Section 1: Team

at_your_own_risc
Patrick De Leo 217220203
Arnav Gupta 219973452

Section 2: Verilator Waveform

Figure 1 - R-type instruction showing decoded values, immediate value, and control signals

Figure 2 - I-type (addi) instruction of max positive immediate value

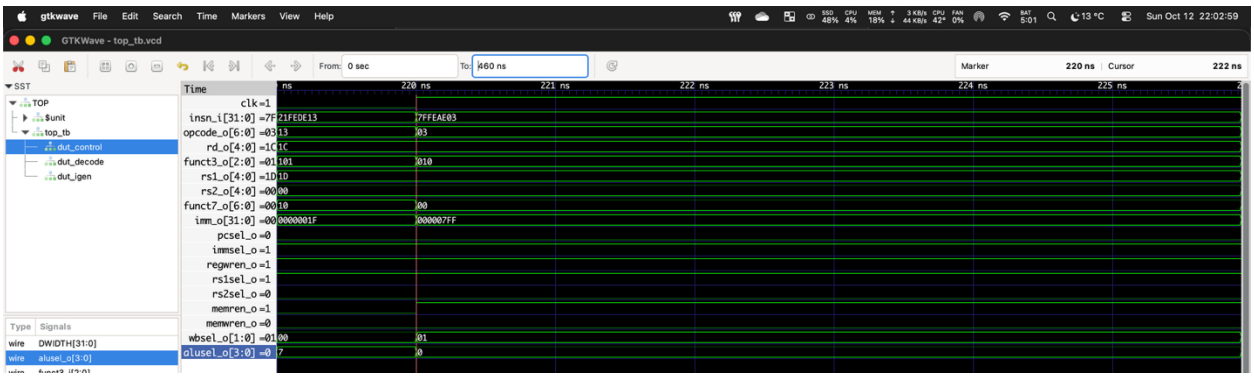

Figure 3 - I-type Load Instruction (lw) of max positive offset

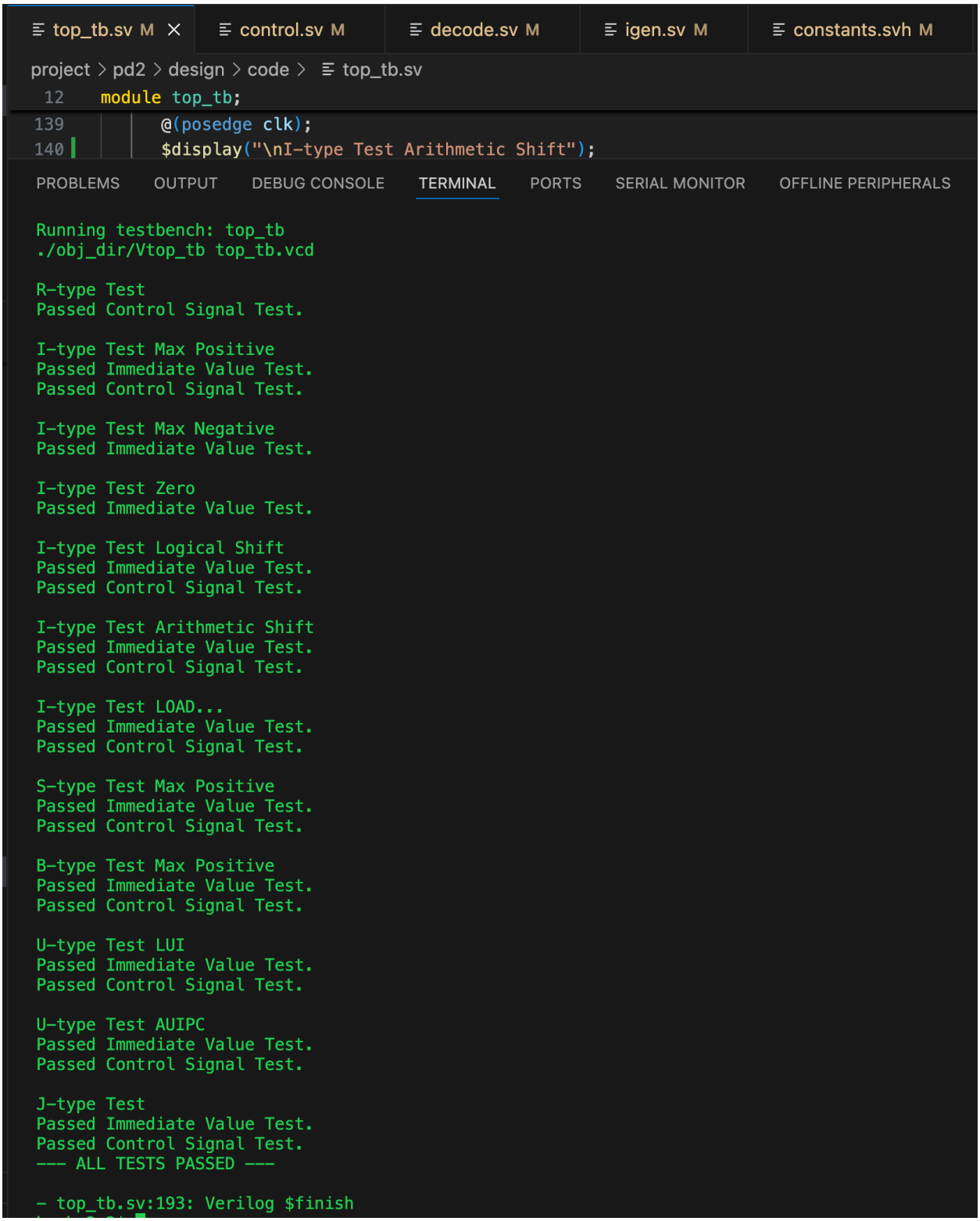

Figure 4 - Terminal showing all test benches have passed.

Section 3: Reflection

This project deliverable came with many challenges exclusively to test benches. The first was the struggle in determining the exact immediate value that should be produced for various instructions. Determining the precise immediate value for shift right arithmetic immediate and shift right logical immediate proved difficult. Additionally, the instructions that had non-contiguous immediate values, such as J-type and B-type instructions were also difficult and required writing out the instructions by hand and going through them bit by bit. For example, J-type and B-type instructions require a left-bit shift since the LSB is always zero to align with program counter increments. The second biggest challenge was figuring out how to create test benches that could run with Verilator. Full disclosure, generative AI was used to produce the C++ file, the makefile, and what lines of code to add to the test bench so that the signals would dump into a .vcd file. These are all noted in comments within the files.