INDRAPRASTHA INSTITUTE of INFORMATION TECHNOLOGY DELHI

Department
of
Electronics & Communication Engineering

Embedded Logic Design(ECE270)

Dr. Sumit J Darak

Lab_6: Design and implement a ROM using Block Memory Generator. Design and implement a FIFO (common block RAM) capable of storing 16 numbers of 4 bits each.

Mohammad Shariq
2020220

08-11-2021

OBJECTIVE:

- Design and implement a ROM using Block Memory Generator. Create a coe file to store the 10 numbers of 4 bits each and find the maximum amongst them.
- Design and implement a FIFO (common block RAM) capable of storing 16 numbers of 4 bits each. Provide the input data using a 4-bit input port. Use two separate push buttons to read and write. Also, see the working of empty, full,almost_empty,almost_full, and data_count signals.

Theory:

- **BROM :-**

The FPGA fabric includes embedded memory elements that can be used as random-access memory (RAM), read-only memory (ROM), or shift registers. These elements are block RAMs (BRAMs), LUTs, and shift registers.

The BRAM is a dual-port RAM module instantiated into the FPGA fabric to provide on-chip storage for a relatively large set of data. The two types of BRAM memories available in a device can hold either 18k or 36k bits, and the available amount of these memories is device specific. The dual-port nature of these memories allows for parallel, same-clock-cycle access to different locations.

In OpenCL code, BRAMs can implement either a RAM or a ROM, covering on-chip, local, and private memory types. In a RAM configuration, the data can be read and written at any time during the runtime of the circuit. In contrast, in a ROM configuration, data can only be read during the runtime of the circuit. The data of the ROM is written as part of the FPGA configuration and cannot be modified in any way.

- **FIFO MEMORY:-**

The acronym FIFO stands for **F**irst **I**n **F**irst **O**ut. FIFOs are used everywhere in FPGA and ASIC designs, they are one of the basic building blocks. And they are very handy! FIFOs can be used for any of these purposes:

A FIFO can be thought of a one-way tunnel that cars can drive through. At the end of the tunnel is a toll with a gate. Once the gate opens, the car can leave the tunnel. If that gate never opens and more cars keep entering the tunnel, eventually the tunnel will fill up with cars. This is called FIFO Overflow and in general it's not a good thing. How deep the FIFO is can be thought of as the length of the tunnel. The deeper the FIFO, the more data can fit into it before it overflows. FIFOs also have a width, which represents the width of the data (in number of bits) that enters the FIFO. Below is an image of the basic interface of any FIFO. These signals will always be found when you look at any FIFO. Often there are more signals that add additional features, such as a count of the number of words in the FIFO. See the figure below:

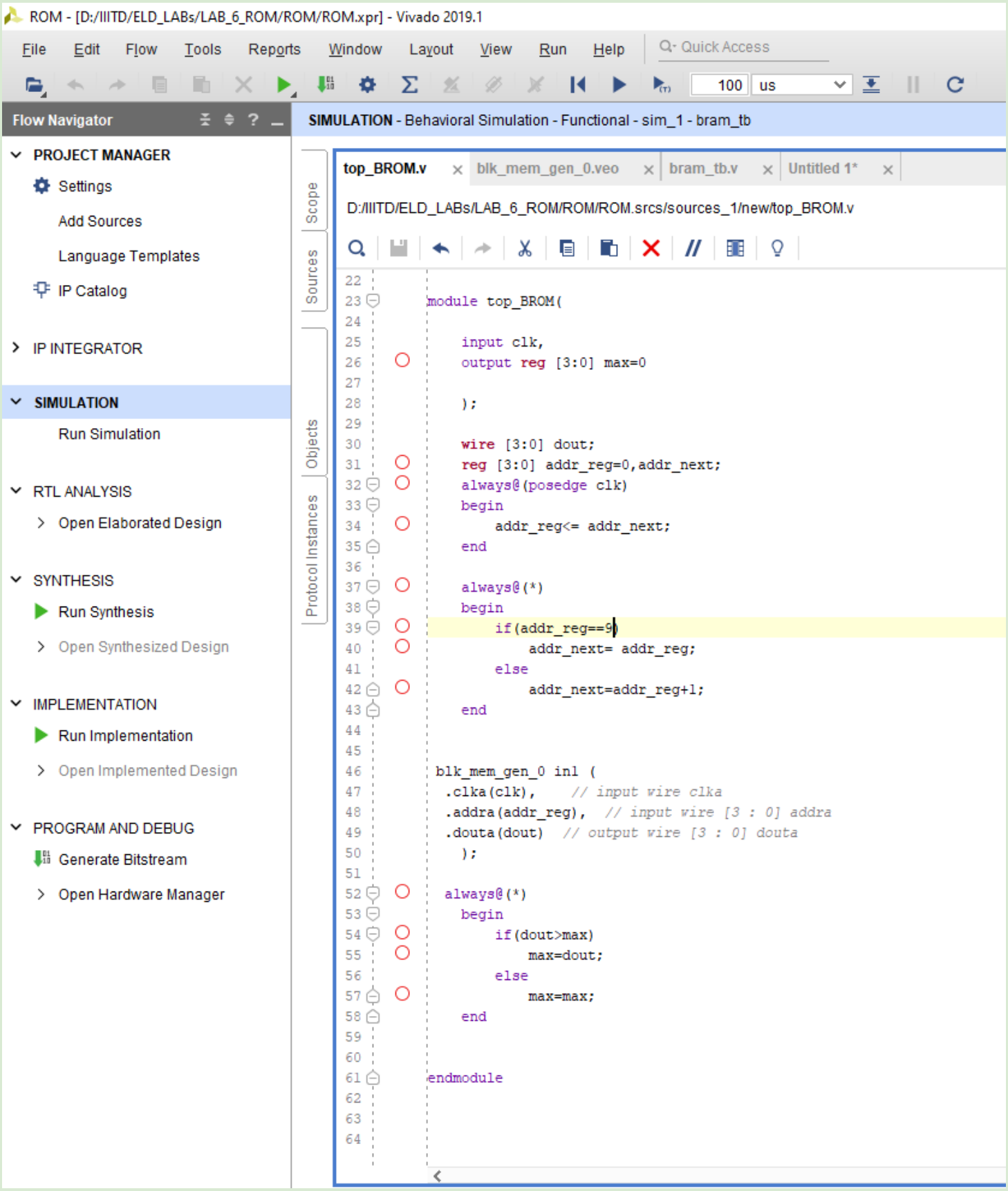
Basic Interface Signals of a FIFO

The FIFO can be divided up into the write half and the read half. The write half has the signals Write Enable, Write Data, and FIFO Full. The designer should **never write to a full FIFO!** Always check the FIFO Full flag to make sure there's room to write another piece of data, otherwise you will lose that data.

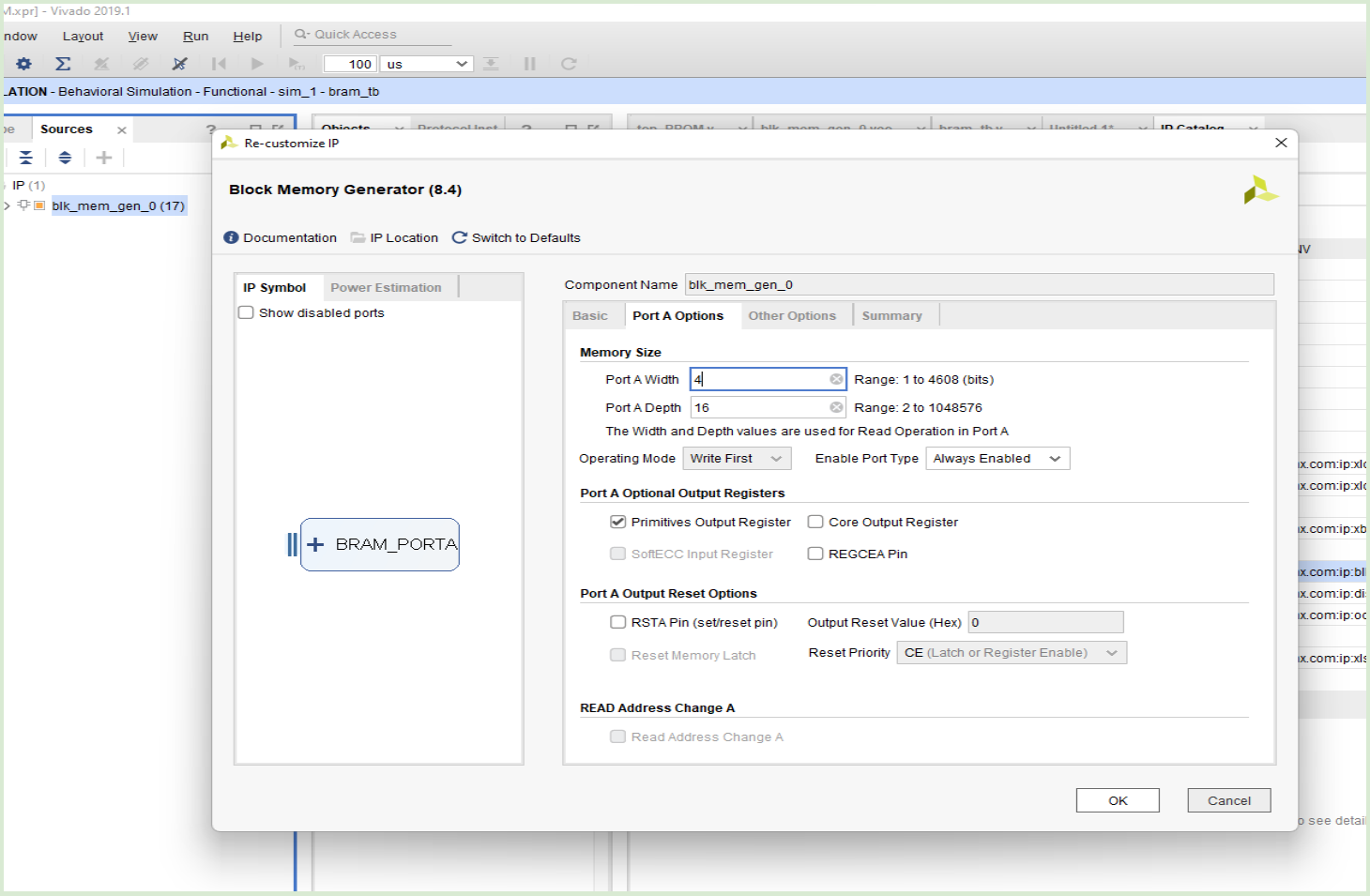
The read half has the signals Read Enable, Read Data, and FIFO Empty. I find it easier when designing code to separate the write-code in one file and the read-code in another file, just to be careful. The designer should **never read from an empty FIFO!** As long as you obey these two basic rules you and FIFOs will get along nicely. I'll restate them again because they're just that important.

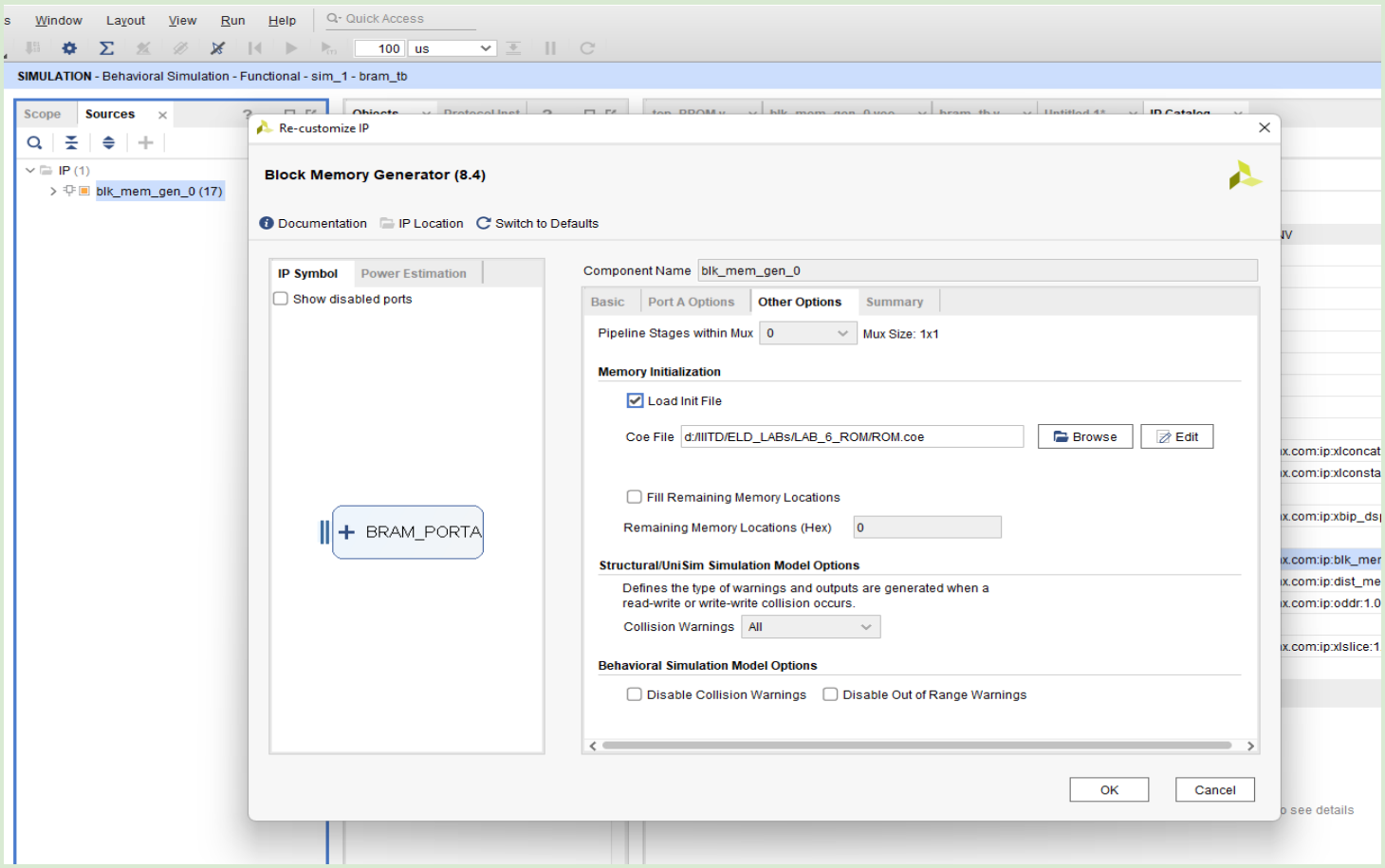
Observations:

