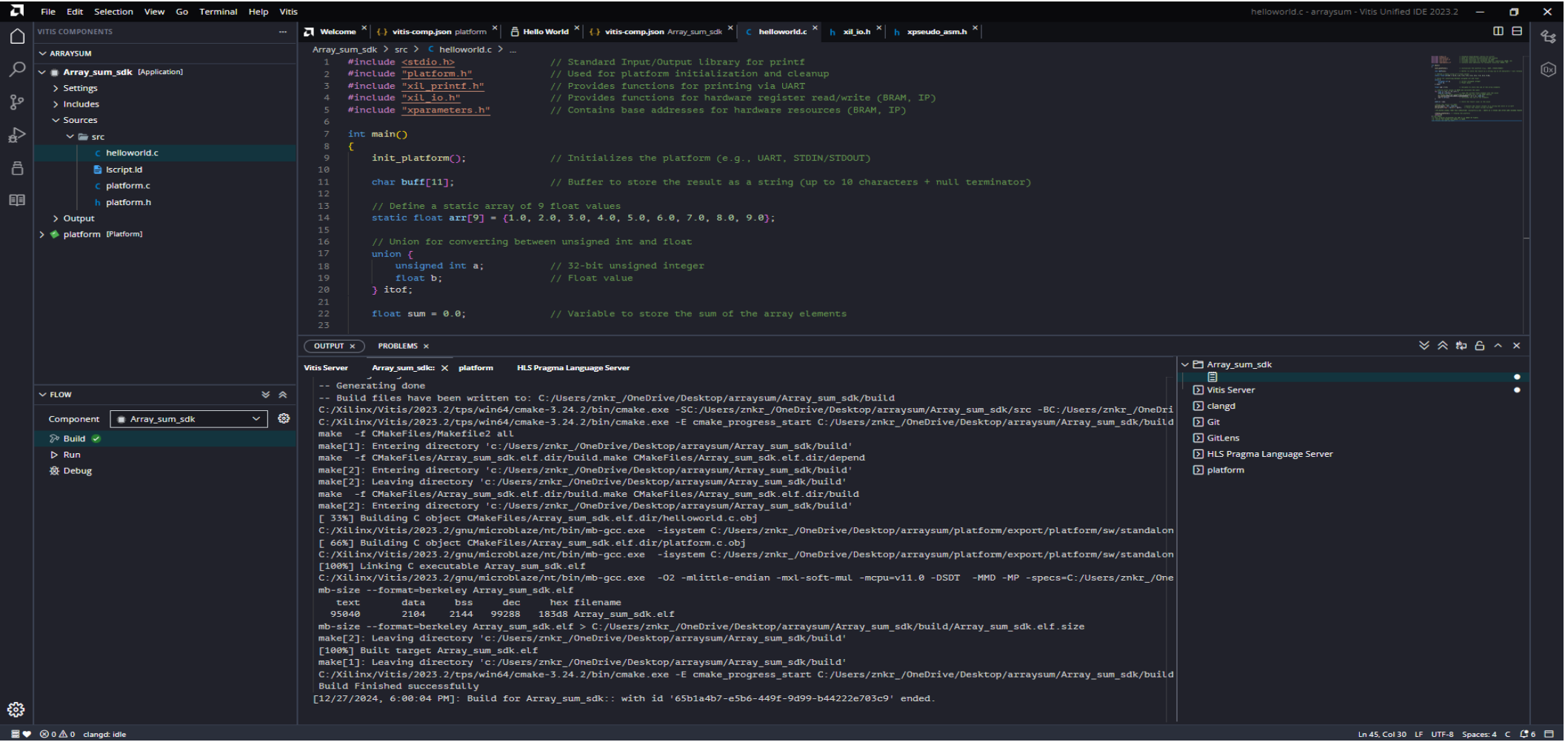
Program : helloworld.c

(Calculates the sum of an array of nine floats, stores the result in BRAM and prints the sum via UART).



