V.GOKUL KUMAR
Wisig Trainee
Indian Institute of Technology Hyderabad
Email: velicharla@outlook.com

Date:24-03-2023

Hls ASSIGNMENT-4

Design a basic integer ALU unit in HLS that takes 3 inputs: Two 8bit operands and one appropriately sized operator. The permitted operations are ADD, SUB, MUL, DIV, AND, OR and XOR. Use AXI-S ports for all I/O ports. The design should be pipelined.

1. Run C simulation first and verify your design with multiple inputs from a file (you need to create this file as well) containing sets of above 3 inputs and 1 reference output to compare your design output in the testbench. The test bench should be a self-checking testbench. It should run the simulation, pass on inputs, compare the outputs with the reference outputs and write the outputs to another file, along with the Pass or Fail status of that particular test as obtained from the comparison. The testbench should also tell (after the simulation output in the console) at the end of the simulation if all the read input sets passed the test or not.
2. Run synthesis and record the resource consumption and timing report.
3. Run C/RTL co-simulation and perform the same steps as mentioned in 1 above.

The purpose of this assignment is to familiarise you with simple FSM implementation and self-checking testbench in HLS. Report all the source code files, input and output files, and the reports from 1, 2 and 3 above

Design:

ALU function:

alu.h | alu_tb2.cpp | input.dat | Synthesis Summary(solution1) | Co-simulation Report(solution1) | integer_alu_csim.log | alu.cpp

```
#include "alu.h"
void integer_alu(
    hls::stream<dina> &in1,
    hls::stream<dinb> &in2,
    hls::stream<dinop> &op,
    hls::stream<dout> &out
) {
    #pragma HLS INTERFACE axis port=in1
    #pragma HLS INTERFACE axis port=in2
    #pragma HLS INTERFACE axis port=op
    #pragma HLS INTERFACE axis port=out
   // #pragma HLS INTERFACE s_axilite port=out bundle=CTRL_BUS
    #pragma HLS PIPELINE II=1

    dina a, b;
    dinop operation;
    dout result;

            a = in1.read();
            b = in2.read();
            operation=op.read();
            switch (operation) {
                case 0: // ADD
                    result = a + b;
                    break;
                case 1: // SUB
                    result= a - b;
                    break;
                case 2: // MUL
                    result= a * b;
                    break;
                case 3: // DIV
                    result = a / b;
                    break;
                case 4: // AND
                    result = a & b;
                    break;
                case 5: // OR
                    result = a | b;
                    break;
                case 6: // XOR
                    result = a ^ b;
                default:
                    //result="INVALID";
                    break;
            }
            out.write(result);
    }

```

Header file:

alu.h | alu_tb2.cpp | input.dat | Synthesis Summary(solution1)

```
#include <hls_stream.h>
#include <ap_axi_sdata.h>
#include "ap_int.h"
typedef ap_int<8> dina;
typedef ap_int<8> dinb;
typedef ap_uint<3> dinop;
typedef ap_int<16> dout;
//typedef ap_axis<8, 1, 1, 1> AXI_VALUE;

void integer_alu(
    hls::stream<dina> &in1,
    hls::stream<dinb> &in2,
    hls::stream<dinop> &op,
    hls::stream<dout> &out
);

```

Test Bench:

alu.h | input.dat | Synthesis Summary(solution1) | Co-simulation Report(solution1) | integer_alu_csim.log | alu.cpp | alu_tb2.cpp

```
#include "alu.h"
int main()
{
    hls::stream<dina> a;
    hls::stream<dinb> b;
    hls::stream<dinop> op;
    hls::stream<dout> output;

    int i=0,j=0,flag=0;
    FILE *fp;
    dina test_inputs[10][2];
    dinop test_opcode[10];
    dout golden_data[10];
    dout out;
    // open the input file to read the data
    fp=fopen("input.dat","r");
     if (fp == NULL) {
       printf("Error opening file\n");
       return 1;
      }
    while (fscanf(fp,"%d %d %d %d",&test_inputs[i][0],&test_inputs[i][1],&test_opcode[i],&golden_data[i]) == 4) {
      i++;
      }
    fclose(fp);

    fp=fopen("output.dat","w");
    for ( j = 0; j < 10; j++)
    {
       a.write(test_inputs[j][0]);
       b.write(test_inputs[j][1]);
       op.write(test_opcode[j]);
       integer_alu(a,b,op,output);
       output >> out;

    if(out==golden_data[j]){
     fprintf(fp,"%d %s\n",out,"Test passed");
     }

   else {
    fprintf(fp,"%d %s\n",out,"Test Failed");
    flag=1;
   }
   }
     fclose(fp);

    if(flag==0){
      std::cout<<"all the read input sets passed"<<std::endl;
     }
      else{
       std::cout<<"few read input sets were not passed"<<std::endl;
     }
}