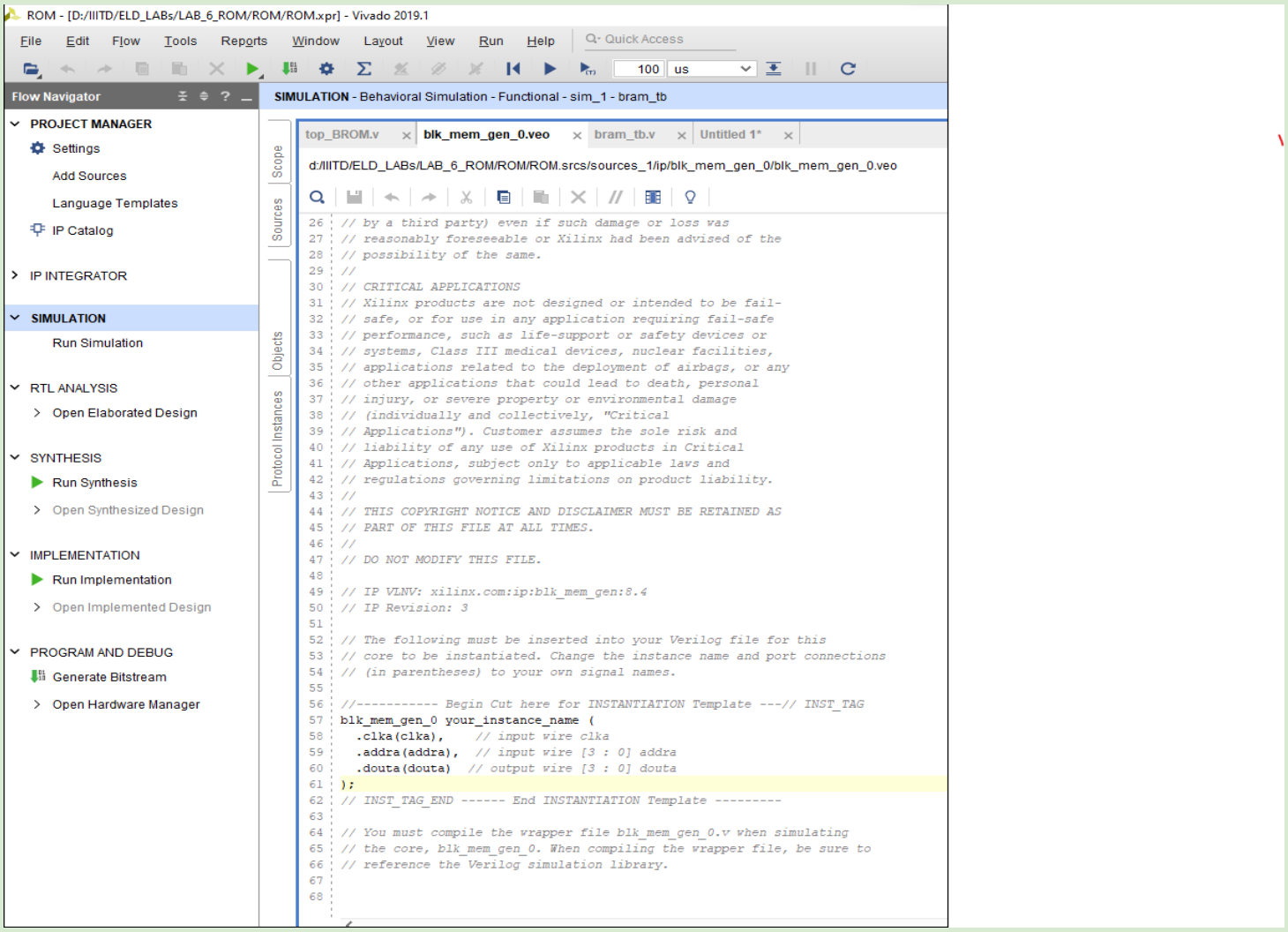
programme for the functionality of BROM(Block ROM):

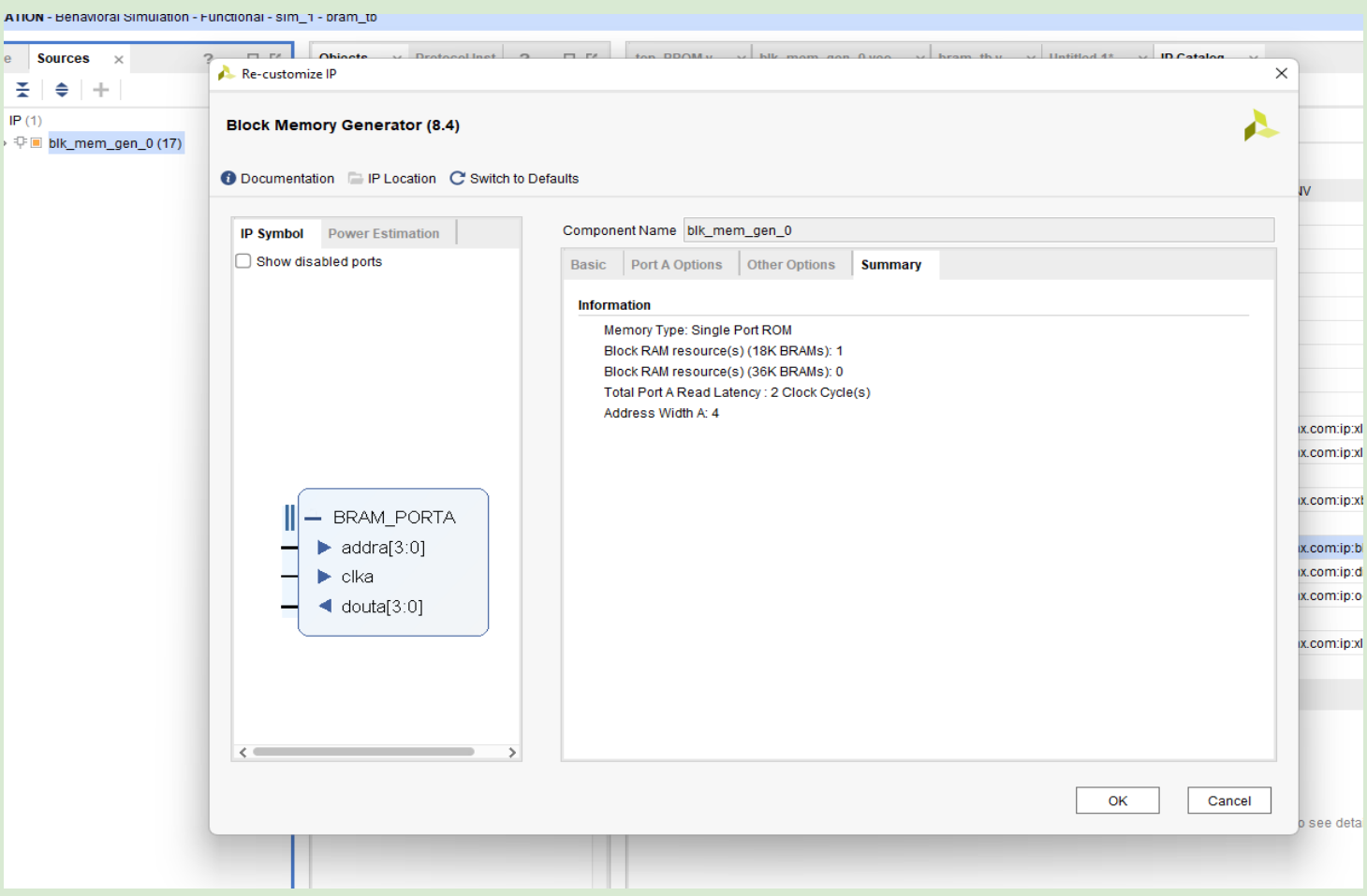


Block Memory Generator IP for BROM:






ROM - [D:/IIITD/ELD_LABs/LAB_6_ROM/ROM/ROM.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Run Help
Quick Access
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
SIMULATION - Behavioral Simulation - Functional - sim_1 - bram_tb
top_BROM.v
blk_mem_gen_0.veo
bram_tb.v
Untitled 1*
d:/IIITD/ELD_LABs/LAB_6_ROM/ROM/ROM.srcs/sources_1/ip/blk_mem_gen_0/blk_mem_gen_0.veo
// by a third party) even if such damage or loss was
// reasonably foreseeable or Xilinx had been advised of the
// possibility of the same.
//
// CRITICAL APPLICATIONS
// Xilinx products are not designed or intended to be fail-
// safe, or for use in any application requiring fail-safe
// performance, such as life-support or safety devices or
// systems, Class III medical devices, nuclear facilities,
// applications related to the deployment of airbags, or any
// other applications that could lead to death, personal
// injury, or severe property or environmental damage
// (individually and collectively, "Critical
// Applications"). Customer assumes the sole risk and
// liability of any use of Xilinx products in Critical
// Applications, subject only to applicable laws and
// regulations governing limitations on product liability.
//
// THIS COPYRIGHT NOTICE AND DISCLAIMER MUST BE RETAINED AS
// PART OF THIS FILE AT ALL TIMES.
//
// DO NOT MODIFY THIS FILE.
// IP VLNV: xilinx.com:ip:blk_mem_gen:8.4
// IP Revision: 3
// The following must be inserted into your Verilog file for this
// core to be instantiated. Change the instance name and port connections
// (in parentheses) to your own signal names.
//----------- Begin Cut here for INSTANTIATION Template ---// INST_TAG
blk_mem_gen_0 your_instance_name (
.clka(clka), // input wire clka
.addra(addra), // input wire [3 : 0] addra
.douta(douta) // output wire [3 : 0] douta
);
// INST_TAG_END ------ End INSTANTIATION Template ---------
// You must compile the wrapper file blk_mem_gen_0.v when simulating
// the core, blk_mem_gen_0. When compiling the wrapper file, be sure to
// reference the Verilog simulation library.


Sources
IP (1)
blk_mem_gen_0 (17)
Re-customize IP
Block Memory Generator (8.4)
Documentation
IP Location
Switch to Defaults
IP Symbol
Power Estimation
Show disabled ports
BRAM_PORTA
addra[3:0]
clka
douta[3:0]
Component Name
blk_mem_gen_0
Basic
Port A Options
Other Options
Summary
Information
Memory Type: Single Port ROM
Block RAM resource(s) (18K BRAMs): 1
Block RAM resource(s) (36K BRAMs): 0
Total Port A Read Latency : 2 Clock Cycle(s)
Address Width A: 4
OK
Cancel

Testbench for BROM:

ROM - [D:/IIITD/ELD_LABs/LAB_6_ROM/ROM/ROM.xpr] - Vivado 2019.1

File Edit Flow Tools Reports Window Layout View Run Help Quick Access

100 us

Flow Navigator

- PROJECT MANAGER
  - Settings
  - Add Sources
  - Language Templates
  - IP Catalog
- IP INTEGRATOR
- SIMULATION
  - Run Simulation
- RTL ANALYSIS
  - Open Elaborated Design
- SYNTHESIS
  - Run Synthesis
  - Open Synthesized Design
- IMPLEMENTATION
  - Run Implementation
  - Open Implemented Design
- PROGRAM AND DEBUG
  - Generate Bitstream
  - Open Hardware Manager

SIMULATION - Behavioral Simulation - Functional - sim_1 - bram_tb

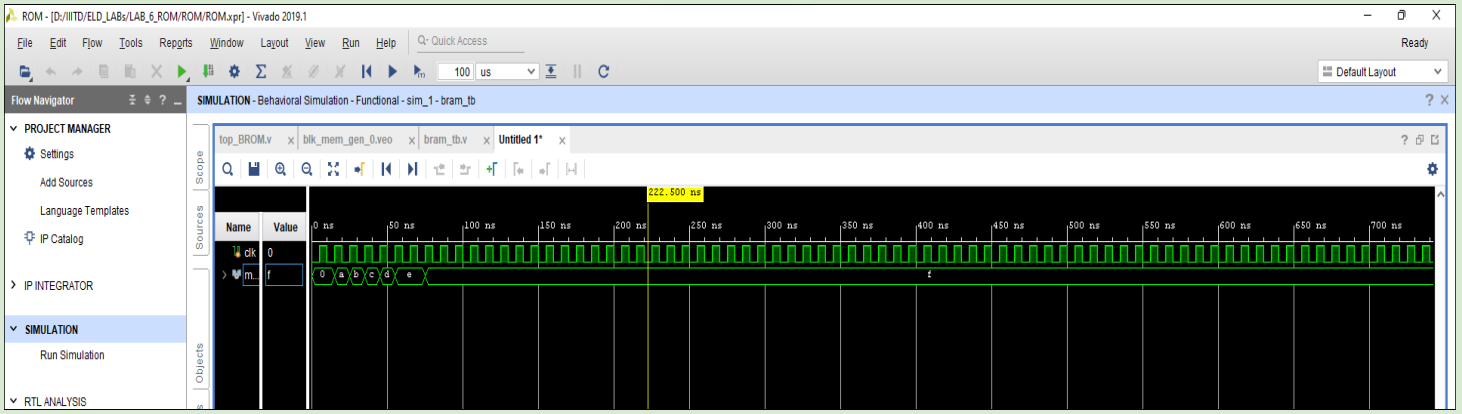
Scope | Sources | Objects | Protocol Instances