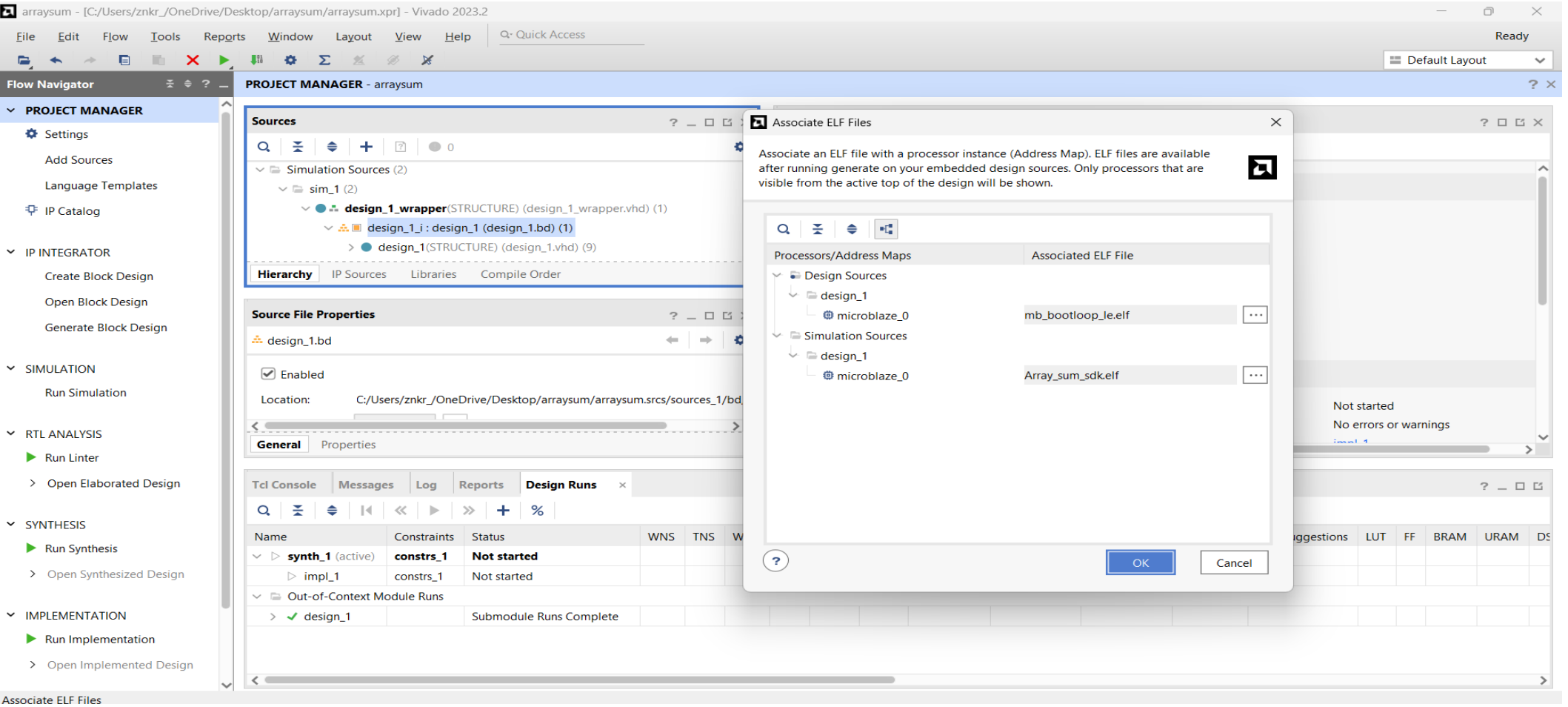
Click : Build.

(Build : To build the Application and generate ELF File).

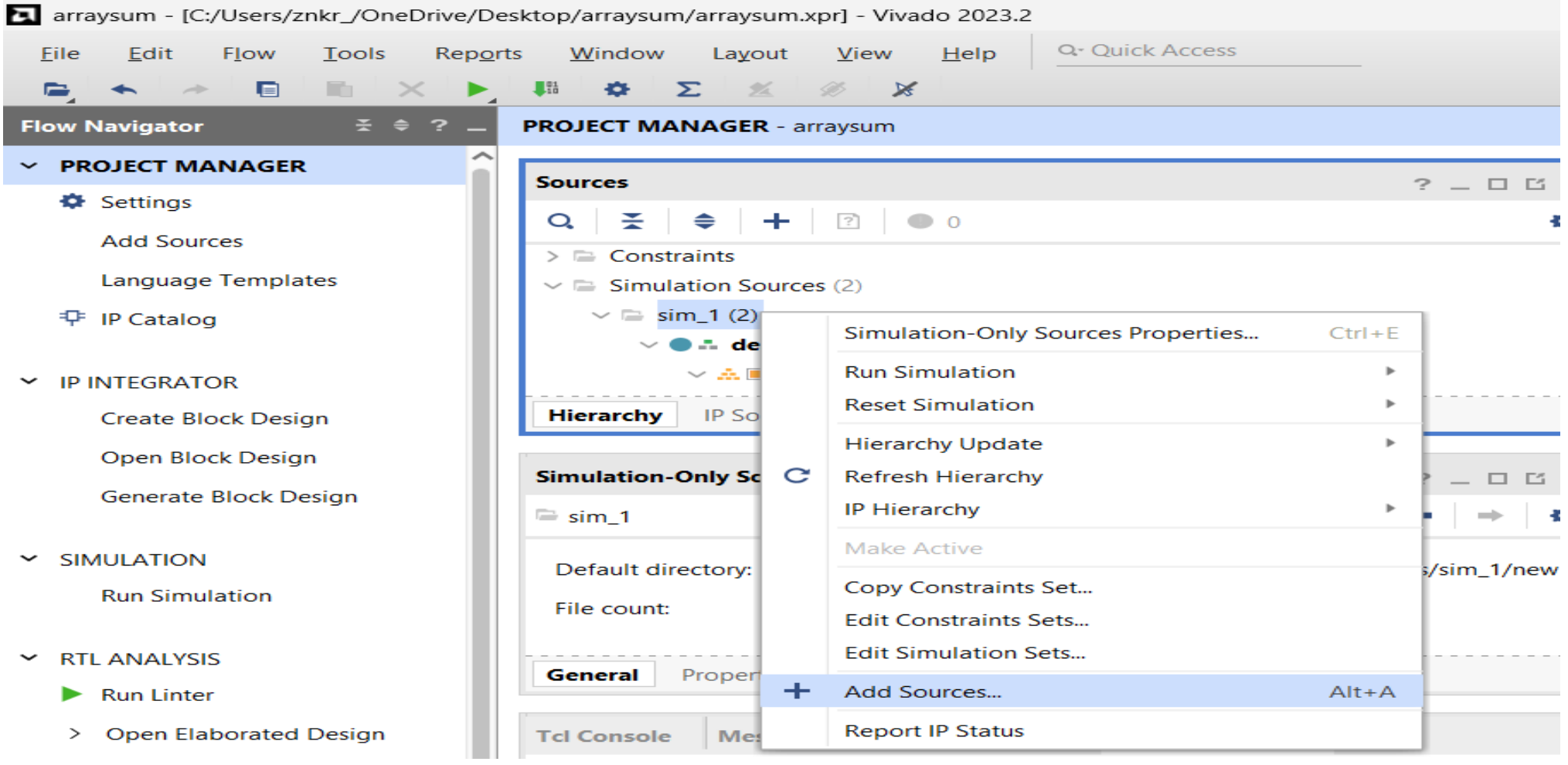


Right-click : Block design and associate ELF files.

