ELEC 522 Assignment 4

Student: Sixu Li (S01435077)

1. Write C/C++ code for the CORDIC circular mode module using VivadoHLS

The CORDIC core is designed with two modes. Hardware resources are reused as a reconfigurable design.

First is the reconfigurable design in I/O;

```
8     if (!select) { // select == 0: sin cos mode
9         x = 0.60735;
10        y = 0;
11        t = CS_I_T;
12    } else { // select == 1; arc tan mode
13        x = IA_I_C;
14        y = IA_I_S;
15        t = 0;
16    }
```

```
35    if (!select) { // select == 0: sin cos mode
36        CS_O_SIN = y;
37        CS_O_COS = x;
38        IA_O_IAT = 0;
39    } else { // select == 1; arc tan mode
40        CS_O_SIN = 0;
41        CS_O_COS = 0;
42        IA_O_IAT = t;
43    }
```

Apart from the reuse in I/O. The main reconfigurable part is the sigma, which will change in different modes.

```
19        int sigma;
20        if (!select) { // select == 0: sin cos mode
21            sigma = (t < 0) ? -1 : 1;
22        } else { // select == 1; arc tan mode
23            sigma = (y < 0) ? 1: -1;
24        }
```

Other computation units are all reused and will not change when in different modes.

```
27        FIXDT x_shift = (x * sigma) >> itr;
28        FIXDT y_shift = (y * sigma) >> itr;
29
30        x = x - y_shift;
31        y = y + x_shift;
32
33        t = t - cordic_phase[itr] * sigma;
```

In this way, the proposed module should consume less hardware resources compared to design two modules separately.

2. Write testbench code to verify the function of the C/C++ code code in VivadoHLS

In the C testbench, we choose $\theta = 30°$ to verify the cos and sin mode. We choose $\theta = 60°$ ($sin = \frac{1}{\sqrt{3}}, cos = 0.5$). The results matched but with some small errors.

```
6 result sin = 0.498718, cos = 0.867004
7 result theta = 60.513073
```

3. Architecture optimization, constraint configurations and advantages of final design.

Vivado HLS has automatically enabled pipeline for the FOR-LOOP. So the default design is optimal considering this FOR-LOOP is the only potential optimization oprtator.

Timing Estimate

Target	Estimated	Uncertainty
10.00 ns	5.630 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
cordic	II Violation			-	34	340.000	-	35	-	no	0	1	184	601	0
VITIS_LOOP_17_1	II Violation	Memory Dependency	1	-	32	320.000	5	3	10	yes	-	-	-	-	-

But we can disable auto pipeline by insert #pragma HLS pipeline off. We can see that if we turn off pileine , we can save less than 10% area but get 60% more latency. So we choose the one with pipeline.

Timing Estimate

Target	Estimated	Uncertainty
10.00 ns	6.573 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
cordic	II Violation			-	51	510.000	-	52	-	no	0	1	160	567	0
VITIS_LOOP_27_1	II Violation	Memory Dependency	1	-	50	500.000	5	-	10	no	-	-	-	-	-

4. Model Composer model using the VivadoHLS block and testing results.

See screenshots in Question 6.

5. Synthesis and place and route implementation report.

The following screenshots are the timing & utilization reports post-synthesis & post-PR.

post-synthesis

Design Timing Summary

Setup		Hold	
Worst Negative Slack (WNS):	5.569 ns	Worst Hold Slack (WHS):	-2.281 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	-59.183 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	72
Total Number of Endpoints:	2933	Total Number of Endpoints:	2833

Timing constraints are not met.

