Group #:
G54

Final Project
Part 1

ENSC 350 2020

Last Name:
Chao

SID: 3 0 1 3 1
0 6 2 4

Last Name:
Kan

SID: 3 0 1 3 0
9 4 1 7

Last Name:
Yonata

SID: 3 0 1 3 0
4 7 9 4

Last Name:

SID:

Table of Contents

Documentation for Design Entities
- Logic Unit 3
- Arithmetic Unit 4

Functional Simulation Waves
- Logic Unit 5
- Arithmetic Unit 6

Timing Simulation Waves
- Logic Unit 8
- Arithmetic Unit 9

Logic Unit RTL **10**

Logic Unit Post Fit **11**

Arithmetic Unit **13**

Arithmetic Unit Post Fit **14**

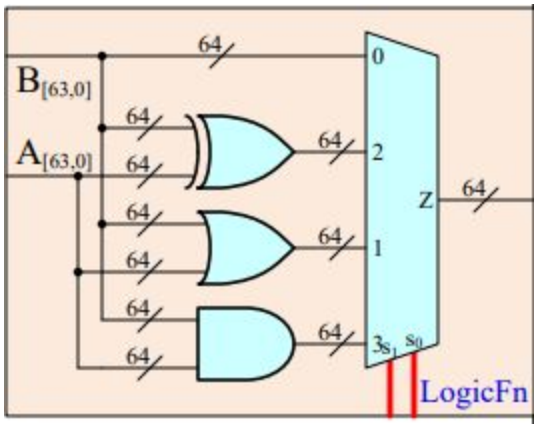
Documentation for Design Entities

Logic Unit

The logic unit performs three logical operations on the two given operands A and B, which are both N bits. N is assigned as 64. The result of the logical operation will be passed as input into the MUX and given the control signal, LogicFn, the desired result will be outputted as Y. Depending on the given control signal, the operands A and B will be XOR, OR or AND together. If the given control signal is 0, the value in operand B will be outputted.

```
Entity LogicUnit is
Generic (N: natural := 64);
Port(   A, B: in std_logic_vector(N-1 downto 0);
        Y: out std_logic_vector(N-1 downto 0);
    LogicFN: in std_logic_vector(1 downto 0));
End Entity LogicUnit;
```

Reference from LogicUnit.vhd



Circuit diagram of Logic Unit

Arithmetic Unit

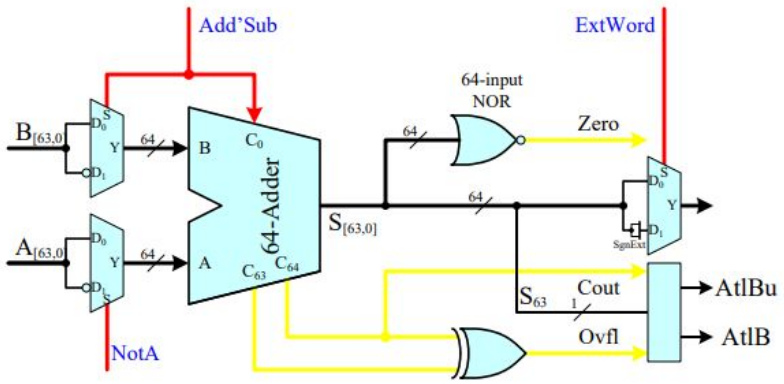
The arithmetic unit is responsible for performing arithmetic and comparison operations in the processor. The arithmetic unit takes in three input control signals: Addn'Sub, ExtWord and NotA. Addn'Sub will determine whether the bits in the B operand will be flipped or not and NotA will determine whether the bits in the A operand will be flipped or not. The Addn'Sub signal also determines whether the two operands will be added with each other or subtracted with each other and it also determines whether the Cin for our 64-bit adder will be a 1 or 0. The resulting sum will be passed on a MUX to determine whether it needs to be sign extended or not. It also goes through a NOR gate to determine whether the Zero signal should be on or off. The resulting carry and overflow goes through the AtlBu and AtlB. AltBu is determined by inverting Cout and AltB is determined by Cout XOR Ovfl. This is for the case of comparing two operands together such as SLT or SLTU. The output of the arithmetic unit is placed into Y.

```
Entity ArithUnit is
    Generic ( N : natural := 64 );
    Port (
        A, B : in std_logic_vector( N-1 downto 0 );
        Y : out std_logic_vector( N-1 downto 0 );
        -- Control signals
        NotA, AddnSub, ExtWord : in std_logic := '0';
        -- Status signals
        Cout, Ovfl, Zero, AltB, AltBu : out std_logic );
End Entity ArithUnit;
```

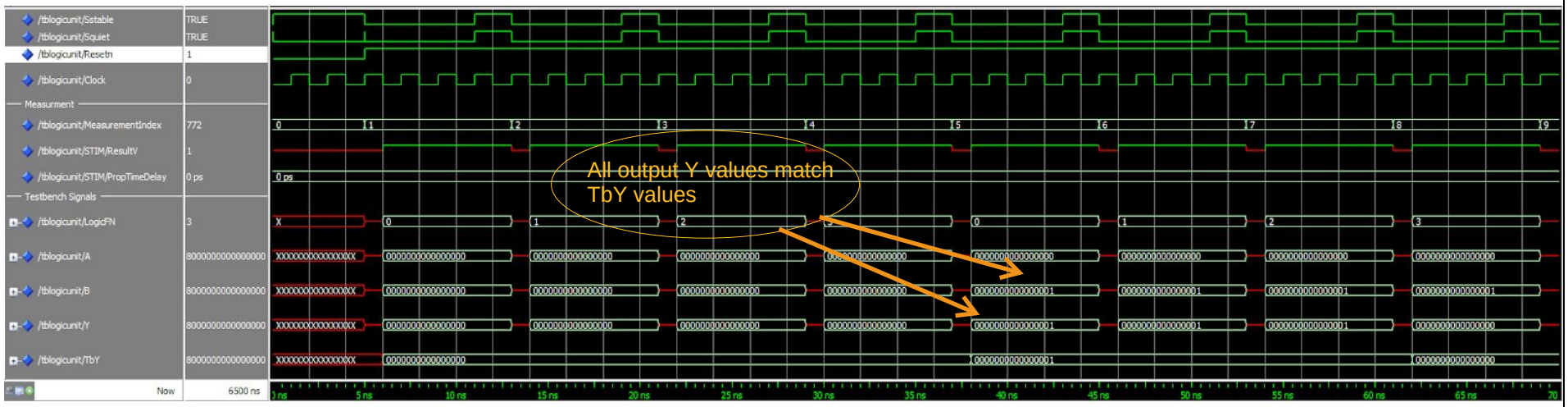
Reference from ArithUnit.vhd

```
entity rippleadder is
    generic ( width : integer := 64 );
    port(
        Xin,Yin     :   in std_logic_vector(width-1 downto 0);
        Cin     :   in std_logic;
        S       :   out std_logic_vector(width-1 downto 0);
        Cout,Cout2  :   out std_logic);
end rippleadder;
```

