Patrick Balcombe, ID- 140387713

# **ECS615U-** Laboratory Session 1

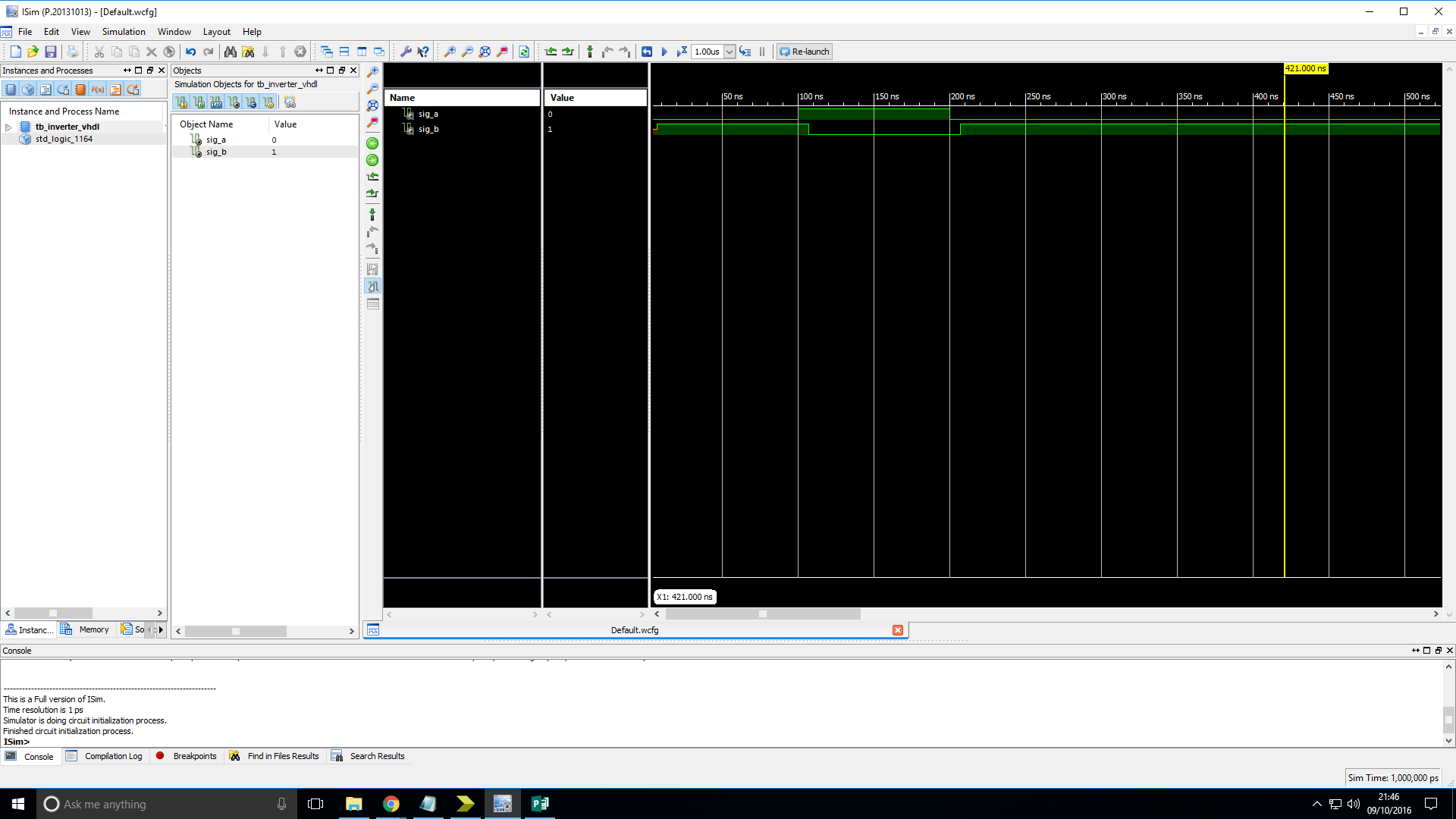
# **Abstract**

In this lab we implement a series of devices including gates, multiplexers and adders in VHDL code. We then go on to test them using a test bench file that cycles through all possible combinations of inputs. Then using the Modelsim software to check the outputs results we confirm that the output is as expected for that particular device.

# Timing Diagrams

In this section we individually run the test bench files for each device in Modelsim to check the timing diagram of the inputs and outputs matches its truth table.