top_BROM.v | blk_mem_gen_0.veo | **bram_tb.v** | Untitled 1*

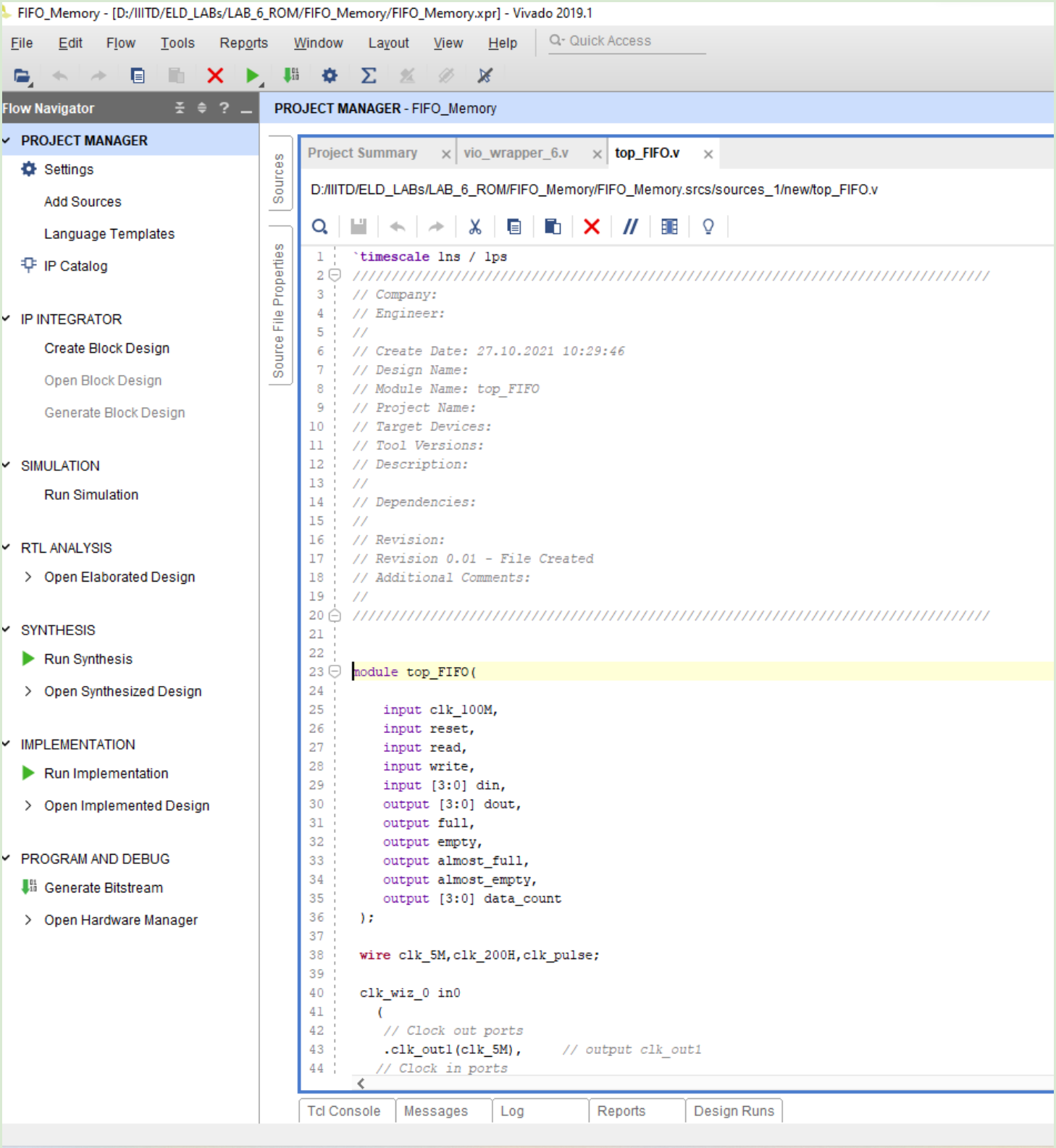
D:/IIITD/ELD_LABs/LAB_6_ROM/ROM/ROM.srcs/sim_1/new/bram_tb.v

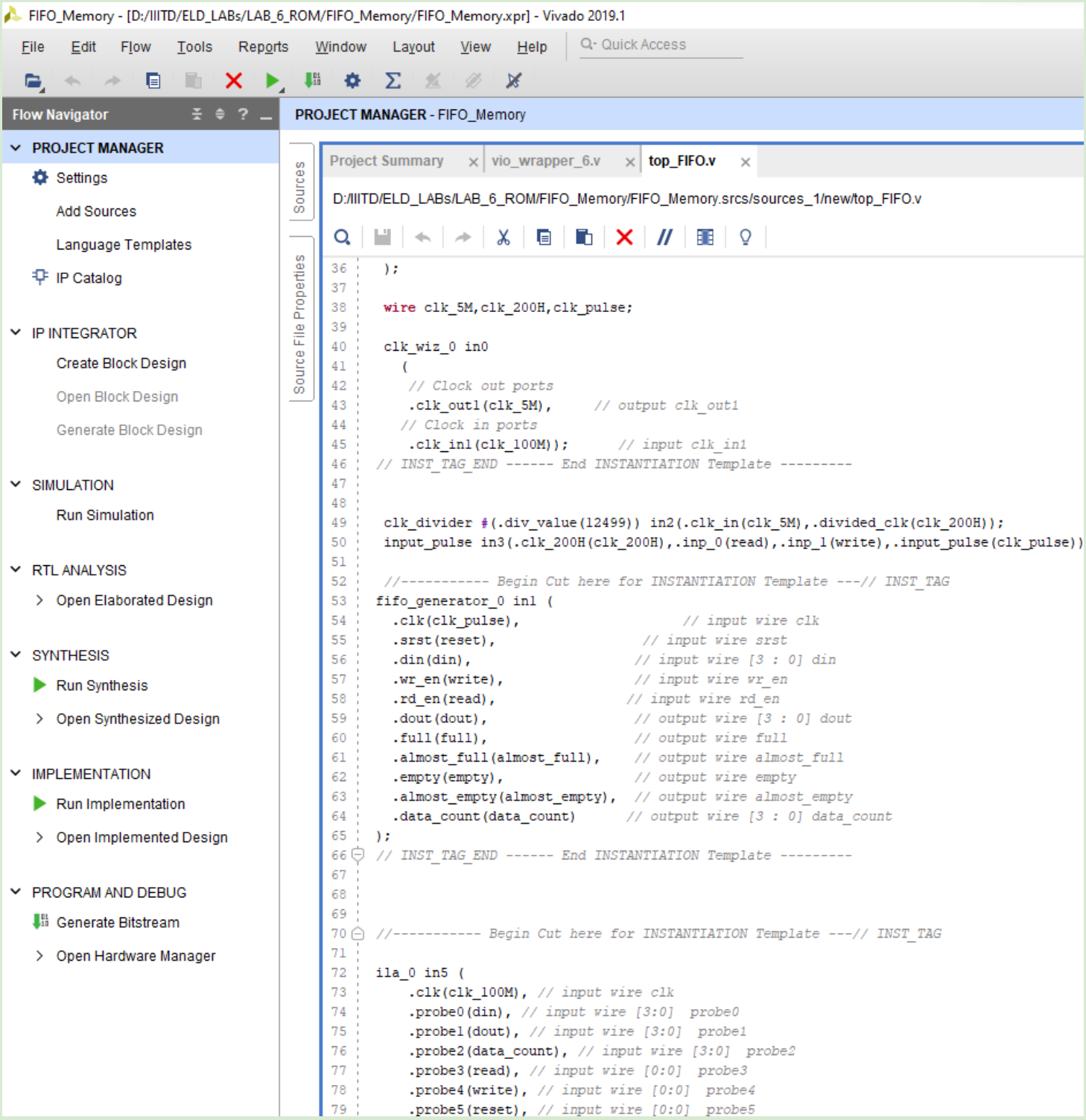
```
1   `timescale 1ns / 1ps
2   //////////////////////////////////////////////////////////////////////////////////
3   // Company:
4   // Engineer:
5   //
6   // Create Date: 08.11.2021 21:32:11
7   // Design Name:
8   // Module Name: bram_tb
9   // Project Name:
10  // Target Devices:
11  // Tool Versions:
12  // Description:
13  //
14  // Dependencies:
15  //
16  // Revision:
17  // Revision 0.01 - File Created
18  // Additional Comments:
19  //
20  //////////////////////////////////////////////////////////////////////////////////
21
22
23  module bram_tb(
24
25      );
26
27      reg clk=0;
28      wire [3:0] max;
29
30      top_BROM tb1(.clk(clk),.max(max));
31
32      always #5 clk=~clk;
33
34  endmodule
35
```

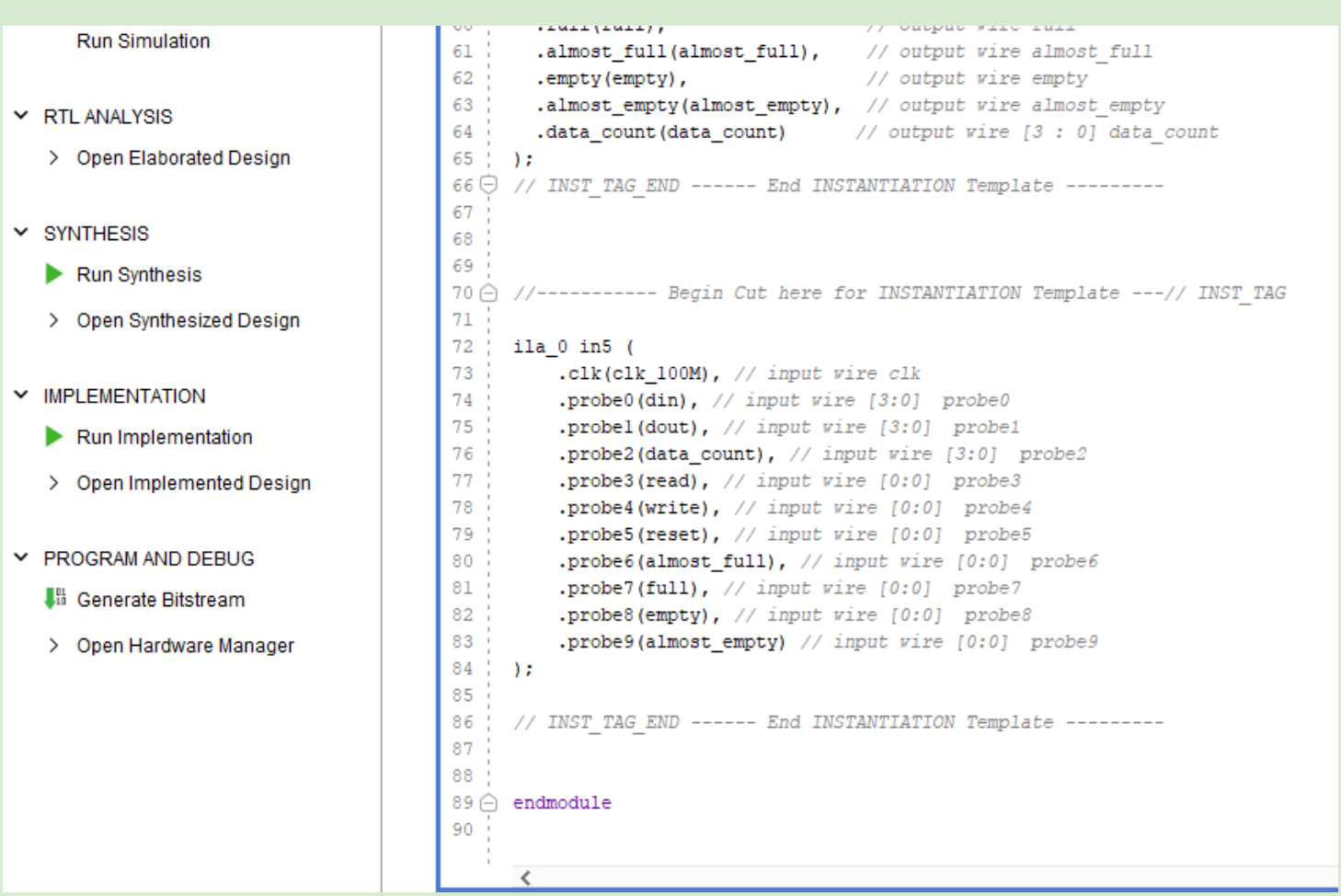
Results for automated testbench of BROM :



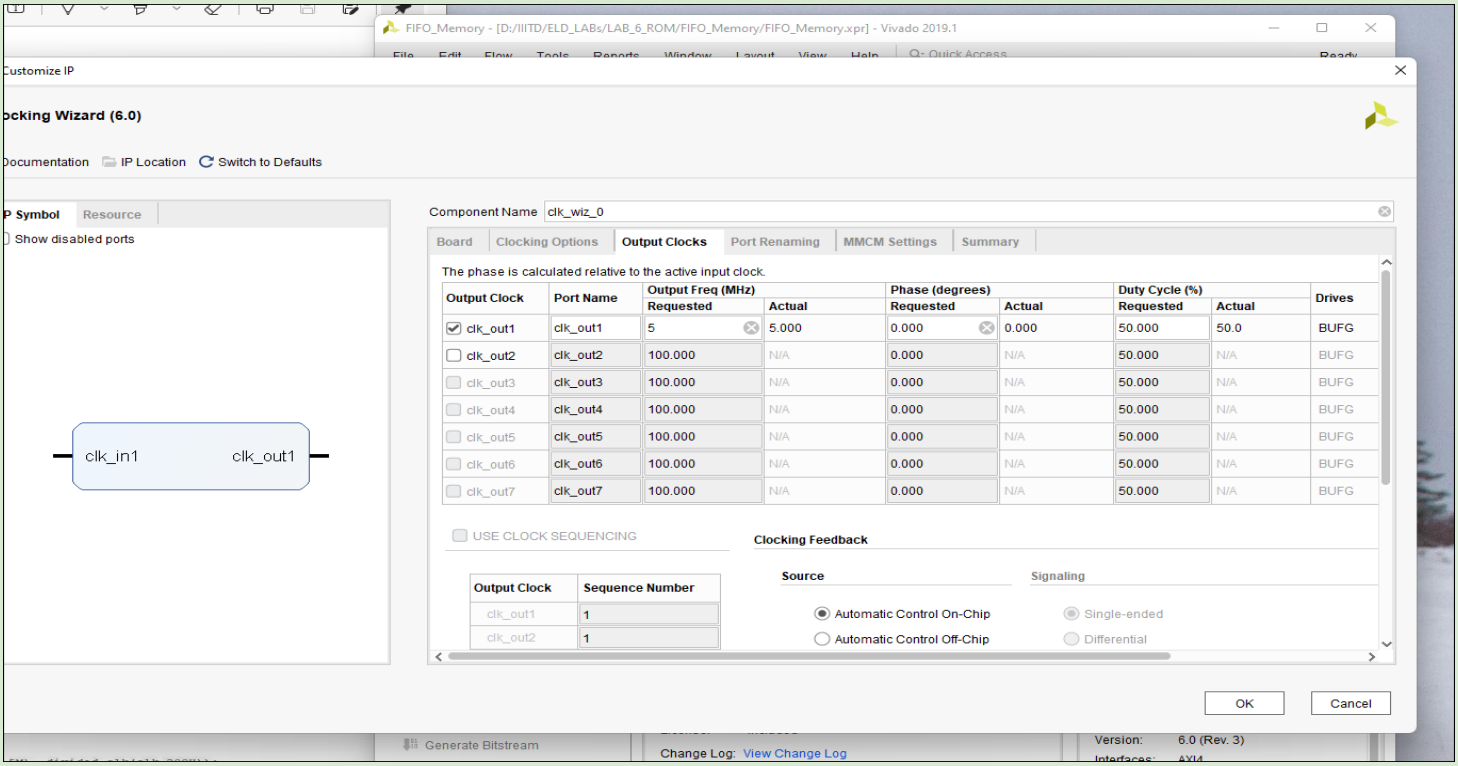
PART _2__Programme for the functionality FIFO