Reference from Adder.vhd

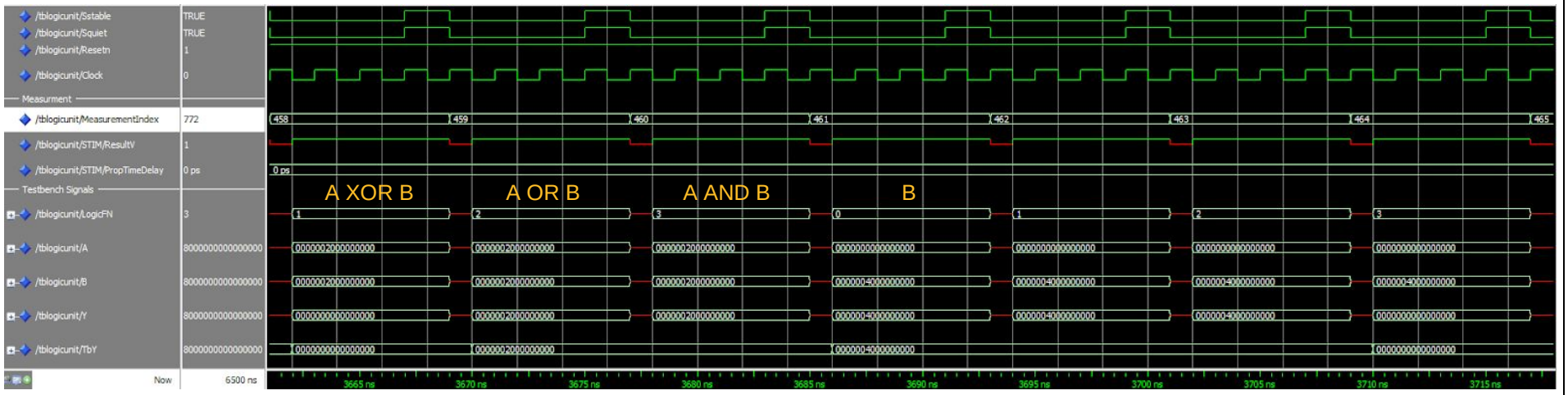


Circuit Diagram of Arithmetic Unit



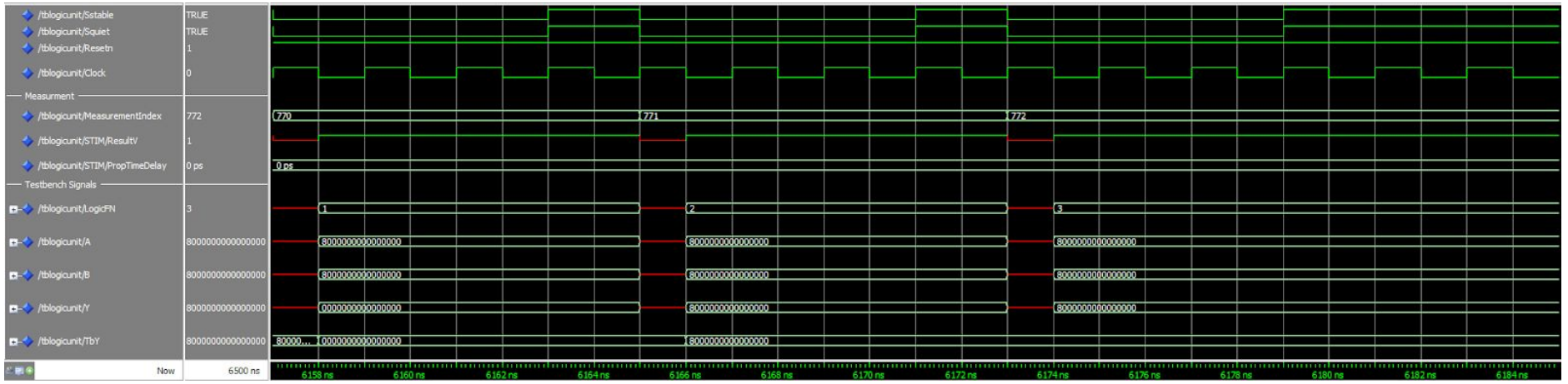
Functional simulation: Logic Unit 1

The diagram shows the values of the first 8 measurements. Each of the Y values and TbY values match for each measurement.



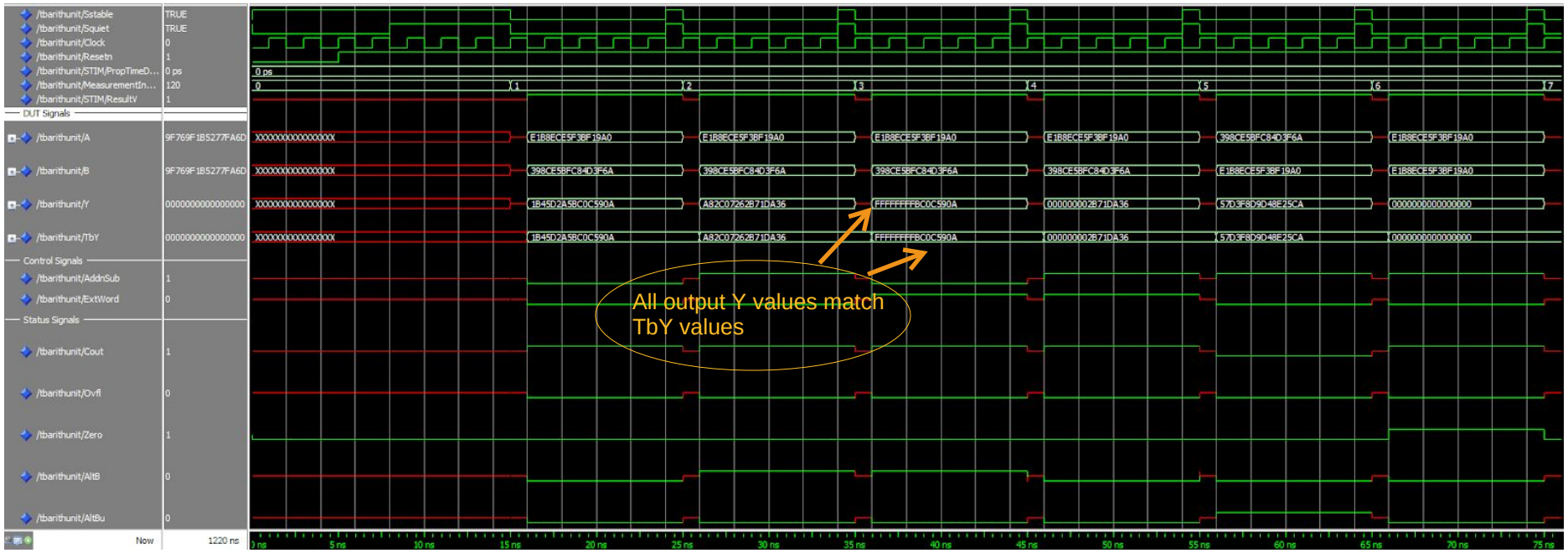
Functional simulation: Logic Unit 2

The diagram shows measurements 458 up to 464.We can also verify, each of the Y values and TbY values match for each measurement.



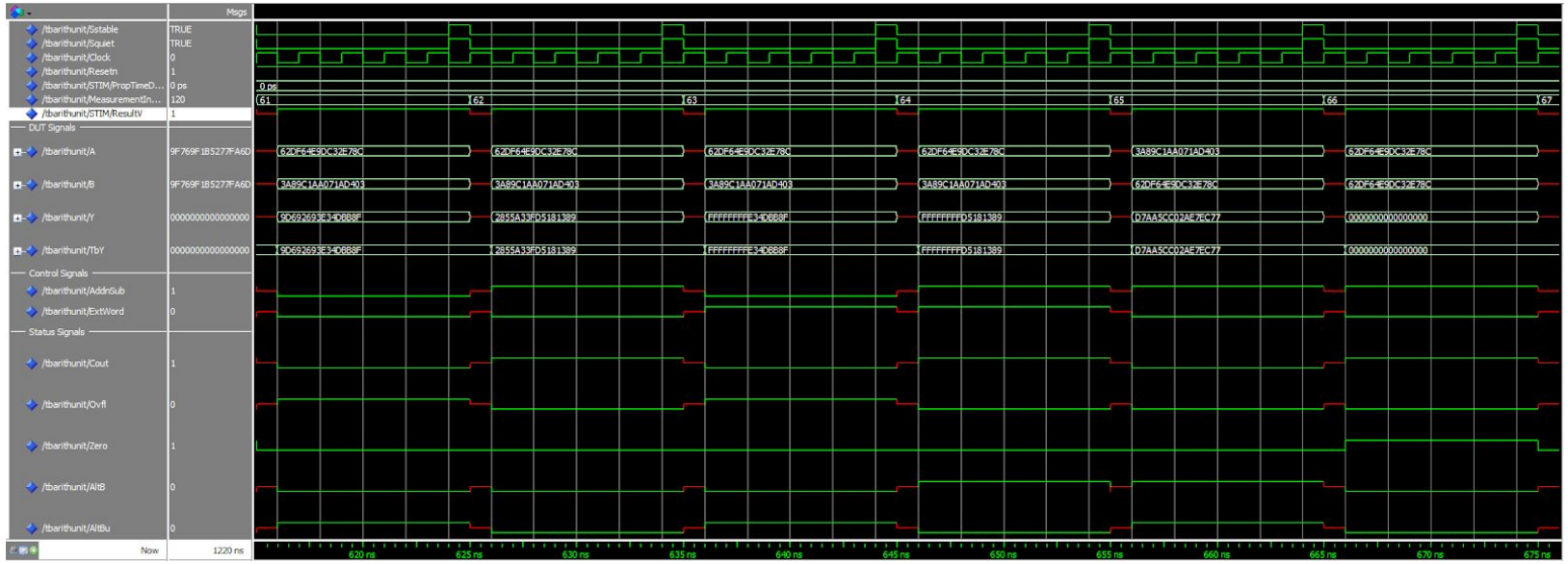
Functional simulation: Logic Unit 3

The diagram shows the last three measurements 770, 771 and 772. From all three diagrams we can verify that our output Y values match all the TbY values which verifies the functionality of our logic unit.



