CS232 lab 5 Q2

Booth multiplier

The working of the Booth multiplier was given in the Sequential Design Files provided to us and the architecture is based on that.

if rst = '1' then result = "00000000"
else, result will be a×b where 'a' and 'b' are 2 four bit numbers.

Architecture:

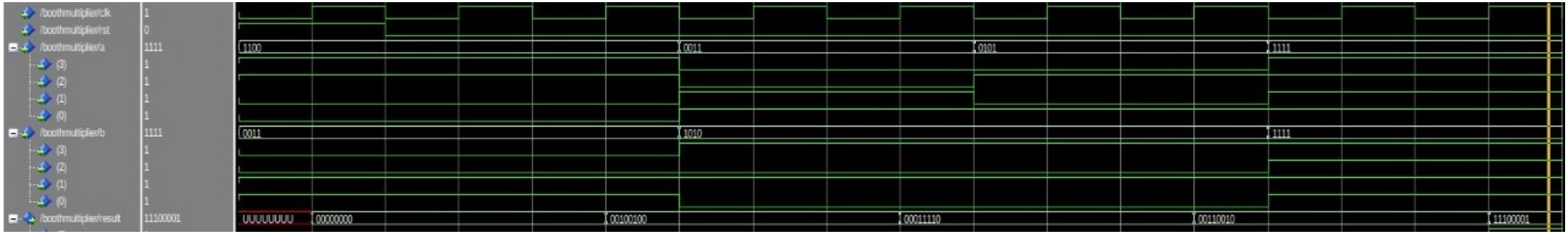
I use 3 processes to build the booth multiplier.

The first process is sensitive to the clock and when
→ if rst = '1', then next_state sets to '0' and sum to "0000 0000"
→ else if rst = '0'
if state = '0' then
next_state is set to '1', partial products are computed. I have employed vector concatenation to calculate the partial products.

if state = '1'
next_state is set to '0', sum of partial products is done and value is stored in the "sum" signal.

In the second process, I update state signal by assigning it value of next_state.
In the third process, the result (output signal) is given the value of sum.

See waveform below for the output.


/boothmultiplier/clk
1
/boothmultiplier/rst
0
/boothmultiplier/a
1111
1100
0011
0101
1111
(3)
1
(2)
1
(1)
1
(0)
1
/boothmultiplier/b
1111
0011
1010
1111
(3)
1
(2)
1
(1)
1
(0)
1
/boothmultiplier/result
11100001
UUUUUUUU
00000000
00100100
00011110
00110010
11100001