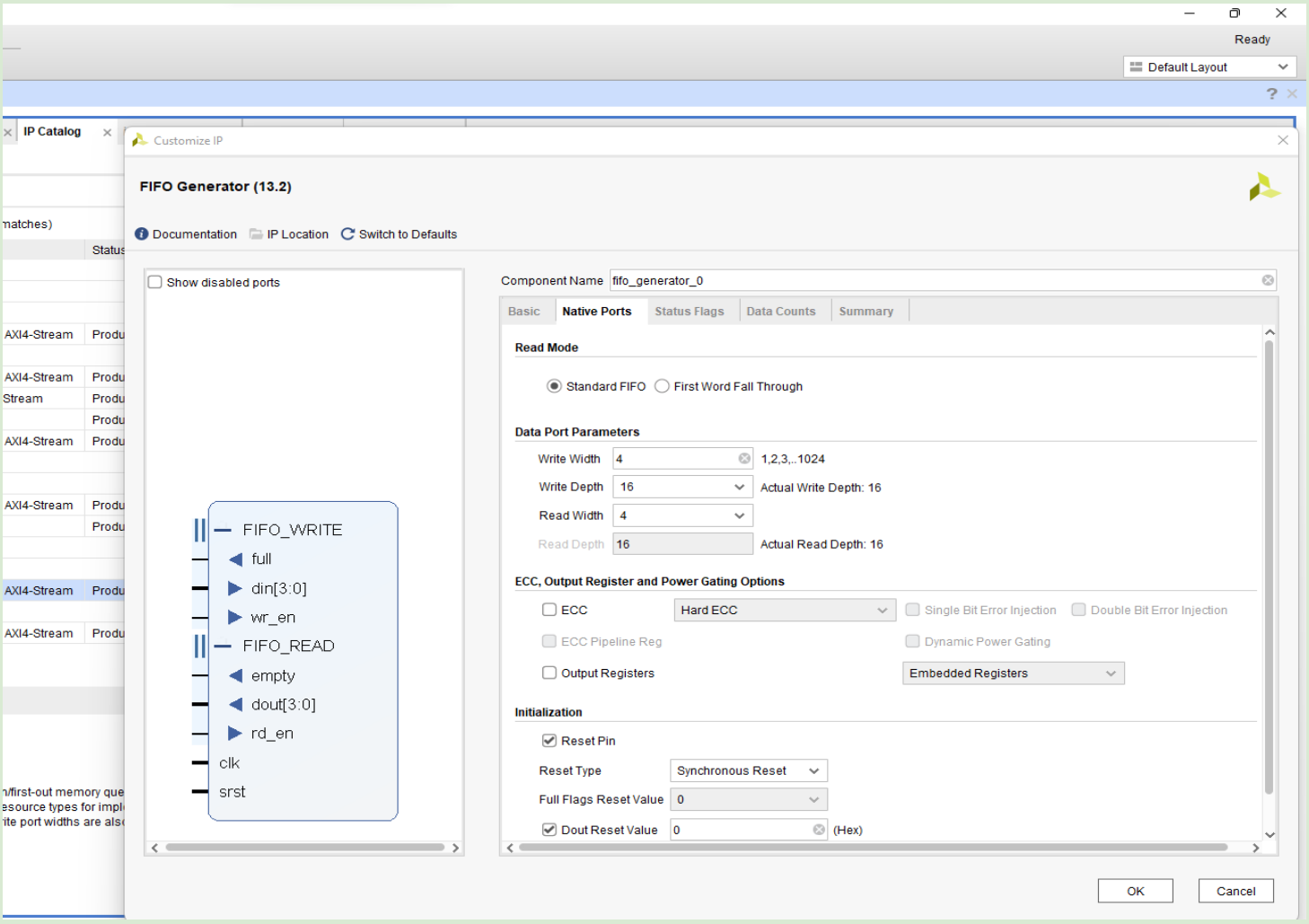

FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
PROJECT MANAGER - FIFO_Memory
Sources
Source File Properties
Project Summary
vio_wrapper_6.v
top_FIFO.v
D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.srcs/sources_1/new/top_FIFO.v
36);
37
38 wire clk_5M,clk_200H,clk_pulse;
39
40 clk_wiz_0 in0
41 (
42 // Clock out ports
43 .clk_out1(clk_5M), // output clk_out1
44 // Clock in ports
45 .clk_in1(clk_100M)); // input clk_in1
46 // INST_TAG_END ------ End INSTANTIATION Template ---------
47
48
49 clk_divider #(.div_value(12499)) in2(.clk_in(clk_5M),.divided_clk(clk_200H));
50 input_pulse in3(.clk_200H(clk_200H),.inp_0(read),.inp_1(write),.input_pulse(clk_pulse))
51
52 //----------- Begin Cut here for INSTANTIATION Template ---// INST_TAG
53 fifo_generator_0 in1 (
54 .clk(clk_pulse), // input wire clk
55 .srst(reset), // input wire srst
56 .din(din), // input wire [3 : 0] din
57 .wr_en(write), // input wire wr_en
58 .rd_en(read), // input wire rd_en
59 .dout(dout), // output wire [3 : 0] dout
60 .full(full), // output wire full
61 .almost_full(almost_full), // output wire almost_full
62 .empty(empty), // output wire empty
63 .almost_empty(almost_empty), // output wire almost_empty
64 .data_count(data_count) // output wire [3 : 0] data_count
65);
66 // INST_TAG_END ------ End INSTANTIATION Template ---------
67
68
69
70 //----------- Begin Cut here for INSTANTIATION Template ---// INST_TAG
71
72 ila_0 in5 (
73 .clk(clk_100M), // input wire clk
74 .probe0(din), // input wire [3:0] probe0
75 .probe1(dout), // input wire [3:0] probe1
76 .probe2(data_count), // input wire [3:0] probe2
77 .probe3(read), // input wire [0:0] probe3
78 .probe4(write), // input wire [0:0] probe4
79 .probe5(reset), // input wire [0:0] probe5

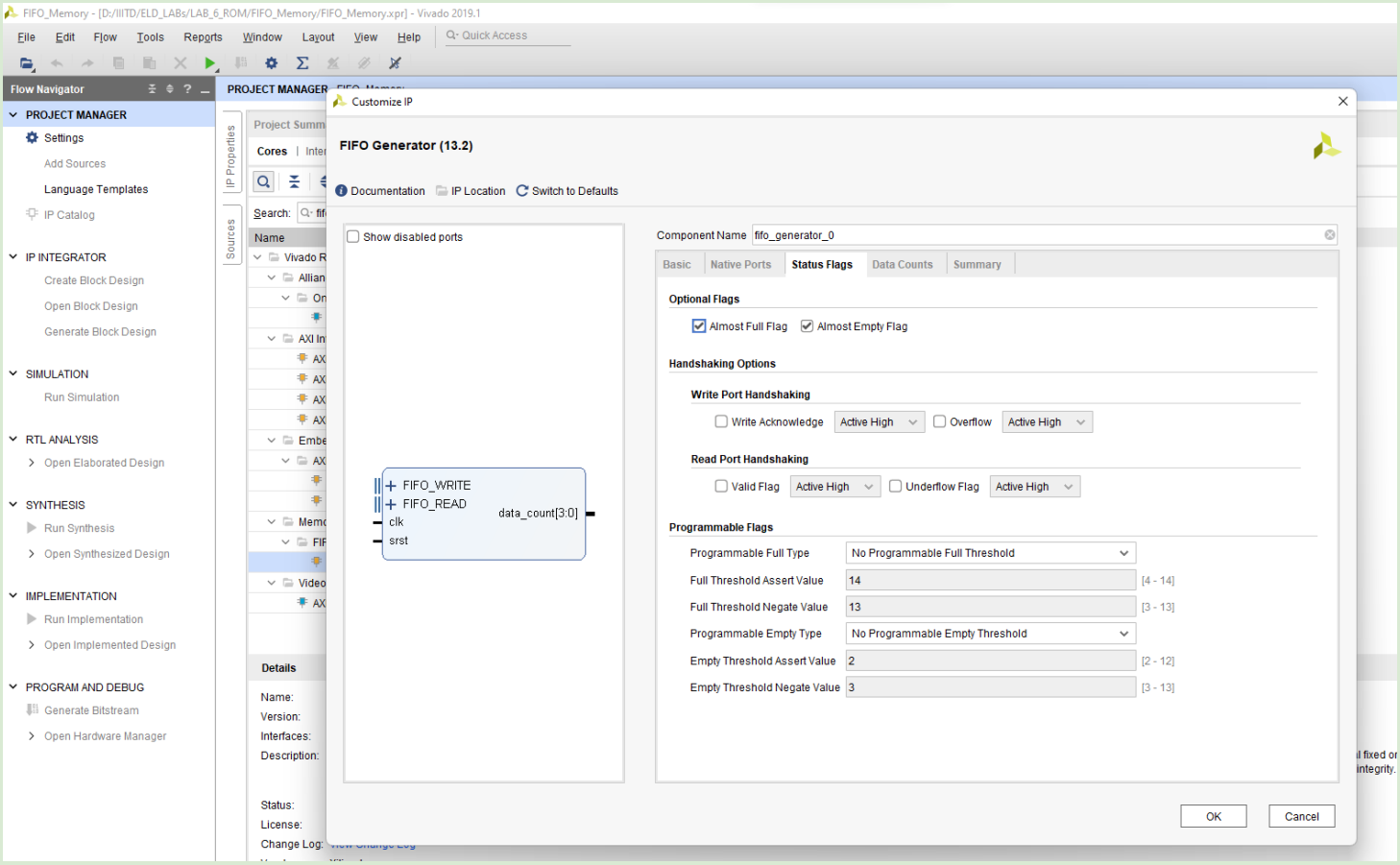


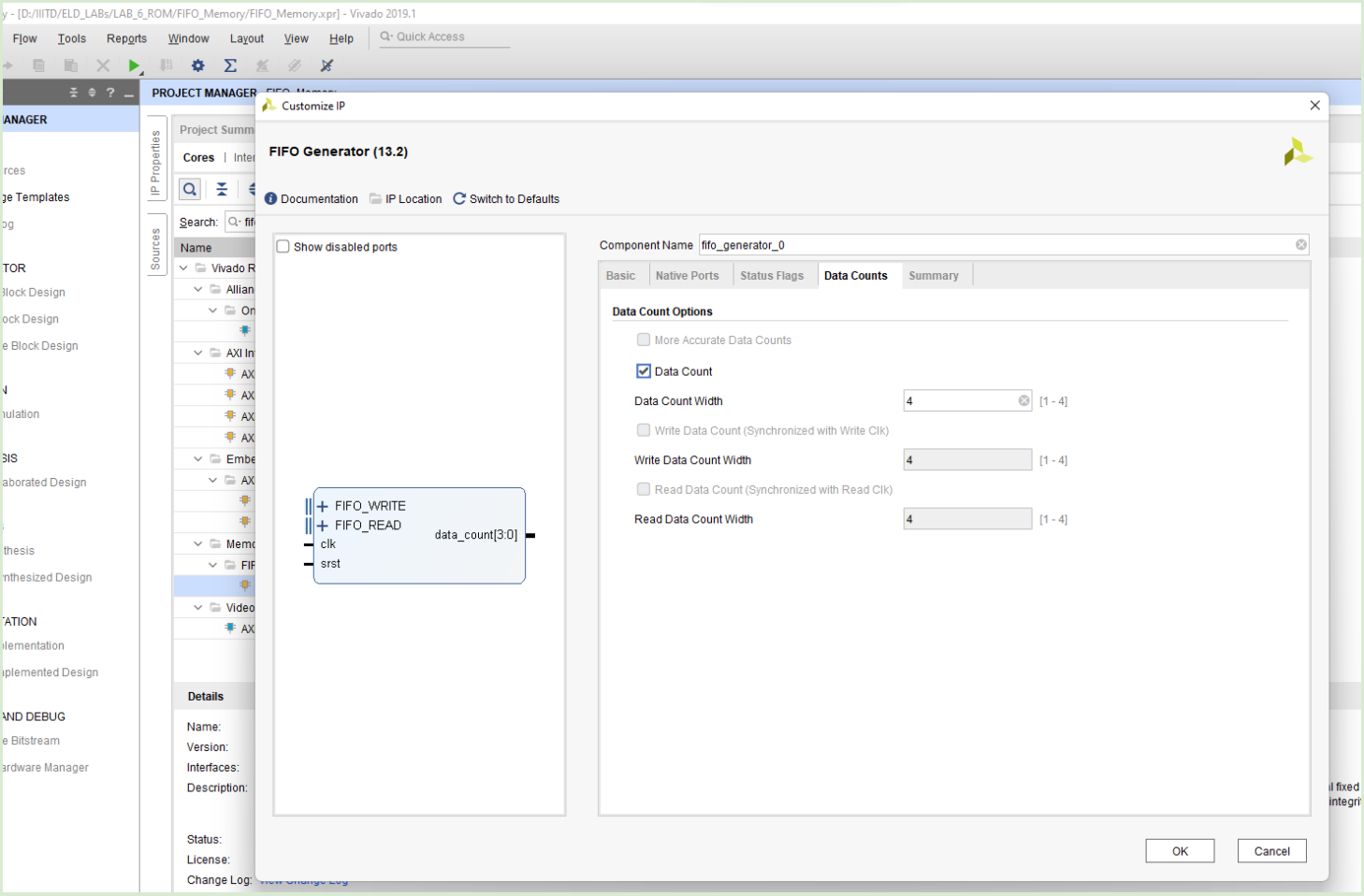
Clocking IP Wizard:

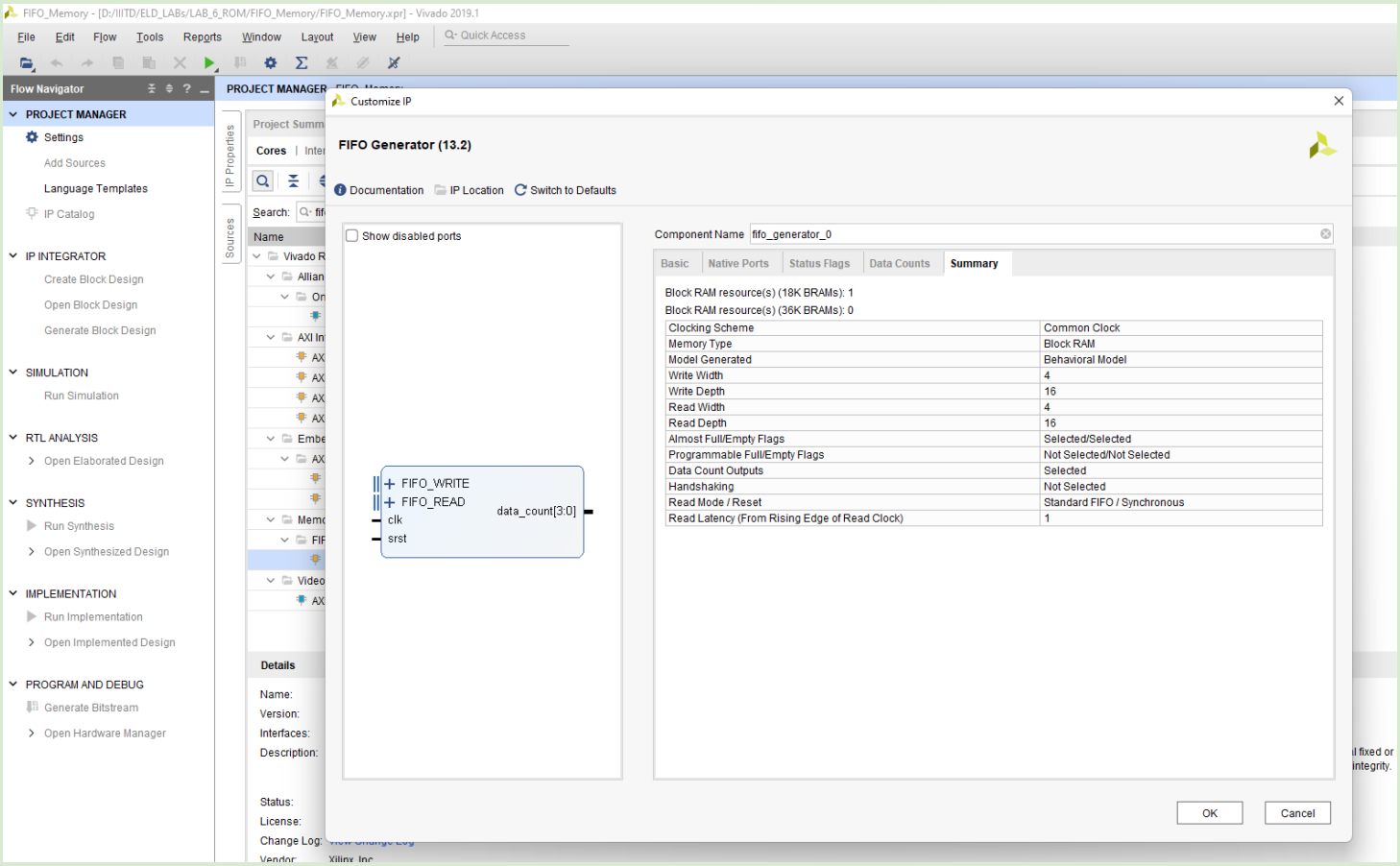


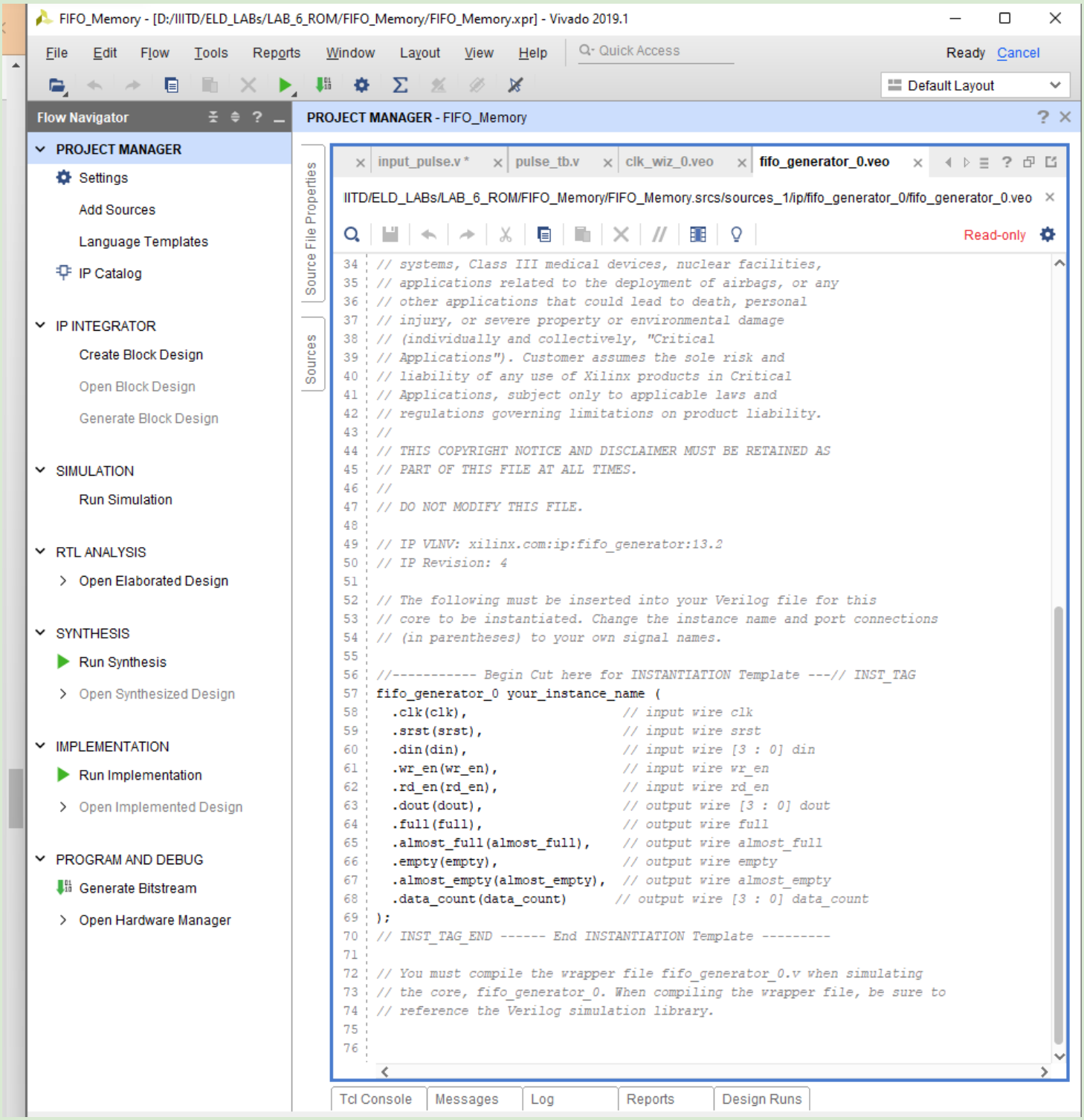
FIFO generator:



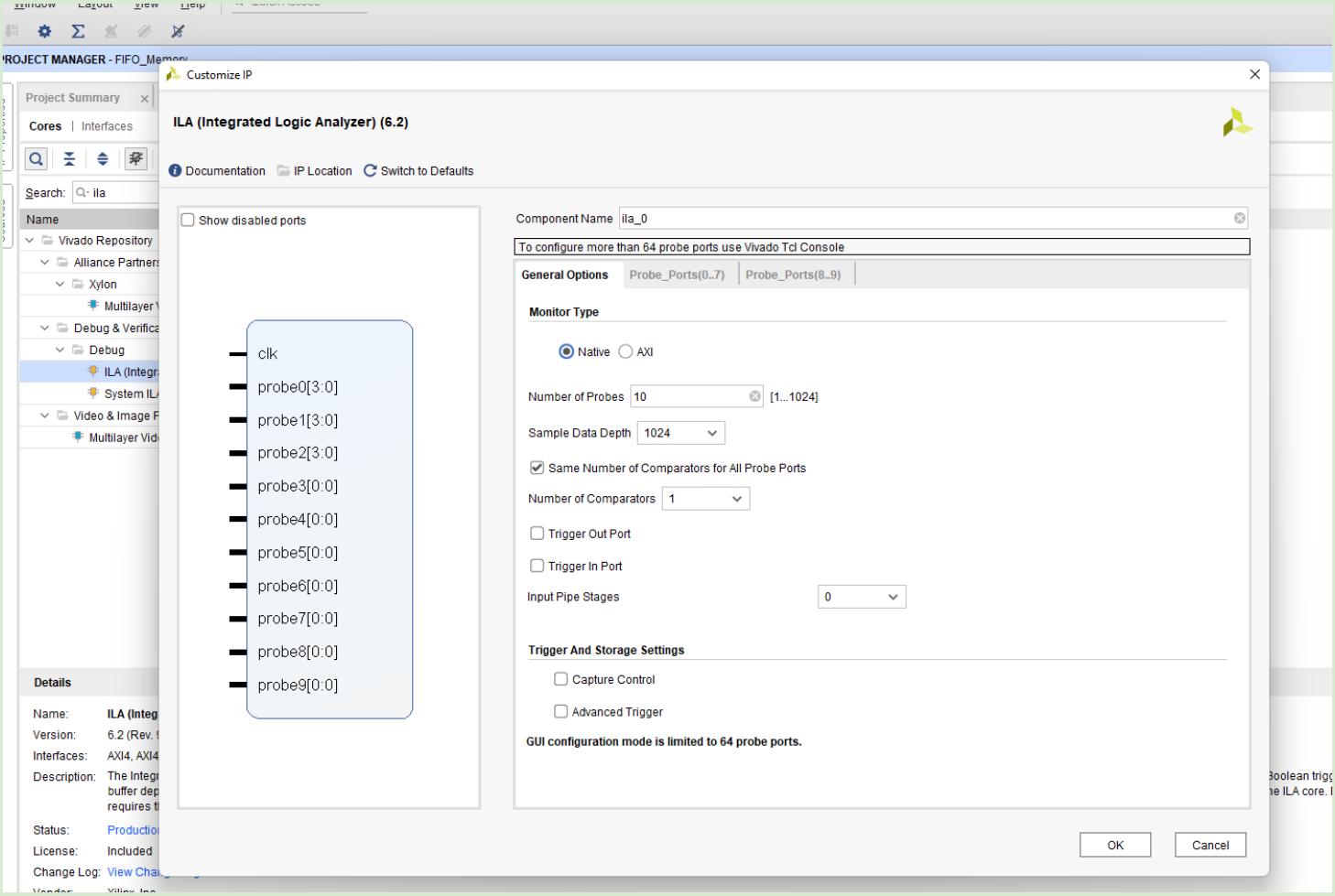


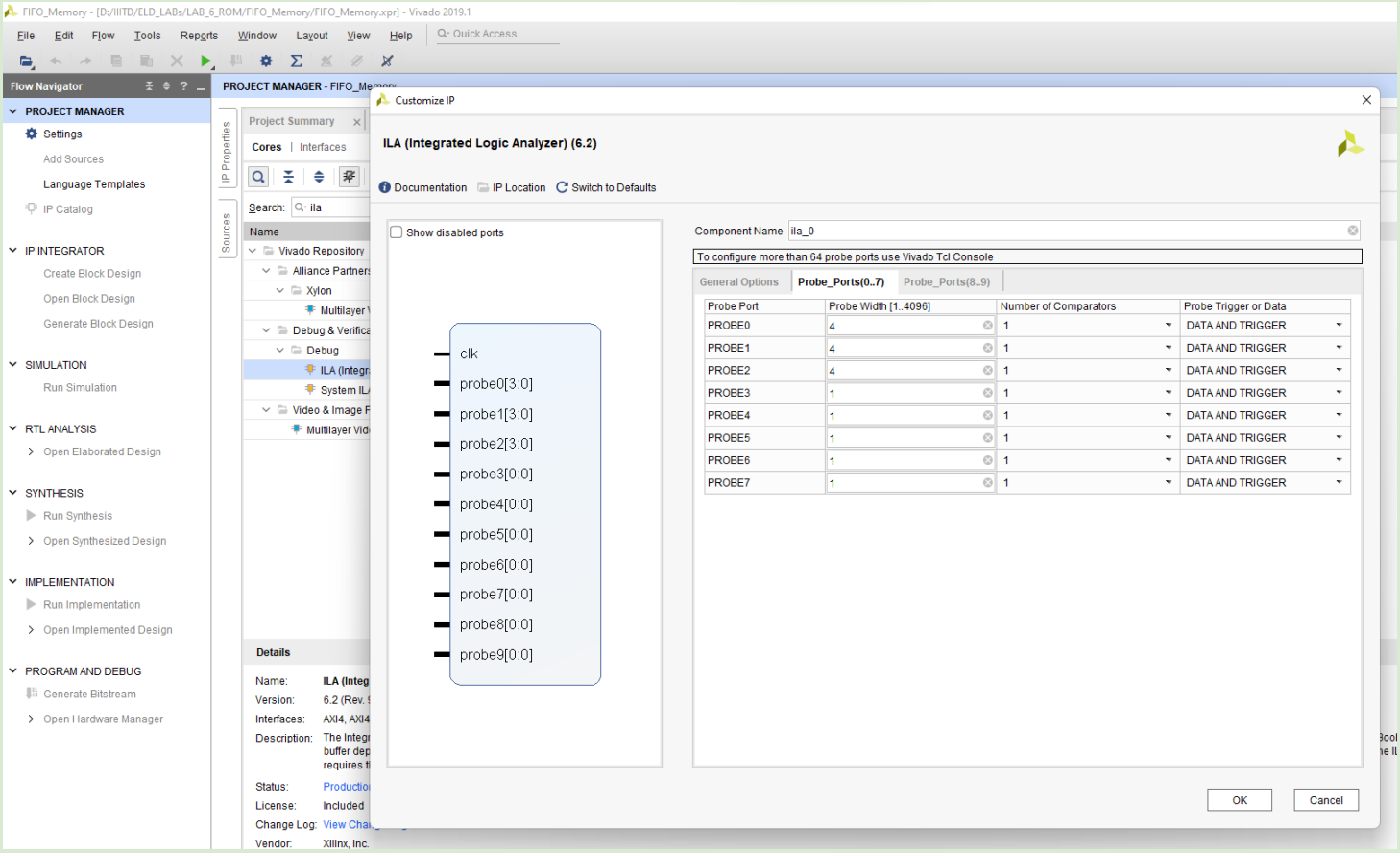

Customize IP
FIFO Generator (13.2)
Documentation
IP Location
Switch to Defaults
Show disabled ports
FIFO_WRITE
FIFO_READ
clk
srst
data_count[3:0]
Component Name
fifo_generator_0
Basic
Native Ports
Status Flags
Data Counts
Summary
Data Count Options
More Accurate Data Counts
Data Count
Data Count Width
4
[1 - 4]
Write Data Count (Synchronized with Write Clk)
Write Data Count Width
Read Data Count (Synchronized with Read Clk)
Read Data Count Width
OK
Cancel

