Real-Time Hardware Sorter (RTHS) Implementation

This project was carried out as part of a research initiative for the **FPGA Design course**, where we explored cutting-edge hardware solutions for real-time sorting. Our main focus was the study, translation, and implementation of the IEEE article titled "RTHS: A Low-Cost High-Performance Real-Time Hardware Sorter."

The research involved thoroughly analyzing the paper, translating its content into Persian for academic use, and implementing the proposed architecture using FPGA simulation tools. By doing so, we aimed to understand the practical and theoretical aspects of hardware-based sorting algorithms, gaining hands-on experience in FPGA design and real-time hardware applications. The final step of the project included validating the design through simulation and comparing its performance with other existing methods.

Overview of the Paper

The RTHS design presented in the article proposes a highly efficient hardware sorter optimized for FPGA implementation. It introduces the Multidimensional Sorting Algorithm (MDSA), which sorts input data in six phases using a matrix-based approach. The key components of the design include:

- **Dual-Mode Pipeline Bitonic Networks (DPBNs):** These enable ascending and descending sorting with reduced critical path delays.
- **Implicit Switches:** Used for matrix transformations without significant hardware overhead.
- **Control Unit:** Oversees the sorting process and manages data flow through the six phases.

The proposed design demonstrates substantial reductions in hardware resource utilization while maintaining high performance. It is particularly suitable for applications requiring Min/Max queues, continuous sorting, or large-scale data handling in real-time systems.

Project Workflow

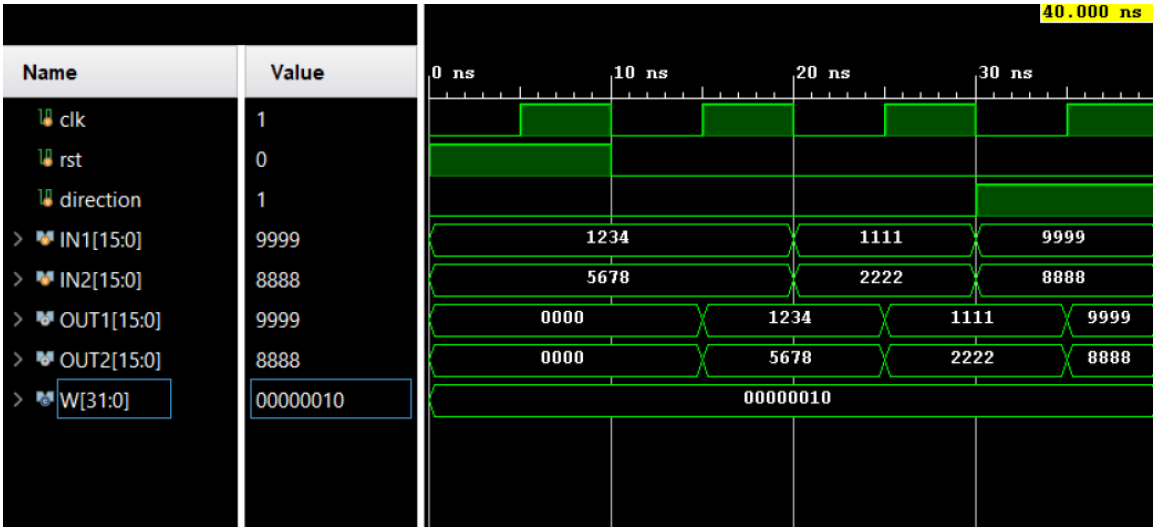
- **Paper Study and Translation:**
  The project began with a detailed study of the RTHS algorithm and architecture as described in the IEEE article. To ensure a deeper understanding and facilitate presentations, the article was fully translated into Persian. This step helped in grasping the core concepts and preparing for the practical phase.

