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

Date:21-03-2023

Hls ASSIGNMENT-3

Repeat the experiment in Assignment 2.3 but by configuring the module as pipelined.

1.

Design:

Multiplier function:

hls_pipe1.cpp | _csim.log | Synthesis Summary(solution1) | Co-simulation Report(solution1)

```
#include"hls_pipe1.h"
void multiply(hls::stream<dinA_t> &A, hls::stream<dinB_t>& B,hls::stream<doutC_t> &C)
{
    #pragma HLS INTERFACE axis port=A
    #pragma HLS INTERFACE axis port=B
    #pragma HLS INTERFACE axis port=C
    #pragma HLS INTERFACE ap_ctrl_none port=return
    #pragma HLS PIPELINE II=1

    dinA_t a;
    dinB_t b;
    doutC_t result;
    a=A.read();
    b=B.read();
    result=a*b;
    C.write(result);
}
```

Header file:

hls_pipe1.cpp | _csim.log | Synthesis Summary(solution1) | Co-simulation Report(solution1) | hls_

```
#include<stdio.h>
#include<hls_stream.h>
#include "ap_int.h"
typedef int dinA_t;
typedef int dinB_t;
typedef long long doutC_t;
void multiply(hls::stream<dinA_t>&A,hls::stream<dinB_t>&B,hls::stream<doutC_t>&C);

```

Test Bench:

hls_pipe1.cpp | _csim.log | Synthesis Summary(solution1) | Co-simulation Report(solution1) | hls_pipe1.h

```
#include"hls_pipe1.h"
int main()
{
    hls::stream<dinA_t> a;
    hls::stream<dinB_t> b;
    hls::stream<doutC_t> c;
    dinA_t test_inputs[10][2] = {
             {0x00000011,0x00000001},
             {0x00000011,0x00000002},
             {0x00000011,0x00000003},
             {0x00000011,0x00000004},
             {0x00000011,0x00000005},
             {0x00000011,0x00000006},
             {0x00000011,0x00000007},
             {0x00000011,0x00000008},
             {0x00000011,0x00000009},
             {0x00000011,0x0000000A},
       };
      for (int i = 0; i < 10; i++)
       {
           a.write(test_inputs[i][0]);
           b.write(test_inputs[i][1]);
           multiply(a,b,c);
           //out >> result;
         std::cout<<  test_inputs[i][0] << "*"<< test_inputs[i][1] << "="<<c.read()<<std::endl;
       }
}

```

C simulation printed output:

hls_pipe1.cpp | _csim.log | Synthesis Summary(solution1) | Co-simulation Report(solution1) | hls_pipe1.h

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/hls_pipe1_tb.cpp in debug mode
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp in debug mode
   Generating csim.exe
17*1=17
17*2=34
17*3=51
17*4=68
17*5=85
17*6=102
17*7=119
17*8=136
17*9=153
17*10=170
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 ***************

```

Synthesis Report:

Synthesis Summary Report of 'multiply'

General Information

Date:	Mon Mar 20 12:57:19 2023	Solution:	solution1 (Vivado IP Flow Target)
Version:	2022.2 (Build 3670227 on Oct 13 2022)	Product family:	zynq
Project:	project_14	Target device:	xc7z020-clg484-1

Timing Estimate

Target	Estimated	Uncertainty
10.00 ns	6.912 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
multiply				-	2	20.000	-	1	-	yes	0	3	168	85	0

Performance Pragma

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

HW Interfaces

AXIS

Interface	Register Mode	TDATA	TREADY	TVALID
A	both	32	1	1
B	both	32	1	1
C	both	64	1	1

TOP LEVEL CONTROL

Interface	Type	Ports
ap_clk	clock	ap_clk
ap_rst_n	reset	ap_rst_n
ap_ctrl	ap_ctrl_none	

SW I/O Information

Top Function Arguments

Argument	Direction	Datatype
A	in	stream<int 0>&
B	in	stream<int 0>&
C	out	stream<long long 0>&

SW-to-HW Mapping

Argument	HW Interface	HW Type
A	A	interface
B	B	interface
C	C	interface

Pragma Report

Valid Pragma Syntax

Type	Options	Location	Function
interface	axis port=A	../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp:4	multiply
interface	axis port=B	../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp:5	multiply
interface	axis port=C	../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp:6	multiply
interface	ap_ctrl_none port=return	../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp:7	multiply
pipeline	II=1	../../../../OneDrive/Desktop/vitis/hls_pipe1.cpp:8	multiply

Bind Op Report

No filter settings

Name	DSP	Pragma	Variable	Op	Impl	Latency
multiply	3					

Cosimulation printed output:

```
////////////////////////////////////////////////////////////////////////////////
// Following axis ports can not be accessed by kernel 'multiply'
//    port 'A' can not be read
//    port 'B' can not be read
//    port 'C' can not be written
//
//    Because the top pipeline FSM is blocked by following axis ports
////////////////////////////////////////////////////////////////////////////////

// RTL Simulation : 1 / 10 [100.00%] @ "175000"
// RTL Simulation : 2 / 10 [100.00%] @ "185000"
// RTL Simulation : 3 / 10 [100.00%] @ "195000"
// RTL Simulation : 4 / 10 [100.00%] @ "205000"
// RTL Simulation : 5 / 10 [100.00%] @ "215000"
// RTL Simulation : 6 / 10 [100.00%] @ "225000"
// RTL Simulation : 7 / 10 [100.00%] @ "235000"
// RTL Simulation : 8 / 10 [100.00%] @ "245000"
// RTL Simulation : 9 / 10 [100.00%] @ "255000"
// RTL Simulation : 10 / 10 [100.00%] @ "265000"
////////////////////////////////////////////////////////////////////////////////
$finish called at time : 325 ns : File "C:/Users/velic/AppData/Roaming/Xilinx/Vitis/project_14/solution1/sim/verilog/multiply.autotb.v" Line 224
## quit
INFO: [Common 17-206] Exiting xsim at Mon Mar 20 12:59:14 2023...
INFO: [COSIM 212-316] Starting C post checking ...
17*1=17
17*2=34
17*3=51
17*4=68
17*5=85
17*6=102
17*7=119
17*8=136
17*9=153
17*10=170
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: 55.685 seconds; current allocated memory: 12.301 MB
INFO: [HLS 200-112] Total CPU user time: 3 seconds. Total CPU system time: 2 seconds. Total elapsed time: 68.19 seconds; peak allocated memory: 108.934 MB.
Finished C/RTL cosimulation.
```

Cosimulation Report:

Cosimulation Report for 'multiply'

General Information

Date:	Mon Mar 20 12:59:20 IST 2023	Solution:	solution1 (Vivado IP Flow Target)
Version:	2022.2 (Build 3670227 on Oct 13 2022)	Product family:	zynq
Project:	project_14	Target device:	xc7z020-clg484-1
Status:	Pass		

Cosim Options

Tool: Vivado XSIM

RTL: Verilog

Performance Estimates

Modules & Loops	Avg II	Max II	Min II	Avg Latency	Max Latency	Min Latency
multiply						

2.
Design:

Multiplier function:

Synthesis Summary(solution1) | hls_pipe2.cpp | hls_pipe2_tb.cpp | hls_pipe2.h | _csim.log

```
#include"hls_stream2.h"
void multiply(hls::stream<dinA_t> &A, hls::stream<dinB_t>& B,hls::stream<doutC_t> &C)
{
    #pragma HLS INTERFACE axis port=A
    #pragma HLS INTERFACE axis port=B
    #pragma HLS INTERFACE axis port=C
    #pragma HLS INTERFACE ap_ctrl_none port=return
    #pragma HLS PIPELINE II=1
    dinA_t a;
    dinB_t b;
    doutC_t result;
    a=A.read();
    b=B.read();
    result=a*b;
    C.write(result);
}

```

Header file:

Synthesis Summary(solution1) | hls_pipe2.cpp | hls_pipe2_tb.cpp | hls_pipe2.h | _csim.log