```

Input File:

input.dat | alu.h | Synthesis Summary(solution1) | Co-simu

```
17 1 0 18
17 2 1 15
17 3 6 18
17 4 5 21
17 5 3 3
17 6 2 102
17 7 0 24
17 8 2 136
17 9 5 25
17 10 3 1
```

C simulation printed output:

If all the outputs of ALU matched with reference output:

input.dat | alu.h | Synthesis Summary(solution1) | Co-simulation Report(solution1) | integer_alu_csim.log

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/alu_tb2.cpp in debug mode
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/alu.cpp in debug mode
   Generating csim.exe
all the read input sets passed
INFO [HLS SIM]: The maximum depth reached by any hls::stream() instance in the design is 1
INFO: [SIM 1] CSim done with 0 errors.
INFO: [SIM 3] *************** CSIM finish ***************
```

Output file with status:

input.dat | alu.h | Synthesis Summary(solution1)

```
18 Test passed
15 Test passed
18 Test passed
21 Test passed
3 Test passed
102 Test passed
24 Test passed
136 Test passed
25 Test passed
1 Test passed
```

input.dat | alu.h | Synthesis Summary(solution1) | Co-simu

```
17 1 0 18
17 2 1 15
17 3 6 18
17 4 5 21
17 5 3 3
17 6 2 102
17 7 0 24
17 8 2 136
17 9 5 25
17 10 3 1
```

If all the outputs of ALU does not match with reference output:

input.dat | alu.h | Synthesis Summary(solution1) | Co-simulation Report(solution1) | alu.cpp | alu_tb2.c

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/alu_tb2.cpp in debug mode
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/alu.cpp in debug mode
   Generating csim.exe
few read input sets were not passed
INFO [HLS SIM]: The maximum depth reached by any hls::stream() instance in the design is 1
INFO: [SIM 1] CSim done with 0 errors.
INFO: [SIM 3] *************** CSIM finish ***************
```

Output file with status:

If reference output is modified for testing

input.dat | alu.h | Synthesis Summary(solution1)

```
18 Test passed
15 Test passed
18 Test passed
21 Test passed
3 Test passed
102 Test passed
24 Test passed
136 Test passed
25 Test passed
1 Test Failed
```

input.dat | alu.h | alu.c

```
17 1 0 18
17 2 1 15
17 3 6 18
17 4 5 21
17 5 3 3
17 6 2 102
17 7 0 24
17 8 2 136
17 9 5 25
17 10 3 2
```

Synthesis Report:

input.dat | alu.h | Co-simulation Report(solution1) | alu.cpp | alu_tb2.cpp | integer_alu_csim.log | Synthesis Summary(solution1)

Synthesis Summary Report of 'integer_alu'

General Information

Date: Thu Mar 23 19:00:56 2023
Version: 2022.2 (Build 3670227 on Oct 13 2022)
Project: project_18
Solution: solution1 (Vivado IP Flow Target)
Product family: zynq
Target device: xc7z020-clg484-1

Timing Estimate

Target	Estimated	Uncertainty
10.00 ns	5.928 ns	2.70 ns

Performance & Resource Estimates

Modules Loops

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT	URAM
integer_alu				-	13	130.000	-	1	-	yes	0	0	655	439	0

Performance Pragma

Modules & Loops	Target TI(cycles)	TI(cycles)	TI met
integer_alu	-	-	-

HW Interfaces

AXIS

Interface	Register Mode	TDATA	TREADY	TVALID
in1	both	8	1	1
in2	both	8	1	1
op	both	8	1	1
out_r	both	16	1	1

TOP LEVEL CONTROL

Interface	Type	Ports
ap_clk	clock	ap_clk
ap_rst_n	reset	ap_rst_n
ap_ctrl	ap_ctrl_hs	ap_done ap_idle ap_ready ap_start

