

ALSU project documentation

Idea of operation:

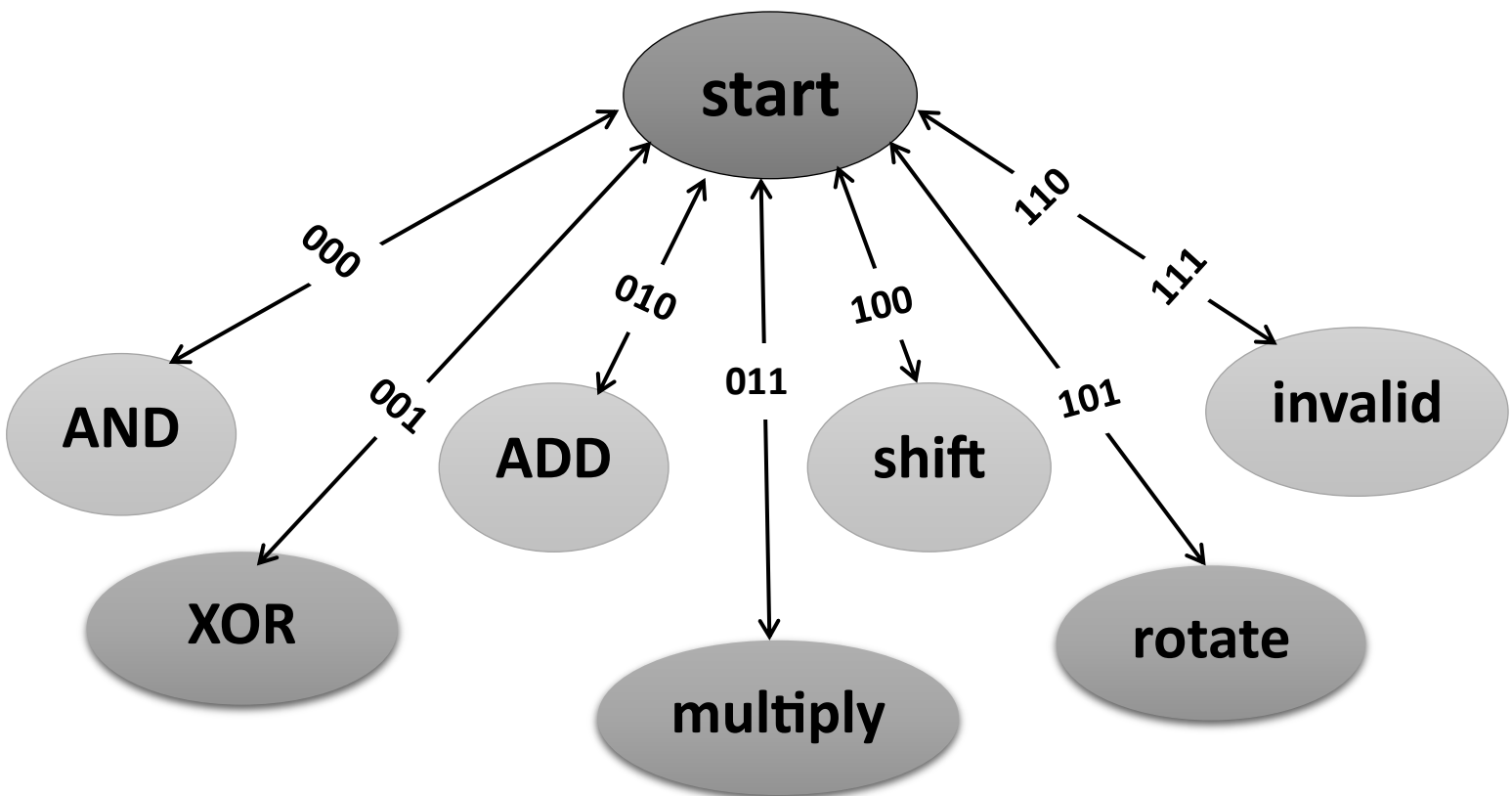
It's a state machine that done arithmetic and shift operations.