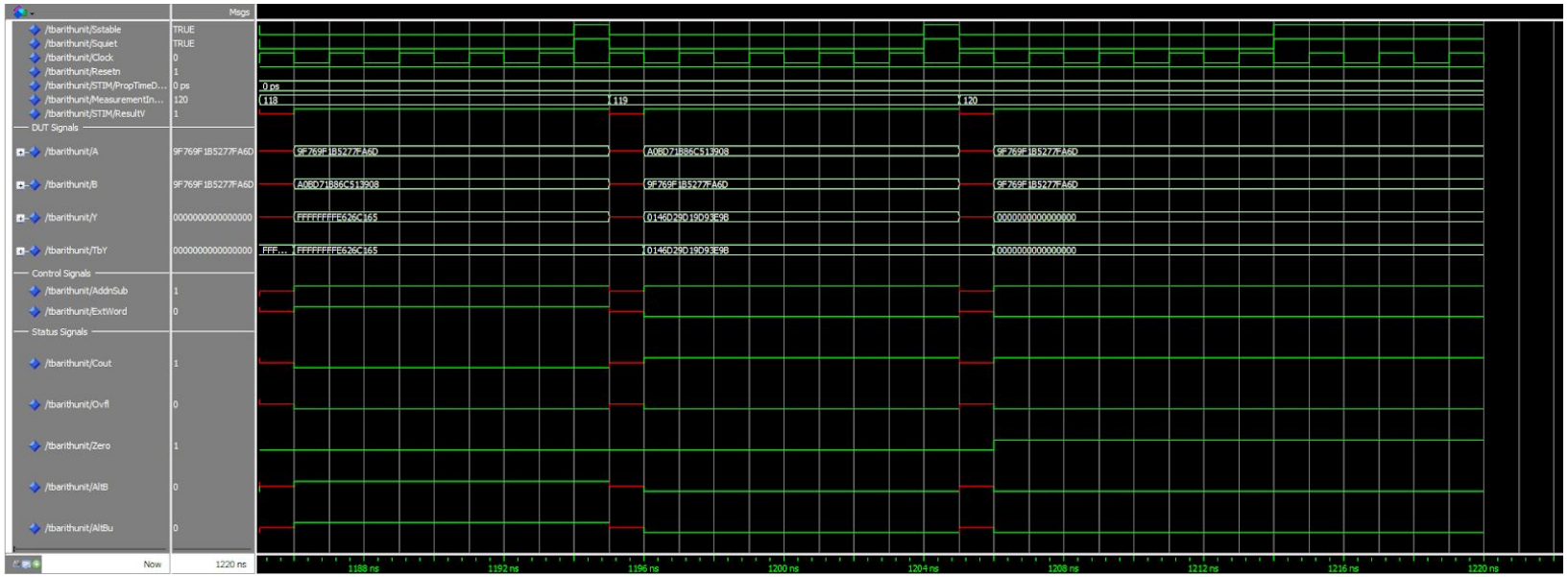
Functional simulation: Arithmetic Unit 1

The diagram shows the first 6 measurements of our arithmetic unit. All our output Y values match the given TbY values and the status signal outputs are correct.



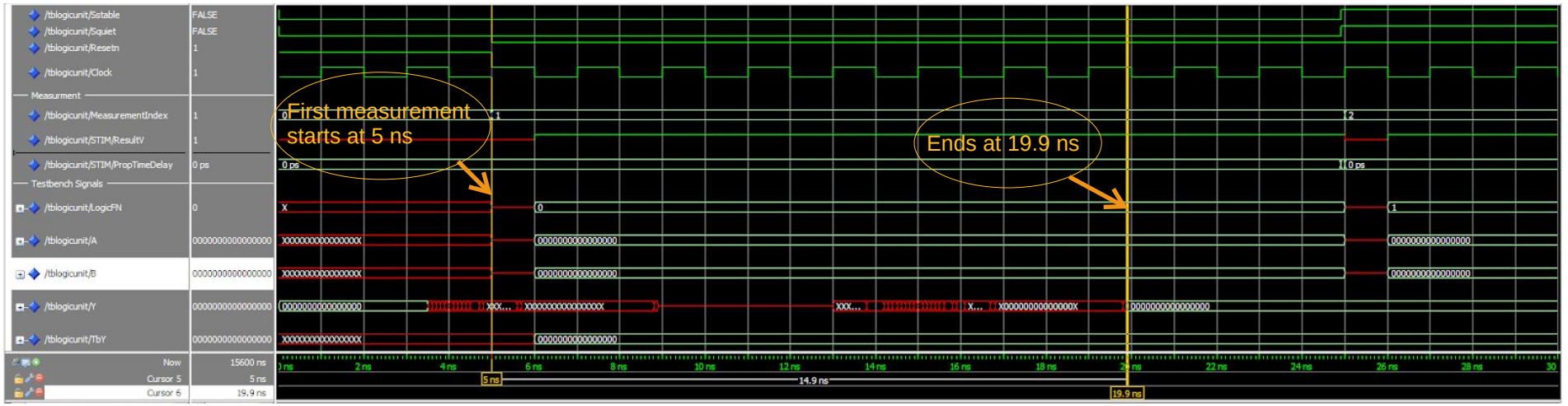
Functional simulation: Arithmetic Unit 2

The diagram shows measurements 61 up to 66 of our arithmetic unit. All our output Y values match the given TbY values and the status signal outputs are correct.



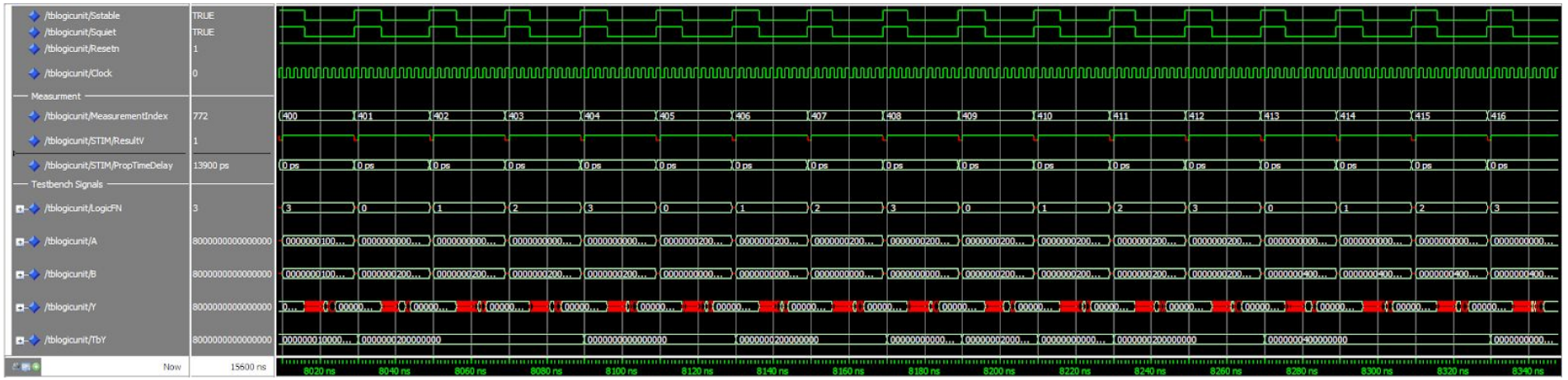
Functional simulation: Arithmetic Unit 3

The diagram shows the last three measurements of our arithmetic unit. All our output Y values match the TbY output values and the status signal outputs are correct. Thus verifying the functionality of our arithmetic unit.