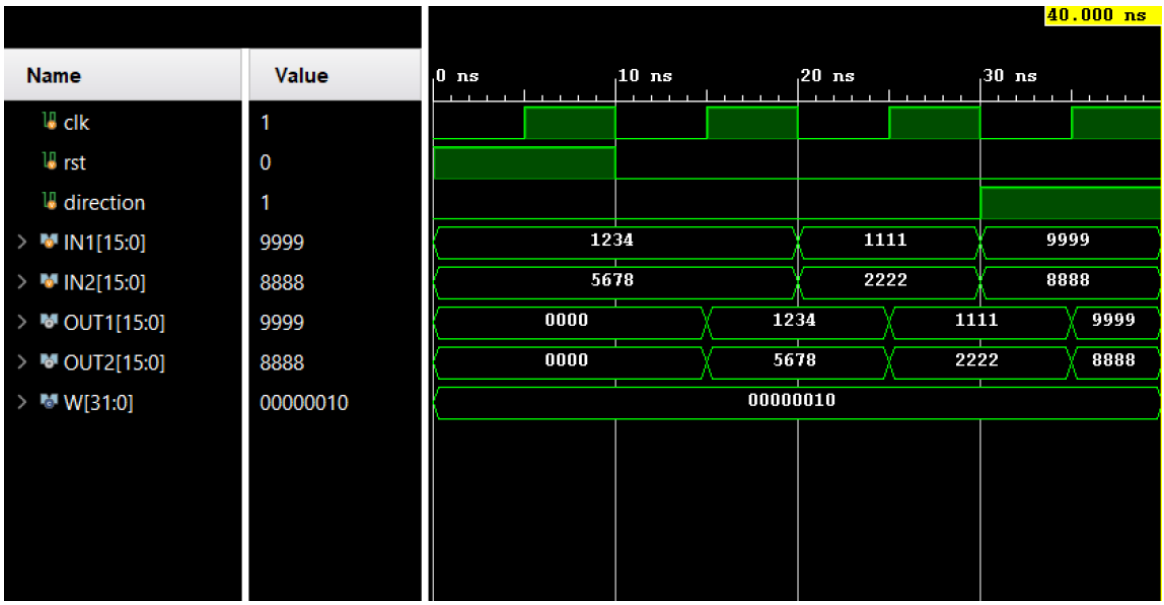
- **FPGA Simulation:**
  While the Verilog code for the RTHS architecture was already available, we focused on its implementation and simulation in the Xilinx Vivado environment. The provided code was used to replicate the proposed hardware sorter, simulating its functionality and verifying its behavior in real-time sorting operations.