Every state in the machine represents an operation.

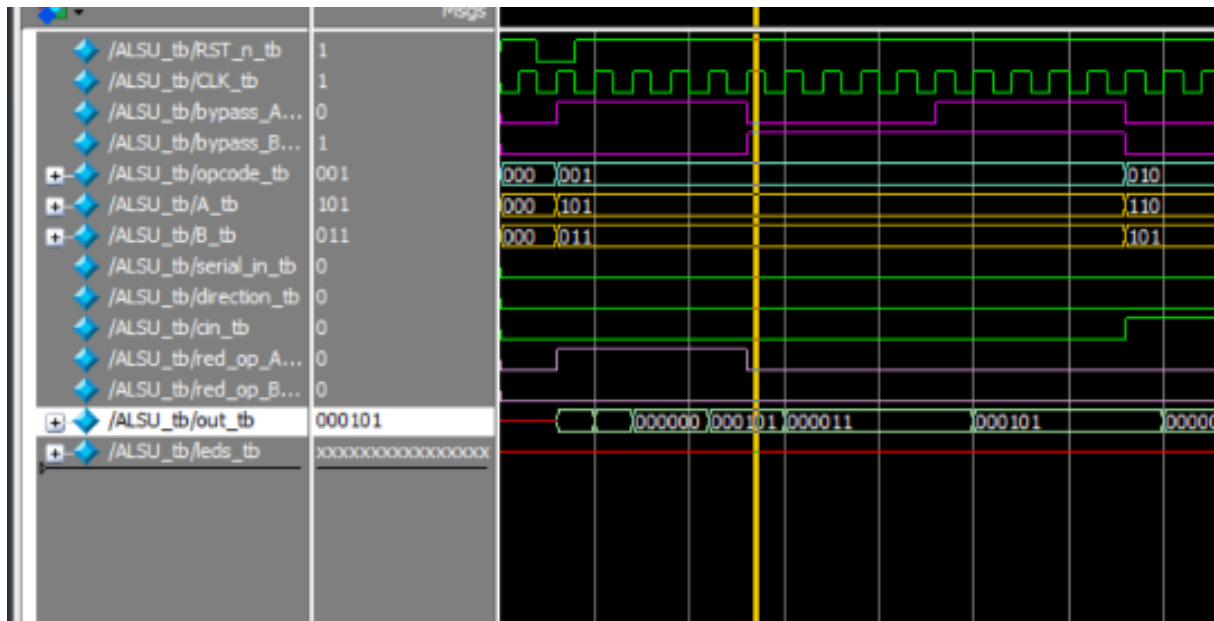
We have a state where we start from.

Start state guiding us to the chosen operation.

