EE 599
Spring 2020
Homework 2
Assigned: March 12, 2020
Due: March 27, 2020
Total Points: 100

1 Barrel Shifter [50 Points]

A Barrel Shifter is a logic circuit for shifting a word by a varying amount. It has a control input (select bits) that specifies the number of bit positions that it shifts. A Barrel Shifter is implemented with a sequence of shift multiplexers, each shifting a word by 2^k bit positions for different values of k. The diagram below (Figure 1) shows a pipeline clock-wise barrel shifter for 8 inputs.

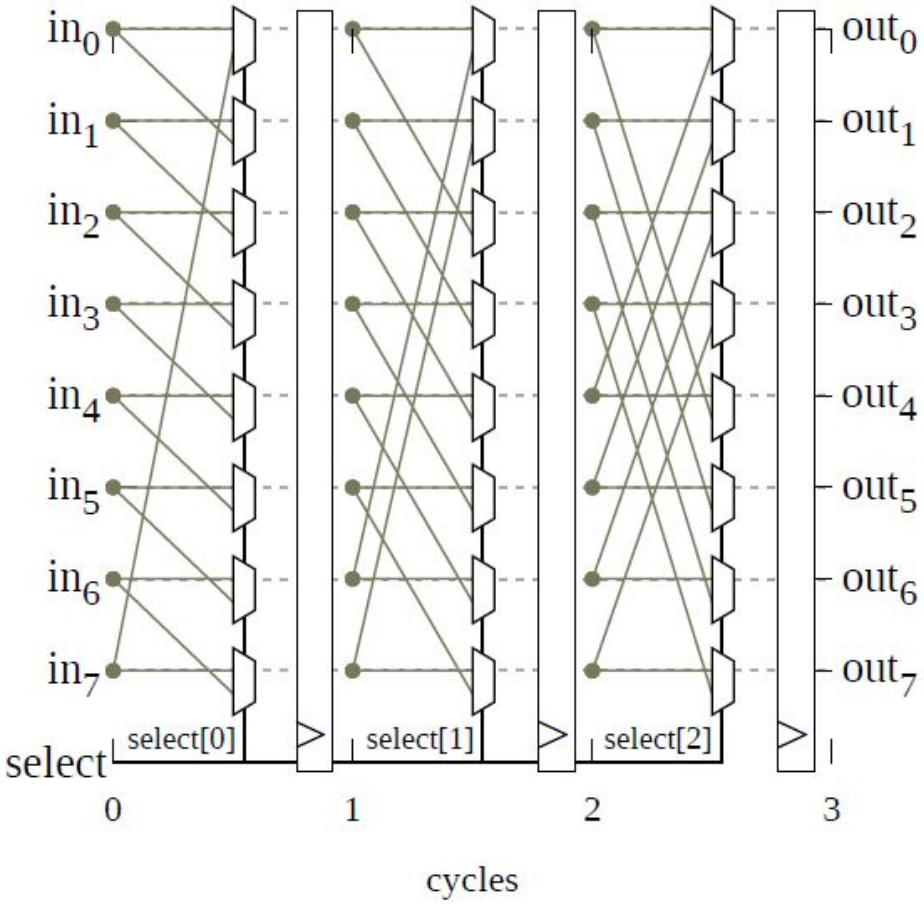

Figure 1: An Example Pipeline Barrel Shifter

Notice that we can shift the input elements by 0 - 7 elements using an implementation like Figure 1. Table 1 shows all the shifting possibilities for the above example.

Similarly, a scalable Barrel Shifter with N inputs and maximum shift of *N-1* can be implemented.

Table 1: All the shifting possibilities for the given example

Select[2]	**Select[1]**	**Select[0]**	**Shift**
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

1.1 Implementation : Using Xilinx Vivado Software (Use Zynq-7000 xc7z007sclg225-2 FPGA)

Consider only 8-bit arithmetic. **You must implement a scalable design.**

1. Implement a **barrel shifter** design in Verilog which takes n inputs with 8 bits and shift them by r (value of r is passed to select bits and $r_{max} = (n - 1)$).
2. For a 16 elements design with a maximum shift of 15, write a testbench and verify the waveforms.
3. Elaborate the design and include all the schematics screenshots of the modules in the report.
4. Synthesis the design and include the schematics screenshots in the report.
5. Generate Resource and timing estimations and include them in the report.
6. Redo parts 3, 4, 5 for 64 elements (with maximum shift of 63).

2 Systolic Array for Dense Matrix-Matrix Multiplication [50 Points]

The Systolic Architecture consists of an array of processing elements, where data flows between neighboring elements, synchronously, from different directions. Processing element takes data from Top, Left, and output the results to Right, Bottom.

