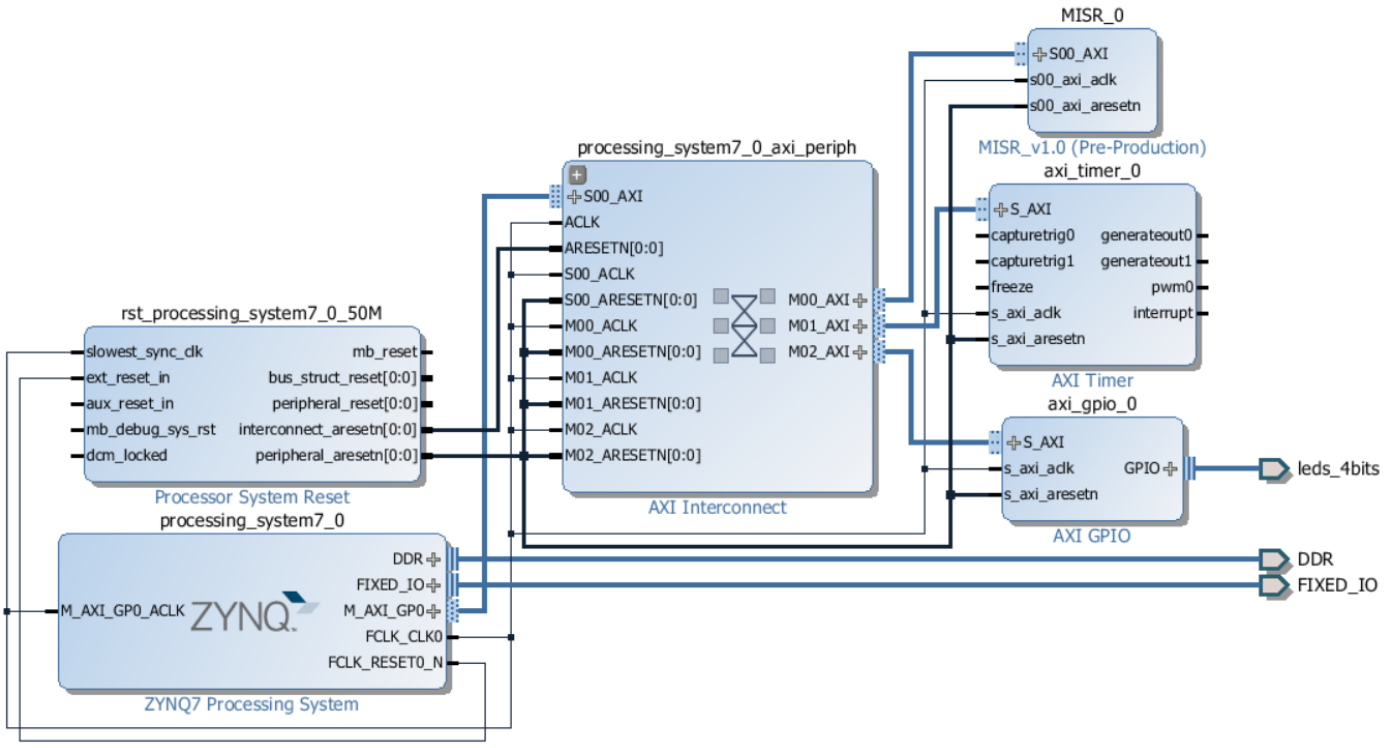
ECE 699 Software/Hardware Codesign
Midterm Exam
Spring 2016

Part 2: Practice

Package the hardware accelerator, called MISR, and build the software/ hardware system using already existing IPs mentioned below. Build the complete system and utilize provided C code (should be modified) to view the performance of the system.

1. Block Design:



2. Configuration of PS:

Configure PS to include the following settings.

1- UART interface to view output on the console.

2- DDR3 memory interface to store the arrays.

3. **PL side:**

   Include the following IPs on the PL side.