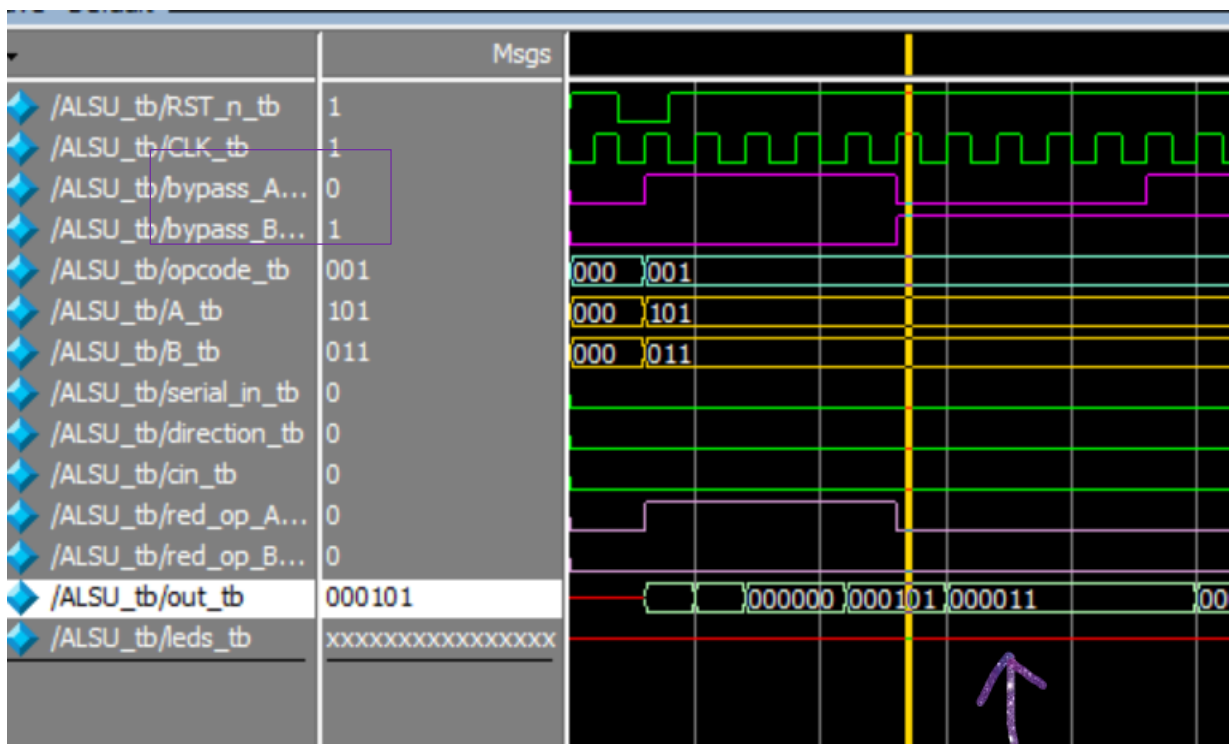
We go to the chosen state and do the operation then return to start state.



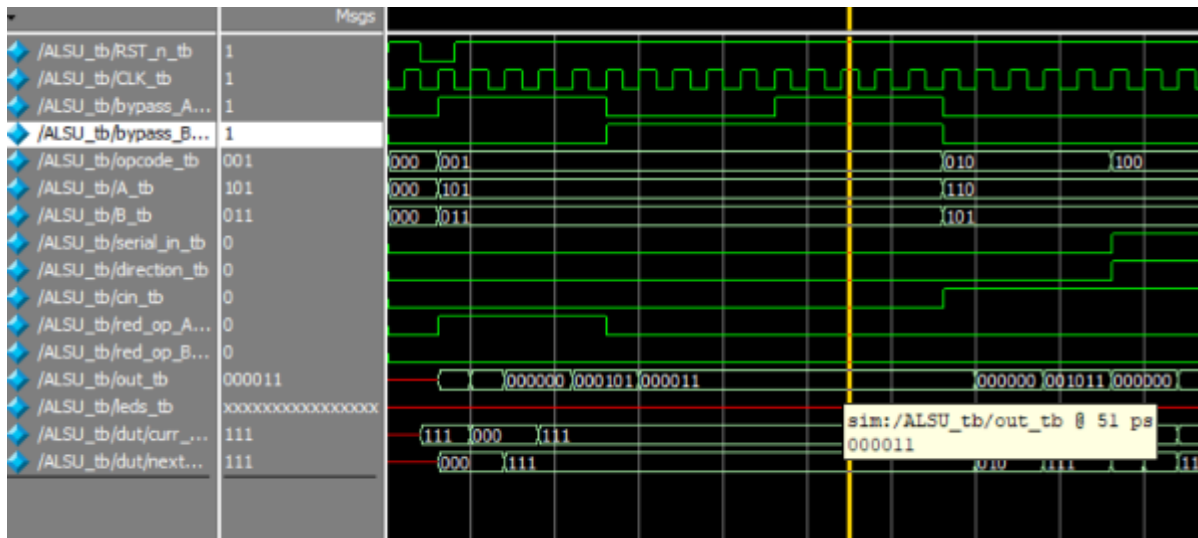
Test cases:



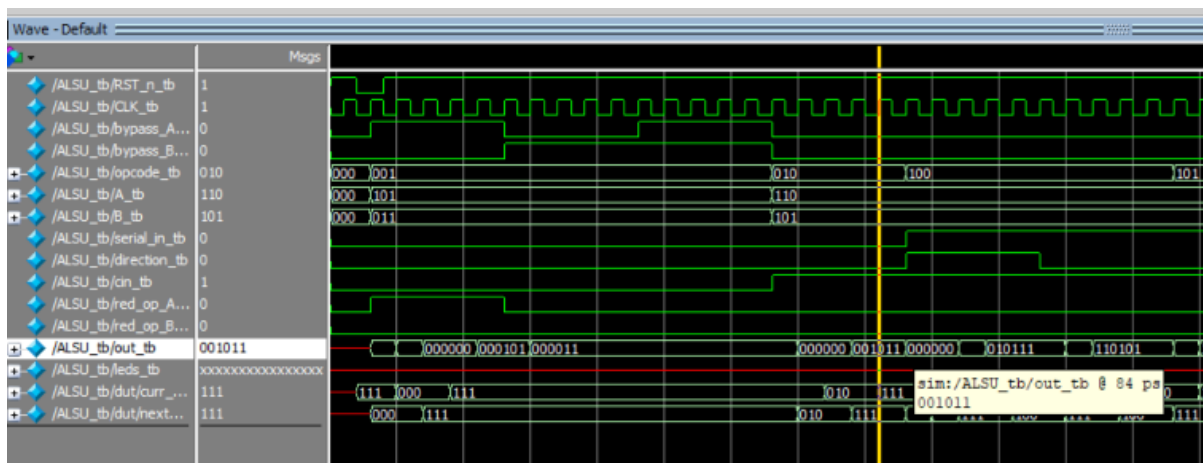
Bypass A



Bypass B



Bypass to priority that selected in this case priority was to B.