(Apply the ELF file from SDK only to the simulation sources).

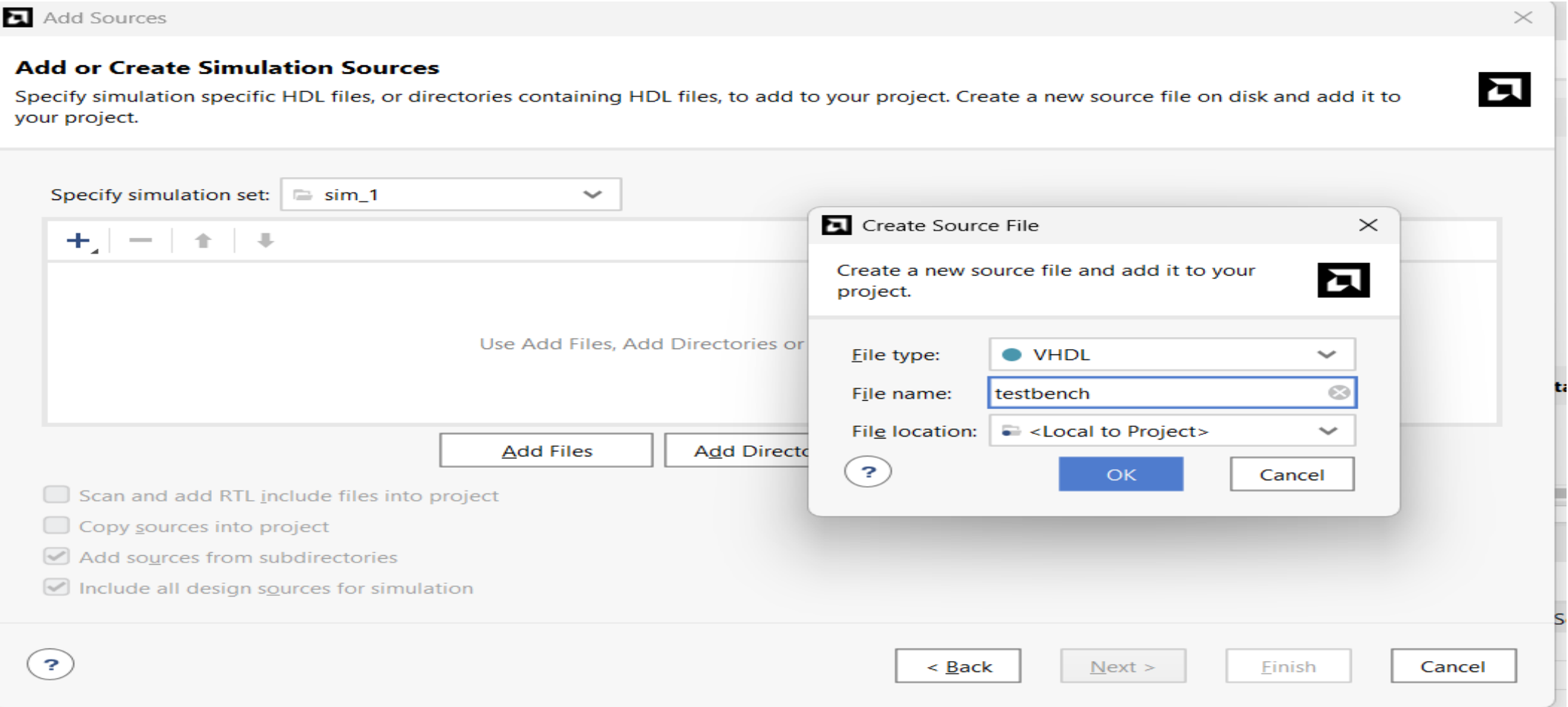


Right-click : Simulation Sources and Add Sources…



