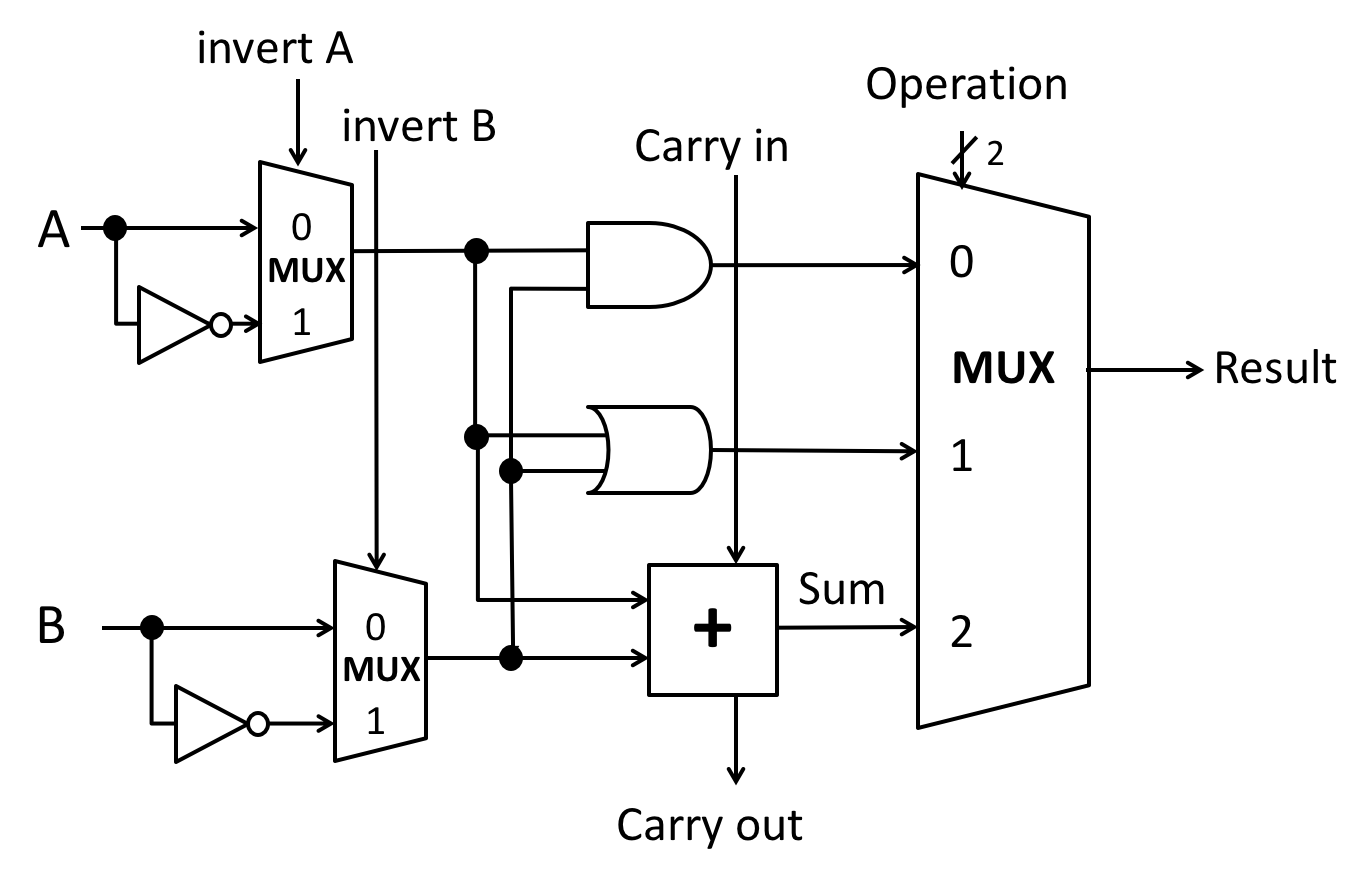
***Question1***

***Shushant Kumar 16CO143***

***Anmol Horo 16CO206***

***Ques:* 1-bit ALU**. Build a 1-bit ALU that can perform the following logical and arithmetic operations: AND, OR, NAND, NOR, ADD, SUBTRACT. Include two additional flags: (a) a flag to indicate if the result of the ALU operation is zero (Z), (b) a flag to indicate if the first input is greater than the second input.



We have implemented the above circuit.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Ainvert | Binvert | CarryIn | Operation |
| AND | 0 | 0 | 0 | 00 |
| OR | 0 | 0 | 0 | 01 |
| ADD | 0 | 0 | 0 | 10 |
| SUB | 0 | 1 | 1 | 10 |
| NAND | 1 | 1 | 0 | 01 |
| NOR | 1 | 1 | 1 | 00 |

The above table represents the ALU Control Signals and this was were used to for carrying out each particular operation.

A flag ‘z’ was included to indicate if the result of the ALU operation is zero.

Another flag ‘flag2’ was included to indicate if the first input is greater than the second input. For this the inputs were compared and the flag value was determined.

The inputs invertA, invertB, carryIn and Operation select lines were used to control the operations flow for data input A and B.

The result is the final result and Carry out returns any carry generated in any operation.