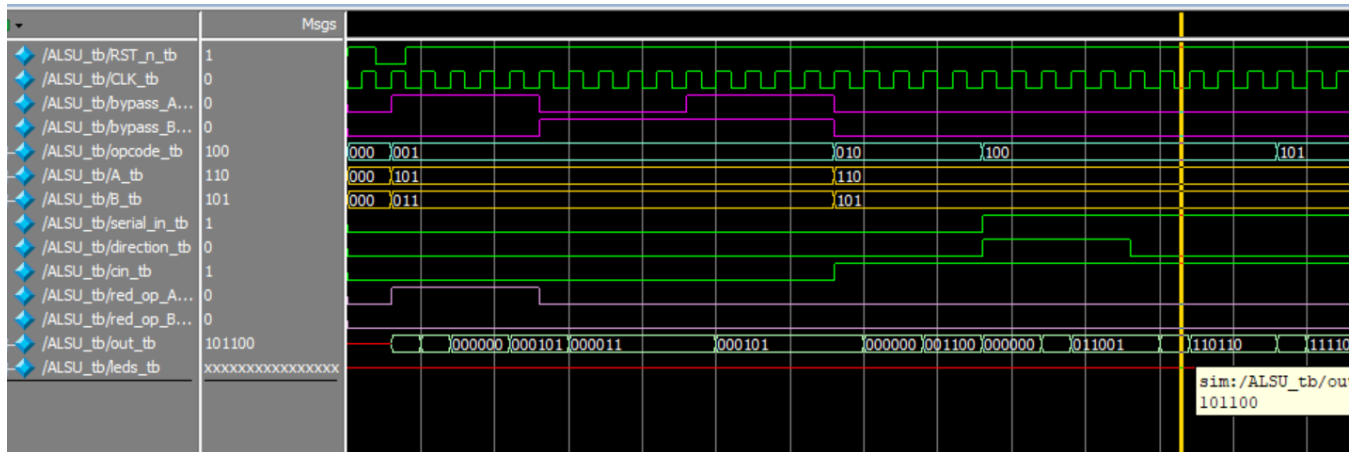


Addition operation

$A+B=out$

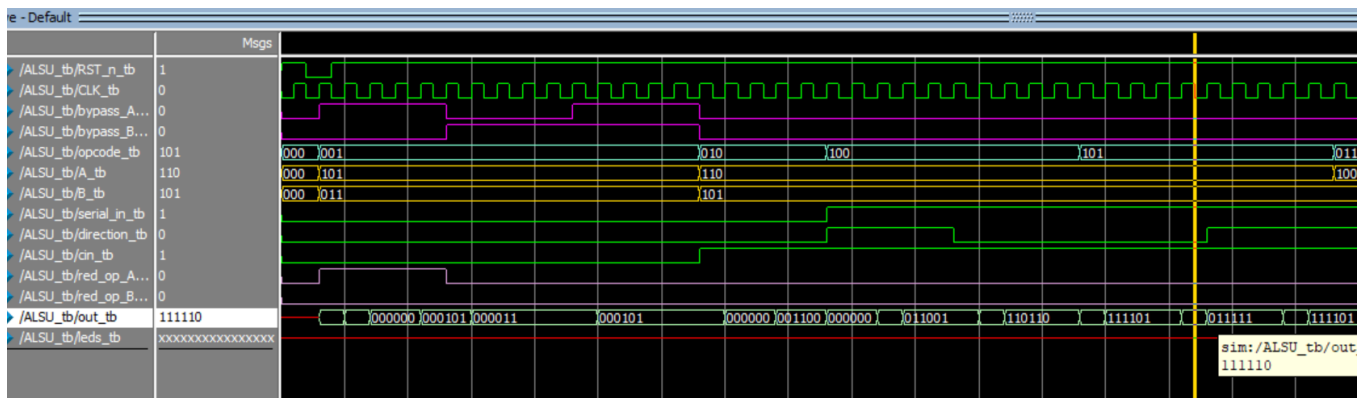
$6+5=11$

Full adder turned off.



Shift right.

011001 becomes 101100.



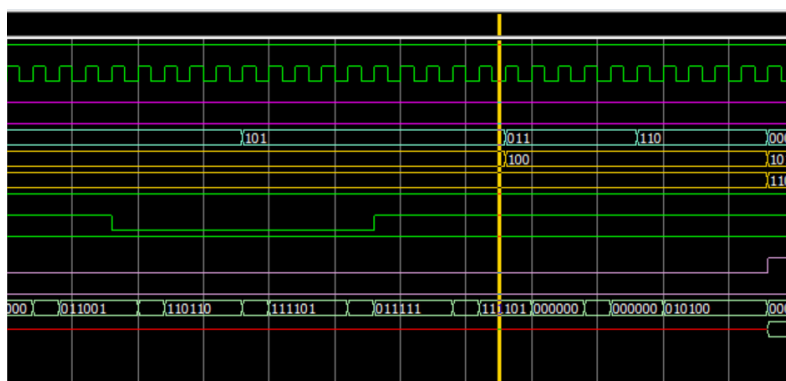
Rotate operation.