Name	Slice LUTs (53200)	Slice Registers (106400)	Block RAM Tile (140)	DSPs (220)	Bonded IOB (200)	BUFGCTRL (32)	MMCME2_ADV (4)	BSCANE2 (4)
N hwcosim_top_wrapper	1023	1527	2	1	1	3	1	1
hwcosim_top_i (hwcosim_top)	1023	1527	2	1	0	3	1	1
axis_data_fifo_rx (hwcosim_top_axis_data_fifo_rx_0)	134	344	1	0	0	0	0	0
axis_data_fifo_tx (hwcosim_top_axis_data_fifo_tx_0)	134	344	1	0	0	0	0	0
axis_dwidth_converter_rx (hwcosim_top_axis_dwidth_converter_rx_0)	20	53	0	0	0	0	0	0
axis_dwidth_converter_tx (hwcosim_top_axis_dwidth_converter_tx_0)	28	50	0	0	0	0	0	0
hls_0 (hwcosim_top_hls_0_0)	461	451	0	1	0	1	0	0
hwc_jtag_axi_transport_0 (hwcosim_top_hwc_jtag_axi_transport_0_0)	64	132	0	0	0	0	0	1
hwcosim_cmd_proc (hwcosim_top_hwcosim_cmd_proc_0)	163	113	0	0	0	0	0	0
reset_gen (reset_gen_imp_1WK4TX6)	19	40	0	0	0	0	0	0
sys_clk_wiz (hwcosim_top_sys_clk_wiz_0)	0	0	0	0	0	2	1	0

post-PR

Design Timing Summary

Setup		Hold	
Worst Negative Slack (WNS):	1.547 ns	Worst Hold Slack (WHS):	0.019 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	2644	Total Number of Endpoints:	2596

All user specified timing constraints are met.