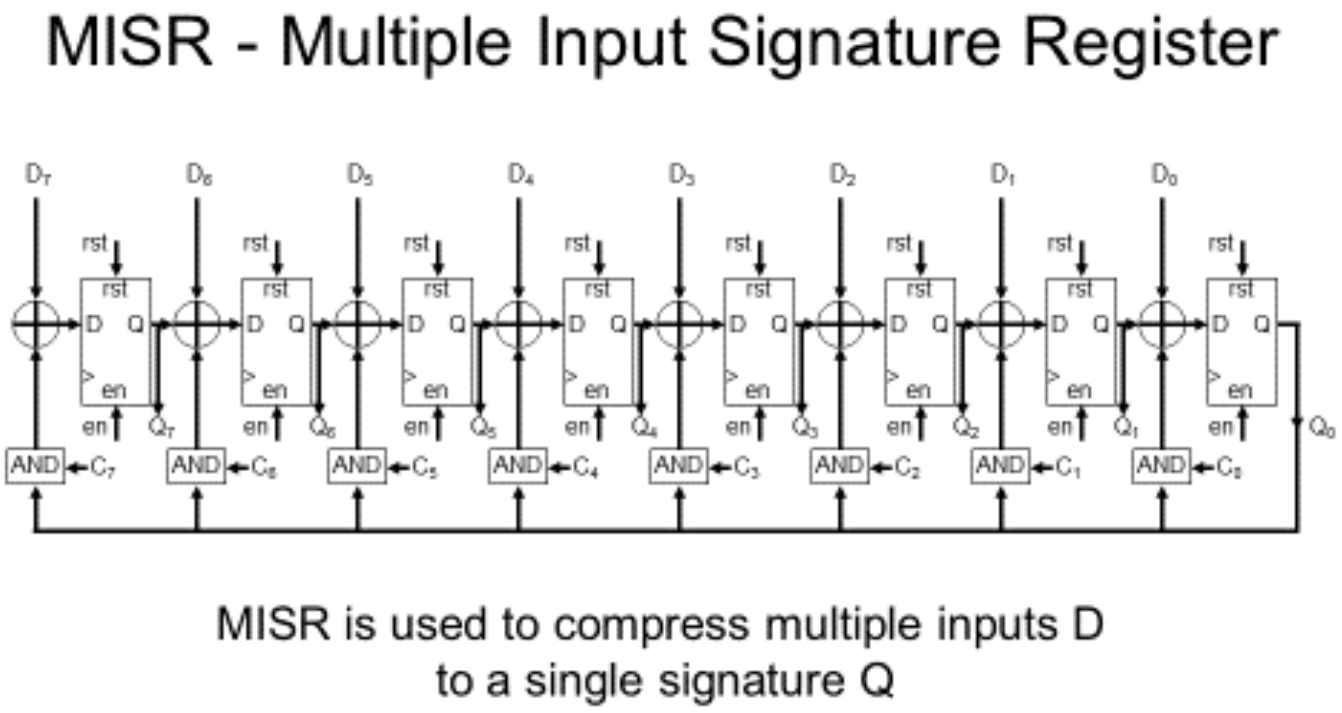
   1- GPIO IP for LED output.

   2- Timer IP to measure the time for HW/ SW codesign approach.

   3- MISR IP to compute the product of two numbers using AXI-Lite interface.

   4- AXI-Interconnect IP to connect PS to PL.

4. **Description of MISR operation**



5. **Functionality of C code:**

   Below is the explanation of the functionality of C code.

   1. Initialize LED.
   2. Initialize the timer.
   3. Initialize D.
   4. Compute software results.
   6. Compute hardware results **(should be done by the students).**
   7. Compare software and hardware results.

Required Tasks:

1. Create and package MISR IP.
2. Build the PL side and export the block design to software.
3. Launch the SDK and include the provided C code.
4. Write the C code for hw_accel function to compute hardware results based on the following description:
   1- Start AXI Timer.
   2- Reset MISR ip by writing 1 to slv_reg3 through AXI Lite interface and then write 0 to the same register.
   3- Write C (constant value) to slv_reg0 through AXI Lite interface.
   4- Write all D values to slv_reg1 through AXI Lite interface using the provided for loop.
   5- Read slv_reg2 through AXI Lite interface and put the result into Q variable.
   6- Stop AXI Timer
   7- Read the Timer and print the timer value.
   8- Write the 4 most significant bits of Q on board LEDs.
   9- Use the provided delay function to generate 1 sec delay.
   10- Write the 4 least significant bits of Q on board LEDs.
   11- Use the provided delay function to generate 1 sec delay.
   12- Turn off all LEDs.

5. Run the code and view the output on the console for different array sizes. Array sizes for A, B and P are directly controlled from the switch input. Array sizes range between 1 to 2^{sw} (switch value = 0 to 15).

List of deliverables for the required task:

1- Run the C code and take snapshot from the SDK console and submit it on blackboard.
2- Submit your C code on BlackBoard
3- 2-minute demo.