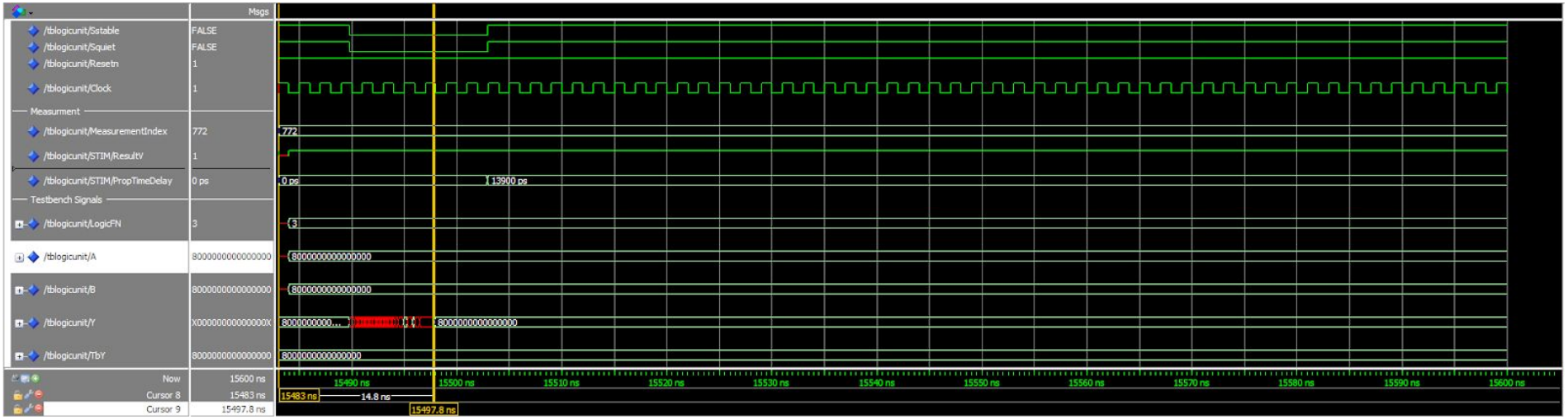
Timing simulation: Logic Unit 1

This diagram shows the close up view of the waves of the timing simulation around measurement #1. For measurement #1, the propagation delay is 14.9ns.



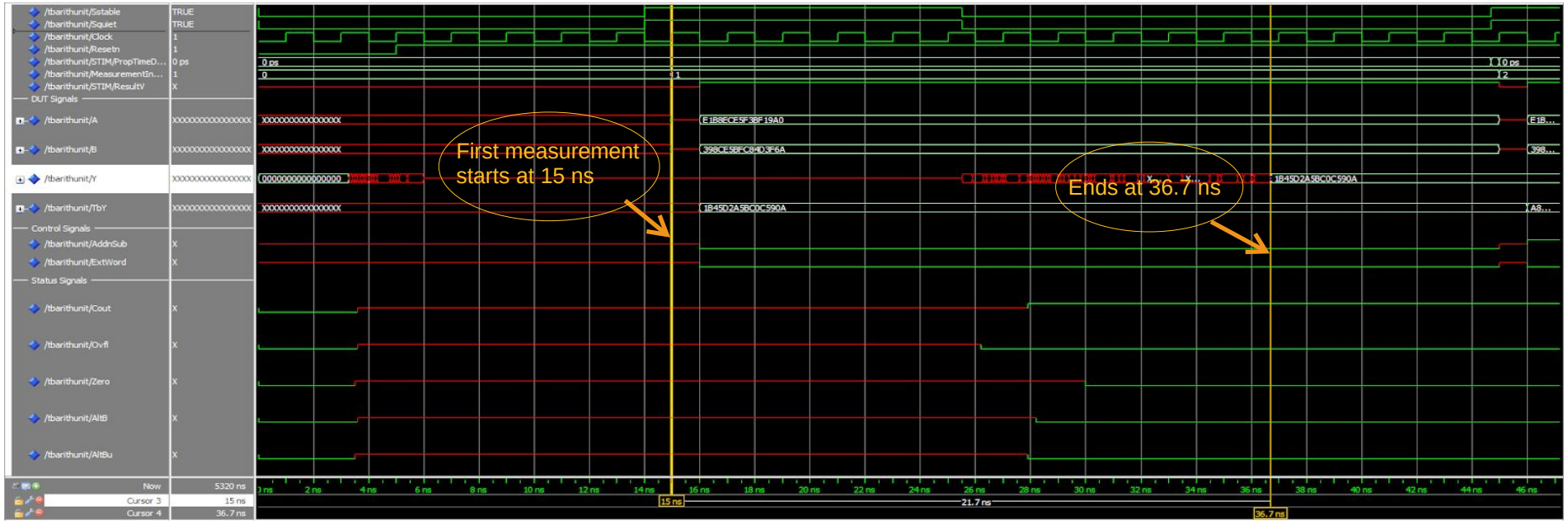
Timing simulation: Logic Unit 2

This diagram shows the close up view of the waves of the timing simulation from measurement #400 to end of #416.



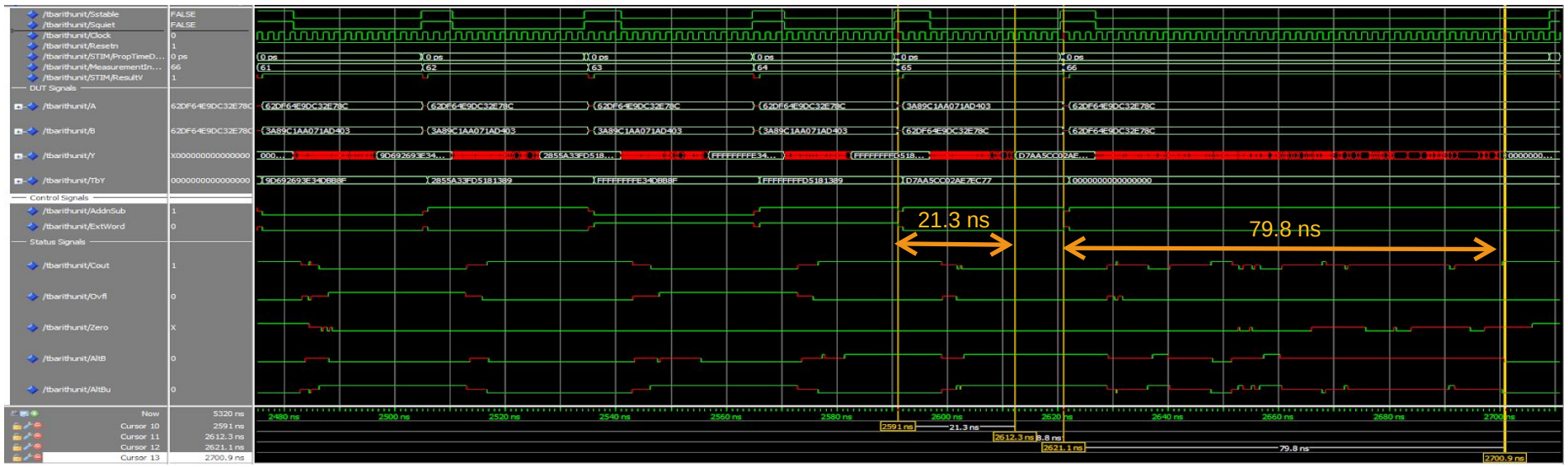
Timing simulation: Logic Unit 3

This diagram shows the close up view of the waves of the timing simulation for measurement #772. For measurement #772, the propagation delay is 14.9ns.



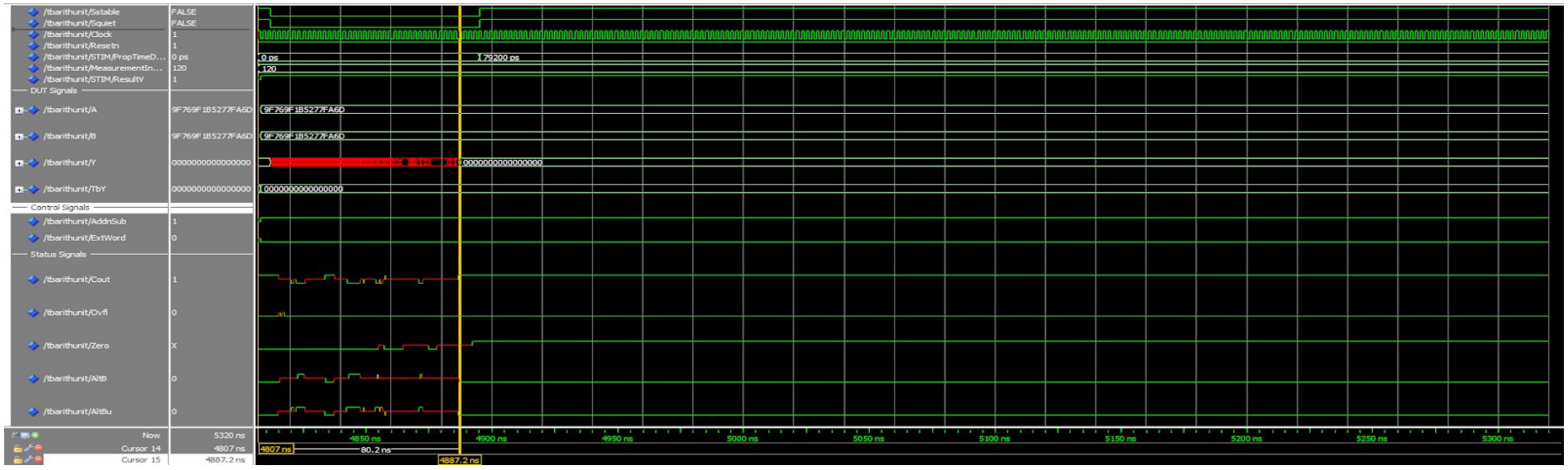
Timing simulation: Arithmetic Unit 1

The diagram shows the measurement of the propagation time delay of measurements #1. The propagation time delay measured is 21.7 ns.