Inverter**:**

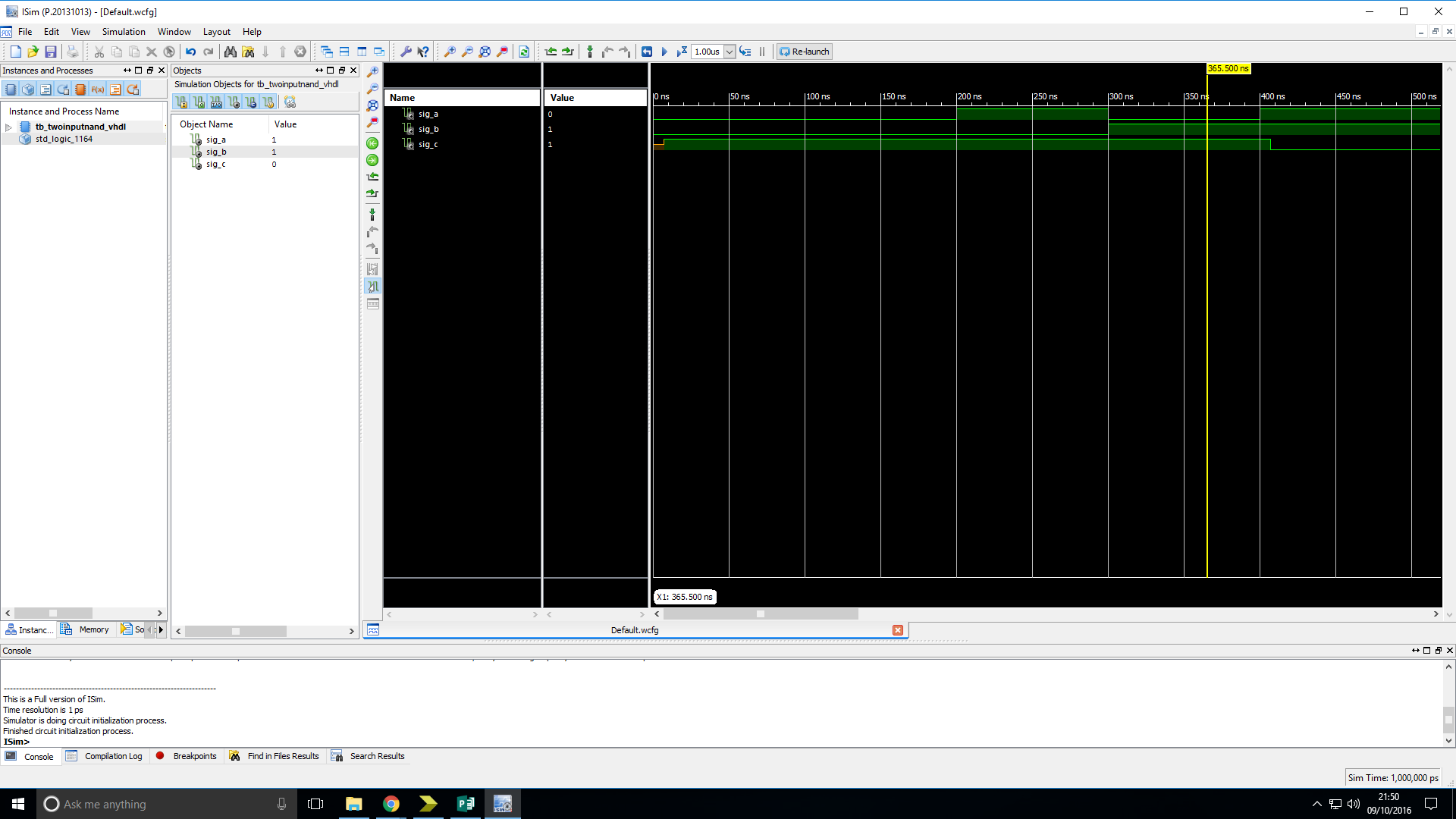
Truth table:

|  |  |
| --- | --- |
| Input | Output |
| a | b |
| 0 | 1 |
| 1 | 0 |

Figure 1: Simulation timing diagram of Inverter gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input NAND gate:

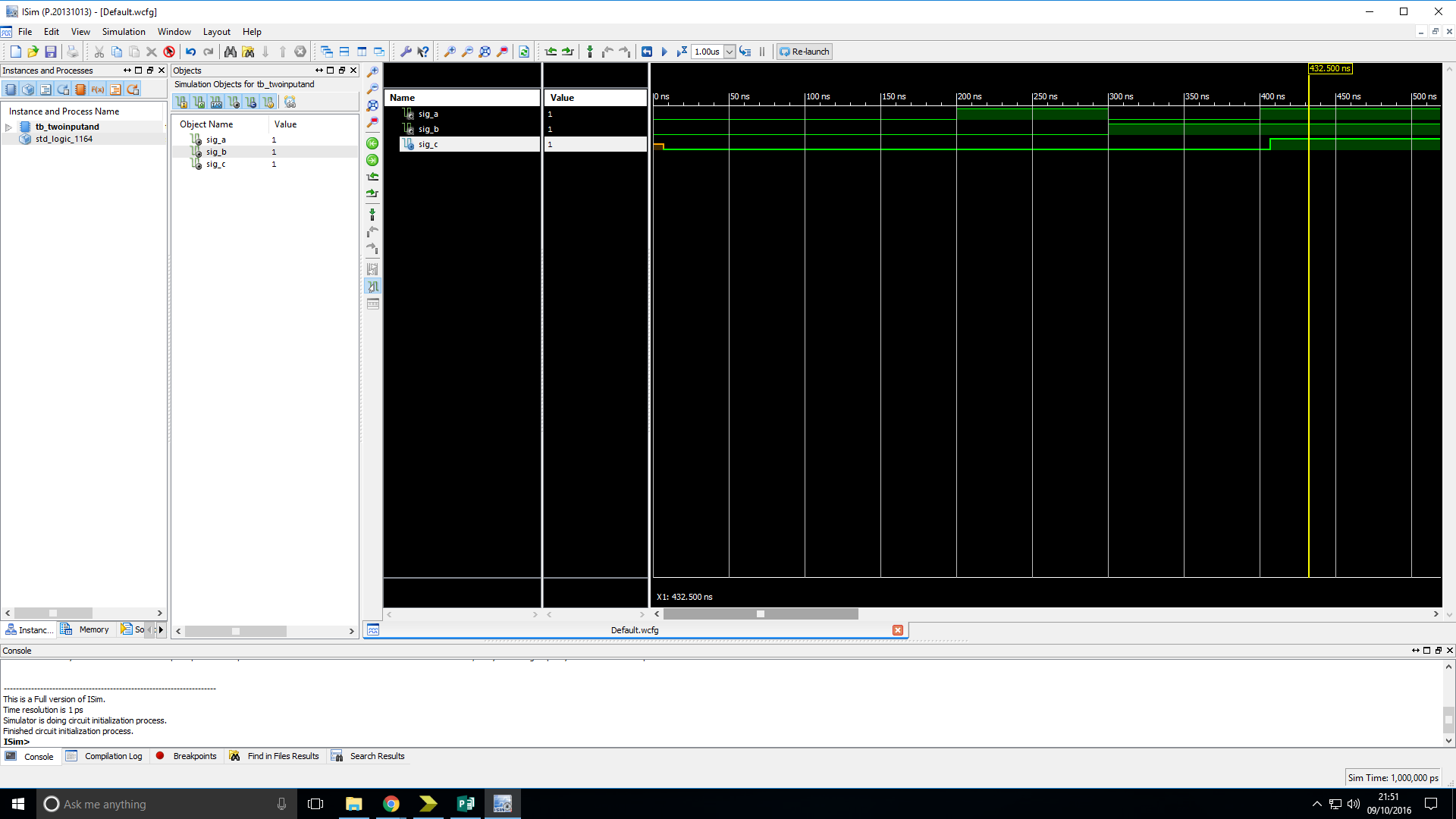
Truth table:

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| a | b | C |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Figure 2: Simulation timing diagram of the two-input NAND gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input AND gate:

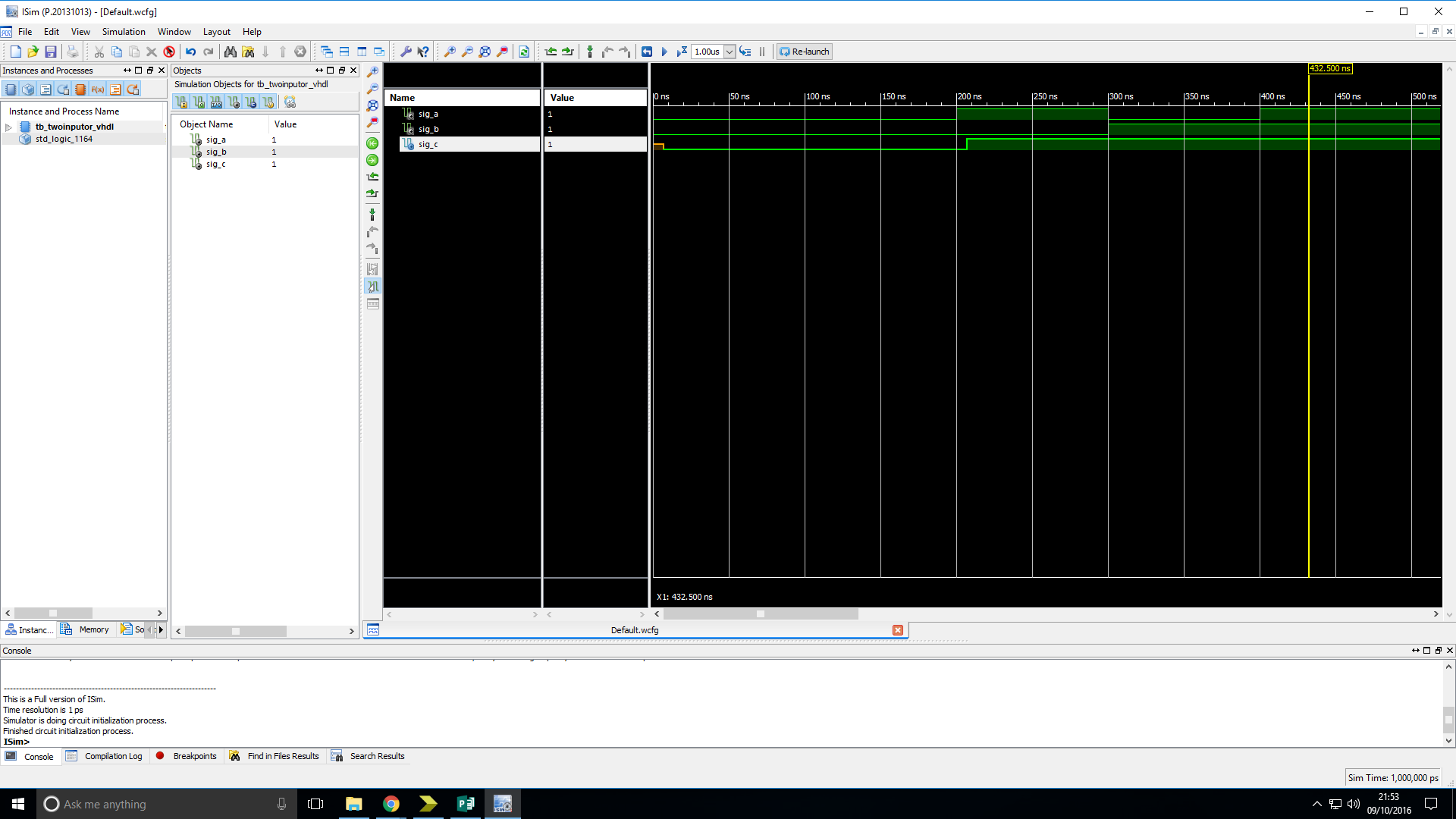
Truth table:

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| a | b | C |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Figure 3: Simulation timing diagram of two-input AND gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input OR gate:

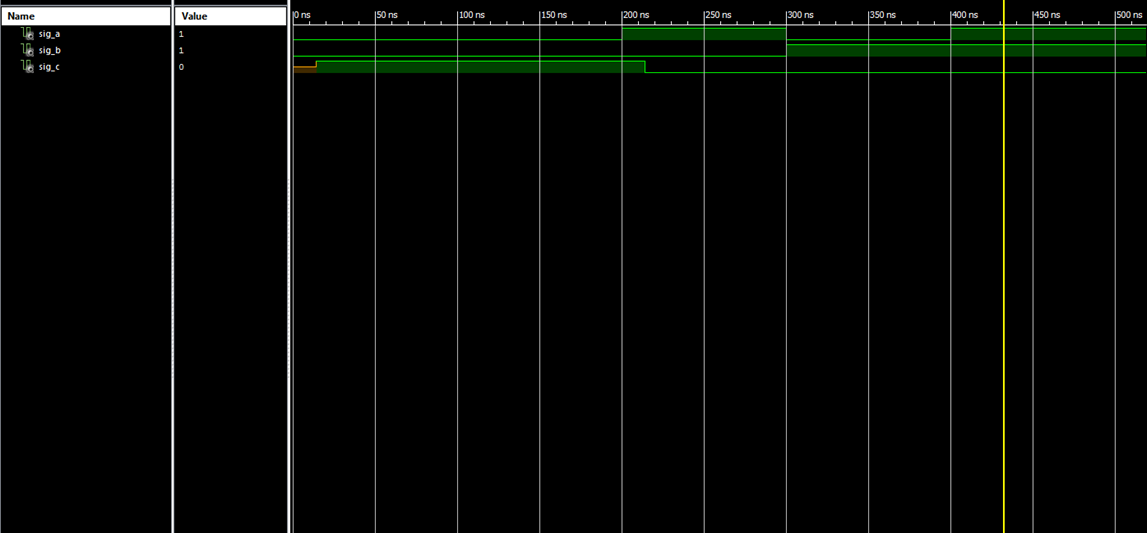
Truth table:

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| a | b | C |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Figure 4: Simulation timing diagram of two-input OR gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input XOR gate

Truth table:

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| a | b | C |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

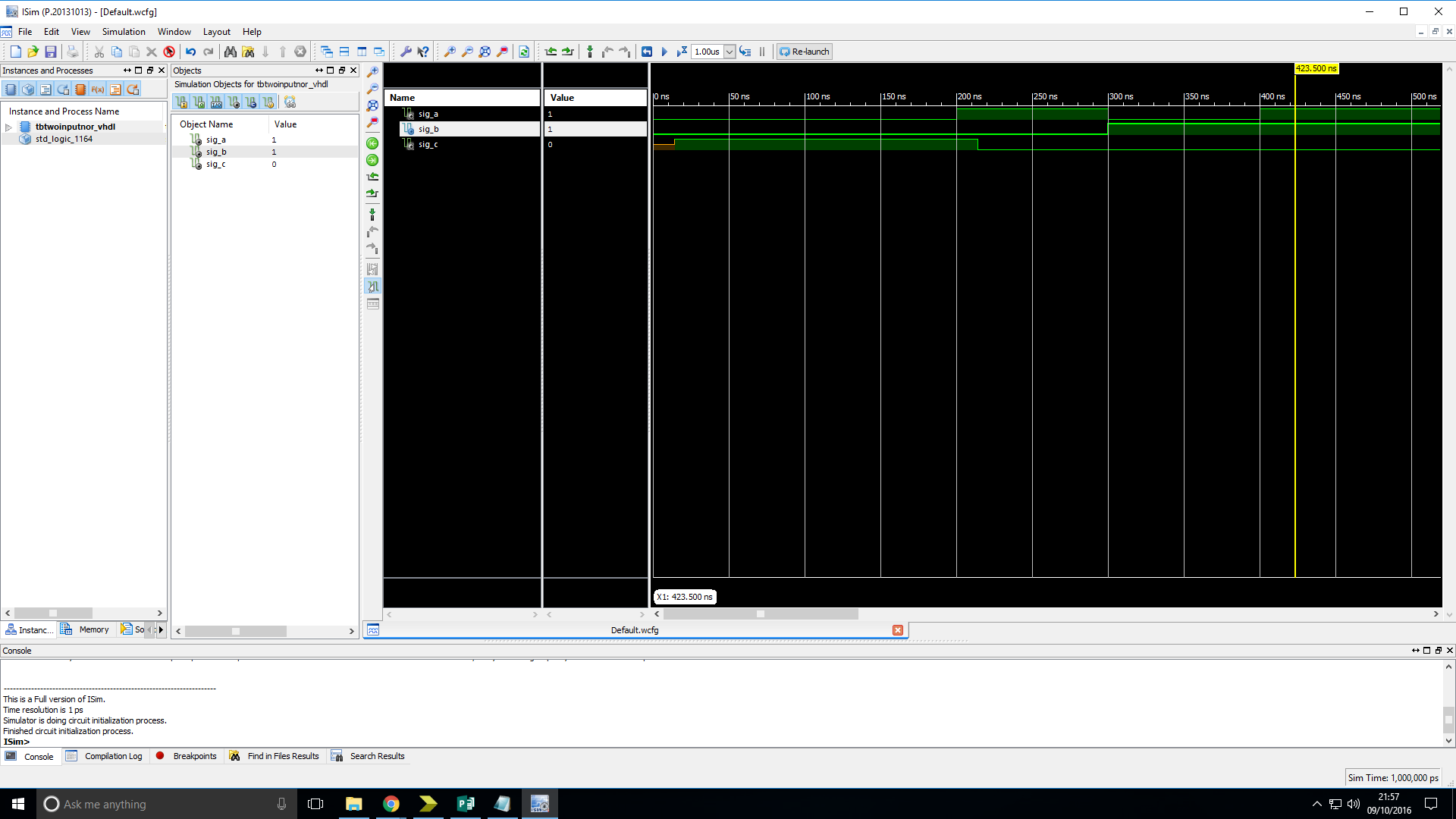
Figure 5: Simulation timing diagram of two-input XOR gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input NOR gate:

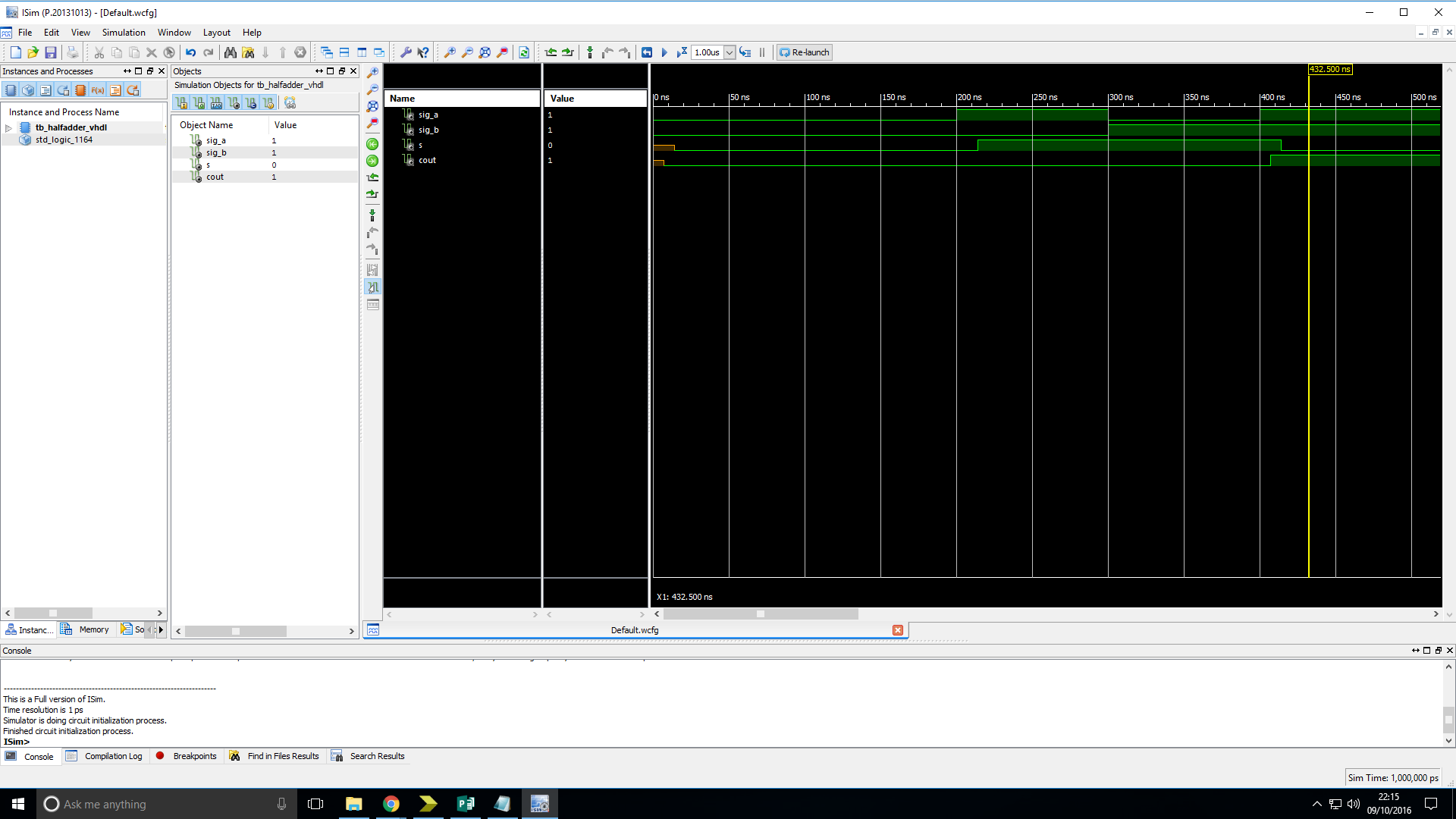
Truth table:

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| a | b | C |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Figure 6: Simulation timing diagram of two-input NOR gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# A Half Adder:

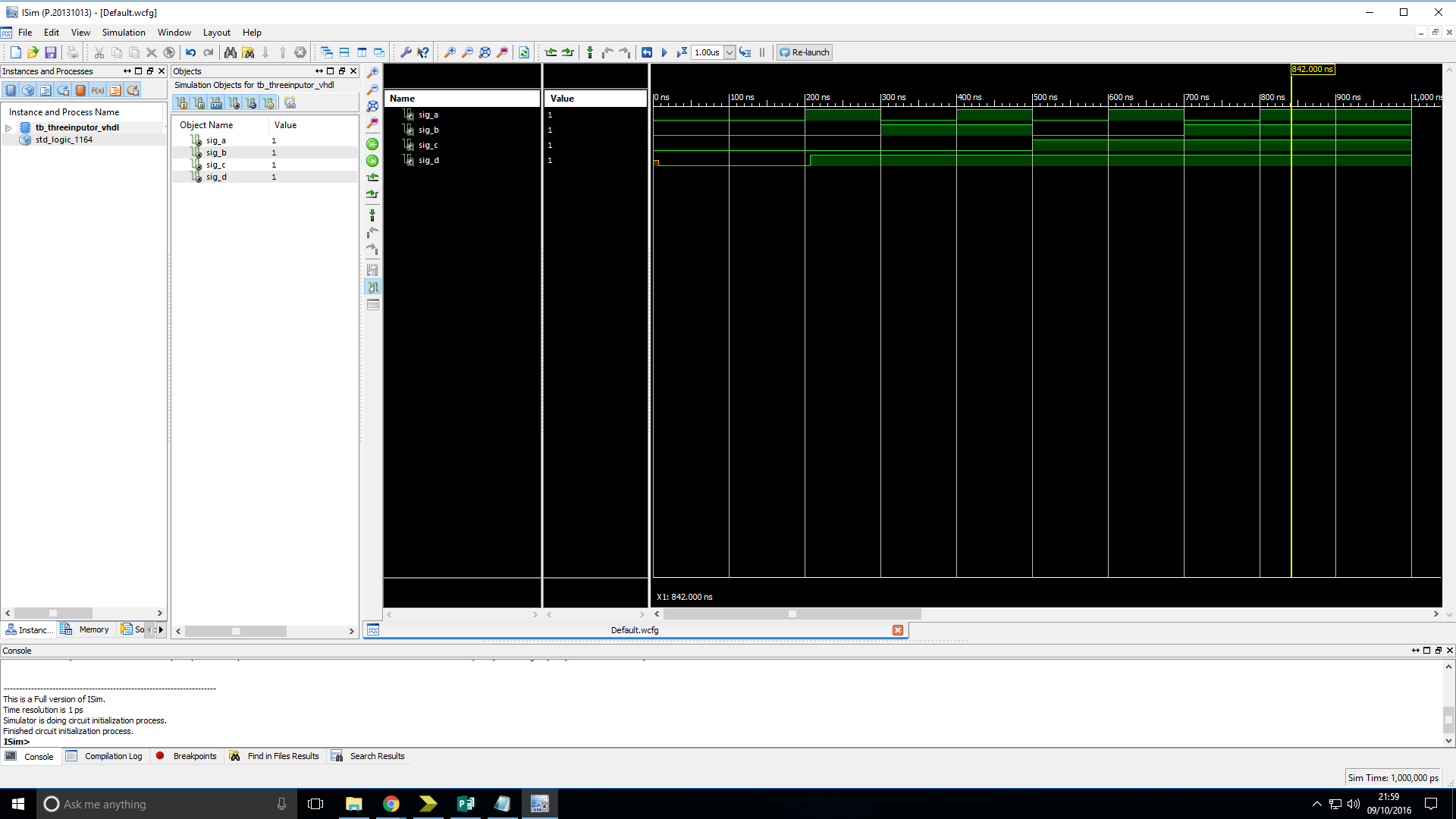
Truth table:

|  |  |  |  |
| --- | --- | --- | --- |
| Input | | Output | |
| a | b | s | Cout |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Figure 7: Simulation timing diagram of Half Adder showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Three-input OR gate:

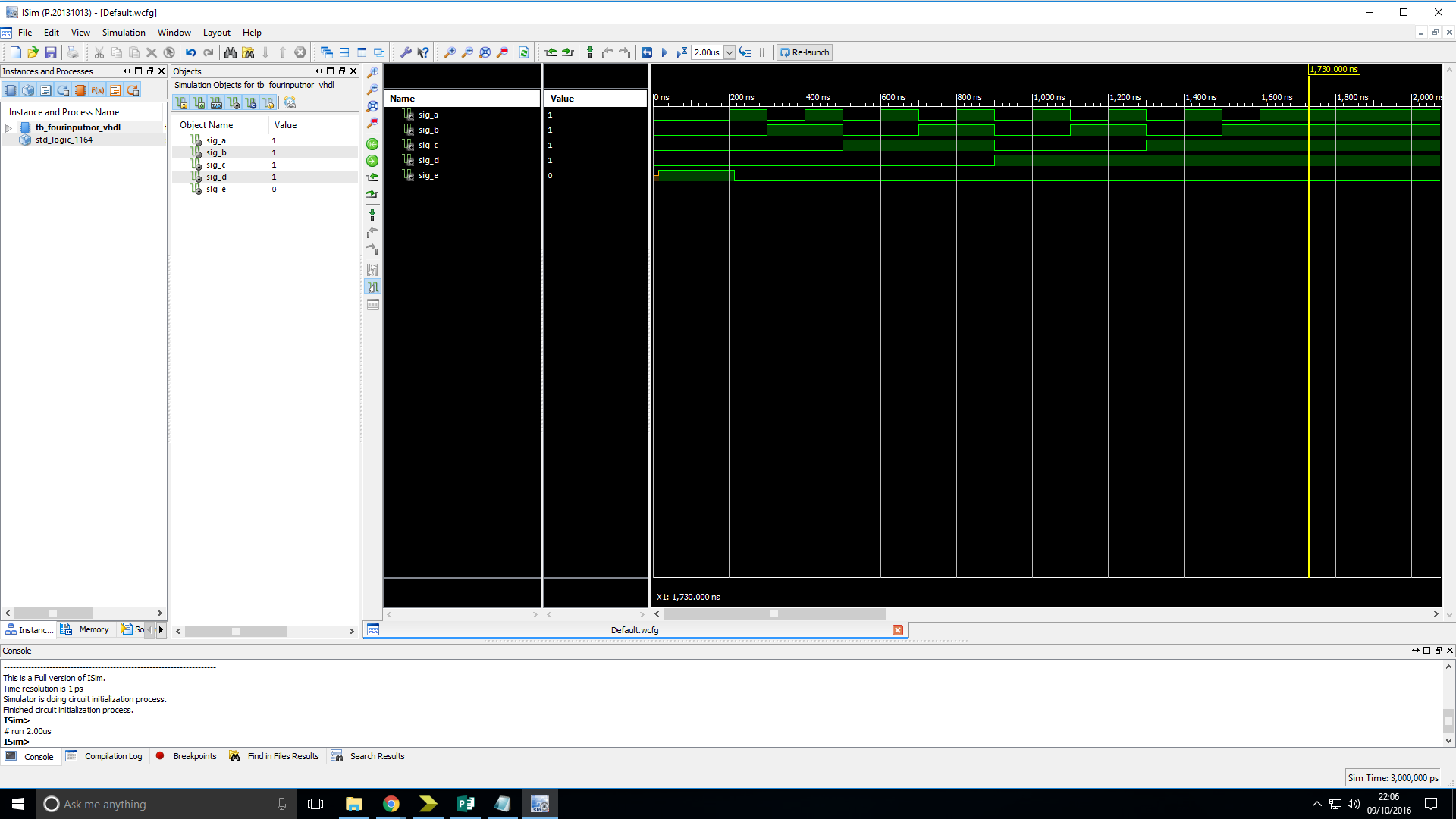
Truth table:

|  |  |  |  |
| --- | --- | --- | --- |
| Input | | | Output |
| a | b | c | d |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Figure 8: Simulation timing diagram of three-input OR gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Four-input NOR gate:

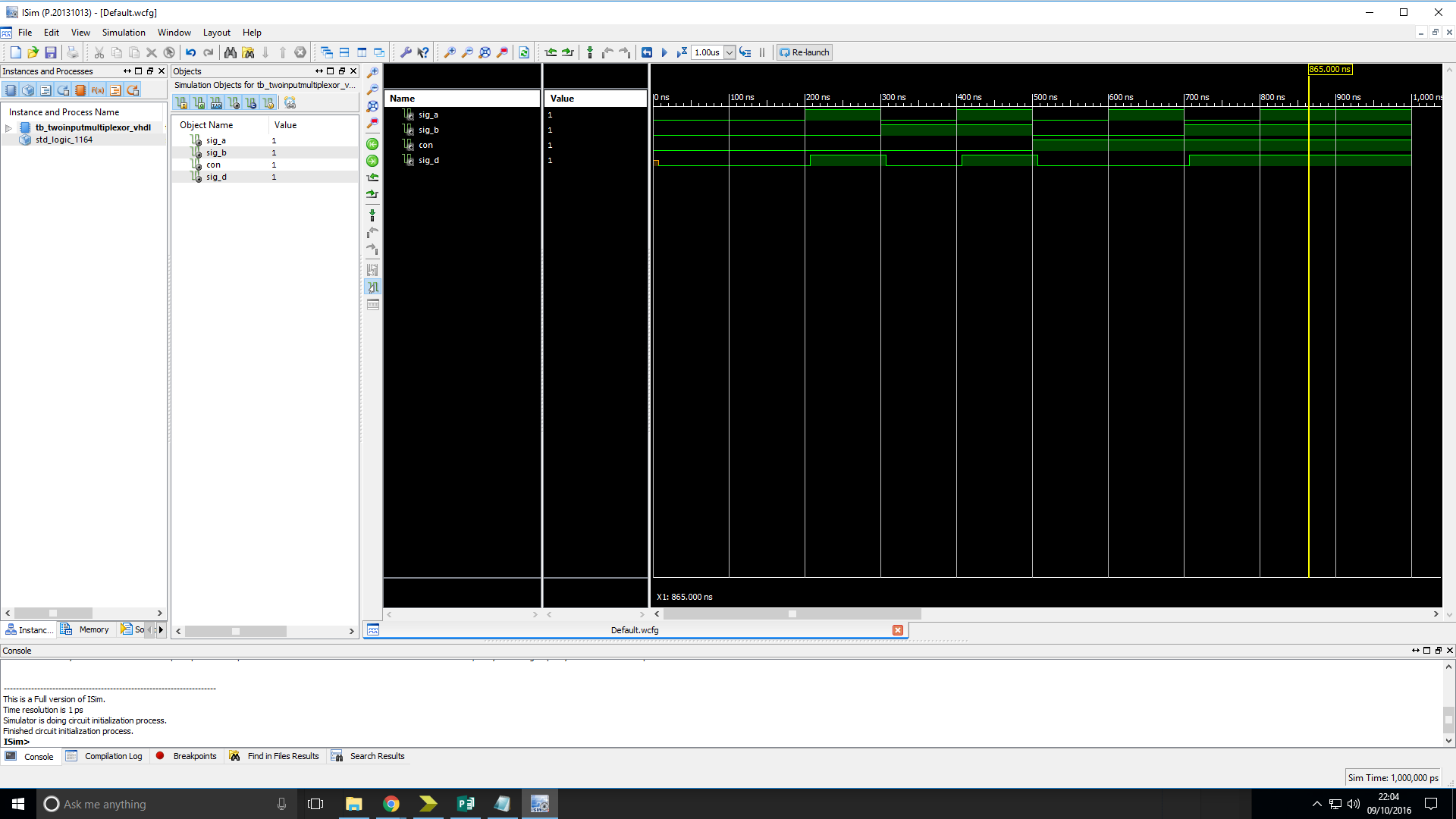
Truth table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input | | | | Output |
| a | b | c | d | e |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