SW I/O Information

Top Function Arguments

Argument	Direction	Datatype
in1	in	stream<ap_int<8> 0>&
in2	in	stream<ap_int<8> 0>&
op	in	stream<ap_uint<3> 0>&
out	out	stream<ap_int<16> 0>&

SW-to-HW Mapping

Argument	HW Interface	HW Type
in1	in1	interface
in2	in2	interface
op	op	interface
out	out_r	interface

Pragma Report

Valid Pragma Syntax

Type	Options	Location	Function
interface	axis port=in1	../../../OneDrive/Desktop/vitis/alu.cpp:8	integer_alu
interface	axis port=in2	../../../OneDrive/Desktop/vitis/alu.cpp:9	integer_alu
interface	axis port=op	../../../OneDrive/Desktop/vitis/alu.cpp:10	integer_alu
interface	axis port=out	../../../OneDrive/Desktop/vitis/alu.cpp:11	integer_alu
pipeline	II=1	../../../OneDrive/Desktop/vitis/alu.cpp:13	integer_alu

Bind Op Report

No filter settings

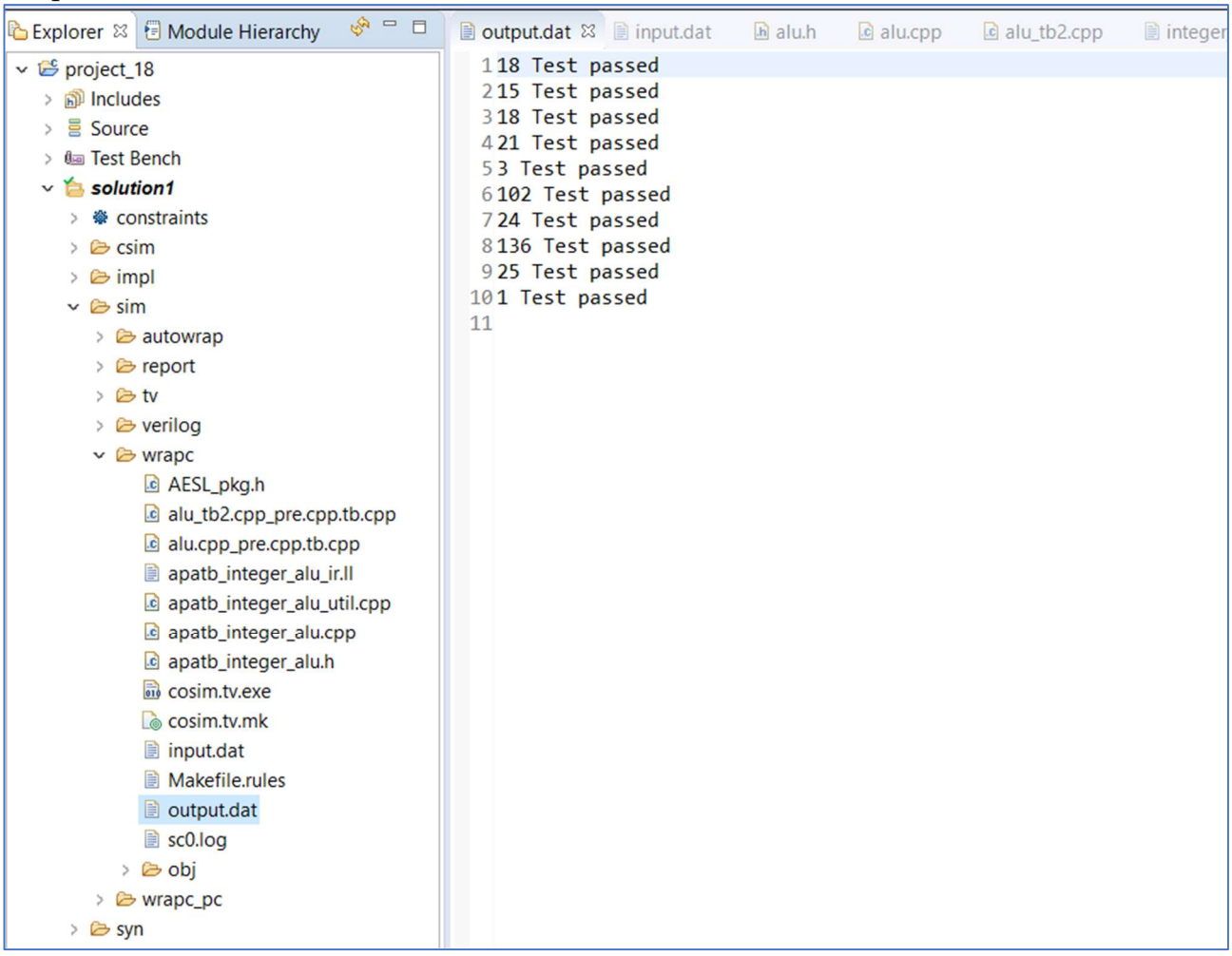
Name	DSP	Pragma	Variable	Op	Impl	Latency	Function	Scope
mul_8s_8s_16_1_1_U2	-		ret_V_2	mul	auto	0	integer_alu	
ret_V_1_fu_161_p2	-		ret_V_1	sub	fabric	0	integer_alu	
ret_V_fu_177_p2	-		ret_V	add	fabric	0	integer_alu	