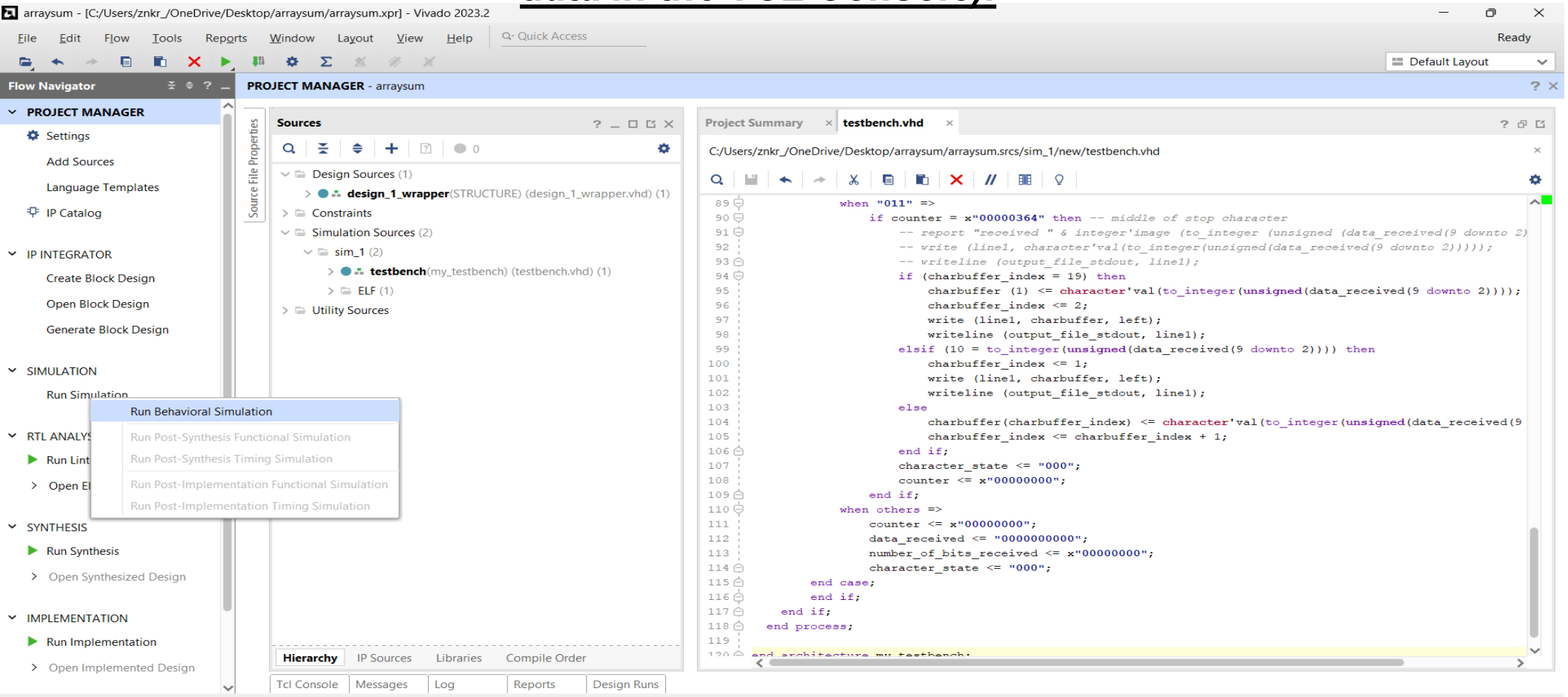
Click : Create Source File.

(File name : testbench).



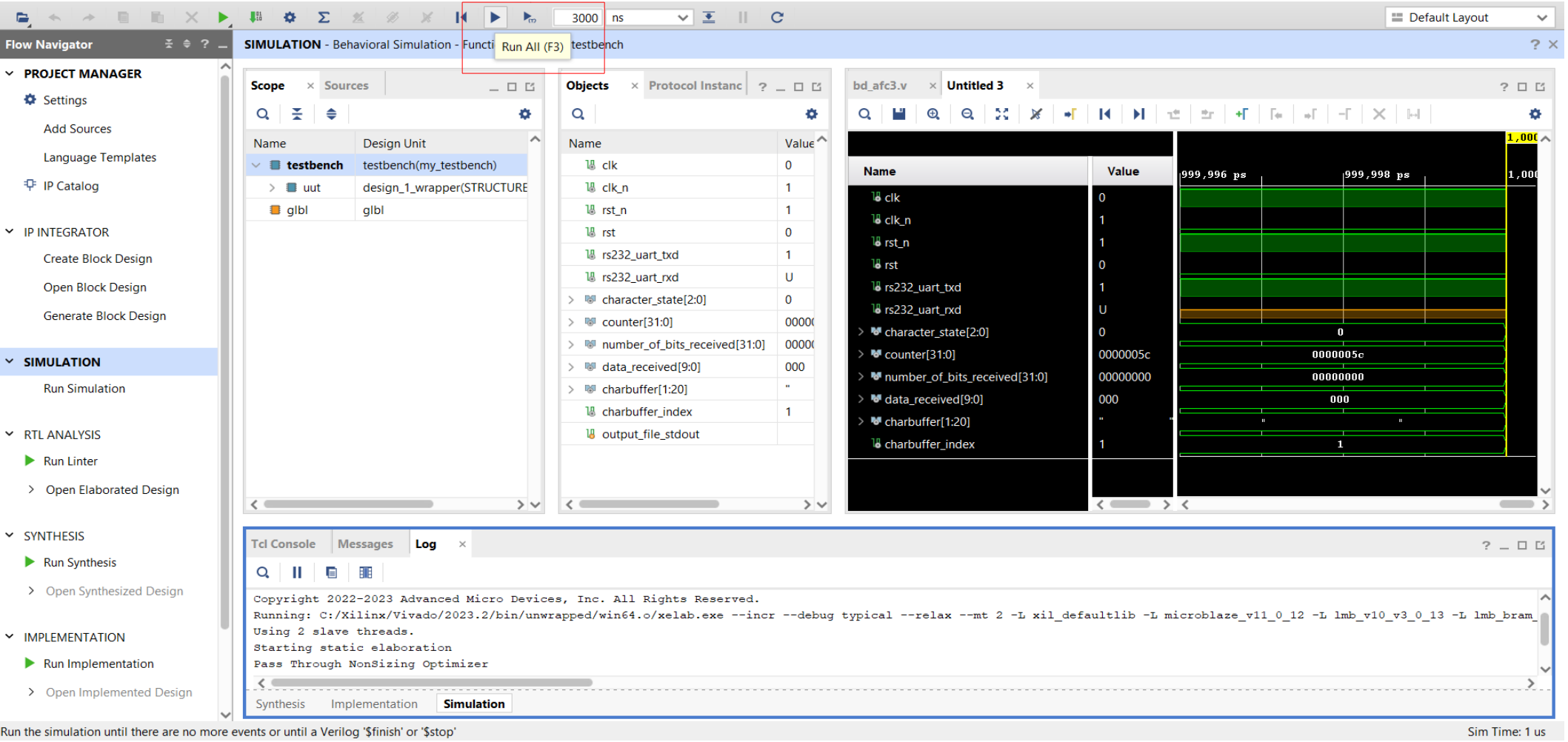
Click : Run Behavioral Simulation.