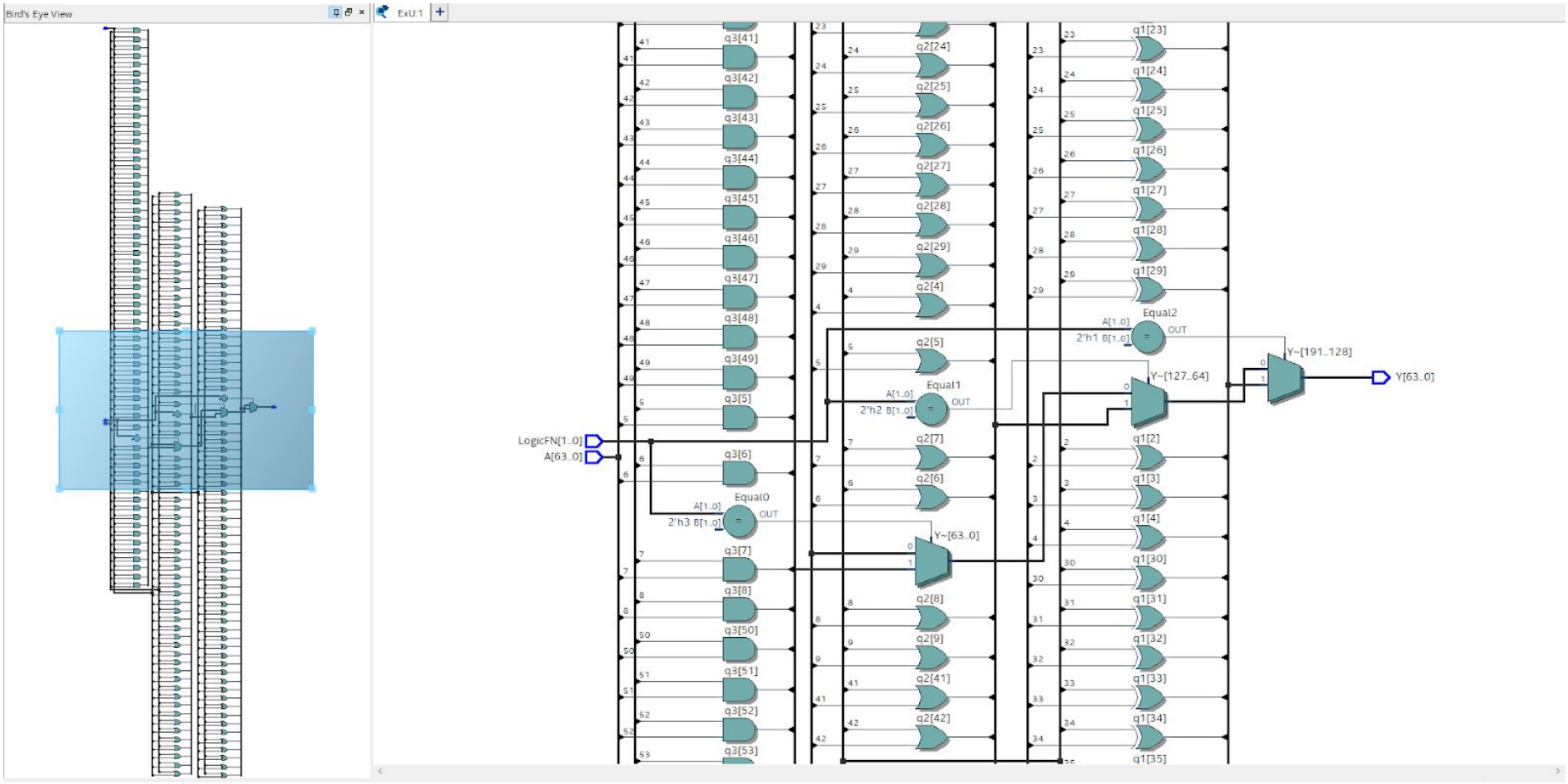
Timing simulation: Arithmetic Unit 2

The diagram shows the measurement of the propagation time delay of measurements #65 and #66. The propagation time delays measured are 21.3ns and 79.8 ns respectively.