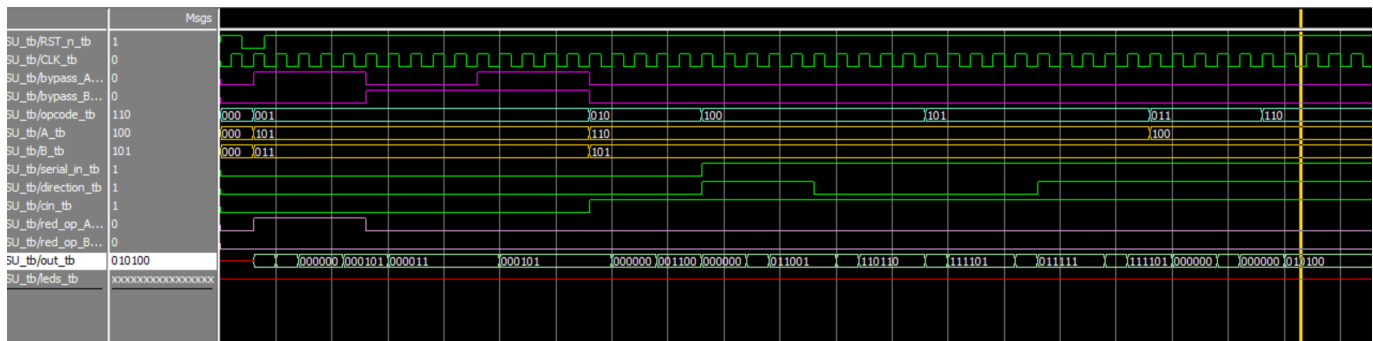
Direction is right.

111101 becomes
111110.

rotate left.

111110 becomes
111101.