(The testbench contains VHDL code for UART communication and displaying data in the TCL Console).



Click : Run All (F3).

(Simulation launched).

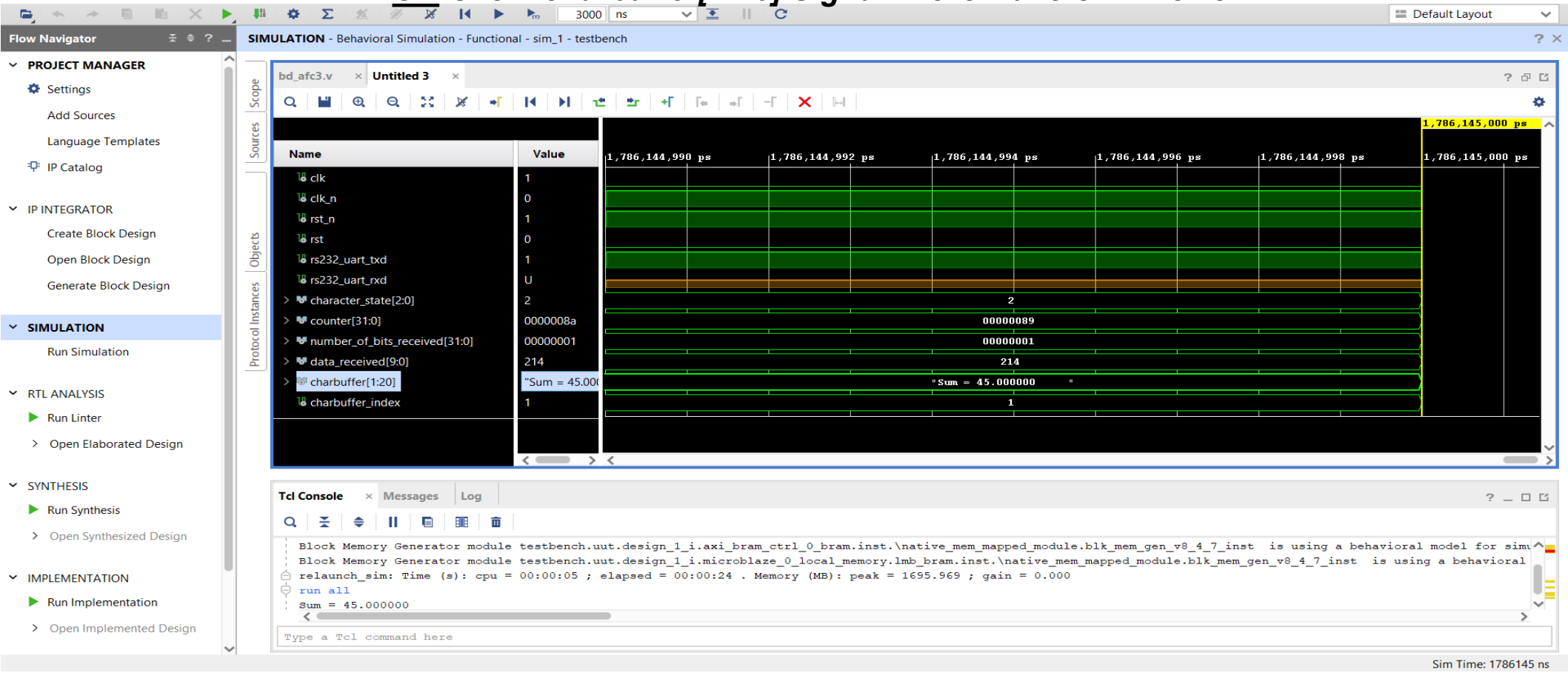


It may take some time; please be patient.

Click : TCL Console to see the result.

(Sum = 45.000000).