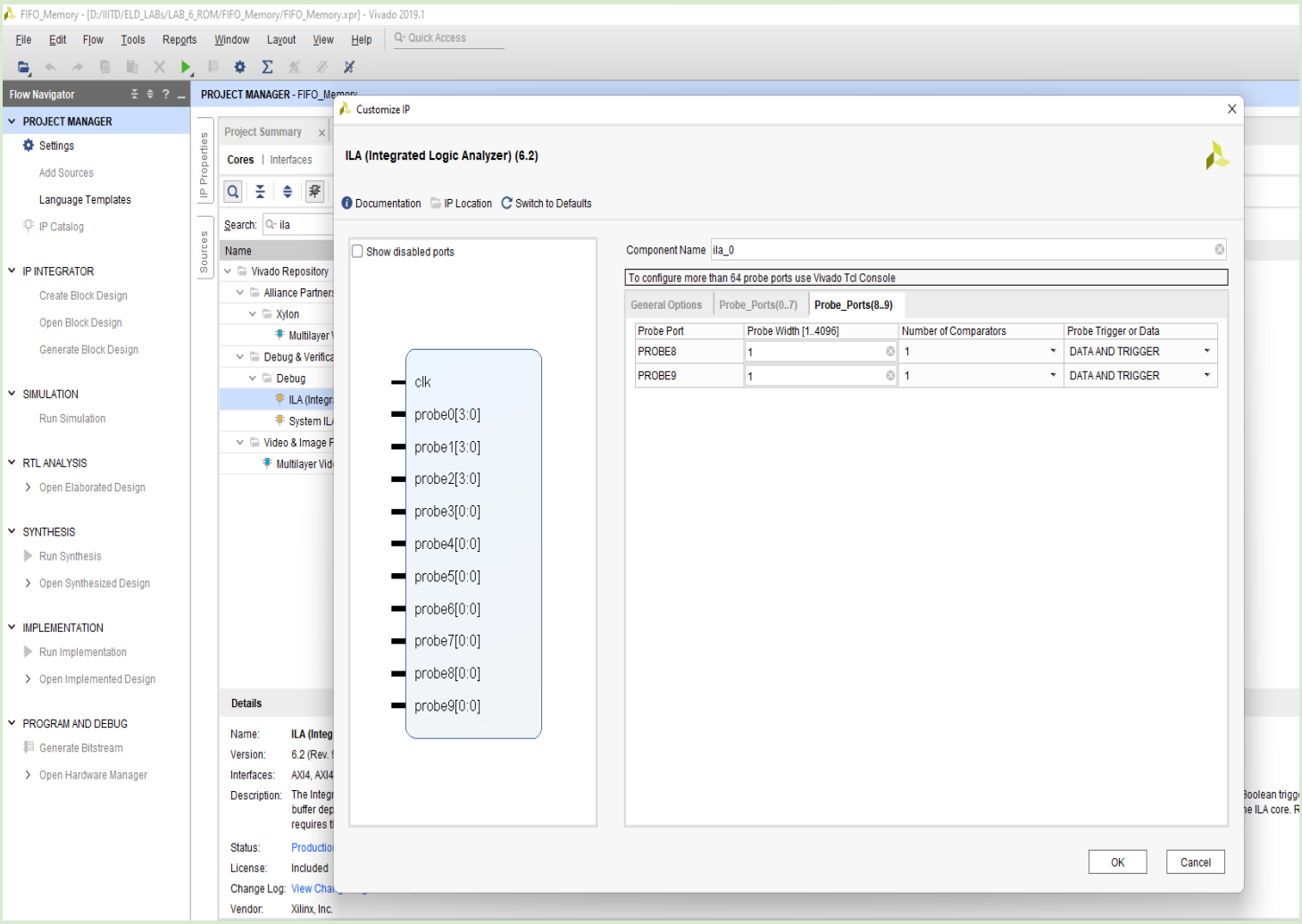

Customize IP
FIFO Generator (13.2)
Documentation
IP Location
Switch to Defaults
Show disabled ports
FIFO_WRITE
FIFO_READ
clk
srst
data_count[3:0]
Component Name
fifo_generator_0
Basic
Native Ports
Status Flags
Data Counts
Summary
Block RAM resource(s) (18K BRAMs): 1
Block RAM resource(s) (36K BRAMs): 0
Clocking Scheme
Common Clock
Memory Type
Block RAM
Model Generated
Behavioral Model
Write Width
4
Write Depth
16
Read Width
4
Read Depth
16
Almost Full/Empty Flags
Selected/Selected
Programmable Full/Empty Flags
Not Selected/Not Selected
Data Count Outputs
Selected
Handshaking
Not Selected
Read Mode / Reset
Standard FIFO / Synchronous
Read Latency (From Rising Edge of Read Clock)
1
OK
Cancel


FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
Ready Cancel
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
PROJECT MANAGER - FIFO_Memory
Source File Properties
Sources
input_pulse.v *
pulse_tb.v
clk_wiz_0.veo
fifo_generator_0.veo
IITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.srcs/sources_1/ip/fifo_generator_0/fifo_generator_0.veo
Read-only
34 // systems, Class III medical devices, nuclear facilities,
35 // applications related to the deployment of airbags, or any
36 // other applications that could lead to death, personal
37 // injury, or severe property or environmental damage
38 // (individually and collectively, "Critical
39 // Applications"). Customer assumes the sole risk and
40 // liability of any use of Xilinx products in Critical
41 // Applications, subject only to applicable laws and
42 // regulations governing limitations on product liability.
43 //
44 // THIS COPYRIGHT NOTICE AND DISCLAIMER MUST BE RETAINED AS
45 // PART OF THIS FILE AT ALL TIMES.
46 //
47 // DO NOT MODIFY THIS FILE.
48
49 // IP VLNV: xilinx.com:ip:fifo_generator:13.2
50 // IP Revision: 4
51
52 // The following must be inserted into your Verilog file for this
53 // core to be instantiated. Change the instance name and port connections
54 // (in parentheses) to your own signal names.
55
56 //----------- Begin Cut here for INSTANTIATION Template ---// INST_TAG
57 fifo_generator_0 your_instance_name (
58 .clk(clk), // input wire clk
59 .srst(srst), // input wire srst
60 .din(din), // input wire [3 : 0] din
61 .wr_en(wr_en), // input wire wr_en
62 .rd_en(rd_en), // input wire rd_en
63 .dout(dout), // output wire [3 : 0] dout
64 .full(full), // output wire full
65 .almost_full(almost_full), // output wire almost_full
66 .empty(empty), // output wire empty
67 .almost_empty(almost_empty), // output wire almost_empty
68 .data_count(data_count) // output wire [3 : 0] data_count
69);
70 // INST_TAG_END ------ End INSTANTIATION Template ---------
71
72 // You must compile the wrapper file fifo_generator_0.v when simulating
73 // the core, fifo_generator_0. When compiling the wrapper file, be sure to
74 // reference the Verilog simulation library.
75
76
Tcl Console
Messages
Log
Reports
Design Runs

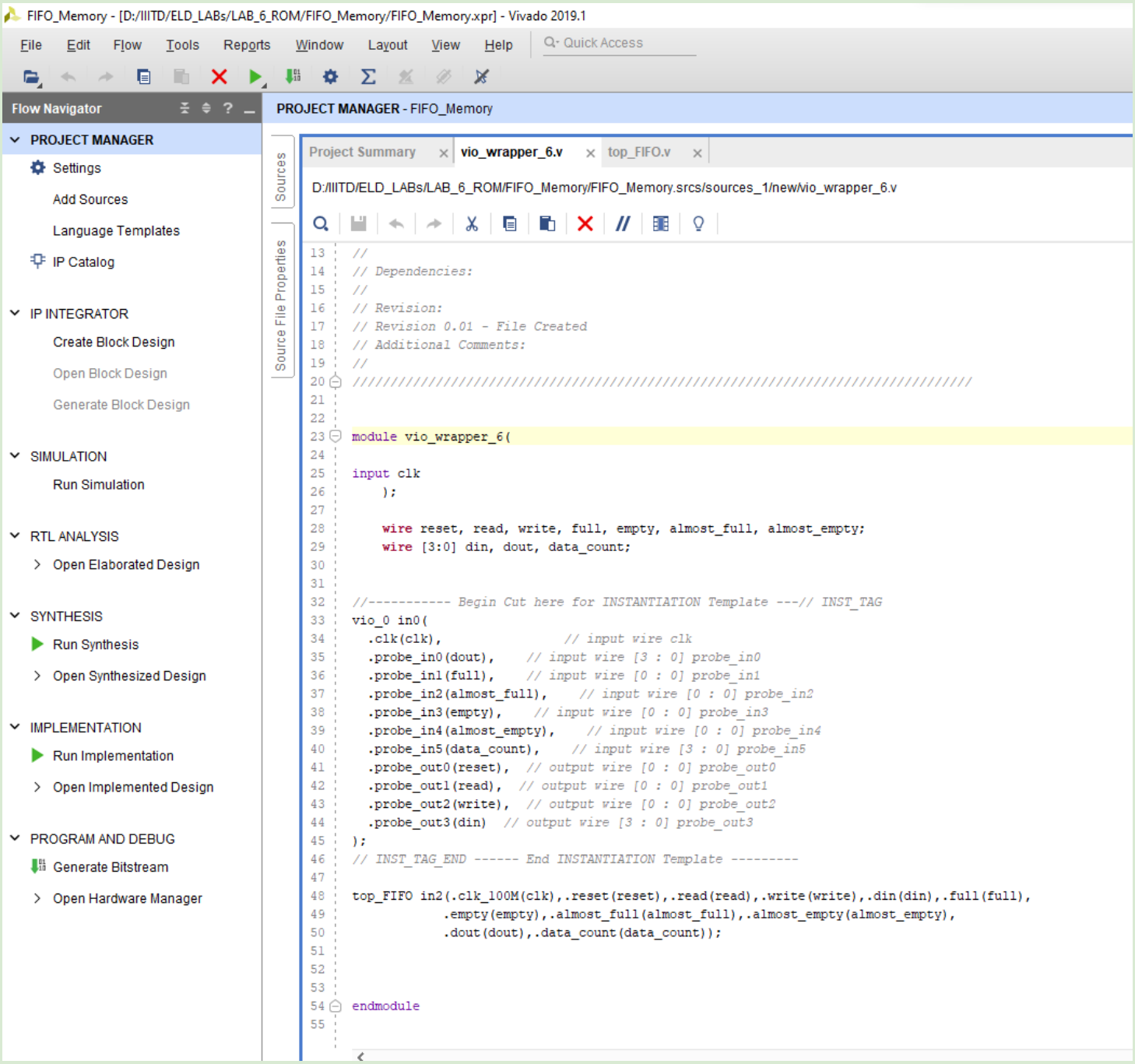
ILA wizard for FIFO:



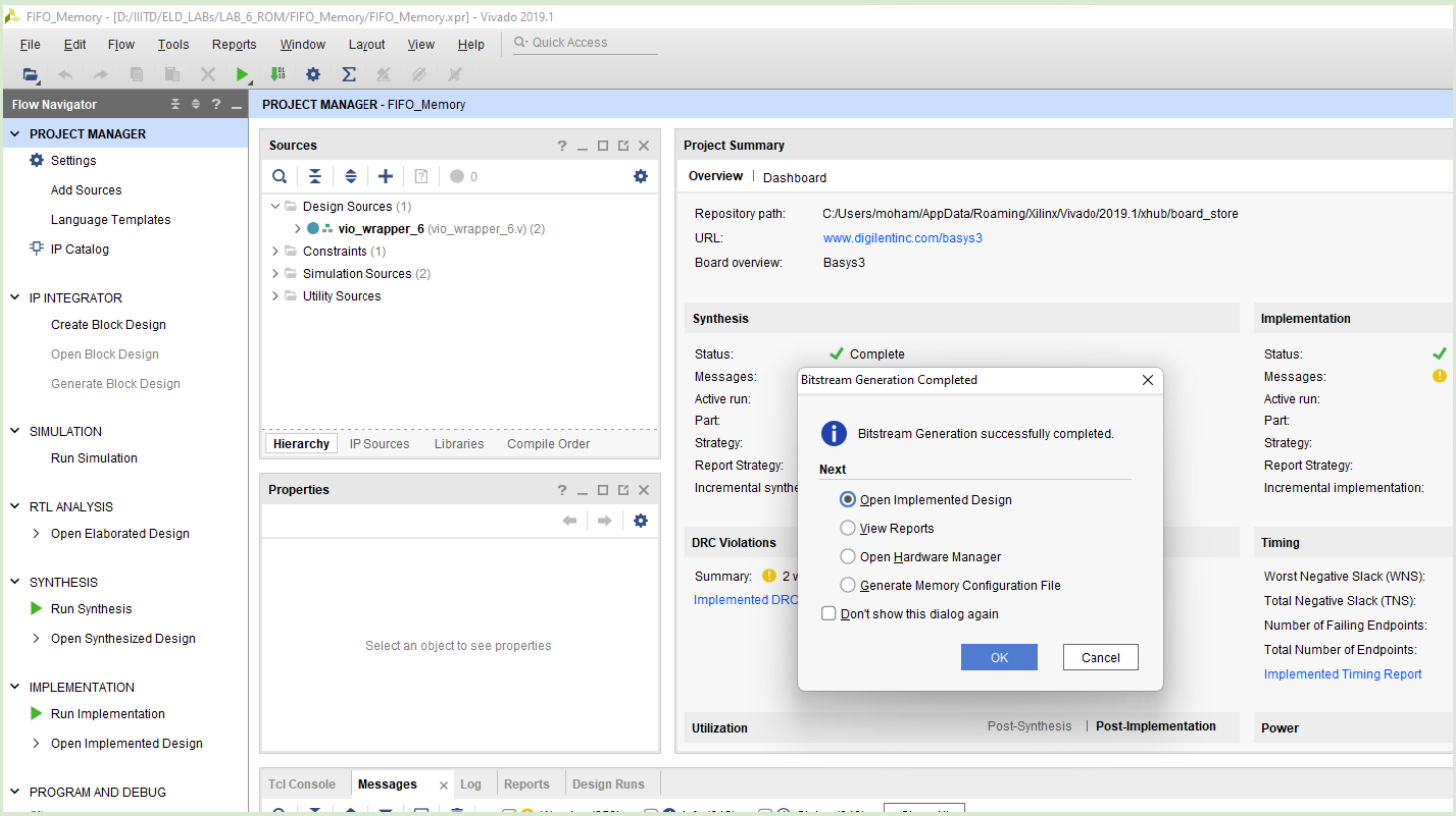



FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
PROJECT MANAGER - FIFO_Memory
IP Properties
Sources
Project Summary
Cores | Interfaces
Search: ila
Name
Vivado Repository
Alliance Partners
Xylon
Debug & Verifica
Debug
ILA (Integr
System ILA
Video & Image P
Details
Name: ILA (Integ
Version: 6.2 (Rev.
Interfaces: AXI4, AXI4
Description: The Integr
buffer dep
requires t
Status: Productio
License: Included
Change Log: View Cha
Vendor: Xilinx, Inc.
Customize IP
ILA (Integrated Logic Analyzer) (6.2)
Documentation
IP Location
Switch to Defaults
Show disabled ports
clk
probe0[3:0]
probe1[3:0]
probe2[3:0]
probe3[0:0]
probe4[0:0]
probe5[0:0]
probe6[0:0]
probe7[0:0]
probe8[0:0]
probe9[0:0]
Component Name ila_0
To configure more than 64 probe ports use Vivado Tcl Console
General Options
Probe_Ports(0..7)
Probe_Ports(8..9)
Probe Port
Probe Width [1..4096]
Number of Comparators
Probe Trigger or Data
PROBE8
1
1
DATA AND TRIGGER
PROBE9
1
1
DATA AND TRIGGER
OK
Cancel

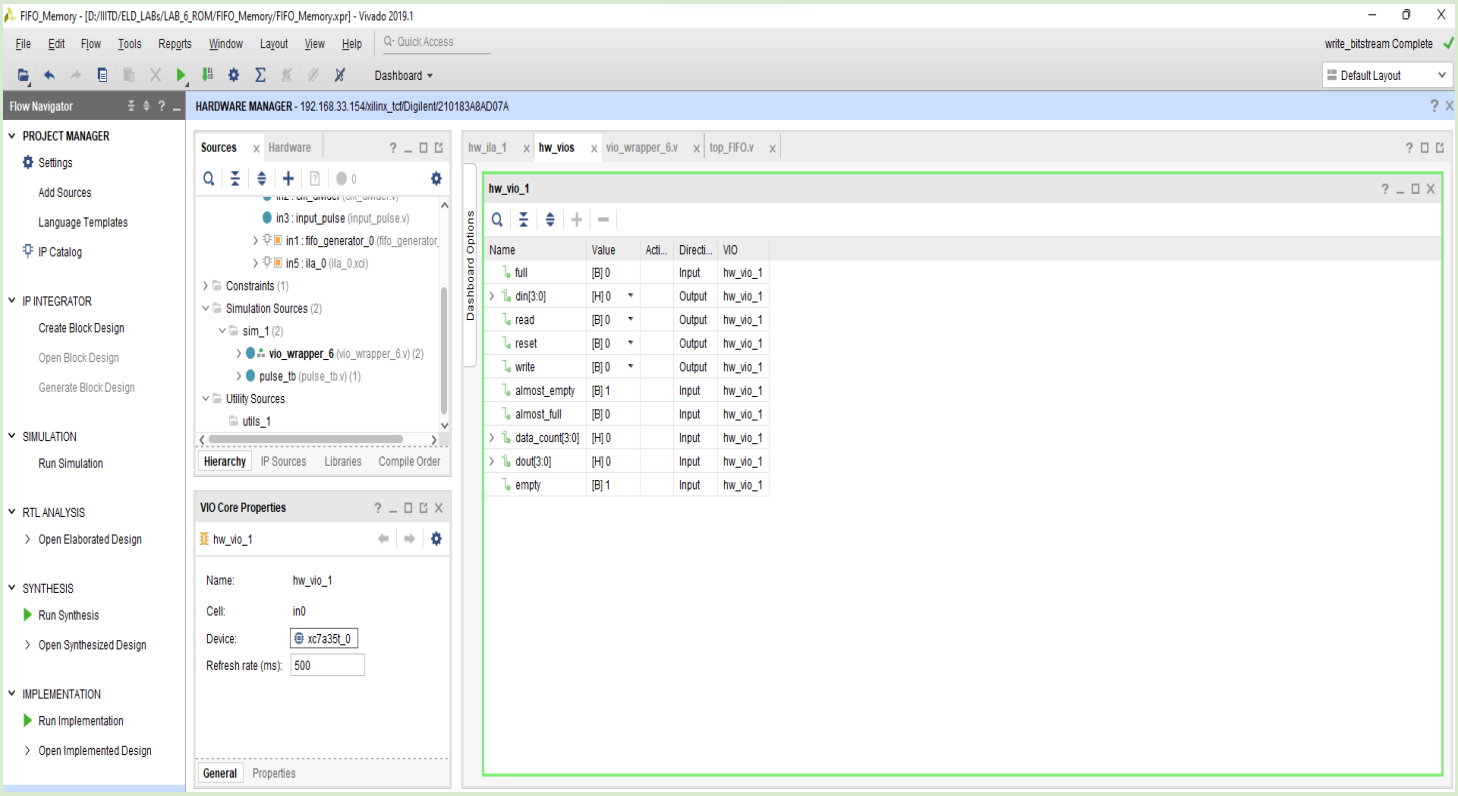
VIO IP Wizard:

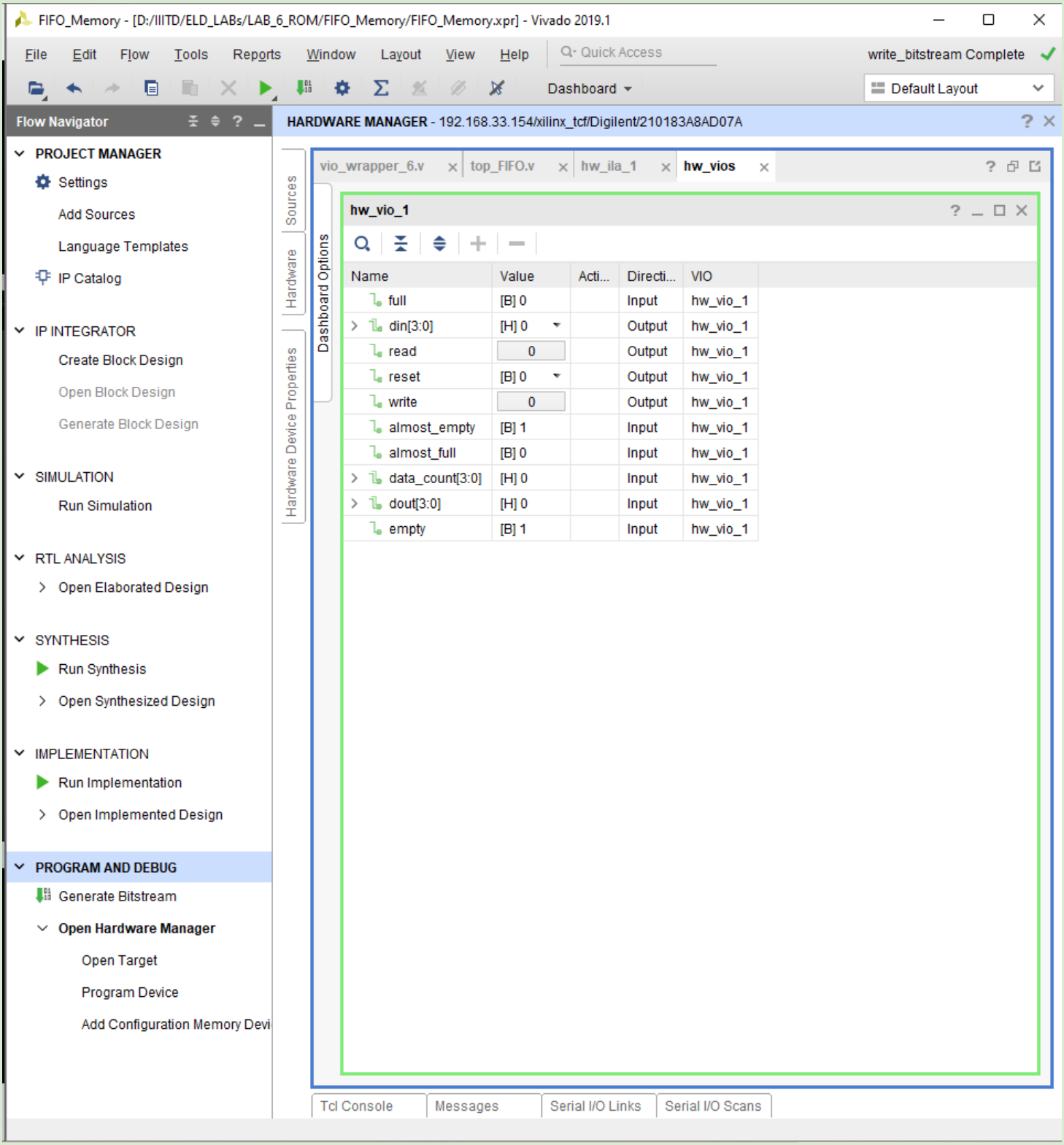


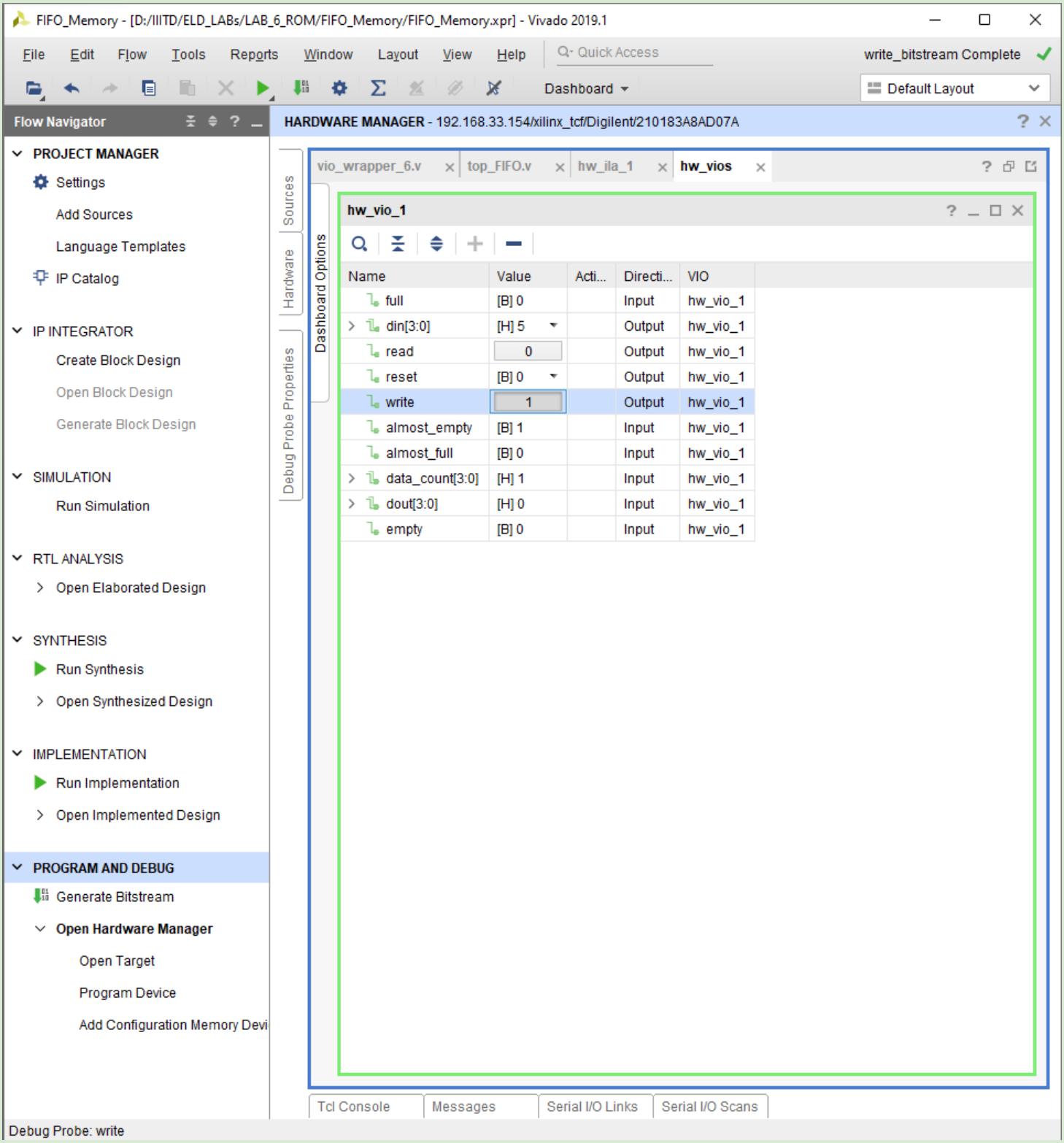
Bitstream generated successfully:

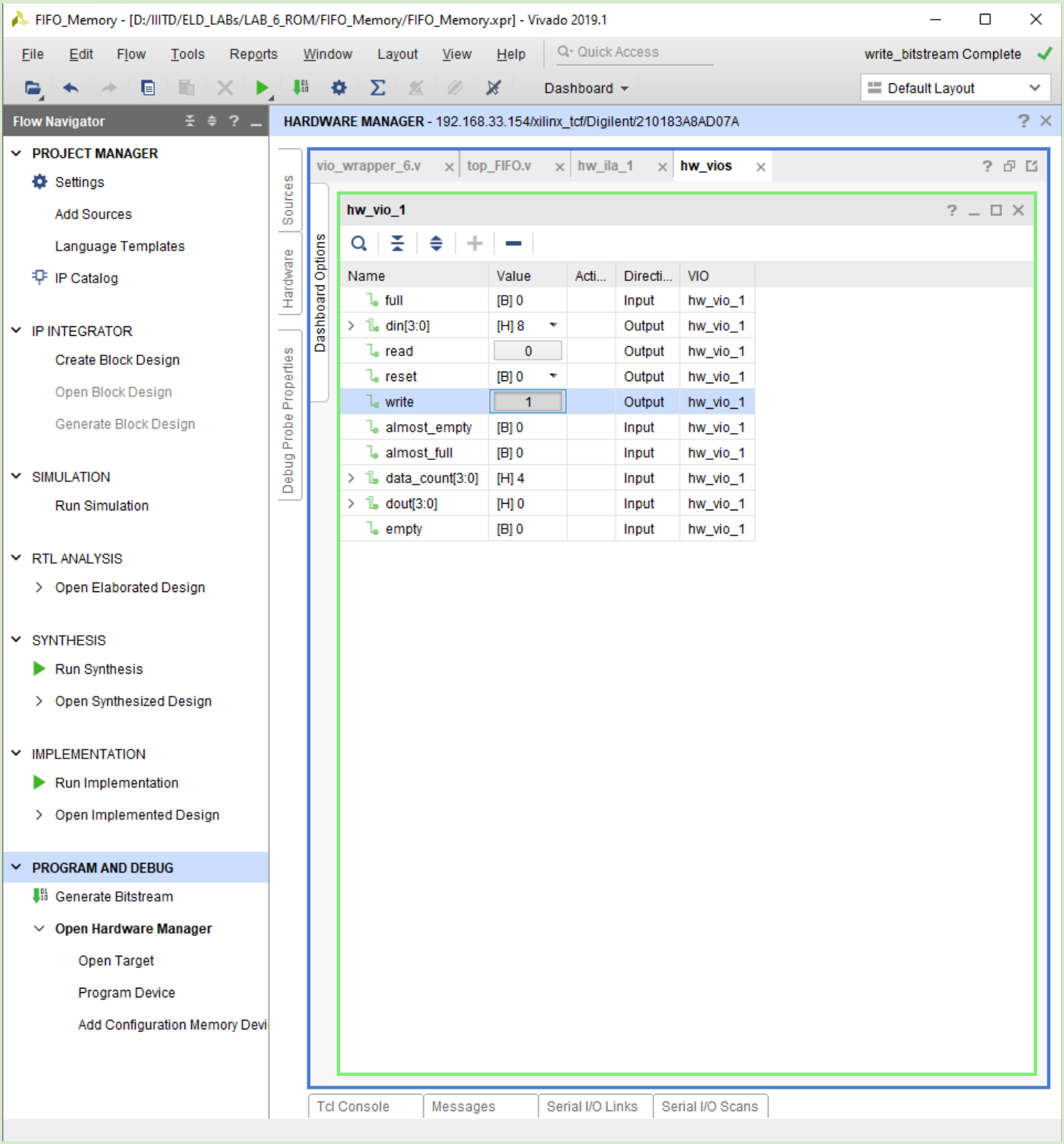


Writing Inputs in FIFO Memory :

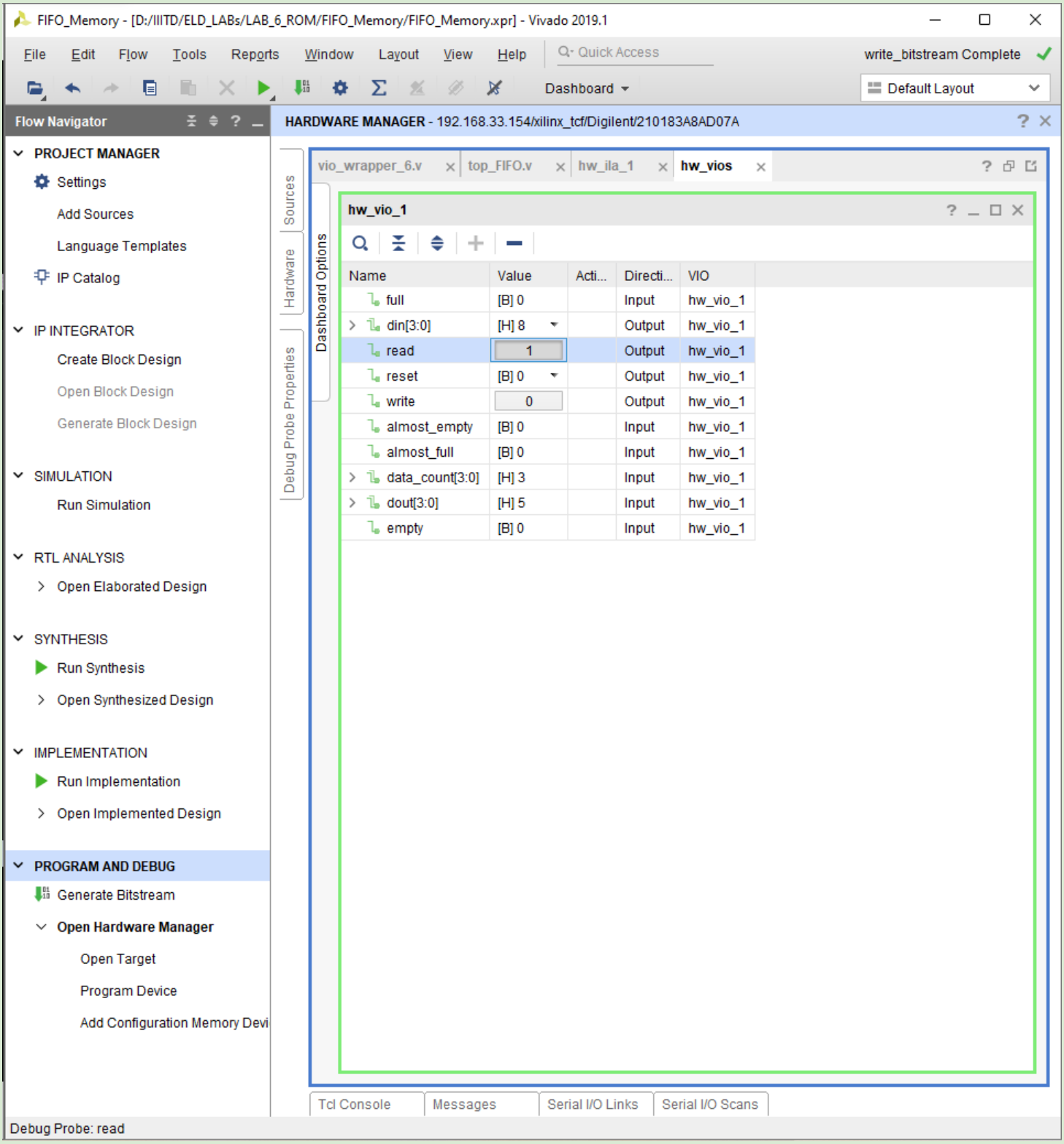



FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
write_bitstream Complete
Dashboard
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
Open Target
Program Device
Add Configuration Memory Devi
HARDWARE MANAGER - 192.168.33.154/xilinx_tcf/Digilent/210183A8AD07A
vio_wrapper_6.v
top_FIFO.v
hw_ila_1
hw_vios
Sources
Hardware
Hardware Device Properties
Dashboard Options
hw_vio_1
Name Value Acti... Directi... VIO
full [B] 0 Input hw_vio_1
din[3:0] [H] 0 Output hw_vio_1
read 0 Output hw_vio_1
reset [B] 0 Output hw_vio_1
write 0 Output hw_vio_1
almost_empty [B] 1 Input hw_vio_1
almost_full [B] 0 Input hw_vio_1
data_count[3:0] [H] 0 Input hw_vio_1
dout[3:0] [H] 0 Input hw_vio_1
empty [B] 1 Input hw_vio_1
Tcl Console
Messages
Serial I/O Links
Serial I/O Scans

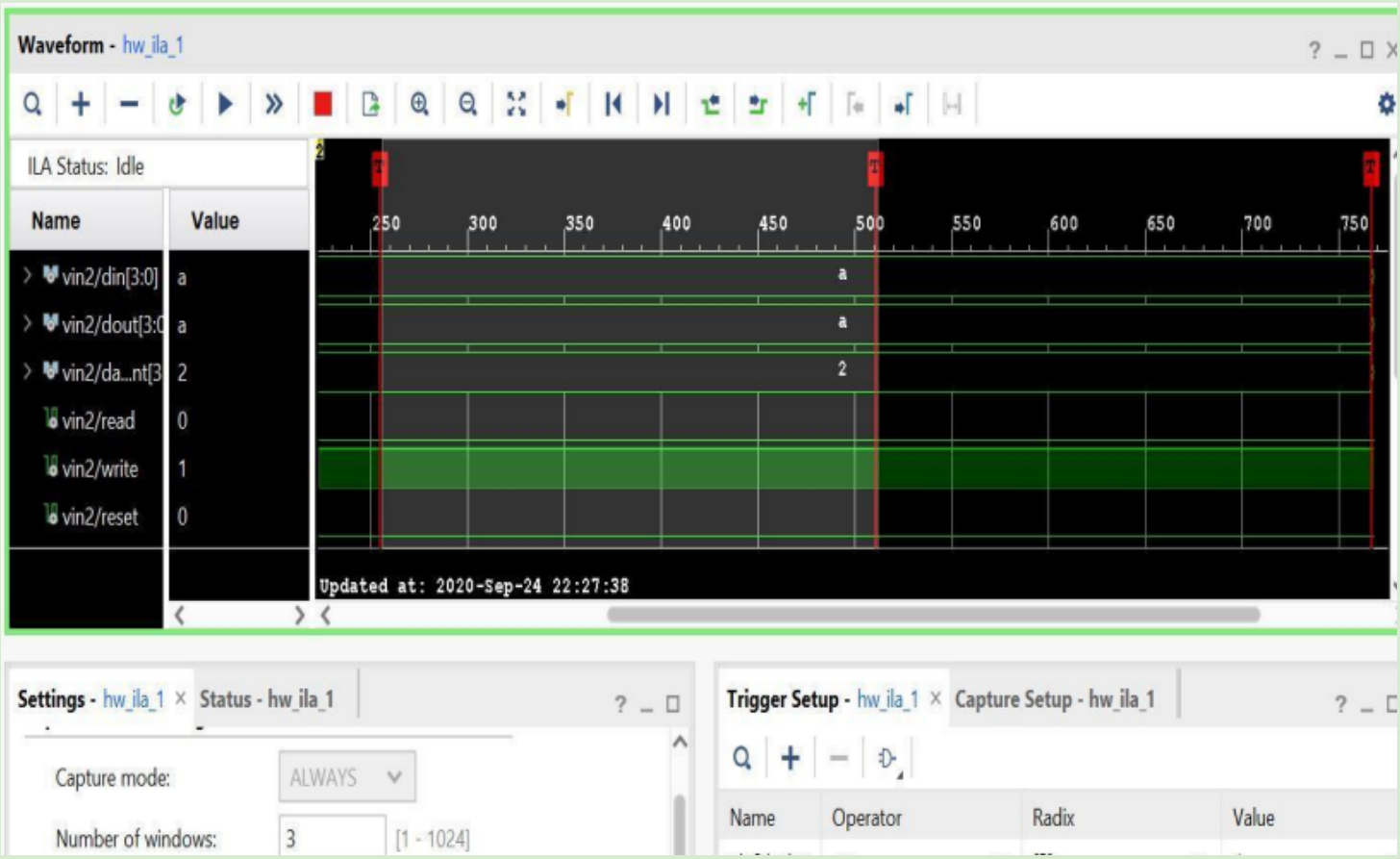

FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
write_bitstream Complete
Dashboard
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
Open Target
Program Device
Add Configuration Memory Devi
HARDWARE MANAGER - 192.168.33.154/xilinx_tcf/Digilent/210183A8AD07A
Sources
Hardware
Debug Probe Properties
Dashboard Options
vio_wrapper_6.v
top_FIFO.v
hw_ila_1
hw_vios
hw_vio_1
Name
Value
Acti...
Directi...
VIO
full
[B] 0
Input
hw_vio_1
din[3:0]
[H] 5
Output
hw_vio_1
read
0
Output
hw_vio_1
reset
[B] 0
Output
hw_vio_1
write
1
Output
hw_vio_1
almost_empty
[B] 1
Input
hw_vio_1
almost_full
[B] 0
Input
hw_vio_1
data_count[3:0]
[H] 1
Input
hw_vio_1
dout[3:0]
[H] 0
Input
hw_vio_1
empty
[B] 0
Input
hw_vio_1
Tcl Console
Messages
Serial I/O Links
Serial I/O Scans
Debug Probe: write


FIFO_Memory - [D:/IIITD/ELD_LABs/LAB_6_ROM/FIFO_Memory/FIFO_Memory.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Help
Quick Access
write_bitstream Complete
Dashboard
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
PROGRAM AND DEBUG
Generate Bitstream
Open Hardware Manager
Open Target
Program Device
Add Configuration Memory Devi
HARDWARE MANAGER - 192.168.33.154/xilinx_tcf/Digilent/210183A8AD07A
Sources
Hardware
Debug Probe Properties
Dashboard Options
vio_wrapper_6.v
top_FIFO.v
hw_ila_1
hw_vios
hw_vio_1
Name
Value
Acti...
Directi...
VIO
full
[B] 0
Input
hw_vio_1
din[3:0]
[H] 8
Output
hw_vio_1
read
0
Output
hw_vio_1
reset
[B] 0
Output
hw_vio_1
write
1
Output
hw_vio_1
almost_empty
[B] 0
Input
hw_vio_1
almost_full
[B] 0
Input
hw_vio_1
data_count[3:0]
[H] 4
Input
hw_vio_1
dout[3:0]
[H] 0
Input
hw_vio_1
empty
[B] 0
Input
hw_vio_1
Tcl Console
Messages
Serial I/O Links
Serial I/O Scans

reading outputs from memory in FIFO format :



ILA results :



Conclusion:

Successfully designed and implemented BROM and FIFO memory.