Assign_7_Q1_Grp27

```
Verilog Assignment 7
Problem No.: 1
Semester: 5
Group No.: 27
Yash Sirvi (21CS10083)
Sanskar Mittal (21CS10057)
```

Question 1

Implemented ALU instructions that take 2 32 bit inputs and performs the corresponding ALU function.

Inputs:

- `in1` , `in2` : 32 bit input values
- `func` : ALU function
- for the FPGA implementation, there is only one 15 bit input `in`

Functions

- `0000: ADD` : returns `in1 + in1`
- `0001: SUBTRACT` : returns `in1 - in2`
- `0010: AND` : returns `in1 & in2` bitwise
- `0011: OR` : returns `in1 || in2` bitwise
- `0100: XOR` : returns `in1 ^ in2` bitwise
- `0101: NOT` : returns `~in`

FPGA implementation

- Since FPGA has limited input switches, we use a single 15 bit `in` to get our 15 bit input
- We then assign the input to given `inp1` or `inp2` data registers according to our states
- state `SI11` takes input to lower 16 bits of `inp1`
- state `SI12` takes input to upper 16 bits of `inp1`
- state `SI21` takes input to lower 16 bits of `inp2`
- state `SI22` takes input to upper 16 bits of `inp2`
- state `Sfunc` waits for user to input the ALU `func`
- state `SCalc` executes the ALU function defined by `func` on `inp1` and `inp2` . It also outputs the lower 16 bits of the output to display
- state `Sout2` displays upper 16 bits of the output
- `nextstate` helps us move between states Manually

- `lock` allows us to change the `state` at most once every press. Avoids multiple state changes if the `nextstate` button is pressed for too long.

Schematic