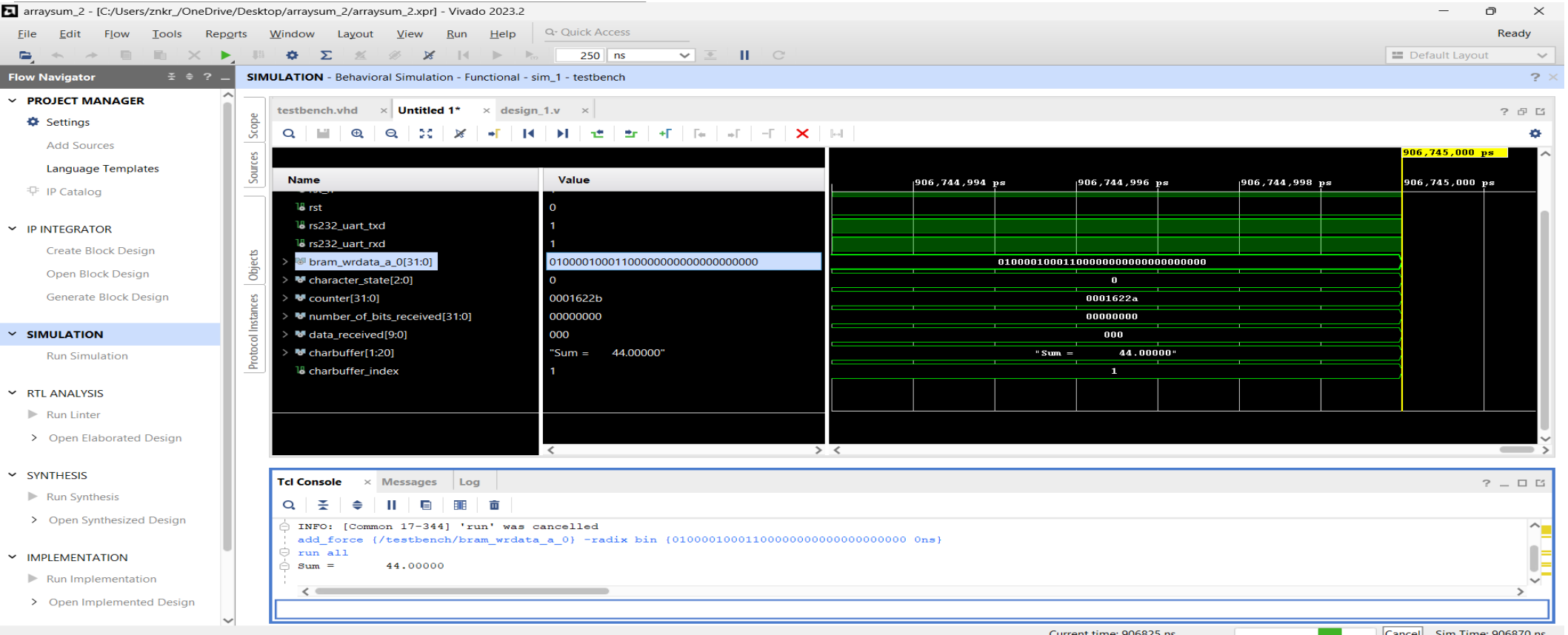
OR Click: charbuffer[1:20] Signal in the waveform viewer.



Example #1 **Input Bit Flips Without TinyAES.**

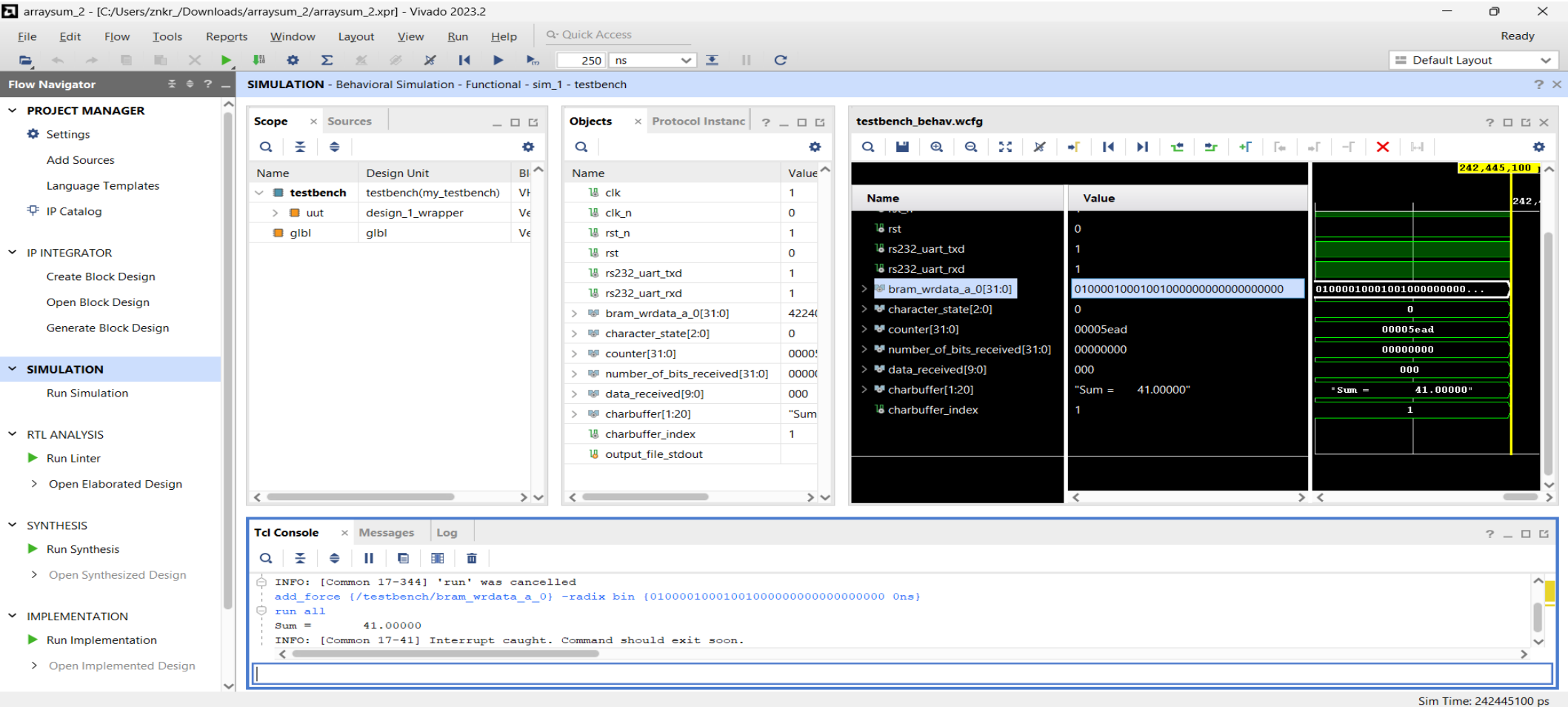
Forcing Different Bit from the original pattern
And getting different result.

Result: 44 Instead of 45.



