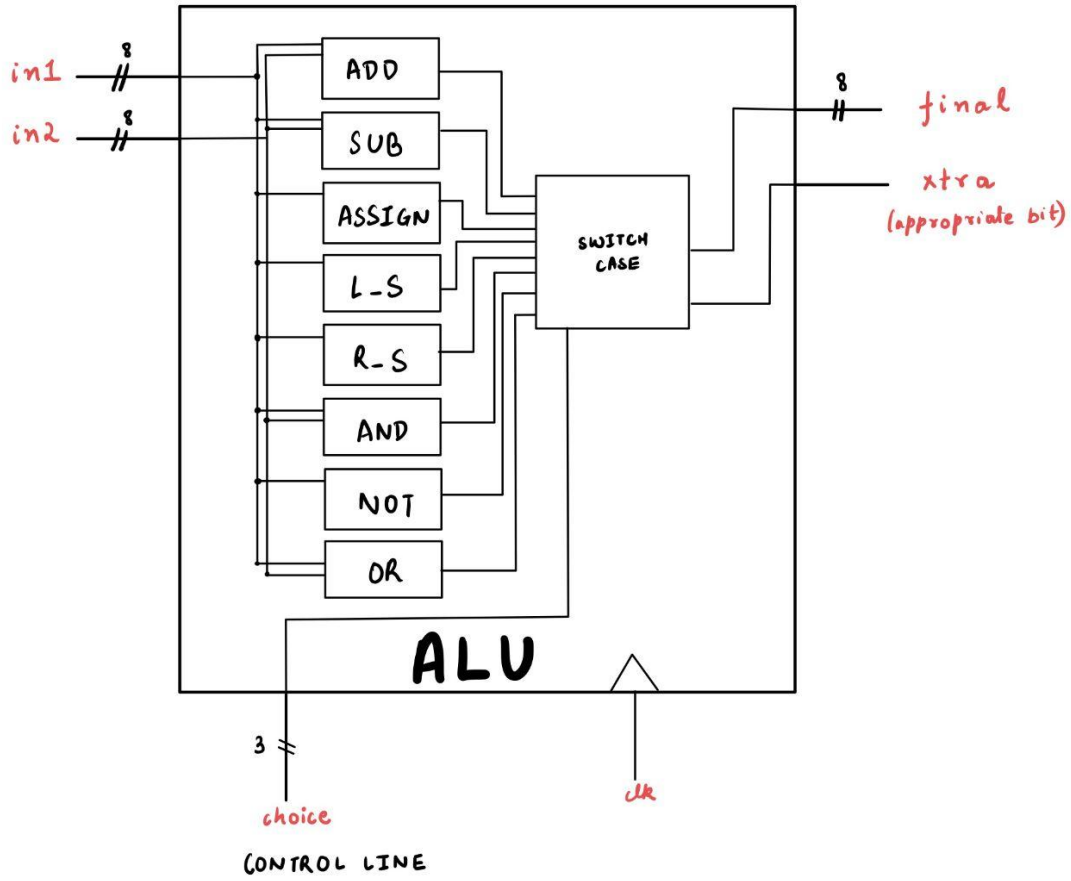
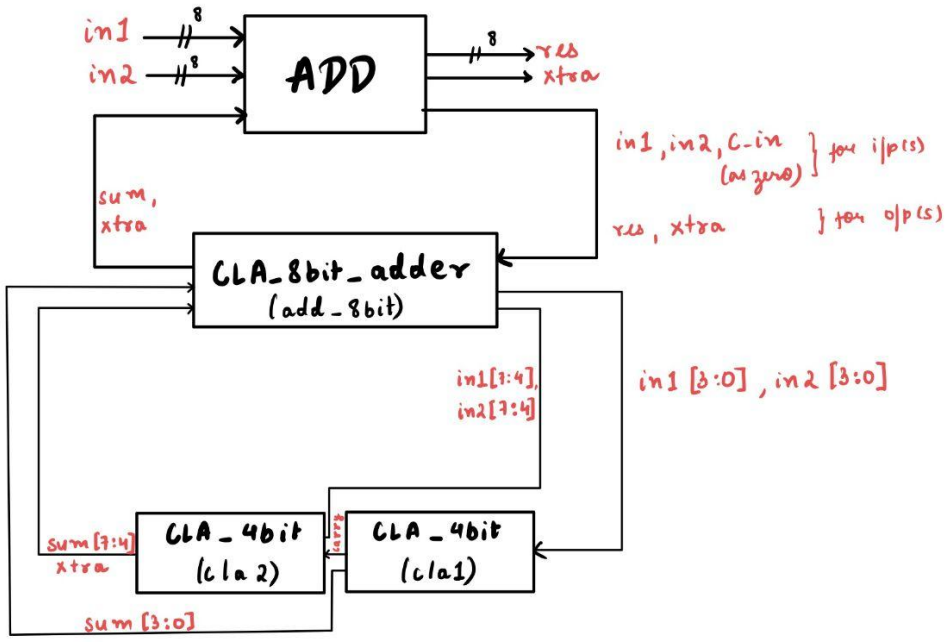
Verilog Assignment 3
Tanishq Prasad (21CS30054)
Krish Khimasia (21CS10037)
Group No. 17

Question 1:



Our ALU module has 8-bit in_1, 8-bit in_2, 3-bit choice (for choice of operation), and clk as input. Output is stored in 8-bit final, and xtra. This module calls instances for each operation.

- We have implemented an 8-bit CLA ripple adder using 2 instances of the 4-bit CLA. The 4-bit CLA uses the normal generate and propagate logic. 1 instance of the 4-bit CLA adds the lower 4 bits of in1 and in2 (with c_in as zero), while the other instance adds the next 4 bits of in1 and in2 (with c_in as c_out of the previous instance). Carry is stored in c_out, and result in sum.



- ADD: Directly uses the 8-bit CLA ripple adder with a as in_1, b as in_2 and carry_in as zero. Carry stored in xtra.
- NOT: Complements in_1.
- AND: Computes A&B.
- OR: Computes A|B.
- SUB: Computes ~in_2 using NOT operation, adds 1 to ~in_2 using ADD, then adds this and in_1, which is the final subtraction result of in_1 and in_2.
- ASSIGN: Assigns res with the value of in_1.
- LEFT_SHIFT: Computes LEFT_SHIFT of in1.
- RIGHT_SHIFT: Computes RIGHT_SHIFT of in1.