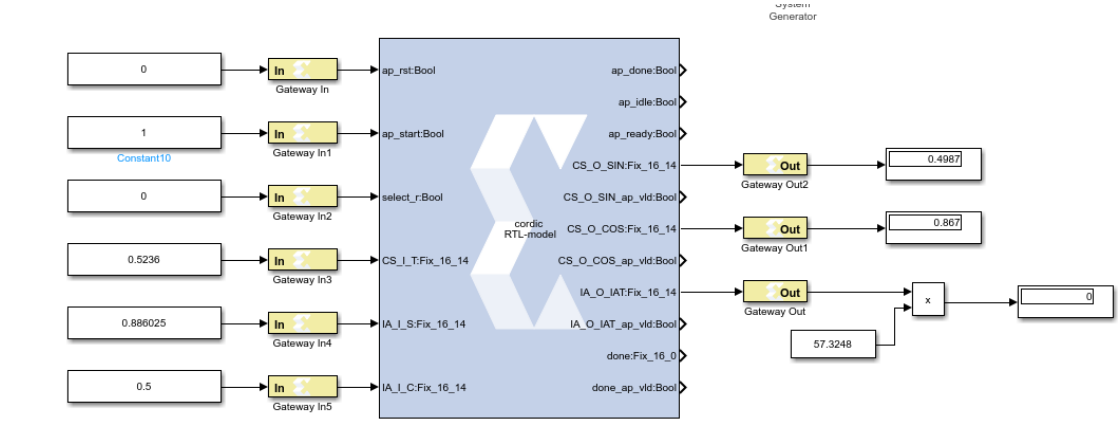
4

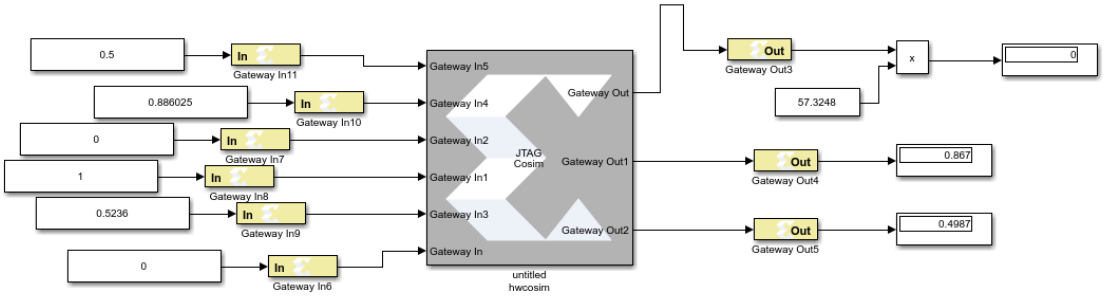
Name	Slice LUTs (53200)	Slice Registers (106400)	Slice (13300)	LUT as Logic (53200)	LUT as Memory (17400)	Block RAM Tile (140)	DSPs (220)	Bonded IOB (200)	BUFGCTRL (32)	MMCME2_ADV (4)	BSCANE2 (4)
N hwcosim_top_wrapper	866	1248	358	865	1	2	1	1	4	1	1
hwcosim_top_i (hwcosim_top)	866	1248	358	865	1	2	1	0	4	1	1
axis_data_fifo_rx (hwcosim_top_axis_data_fifo_rx_0)	79	210	52	79	0	1	0	0	0	0	0
axis_data_fifo_tx (hwcosim_top_axis_data_fifo_tx_0)	79	210	53	79	0	1	0	0	0	0	0
axis_dwidth_converter_rx (hwcosim_top_axis_dwidth_converter_rx_0)	15	50	13	15	0	0	0	0	0	0	0
axis_dwidth_converter_tx (hwcosim_top_axis_dwidth_converter_tx_0)	27	50	15	27	0	0	0	0	0	0	0
hls_0 (hwcosim_top_hls_0_0)	434	450	151	434	0	0	1	0	1	0	0
hwc_jtag_axi_transport_0 (hwcosim_top_hwc_jtag_axi_transport_0_0)	58	132	33	58	0	0	0	0	1	0	1
hwcosim_cmd_proc (hwcosim_top_hwcosim_cmd_proc_0)	163	113	61	163	0	0	0	0	0	0	0
reset_gen (reset_gen_imp_1WK4TX6)	16	33	13	15	1	0	0	0	0	0	0
sys_clk_wiz (hwcosim_top_sys_clk_wiz_0)	0	0	0	0	0	0	0	0	2	1	0

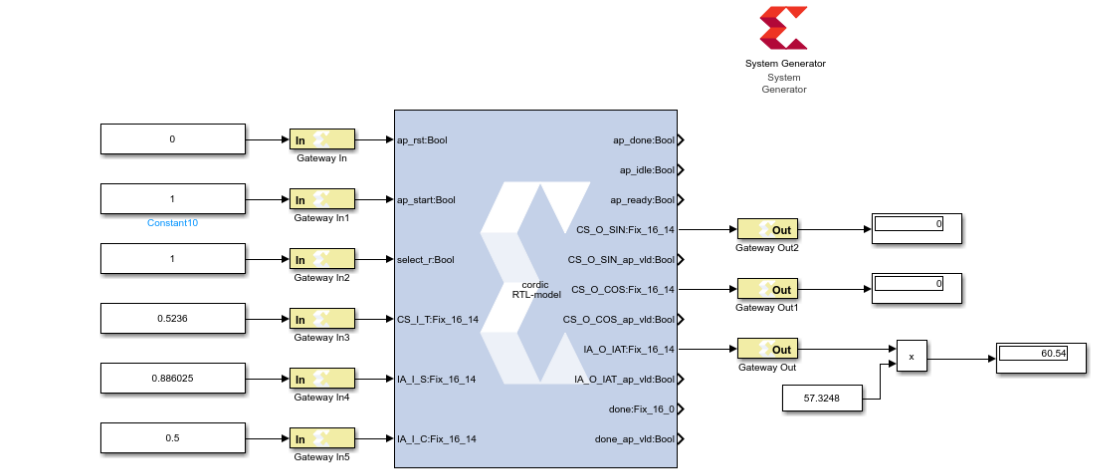
6. Hardware in the loop Co-Simulation to verify performance of VivadoHLS code on the ZedBoard in the lab.

