

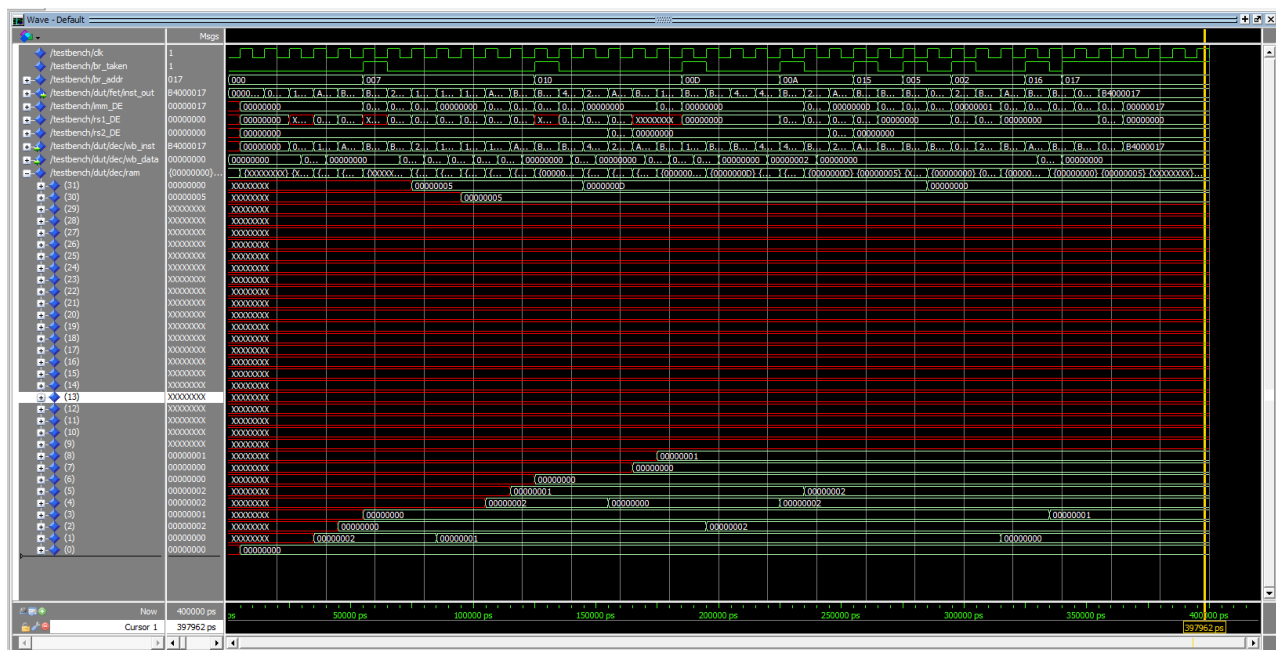
Project 04 - Decode

Procedure

The goal of this project was to take a whole command from the Fetch stage and decode it to determine what the command was, and read values out of the registers when necessary. This required that we implement a RAM in the stage, as well as capability to read and write to it. The opcode decoding part is just a large switch case, which could be condensed to about 5 options but we have kept separated into 19 cases for an easier time adjusting it later if necessary.

We created a top-level project called 'DLX' that will be used from now on as our top-level project, and then we will use GitHub as version control if we ever need to go back and retrieve code from previous labs. We then copied the Fetch stage over and implemented the Decode stage, connecting them in the DLX top file.

Once the two stages were in place, we implemented a testbench to prove out our code and demonstrate it's functionality. It was a tricky testbench because of having to time the jumps for the fetch stage as well as setting the correct values from register write back. See Figure 1 for an overview of the evaluation of 2!.



Results

We were successful in creating the Decode stage, connecting it to the Fetch stage, and creating a testbench that demonstrates the current functionality.

Conclusions

This project was a successful next step in creating the whole DLX processor, has set us up well for completing the rest of the processor, and increased our understanding of processor decode stages and well as testbenches.