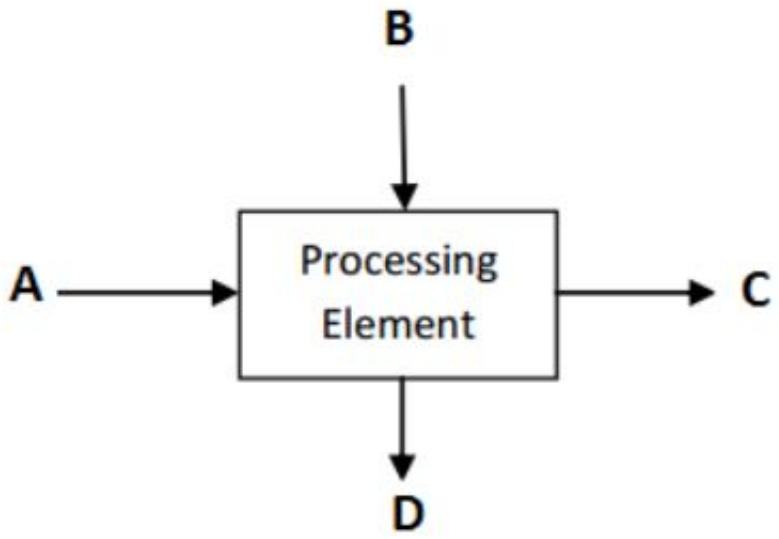

Figure 2: Inputs and outputs to the Processing Elements

One of the main applications of Systolic Architecture is matrix multiplication. As the following figure depicts, in_a, in_b are inputs to the processing element and out_a, out_b are output to the processing element. out_c is to get the output result of each processing element.

Processing elements are arranged in the form of an array. In the following example, we

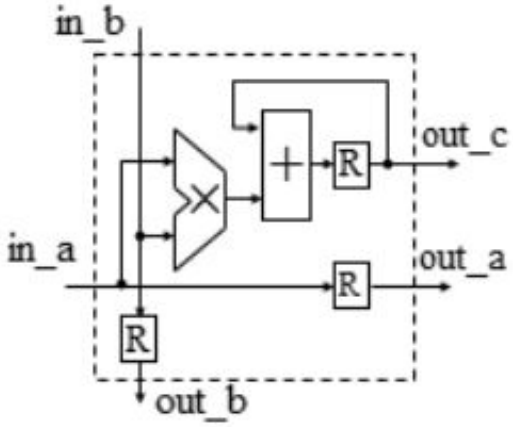

Figure 3: Internals of the PE

analyze, multiplication of 3×3 matrices, which can be easily extended. Let say the two matrices are A and B. Figure 4 depicts how matrix A and B are fed into PE array.

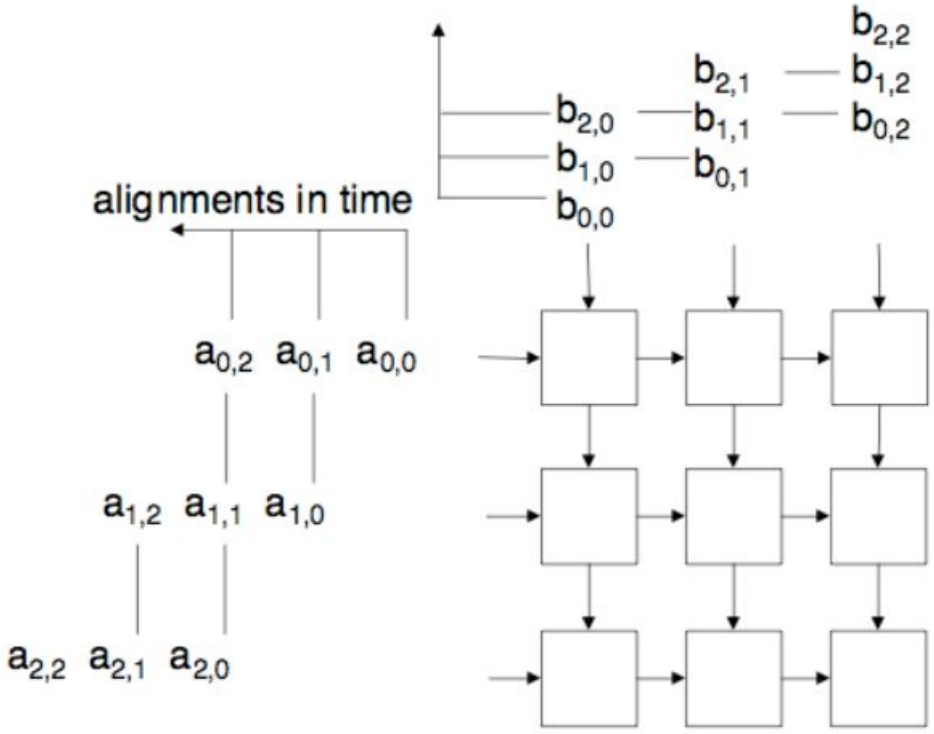

Figure 4: Example 3×3 Systolic Array

2.0.1 Implementation : Using Xilinx Vivado Software (Use Zynq-7000 xc7z007sclg225-2 FPGA)

Consider only 8-bit arithmetic. All the values are already stored in the BRAM.
You must implement a scalable design.

1. Implement a **Systolic Array** design in Verilog, which takes two $n \times n$ matrices and multiply them together.
2. Write a testbench that takes two 16×16 matrices to the design and provide an output of 16×16.
3. Simulate the design using the Vivado simulator and include the waveform in the report (clearly indicate the locations where final outputs produced).
4. Elaborate the design for 16×16 and include all the schematic screenshots of the modules in the report.
5. Synthesis the design and include the schematics screenshots in the report.
6. Generate Resource and timing estimations and include them in the report.
7. Generate power estimation reports and include them in the report.
8. Similarly provides reports for 32×32 matrix multiplication.