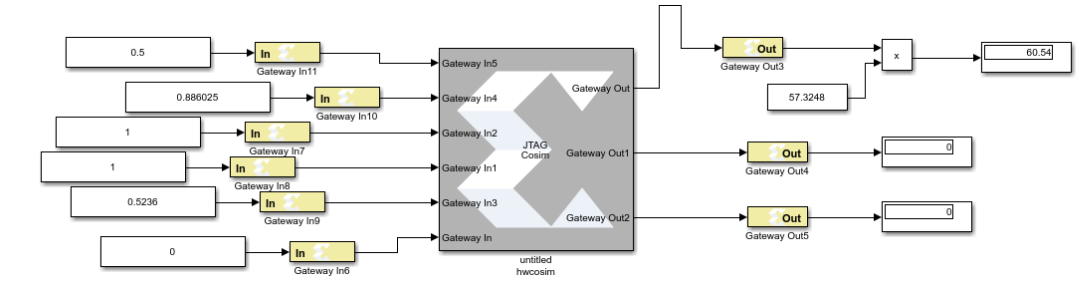
The following figure are the test results for mode 0 (sin&cos) and mode 1 (arctan).

We can see that the Simulink result have some errors compared to HLS C simulation, but within 0.05% which can be ignored.


System Generator
0
In
Gateway In
ap_rst:Bool
ap_done:Bool
ap_idle:Bool
1
Constant10
In
Gateway In1
ap_start:Bool
ap_ready:Bool
0
In
Gateway In2
select_r:Bool
CS_O_SIN:Fix_16_14
CS_O_SIN_ap_vld:Bool
cordic
RTL-model
CS_O_COS:Fix_16_14
0.5236
In
Gateway In3
CS_I_T:Fix_16_14
CS_O_COS_ap_vld:Bool
IA_O_IAT:Fix_16_14
0.886025
In
Gateway In4
IA_I_S:Fix_16_14
IA_O_IAT_ap_vld:Bool
done:Fix_16_0
0.5
In
Gateway In5
IA_I_C:Fix_16_14
done_ap_vld:Bool
Out
Gateway Out2
0.4987
Out
Gateway Out1
0.867
Out
Gateway Out
x
0
57.3248

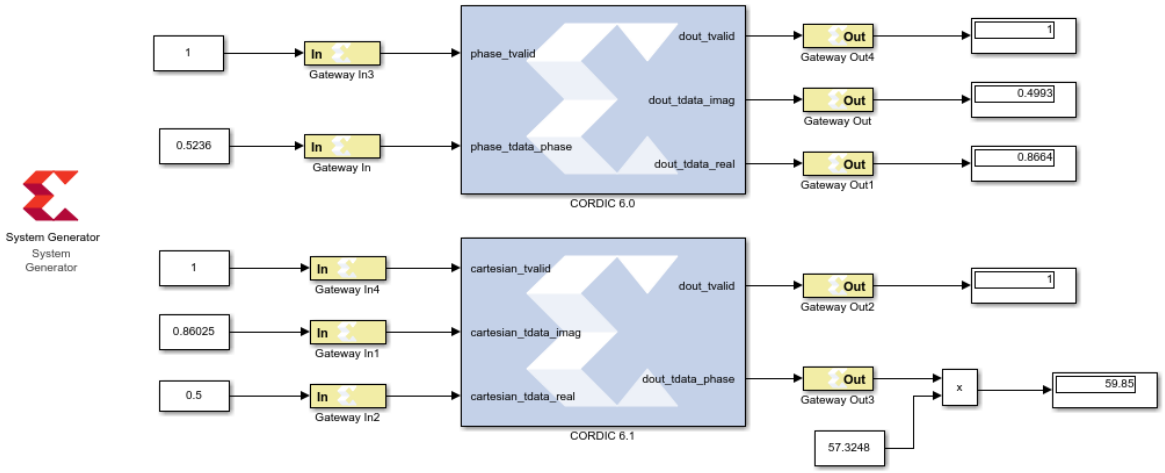

0.5
In
Gateway In11
Gateway In5
Gateway Out
Out
Gateway Out3
x
0
0.886025
In
Gateway In10
Gateway In4
57.3248
0
In
Gateway In7
Gateway In2
JTAG
Cosim
1
In
Gateway In8
Gateway In1
Gateway Out1
Out
Gateway Out4
0.867
0.5236
In
Gateway In9
Gateway In3
Gateway Out2
Out
Gateway Out5
0.4987
0
In
Gateway In6
Gateway In
untitled
hwcosim