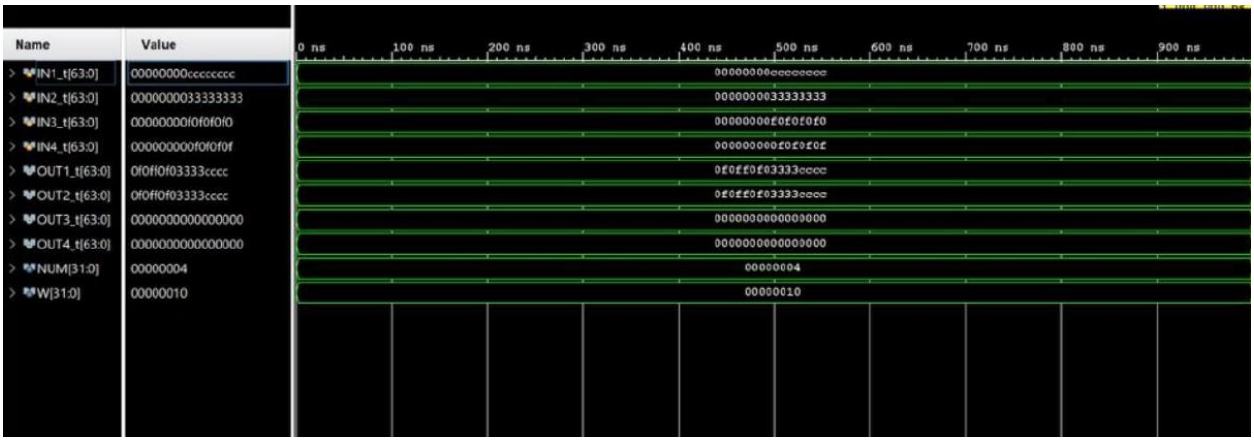
Simulation Outputs

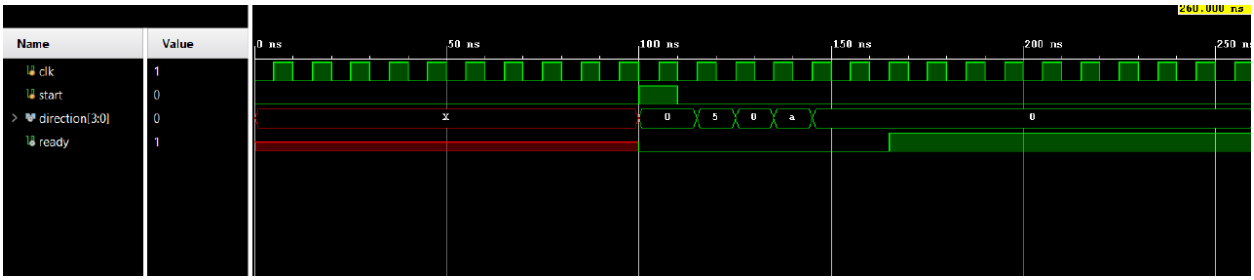
The simulation in Vivado provided valuable insights into the behavior of the RTHS design.

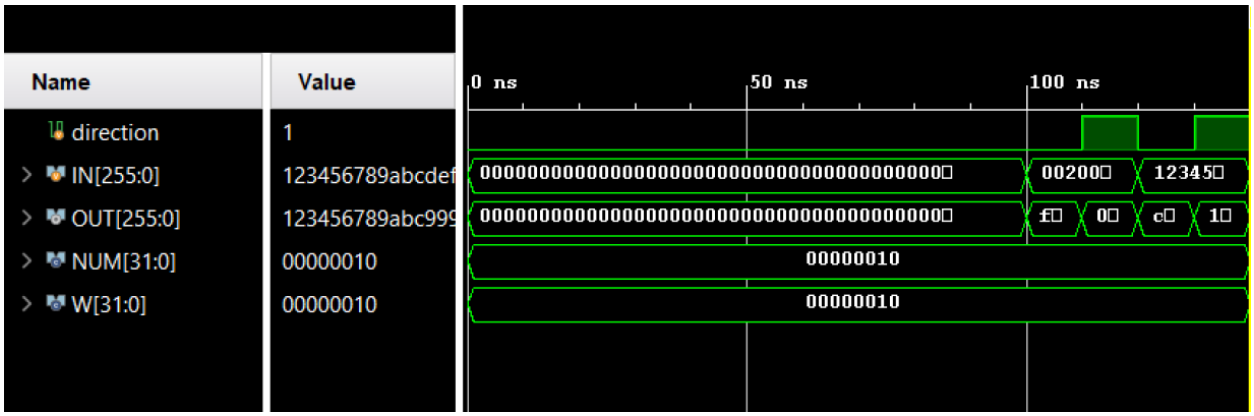
The detailed results of the simulation are presented in the images below.