```
#include<stdio.h>
#include<ap_fixed.h>
#include<hls_stream.h>
typedef ap_fixed <28,4> dinA_t;
typedef ap_fixed <28,4> dinB_t;
typedef ap_fixed <56,8> doutC_t;
void multiply(hls::stream<dinA_t>&A,hls::stream<dinB_t>&B,hls::stream<doutC_t>&C);

```

Test Bench:

Synthesis Summary(solution1) | hls_pipe2.cpp | hls_pipe2_tb.cpp | hls_pipe2.h | _csim.log

```
#include"hls_stream2.h"
int main()
{
    hls::stream<dinA_t> a;
    hls::stream<dinB_t> b;
    hls::stream<doutC_t> c;
    dinA_t test_inputs[10][2] = {
             {4.125,3.264589},
             {4.56215,2.4568912},
             {3.69845632,2.456891},
             {3.69845632,2.456891},
             {1.25698456,3.569874},
             {1.4567896512,5.623366},
             {5.6684258,6.54566321},
             {6.2255620,5.636115},
             {4.6985230,2.5698523},
             {1.563995,3.2656343}
       };
      for (int i = 0; i < 10; i++)
       {
           a.write(test_inputs[i][0]);
           b.write(test_inputs[i][1]);
           multiply(a,b,c);
           //out >> result;
         std::cout<<  test_inputs[i][0] << "*"<< test_inputs[i][1] << "="<<c.read()<<std::endl;
       }
}
```

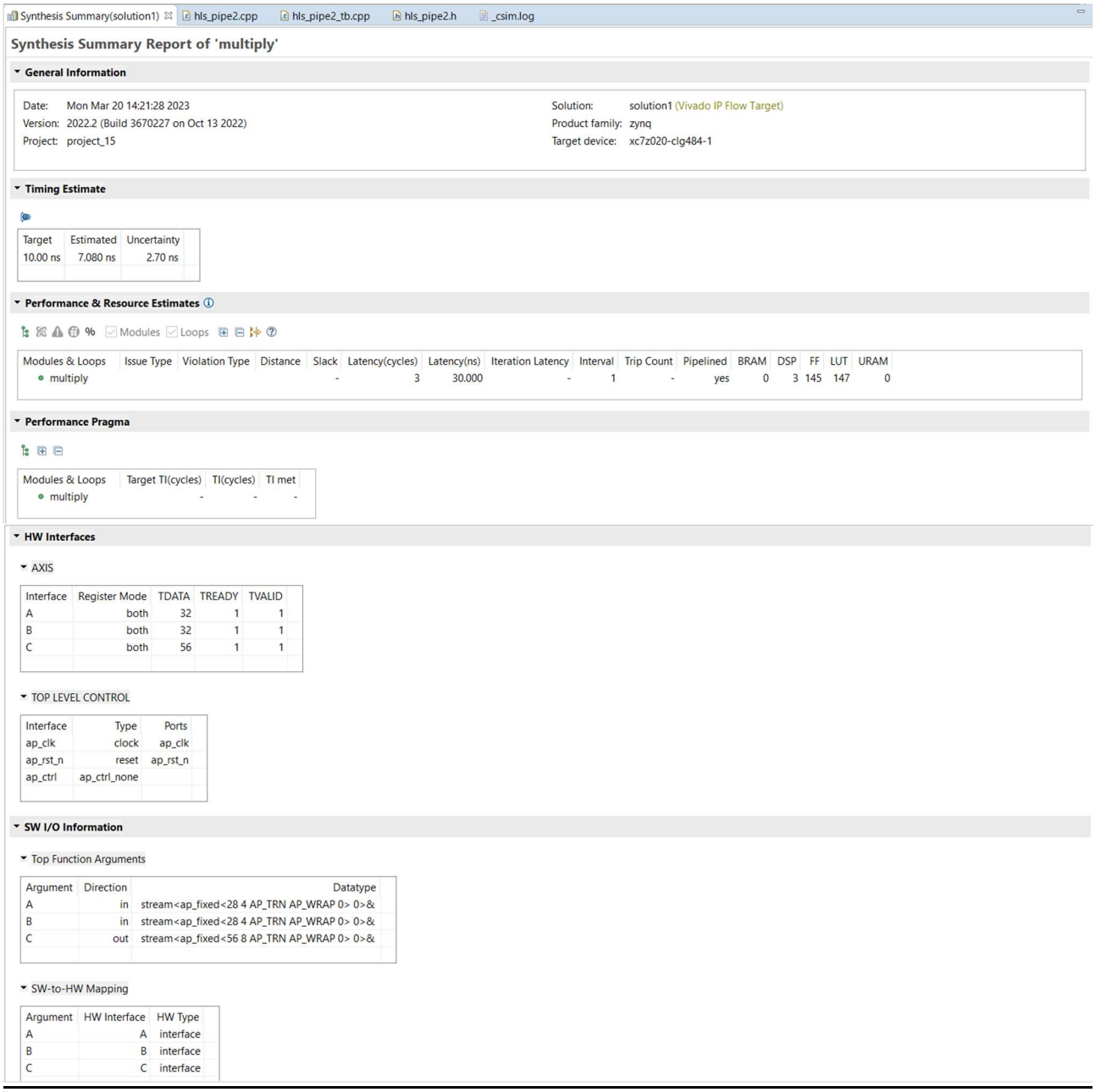
C simulation printed output:

Synthesis Summary(solution1) | hls_pipe2.cpp | hls_pipe2_tb.cpp | hls_pipe2.h | _csim.log

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/hls_pipe2_tb.cpp in debug mode
   Compiling ../../../../../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp in debug mode
   Generating csim.exe
4.125*3.26459=13.4664
4.56215*2.45689=11.2087
3.69846*2.45689=9.0867
3.69846*2.45689=9.0867
1.25698*3.56987=4.48728
1.45679*5.62337=8.19206
5.66843*6.54566=37.1036
6.22556*5.63611=35.088
4.69852*2.56985=12.0745
1.56399*3.26563=5.10744
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 ***************