System Generator
System
Generator
0
In
Gateway In
ap_rst:Bool
ap_done:Bool
ap_idle:Bool
1
Constant10
In
Gateway In1
ap_start:Bool
ap_ready:Bool
1
In
Gateway In2
select_r:Bool
CS_O_SIN:Fix_16_14
CS_O_SIN_ap_vld:Bool
cordic
RTL-model
CS_O_COS:Fix_16_14
0.5236
In
Gateway In3
CS_I_T:Fix_16_14
CS_O_COS_ap_vld:Bool
IA_O_IAT:Fix_16_14
0.886025
In
Gateway In4
IA_I_S:Fix_16_14
IA_O_IAT_ap_vld:Bool
done:Fix_16_0
0.5
In
Gateway In5
IA_I_C:Fix_16_14
done_ap_vld:Bool
Out
Gateway Out2
0
Out
Gateway Out1
0
Out
Gateway Out
x
60.54
57.3248


0.5
In
Gateway In11
Gateway In5
Gateway Out
Out
Gateway Out3
x
60.54
0.886025
In
Gateway In10
Gateway In4
57.3248
1
In
Gateway In7
Gateway In2
JTAG
Cosim
1
In
Gateway In8
Gateway In1
Gateway Out1
Out
Gateway Out4
0
0.5236
In
Gateway In9
Gateway In3
Gateway Out2
Out
Gateway Out5
0
0
In
Gateway In6
Gateway In
untitled
hwcosim

7. Results comparison with the built-in Xilinx CORDIC modules in SysGen in terms of numerical results and also FPGA area usage.

The iteration is configured to 10 which is the same as the HLS verision.



The numerical results in the built in CORDIC module is slightly different compared to my HLS module, but the relative error is smaller than 2%.

Name	Slice LUTs (53200)	Slice Registers (106400)	Slice (13300)	LUT as Logic (53200)	LUT as Memory (17400)	Block RAM Tile (140)	Bonded IOB (200)	BUFGCTRL (32)	MMCME2_ADV (4)	BSCANE2 (4)
hwcosim_top_wrapper	1990	2507	705	1979	11	2	1	4	1	1
hwcosim_top_i (hwcosim_top)	1990	2507	705	1979	11	2	0	4	1	1
axis_data_fifo_rx (hwcosim_to	79	210	56	79	0	1	0	0	0	0
axis_data_fifo_tx (hwcosim_to	79	210	56	79	0	1	0	0	0	0
axis_dwidth_converter_rx (hwc	15	50	16	15	0	0	0	0	0	0
axis_dwidth_converter_tx (hwc	27	50	17	27	0	0	0	0	0	0
hwc_jtag_axi_transport_0 (hw	58	132	34	58	0	0	0	1	0	1
hwcosim_cmd_proc (hwcosim	163	113	62	163	0	0	0	0	0	0
reset_gen (reset_gen_imp_1\	16	33	11	15	1	0	0	0	0	0
sys_clk_wiz (hwcosim_top_sy	0	0	0	0	0	0	0	2	1	0
xilinxip_0 (hwcosim_top_xilinx	1558	1709	483	1548	10	0	0	1	0	0

We can see that the Xilinx CORDIC consume two times of hardware resources. This makes sense considering we use two CORDIC unit for SINCOS and ARCTAN which will cost more area compared to reconfigurable design.

8. IP block export to SDK and Zynq ARM program control

The following figure shows the UART output of vitis arm baremetal program, we can see that the screenshot matched the C testbench.

```
Connected to COM5 at 115200
--- Start of the Program ---

Ap_fixed variables initialized for software sim, mode=0 t=0.52356 ts=0.885986 tc=0.5

Done signal from hardware = 1

result sin = 0.498718, cos = 0.867004

Ap_fixed variables initialized for software sim, mode=1 t=0.52356 ts=0.885986 tc=0.5

Done signal from hardware = 1

result theta = 60.513073

--- End of the Program ---
```

9. Turning in files

All the files are attached in this zip file.