Figure 9: Simulation timing diagram of four-input NOR gate showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Two-input Multiplexer:

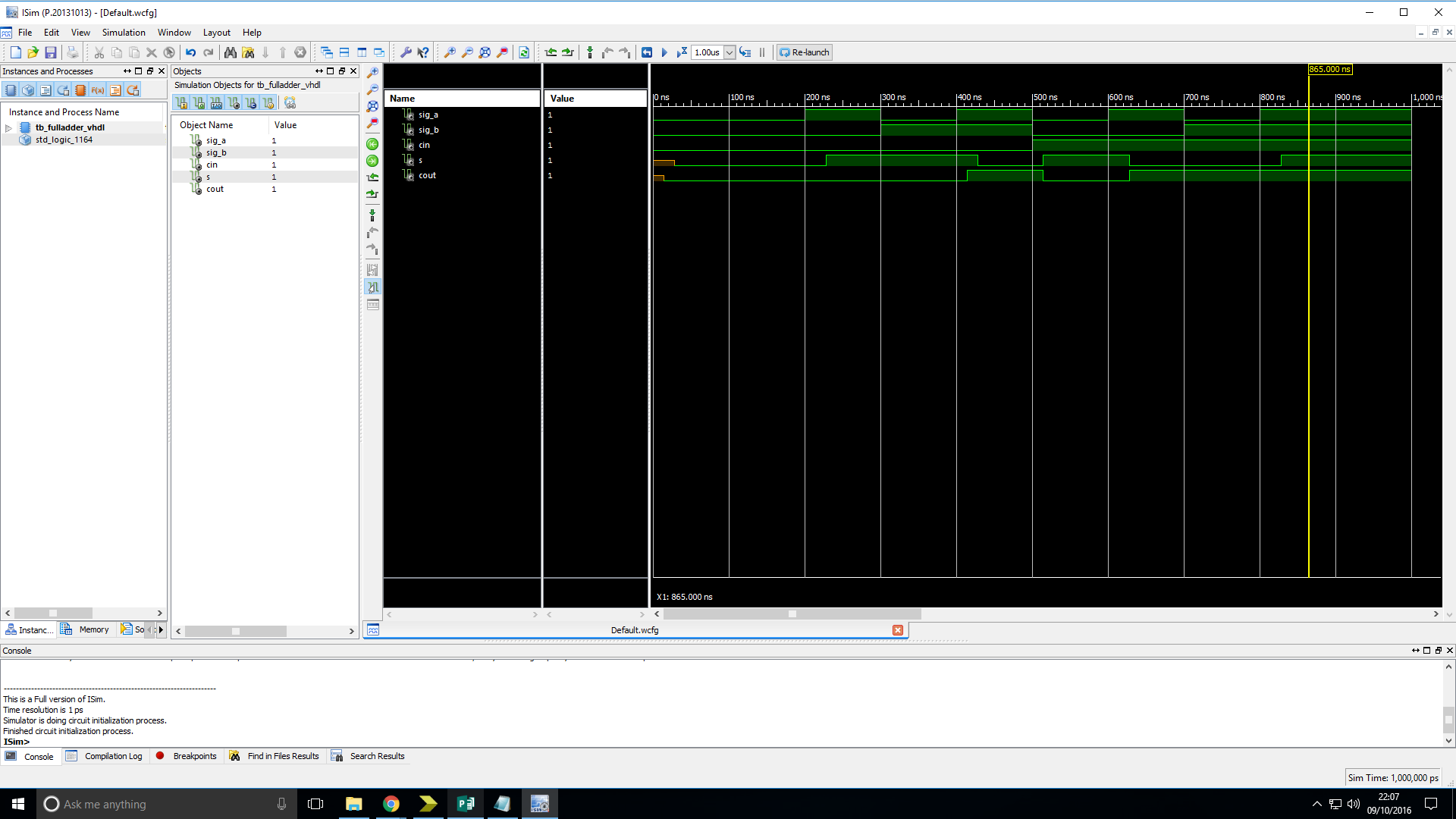
Truth table:

|  |  |  |  |
| --- | --- | --- | --- |
| Input | | | Output |
| a | b | Con | d |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Figure 10: Timing diagram of two-input Multiplexer showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# ­­­­­Full Adder:

Truth table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input | | | Output | |
| a | b | Cin | s | Cout |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Figure 11: Timing diagram of Full Adder showing its behaviour when tested with all possible inputs.

Analysis: The output of the device is as expected.

# Overall Analysis

All devices that were tested matched their truth tables for all possible inputs. The delay in some outputs taking a value at the start (meaning they are uninitialized) for the first few Nano seconds is due to the delays put into the code in order to make it a more realistic model.

# Conclusion

For this lab we implemented various simple devices in VHDL code as well as an associated test bench file that cycled through all possible combinations of inputs. Then using the Modelsim software we checked the output and confirmed that they were as expected for each devices truth table.