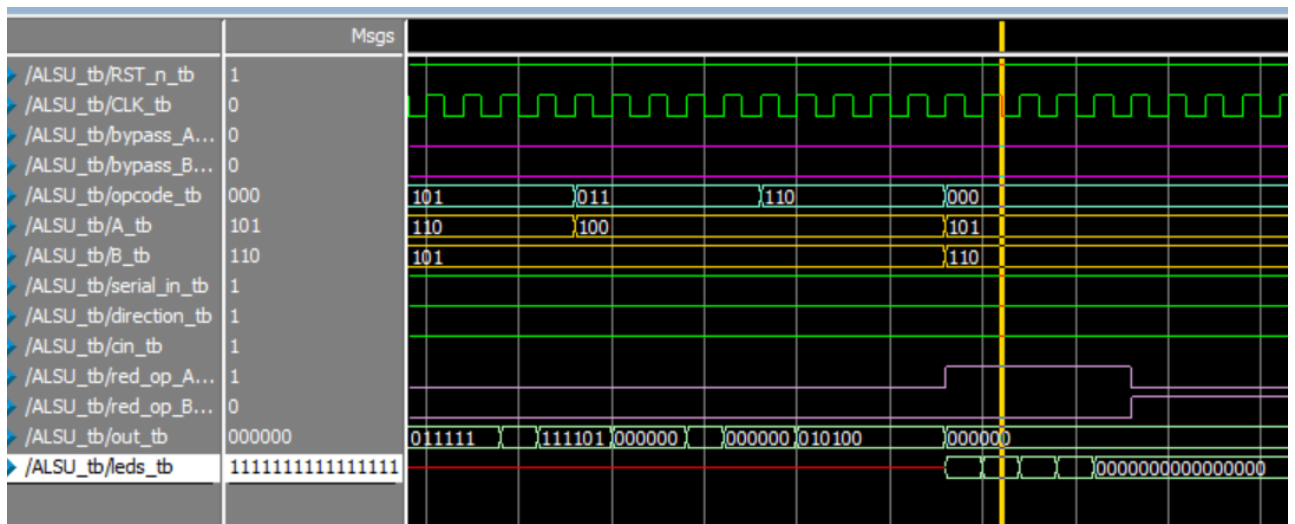




multiplication operation

$A * B = \text{out}$

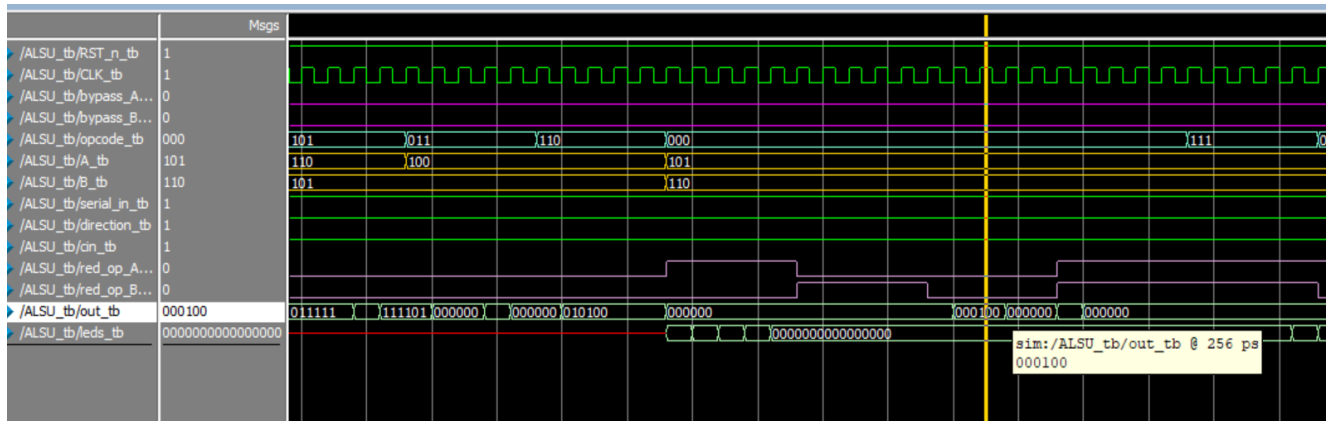
$4 * 5 = 20$



Invalid case

Out become 000000.

And the LEDs start to toggle.



AND operation

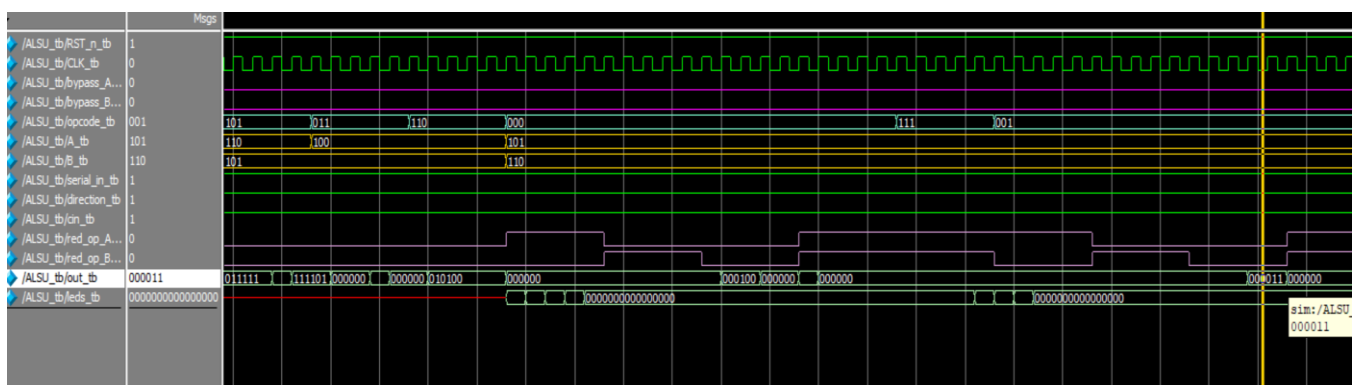
$A \& B = \text{out}$

$101 \& 110 = 000100$

Before ANDing A&B

Red_op_A was set to one and $\&A = 000000$

Then Red_op_B was set to one and $\&B = 000000$



XOR operation

$A \wedge B = \text{out}$

$101 \wedge 110 = 000011$

Before XORing A,B Red_op_A was set high $\wedge A = 000000$

then Red_op_B was set high and $\wedge B = 000000$