Cosimulation printed output:

If all the outputs of ALU matched with reference output:

```
Built simulation snapshot integer_alu

****** xsim v2022.2 (64-bit)
  **** SW Build 3671981 on Fri Oct 14 05:00:03 MDT 2022
  **** IP Build 3669848 on Fri Oct 14 08:30:02 MDT 2022
    ** Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.

source xsim.dir/integer_alu/xsim_script.tcl
# xsim {integer_alu} -autoloadwcfg -tclbatch {integer_alu.tcl}
Time resolution is 1 ps
source integer_alu.tcl
## run all
////////////////////////////////////////////////////////////////////////////////
// Inter-Transaction Progress: Completed Transaction / Total Transaction
// Intra-Transaction Progress: Measured Latency / Latency Estimation * 100%
//
// RTL Simulation : "Inter-Transaction Progress" ["Intra-Transaction Progress"] @ "Simulation Time"
////////////////////////////////////////////////////////////////////////////////
// RTL Simulation : 0 / 10 [0.00%] @ "125000"
find kernel block.
// RTL Simulation : 1 / 10 [92.31%] @ "285000"
// RTL Simulation : 2 / 10 [92.31%] @ "295000"
// RTL Simulation : 3 / 10 [92.31%] @ "305000"
// RTL Simulation : 4 / 10 [92.31%] @ "315000"
// RTL Simulation : 5 / 10 [92.31%] @ "325000"
// RTL Simulation : 6 / 10 [92.31%] @ "335000"
// RTL Simulation : 7 / 10 [92.31%] @ "345000"
// RTL Simulation : 8 / 10 [92.31%] @ "355000"
// RTL Simulation : 9 / 10 [92.31%] @ "365000"
// RTL Simulation : 10 / 10 [100.00%] @ "375000"
////////////////////////////////////////////////////////////////////////////////
$finish called at time : 435 ns : File "C:/Users/velic/AppData/Roaming/Xilinx/Vitis/project_18/solution1/sim/verilog/integer_alu.autotb.v" Line 321
## quit
INFO: [Common 17-206] Exiting xsim at Thu Mar 23 19:04:43 2023...
INFO: [COSIM 212-316] Starting C post checking ...
all the read input sets passed
INFO [HLS SIM]: The maximum depth reached by any hls::stream() instance in the design is 1
INFO: [COSIM 212-1000] *** C/RTL co-simulation finished: PASS ***
INFO: [HLS 200-111] Finished Command cosim_design CPU user time: 1 seconds. CPU system time: 1 seconds. Elapsed time: 40.289 seconds; current allocated memory: 13.230 MB
INFO: [HLS 200-112] Total CPU user time: 3 seconds. Total CPU system time: 2 seconds. Total elapsed time: 52.804 seconds; peak allocated memory: 111.480 MB.
Finished C/RTL cosimulation.
```