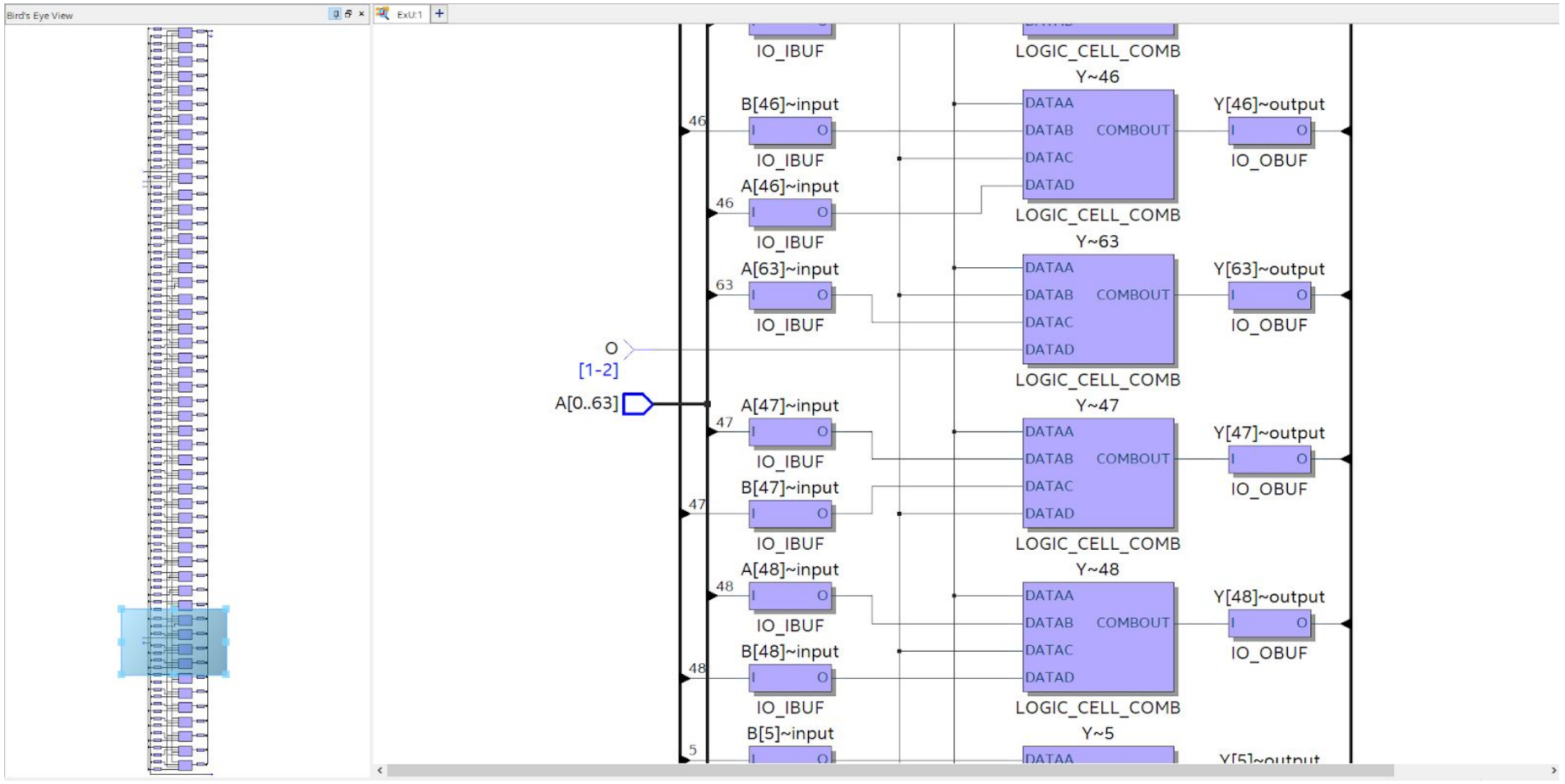


Timing simulation: Arithmetic Unit 3

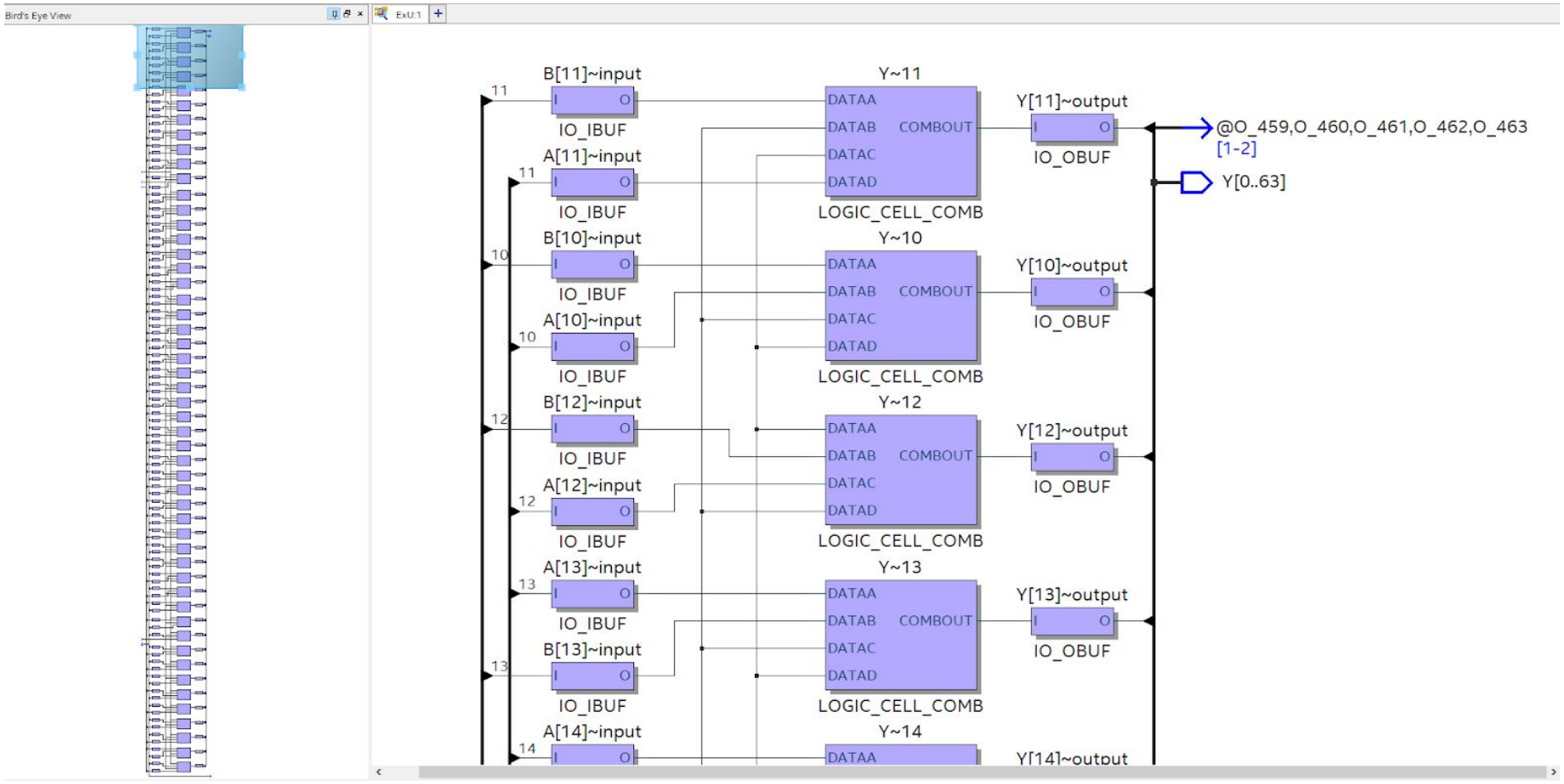
The diagram shows the measurement of the propagation time delay of measurements #120. The propagation time delay measured is 80.2 ns.



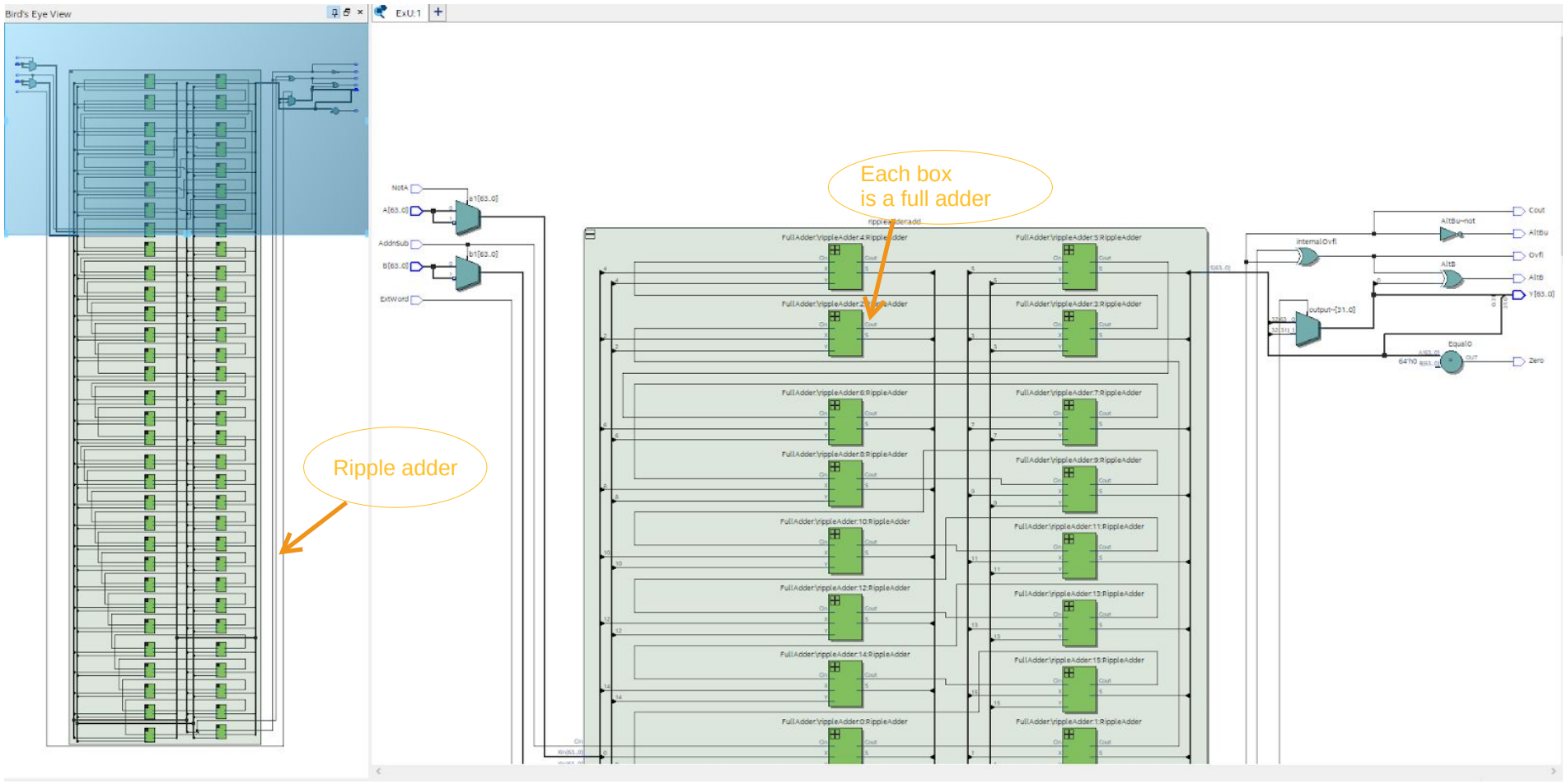
Logic Unit RTL:
The zoomed in picture shows the input LogicFN signal with 64-bit input operand A, into the overall general structure of the circuit. The result is shown to pass on to a MUX into output Y at the end.



