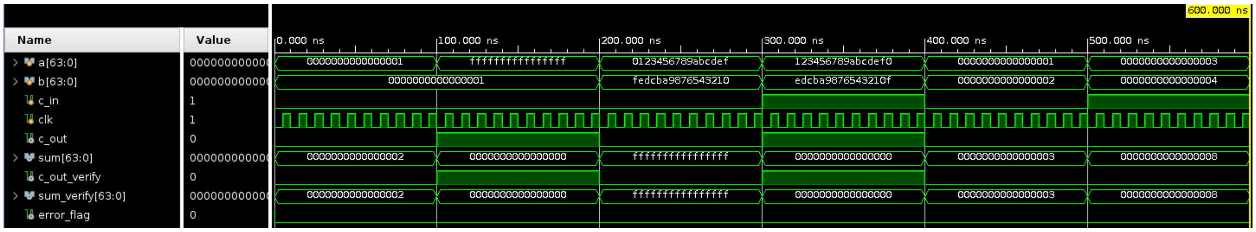
Rawisara Chairat

U65053623

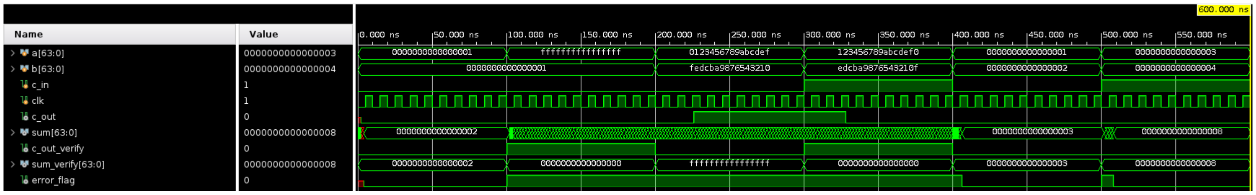
Lab 3

For this lab, I have created two adders, 64-bit Ripple Carry Adder (RCA_64bit.v) and 64-bit 2-stage Carry Select Adder (CSA_64bit.v).

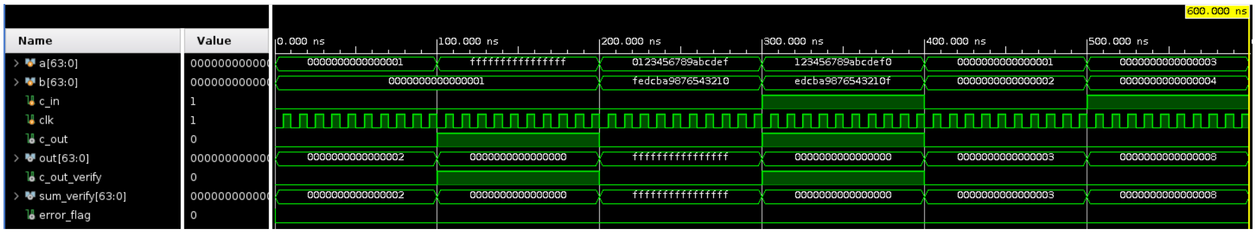
1. **RCA_64bit.v:** Construct a 64-bit ripple carry adder by connecting four 16-bit full adders (RCA_16bit.v) together, linking the carry out of each to the carry in of the next. The adder will accept two 64-bit inputs and a carry in, producing a 64-bit sum and a carry out.
    a. **RCA_16bit.v:** Build a 16-bit adder by connecting four 4-bit full adders (Adder_4bit.v) together, linking the carry out of each to the carry in of the next. The adder will accept two 16-bit inputs and a carry in, producing a 16-bit sum and a carry out.
        i. **Adder_4bit.v:** Create a 4-bit adder by connecting four 1-bit full adders (FA_str.v) together, linking the carry out of each to the carry in of the next. The adder will take two 4-bit inputs and a carry in, producing a 4-bit sum and a carry out.
2. **CSA_64bit.v:** Construct a 64-bit carry select adder by having three 32-bit full adders (RCA_32bit.v). The first adder will process the first 32-bit input to produce the first 32-bit output and a carry out, which will be connected to the select input of the multiplexer The second adder will have its carry-in wired to 0, using the last 32-bit input; its output and carry-out will be sent to the multiplexer. The third adder will have its carry-in wired to 1, also using the last 32-bit input, with its output and carry-out directed to the multiplexer. The multiplexer will then produce the last 32-bit output with a carry out.
    a. **RCA_32bit.v:** Construct a 32-bit adder by connecting two 16-bit full adders (RCA_16bit.v) together, linking the carry out of each to the carry in of the next. The adder will accept two 32-bit inputs and a carry in, producing a 32-bit sum and a carry out.
        i. **RCA_16bit.v:** Build a 16-bit adder by connecting four 4-bit full adders (Adder_4bit.v) together, linking the carry out of each to the carry in of the next. The adder will accept two 16-bit inputs and a carry in, producing a 16-bit sum and a carry out.
            1. **Adder_4bit.v:** Create a 4-bit adder by connecting four 1-bit full adders (FA_str.v) together, linking the carry out of each to the carry in of the next. The adder will take two 4-bit inputs and a carry in, producing a 4-bit sum and a carry out.
    b. **MUX:** Create a 33-bit multiplexer that accepts two 33-bit inputs, a select, a clock and produces a 33-bit output. The multiplexer will use the select signal to determine which input is passed to the output, and the operation will be synchronized with the clock.



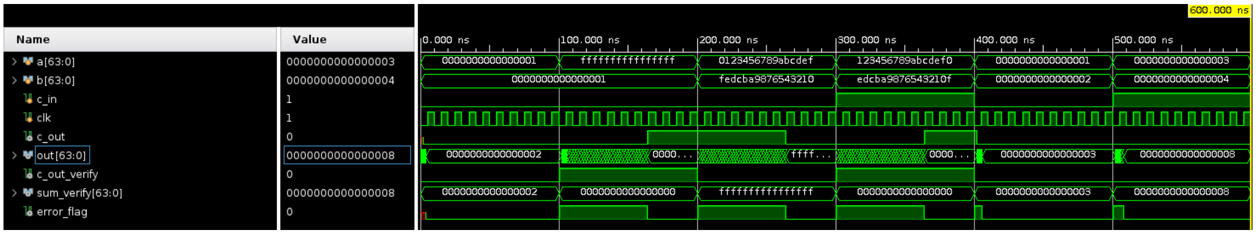
Waveform of Ripple Carry Adder without delay



Waveform of Ripple Carry Adder with delay



Waveform of Carry Select Adder without delay



Waveform of Carry Select Adder without delay

Gate Delay

1. 64-bit Ripple Carry Adder
   a. Without Delay: 0 gate delay (Since there is no delay in the gate)
   b. With Delay: 128 gate delays (Since a ripple carry adder consists of multiple full adders connected in series, and each full adder has 2 gate delays, the total for 64 full adders is 64 x 2 = 128 gate delays.)
2. 64-bit 2-stage Carry Select Adder
   a. Without Delay: 0 gate delay (Since there is no delay in the gate)
   b. With Delay: 64 gate delays (Since the 32-bit adders are producing the sum in parallel. Each 32-bit adder contributes delays from its full adders, leading to a total of 32 x 2 = 64 gate delays.)