# Lab 3

1. Design 16:4 Priority Encoder using if-else Statement. Use $write to display the output with respect to current input on new line of the transcript.

2. Design an 8-bit ALU. Features include:
   a. Two 8-bit inputs A and B.
   b. One 8-bit output C.
   c. Five 1-bit outputs Borrow, Carry, Equal, less (A<B) and more (A>B).
   d. Supports Addition, Subtraction, XOR, AND, NOR, NAND and Comparison.

   Also Display Inputs, Outputs and Operation performed on transcript. Output (8-bit) will be zero for comparison operation. Use case equality for comparison of A and B.

3. Design a 16 bit parallel in parallel out universal Shift Register which performs following operations:
   a. Load, Shift_enable.
   b. Left Shift.
   c. Right Shift.
   d. Rotational Left Shift.
   e. Rotational Right Shift.

4. Design a circuit which accepts two 3-bit inputs (A and B) and produces 5 1-bit outputs (gray, Excess-3, More, Less, no_relation). The outputs are set high when:
   a. Gray: A and B are gray codes to each other
   b. Excess-3: A and B differ by 3.
   c. More: A is one more than B.
   d. Less: A is one less than B.
   e. no_relation: If none of the above relation is true.

5. Write a function to perform XOR operation on two inputs A and B (1- bit each). Using this function, write a parity function which accepts m-bit input and returns parity. Design a module which accepts 16-bit input din and returns 1-bit output Parity using parity function.