```

Synthesis Report:



Synthesis Summary(solution1) | hls_pipe2.cpp | hls_pipe2_tb.cpp | hls_pipe2.h | _csim.log

Synthesis Summary Report of 'multiply'

General Information

Date: Mon Mar 20 14:21:28 2023
Version: 2022.2 (Build 3670227 on Oct 13 2022)
Project: project_15
Solution: solution1 (Vivado IP Flow Target)
Product family: zynq
Target device: xc7z020-clg484-1

Timing Estimate

Target	Estimated	Uncertainty
10.00 ns	7.080 ns	2.70 ns

Performance & Resource Estimates

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT	URAM
multiply				-	3	30.000	-	1	-	yes	0	3	145	147	0

Performance Pragma

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

HW Interfaces

AXIS

Interface	Register Mode	TDATA	TREADY	TVALID
A	both	32	1	1
B	both	32	1	1
C	both	56	1	1

TOP LEVEL CONTROL

Interface	Type	Ports
ap_clk	clock	ap_clk
ap_rst_n	reset	ap_rst_n
ap_ctrl	ap_ctrl_none	

SW I/O Information

Top Function Arguments

Argument	Direction	Datatype
A	in	stream<ap_fixed<28 4 AP_TRN AP_WRAP 0> 0>&
B	in	stream<ap_fixed<28 4 AP_TRN AP_WRAP 0> 0>&
C	out	stream<ap_fixed<56 8 AP_TRN AP_WRAP 0> 0>&

SW-to-HW Mapping

Argument	HW Interface	HW Type
A	A	interface
B	B	interface
C	C	interface

Pragma Report

Valid Pragma Syntax

Type	Options	Location	Function
interface	axis port=A	../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp:4	multiply
interface	axis port=B	../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp:5	multiply
interface	axis port=C	../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp:6	multiply
interface	ap_ctrl_none port=return	../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp:7	multiply
pipeline	II=1	../../../../OneDrive/Desktop/vitis/hls_pipe2.cpp:8	multiply