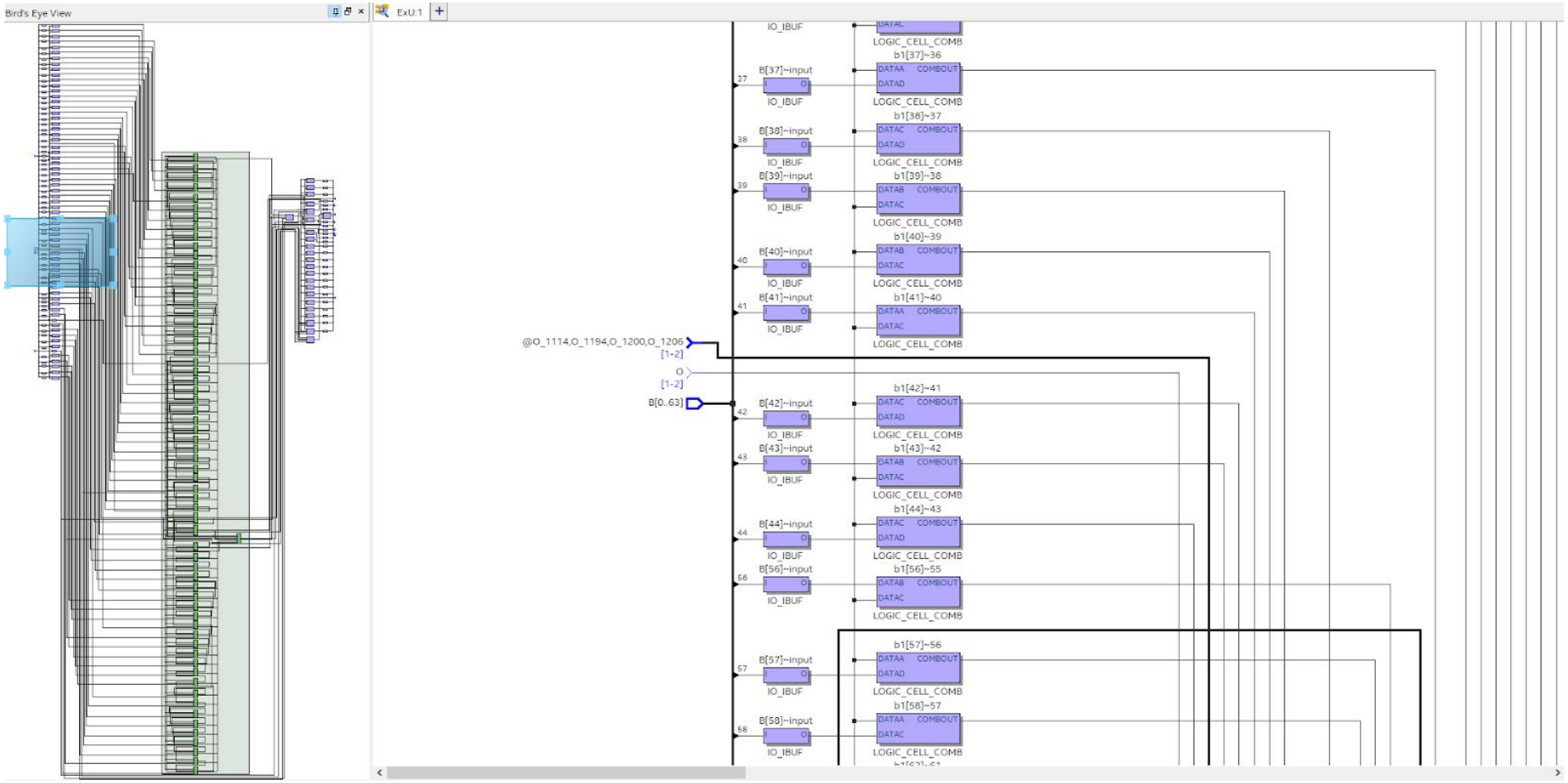
Logic Unit Post Fit:
Shows the 64-bit input of operand A into the logic unit and control signal input.



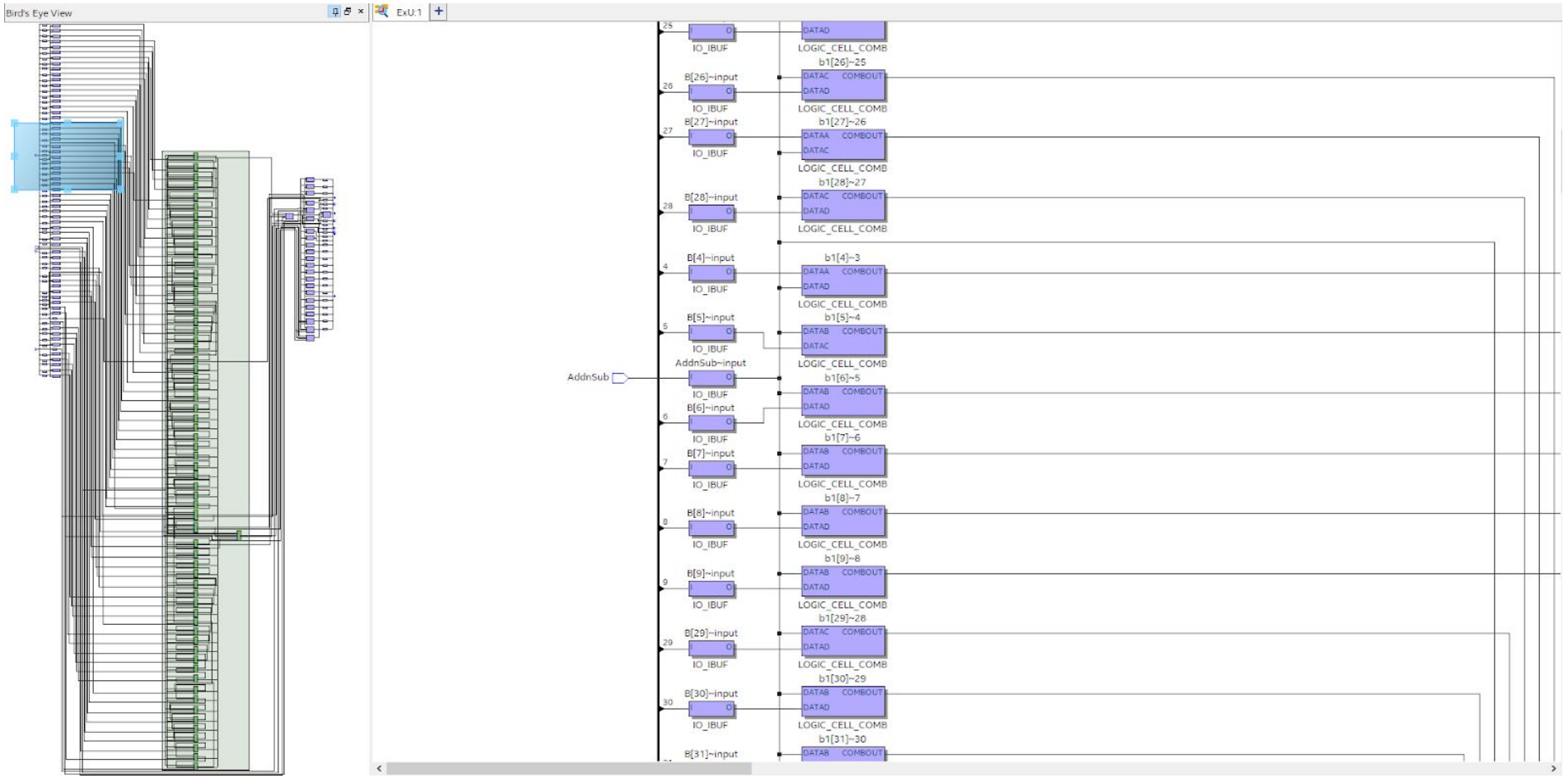
Logic Unit Post Fit:
The diagram above shows the general structure of the circuit and the resulting output Y at the top right.