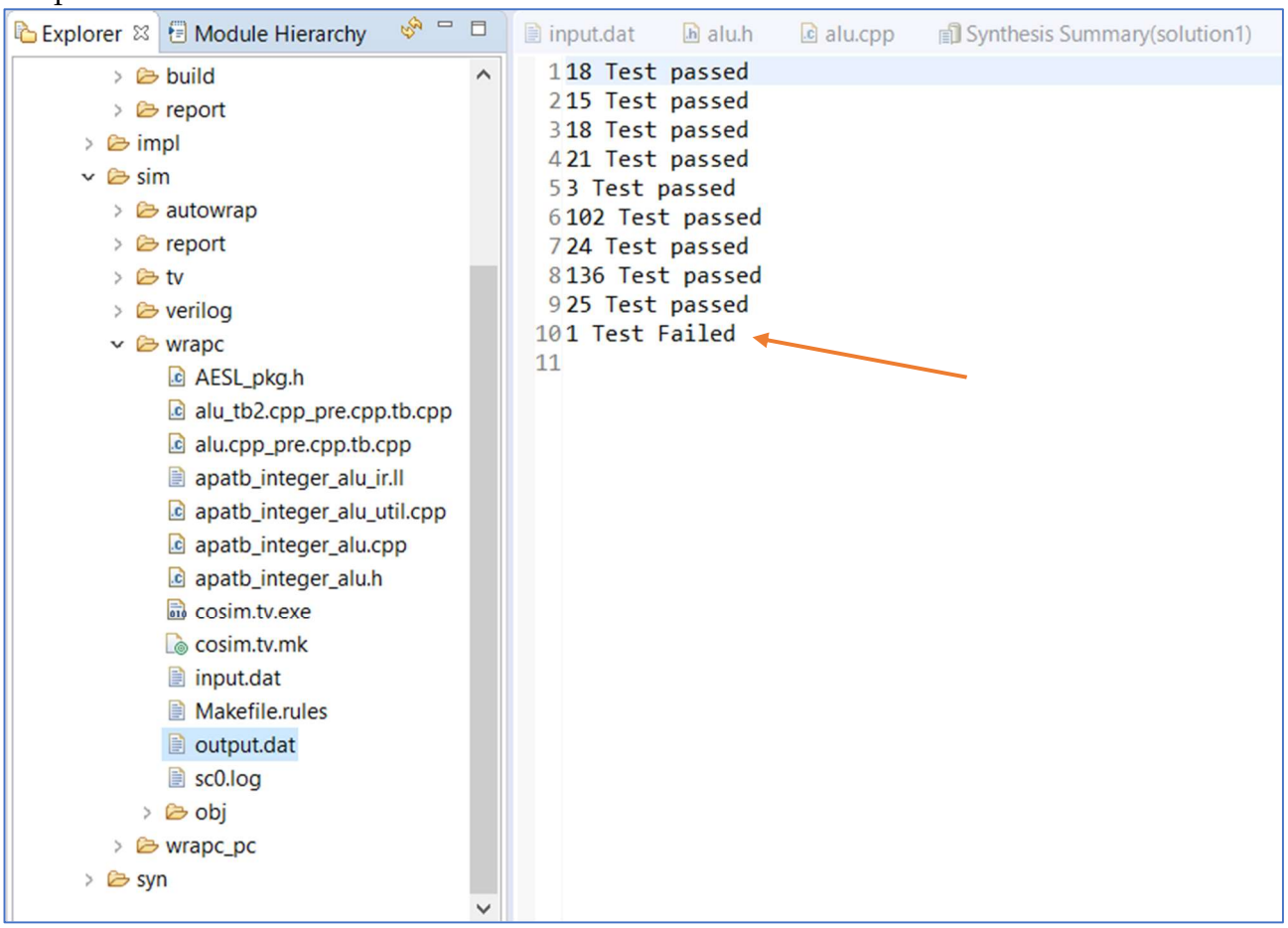
Output file with status:



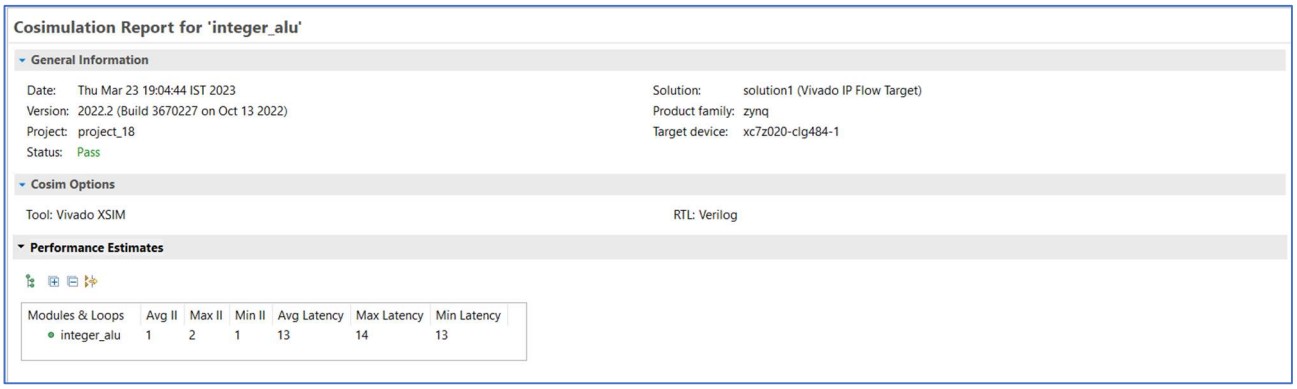
If all the outputs of ALU does not match with reference output:

```
Built simulation snapshot integer_alu

****** xsim v2022.2 (64-bit)
  **** SW Build 3671981 on Fri Oct 14 05:00:03 MDT 2022
  **** IP Build 3669848 on Fri Oct 14 08:30:02 MDT 2022
    ** Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.

source xsim.dir/integer_alu/xsim_script.tcl
# xsim {integer_alu} -autoloadwcfg -tclbatch {integer_alu.tcl}
Time resolution is 1 ps
source integer_alu.tcl
## run all
////////////////////////////////////////////////////////////////////////////////
// Inter-Transaction Progress: Completed Transaction / Total Transaction
// Intra-Transaction Progress: Measured Latency / Latency Estimation * 100%
//
// RTL Simulation : "Inter-Transaction Progress" ["Intra-Transaction Progress"] @ "Simulation Time"
////////////////////////////////////////////////////////////////////////////////
// RTL Simulation : 0 / 10 [0.00%] @ "125000"
find kernel block.
// RTL Simulation : 1 / 10 [92.31%] @ "285000"
// RTL Simulation : 2 / 10 [92.31%] @ "295000"
// RTL Simulation : 3 / 10 [92.31%] @ "305000"
// RTL Simulation : 4 / 10 [92.31%] @ "315000"
// RTL Simulation : 5 / 10 [92.31%] @ "325000"
// RTL Simulation : 6 / 10 [92.31%] @ "335000"
// RTL Simulation : 7 / 10 [92.31%] @ "345000"
// RTL Simulation : 8 / 10 [92.31%] @ "355000"
// RTL Simulation : 9 / 10 [92.31%] @ "365000"
// RTL Simulation : 10 / 10 [100.00%] @ "375000"
////////////////////////////////////////////////////////////////////////////////
$finish called at time : 435 ns : File "C:/Users/velic/AppData/Roaming/Xilinx/Vitis/project_18/solution1/sim/verilog/integer_alu.autotb.v" Line 321
## quit
INFO: [Common 17-206] Exiting xsim at Thu Mar 23 19:13:31 2023...
INFO: [COSIM 212-316] Starting C post checking ...
few read input sets were not passed
INFO [HLS SIM]: The maximum depth reached by any hls::stream() instance in the design is 1
INFO: [COSIM 212-1000] *** C/RTL co-simulation finished: PASS ***
INFO: [HLS 200-111] Finished Command cosim_design CPU user time: 2 seconds. CPU system time: 1 seconds. Elapsed time: 39.791 seconds; current allocated memory: 11.949 MB
INFO: [HLS 200-112] Total CPU user time: 4 seconds. Total CPU system time: 2 seconds. Total elapsed time: 52.27 seconds; peak allocated memory: 110.066 MB.
Finished C/RTL cosimulation.
```

Output file with status:



Cosimulation Report:



Cosimulation Report for 'integer_alu'

General Information

Date: Thu Mar 23 19:04:44 IST 2023
Version: 2022.2 (Build 3670227 on Oct 13 2022)
Project: project_18
Status: Pass

Solution: solution1 (Vivado IP Flow Target)
Product family: zynq
Target device: xc7z020-clg484-1

Cosim Options

Tool: Vivado XSIM
RTL: Verilog

Performance Estimates

Modules & Loops	Avg II	Max II	Min II	Avg Latency	Max Latency	Min Latency
integer_alu	1	2	1	13	14	13

Github:

https://github.com/velicharlagokulkumar/vitis-hls/tree/main/project_18