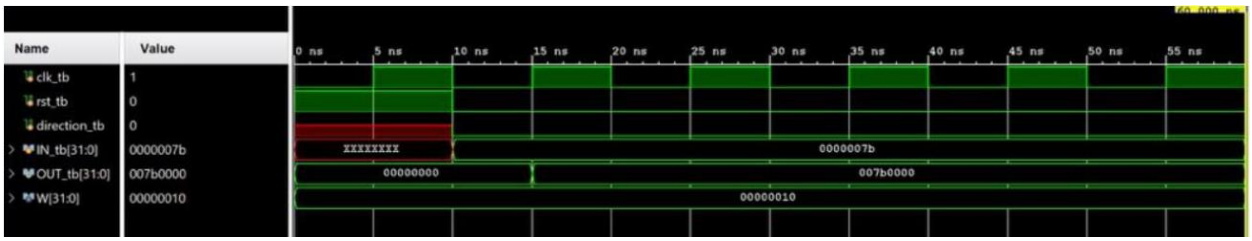


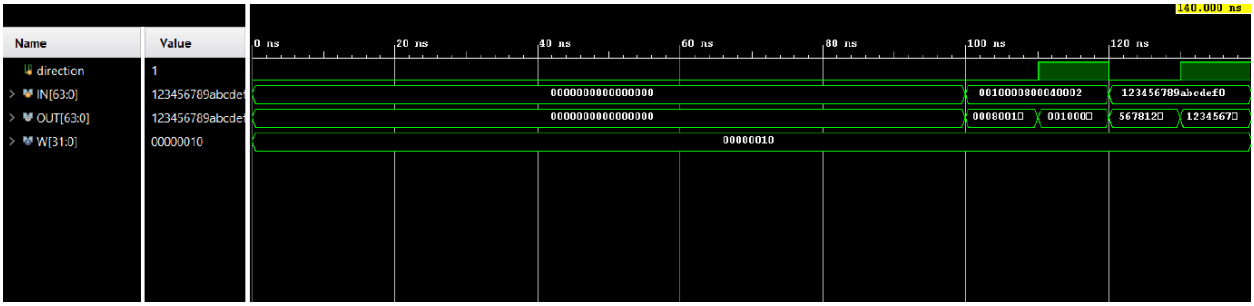


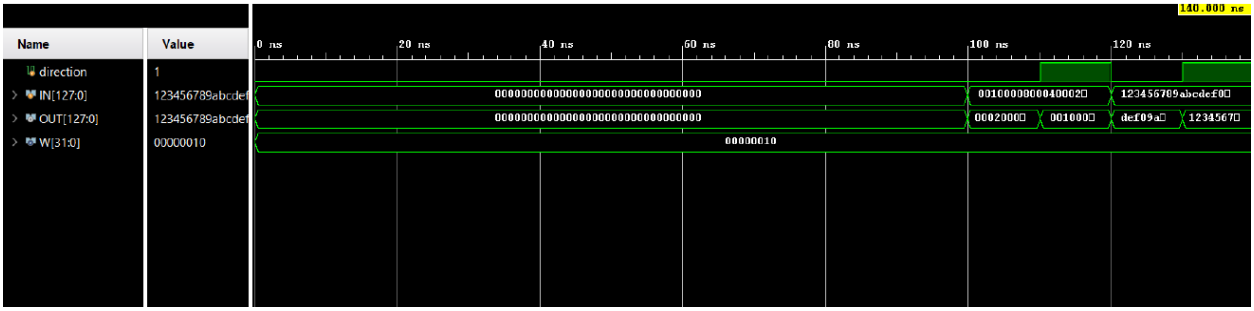

Name
Value
IN1_t[63:0]
IN2_t[63:0]
IN3_t[63:0]
IN4_t[63:0]
OUT1_t[63:0]
OUT2_t[63:0]
OUT3_t[63:0]
OUT4_t[63:0]
NUM[31:0]
W[31:0]
00000004
00000010


Name
Value
clk
start
direction[3:0]
ready
0 ns
50 ns
100 ns
150 ns
200 ns
250 ns


Name
Value
direction
1
IN[255:0]
123456789abcdef
OUT[255:0]
123456789abc999
NUM[31:0]
00000010
W[31:0]
00000010
0 ns
50 ns
100 ns


Name
Value
clk_tb
rst_tb
direction_tb
IN_tb[31:0]
OUT_tb[31:0]
W[31:0]
0000007b
007b0000
00000010


Name
Value
direction
1
IN[63:0]
123456789abcdef
OUT[63:0]
123456789abcdef
W[31:0]
00000010
140.000 ns
0 ns
20 ns
40 ns
60 ns
80 ns
100 ns
120 ns
0000000000000000
0010000000040002
123456789abcdef0
0008001□
001000□
567812□
1234567□
00000010


Name
Value
direction
1
IN[127:0]
123456789abcdef
OUT[127:0]
123456789abcdef
W[31:0]
00000010
140.000 ns
0 ns
20 ns
40 ns
60 ns
80 ns
100 ns
120 ns
00000000000000000000000000000000
0010000000040002□
123456789abcdef0□
00000000000000000000000000000000
0002000□
001000□
def09a□
1234567□
00000010