Bind Op Report

No filter settings

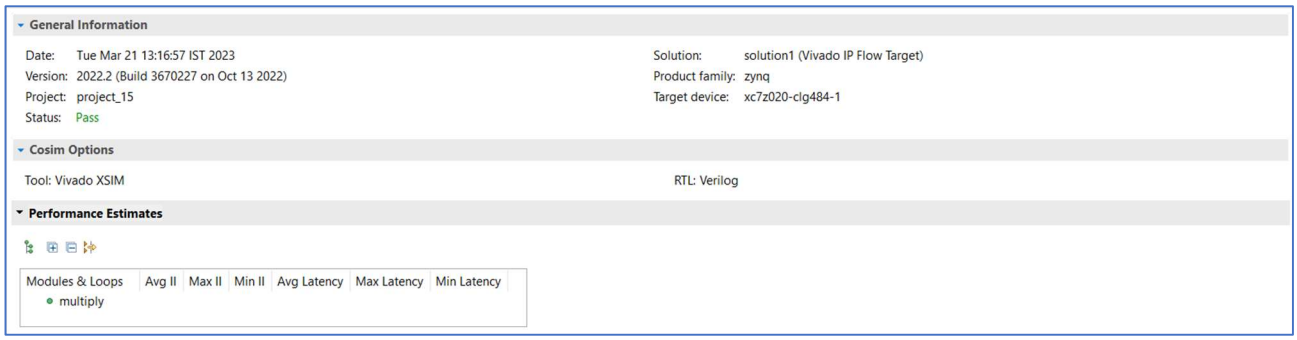
Name	DSP	Pragma	Variable	Op	Impl	Latency
multiply	3					

Cosimulation printed output:

```
////////////////////////////////////////////////////////////////////////////////
// Following axis ports can not be accessed by kernel 'multiply'
//    port 'A' can not be read
//    port 'B' can not be read
//    port 'C' can not be written
//
//    Because the top pipeline FSM is blocked by following axis ports
////////////////////////////////////////////////////////////////////////////////

// RTL Simulation : 1 / 10 [100.00%] @ "185000"
// RTL Simulation : 2 / 10 [100.00%] @ "195000"
// RTL Simulation : 3 / 10 [100.00%] @ "205000"
// RTL Simulation : 4 / 10 [100.00%] @ "215000"
// RTL Simulation : 5 / 10 [100.00%] @ "225000"
// RTL Simulation : 6 / 10 [100.00%] @ "235000"
// RTL Simulation : 7 / 10 [100.00%] @ "245000"
// RTL Simulation : 8 / 10 [100.00%] @ "255000"
// RTL Simulation : 9 / 10 [100.00%] @ "265000"
// RTL Simulation : 10 / 10 [100.00%] @ "275000"
////////////////////////////////////////////////////////////////////////////////
$finish called at time : 335 ns : File "C:/Users/velic/AppData/Roaming/Xilinx/Vitis/project_15/solution1/sim/verilog/multiply.autotb.v" Line 224
## quit
INFO: [Common 17-206] Exiting xsim at Tue Mar 21 13:16:50 2023...
INFO: [COSIM 212-316] Starting C post checking ...
4.125*3.26459=13.4664
4.56215*2.45689=11.2087
3.69846*2.45689=9.0867
1.25698*3.56987=4.48728
1.45679*5.62337=8.19206
5.66843*6.54566=37.1036
6.22556*5.63611=35.088
4.69852*2.56985=12.0745
1.56399*3.26563=5.10744
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: 95.793 seconds; current allocated memory: 11.770 MB
INFO: [HLS 200-112] Total CPU user time: 3 seconds. Total CPU system time: 2 seconds. Total elapsed time: 108.324 seconds; peak allocated memory: 109.242 MB.
Finished C/RTL cosimulation.
```

Cosimulation Report:



- Pipelining helps in reduce the amount of hardware resources required for implementing a stream interface. By breaking the processing into smaller stages
- pipelining introduces additional latency, as data needs to pass through multiple stages before being processed
- Using a pipelined, blocking stream interface can help to improve the performance of the design by enabling higher throughput and reducing the latency of communication.

Github:

https://github.com/velicharlagokulkumar/vitis-hls/tree/main/project_14
https://github.com/velicharlagokulkumar/vitis-hls/tree/main/project_15