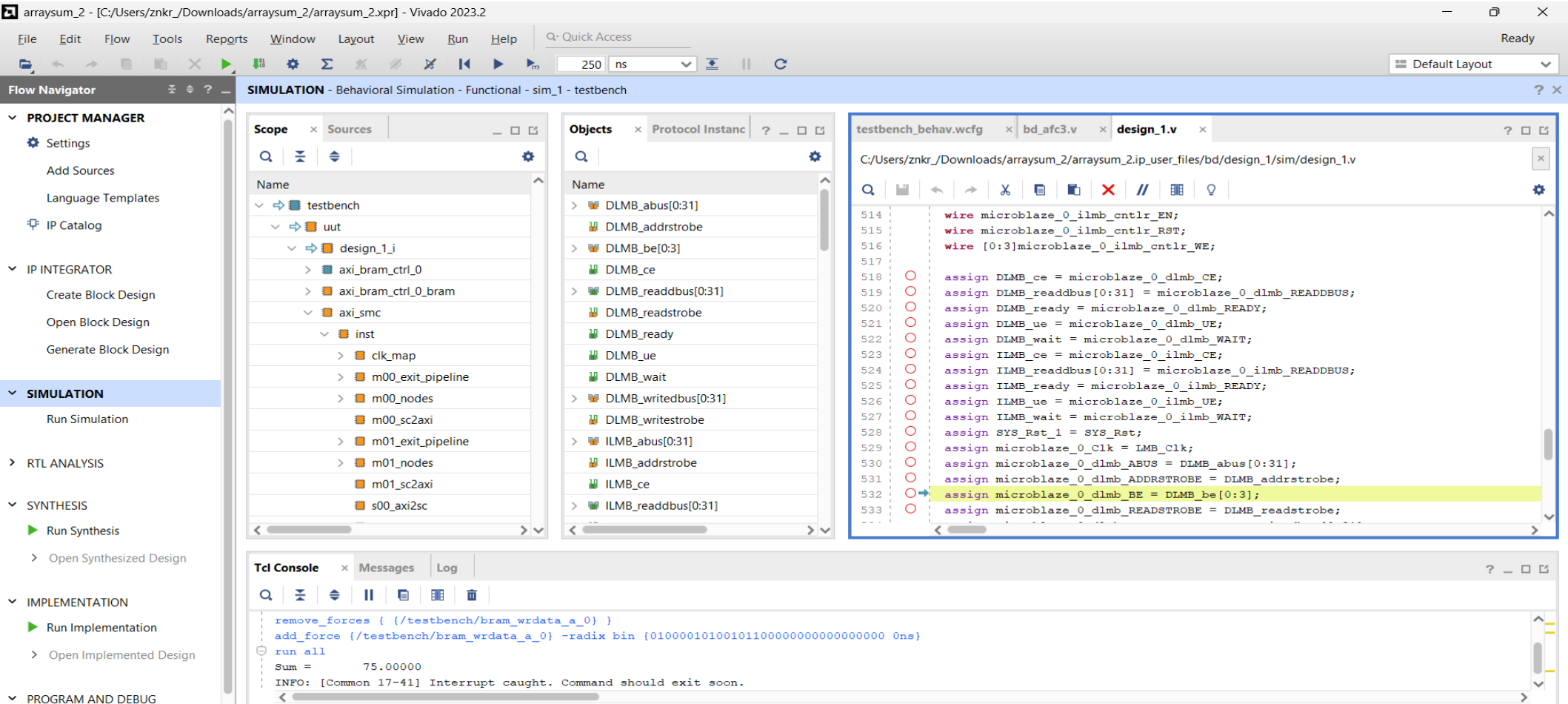
Example 2#

Result 41 Instead of 45.