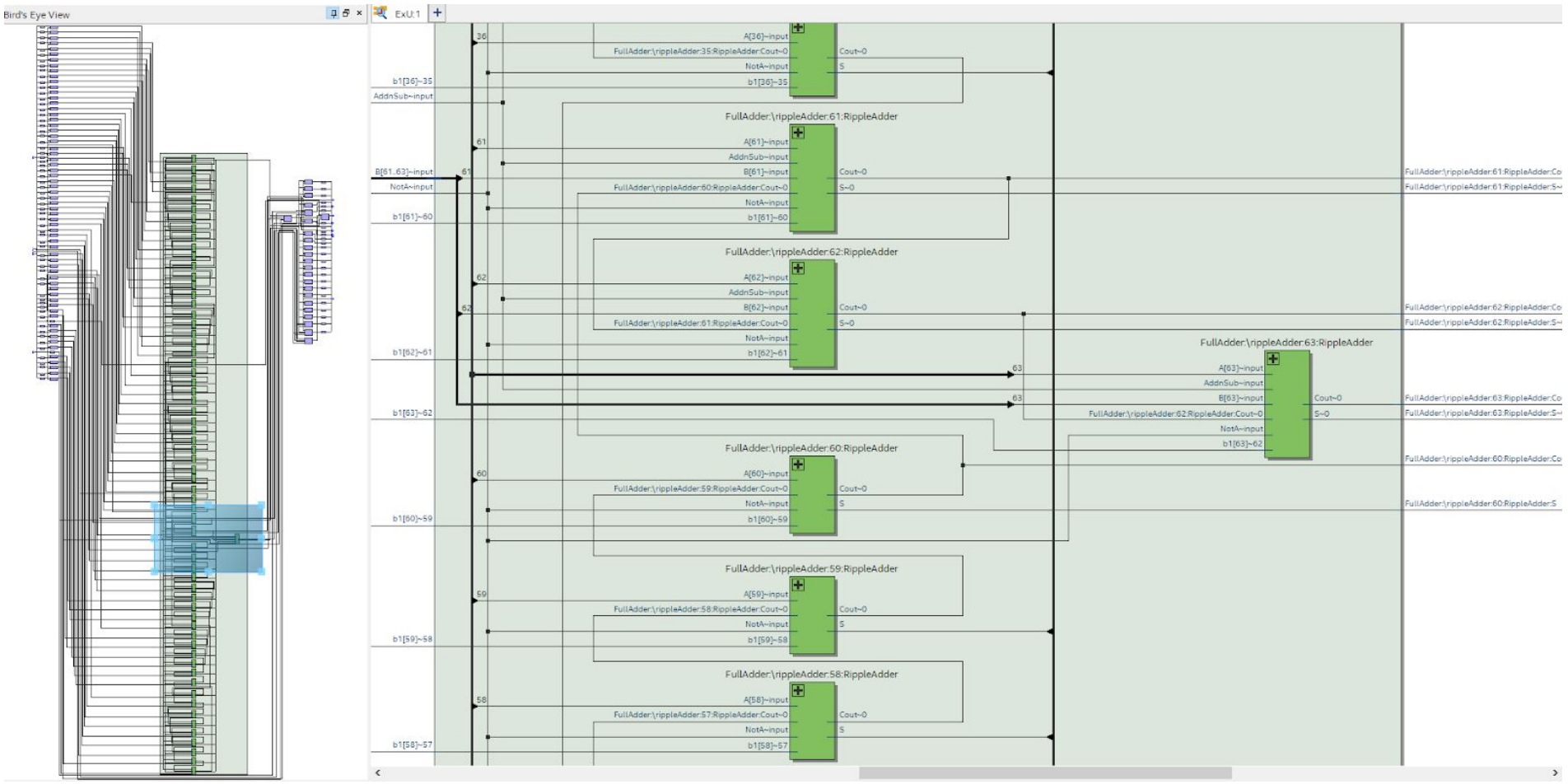
Arithmetic Unit RTL viewer: Ripple adder and general structure of the arithmetic unit.



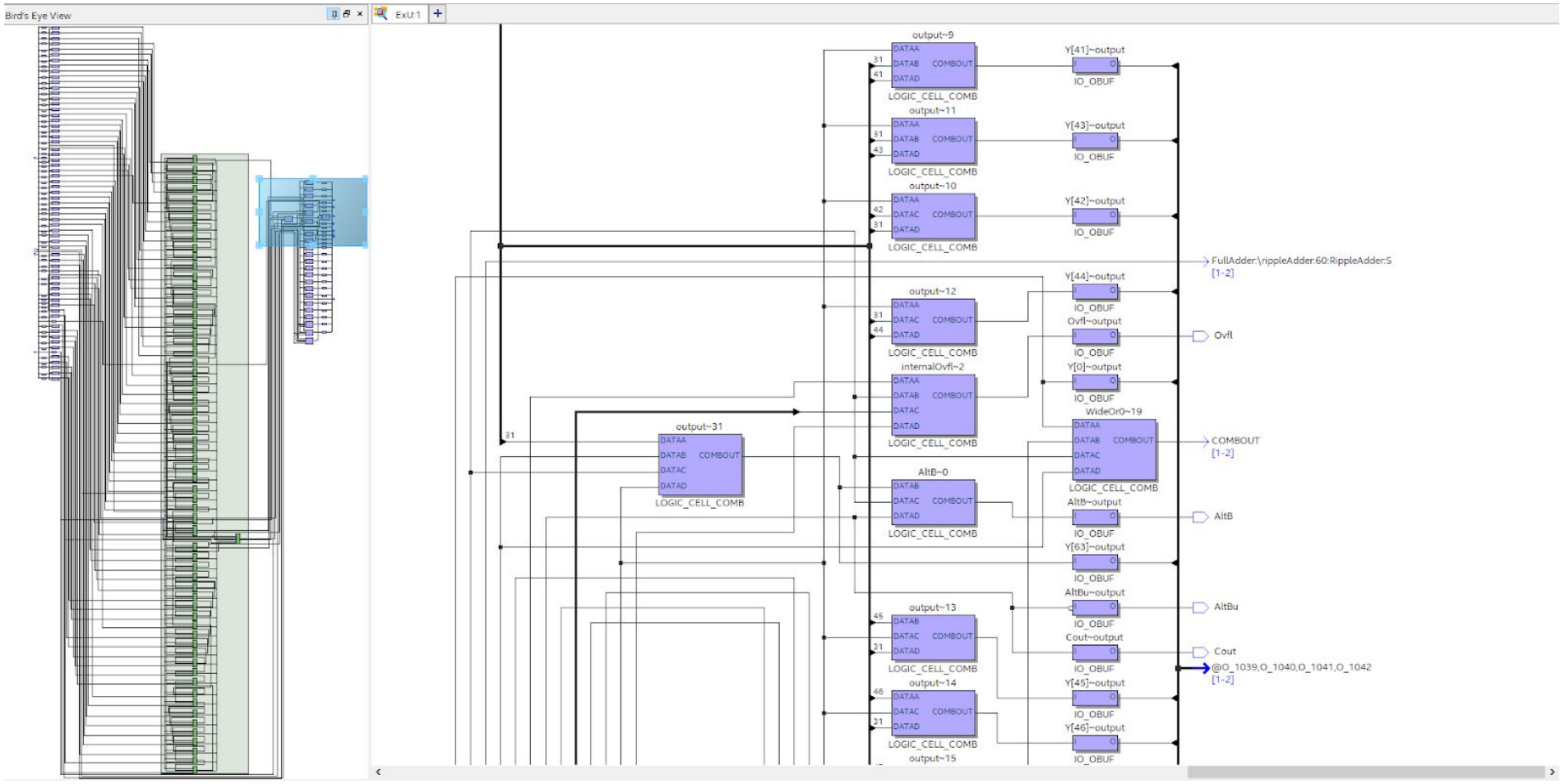
Arithmetic Unit Post Fit:
The following circuit shows the 64-bit input operand B into the arithmetic unit.



Arithmetic Unit Post Fit:
The diagram above shows the input of the Addn'Sub control signal into the circuit.



Arithmetic Unit Post Fit: Ripple adder output



Arithmetic Unit Post Fit:
This screenshot shows the outputs of the Status signals. The Ovfl and Cout signal coming out from the ripple adder is passed on the above circuit to help calculate the output of the AltBu and AltB signals.