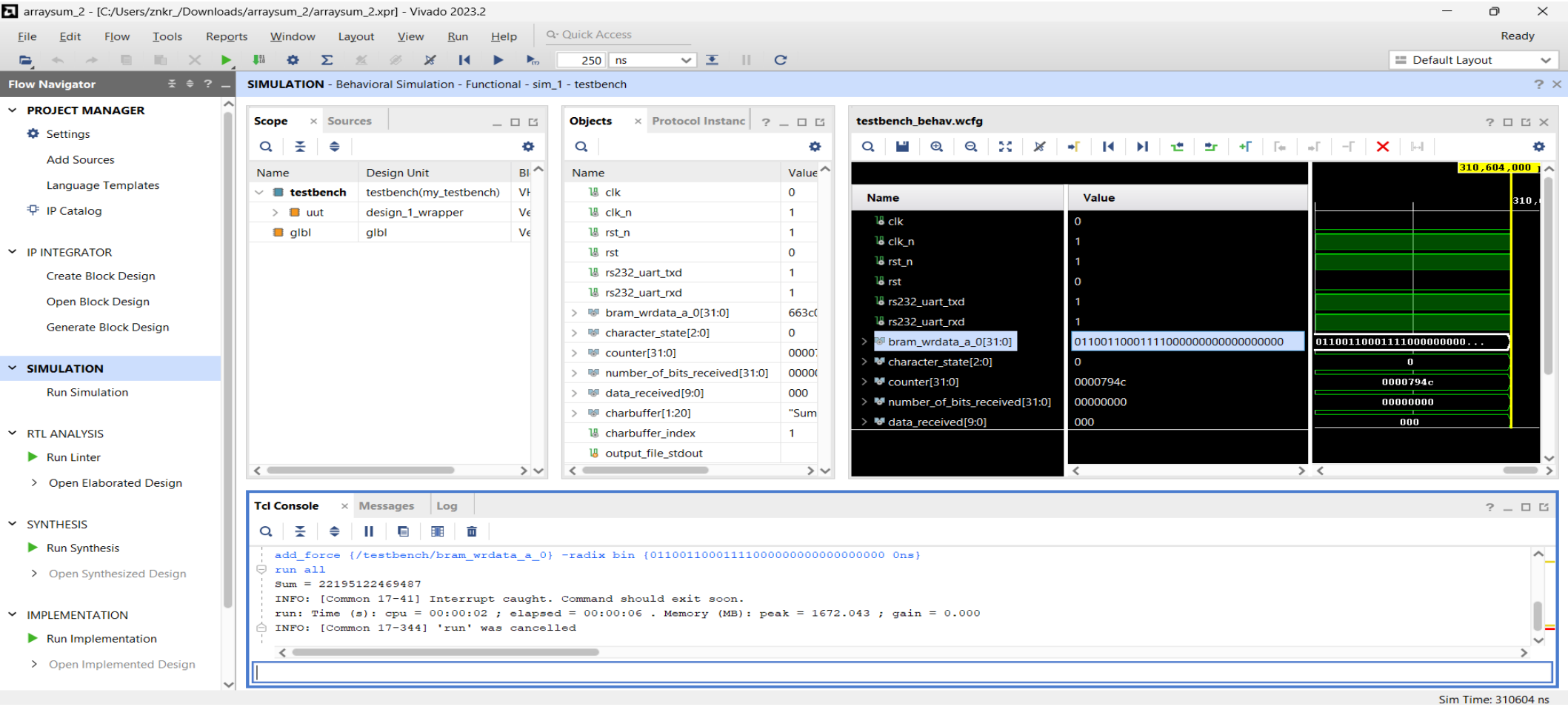
Example #3

Result: 75 Instead of 45



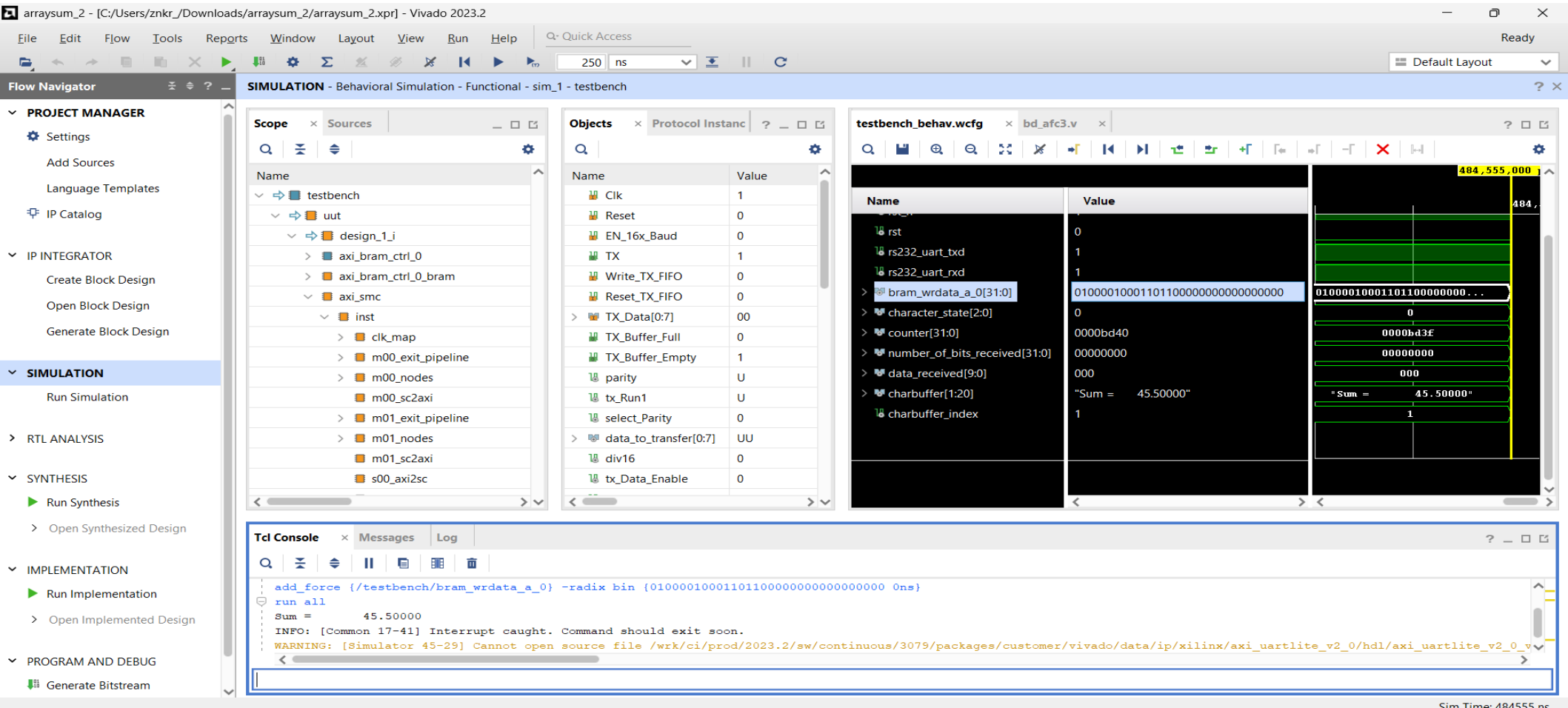
Example 4#:

Result: Significantly Different from 45.



Example 5#:

Result: 45.50 instead of 45.00 (very close to the original).



Thank you for your participation.