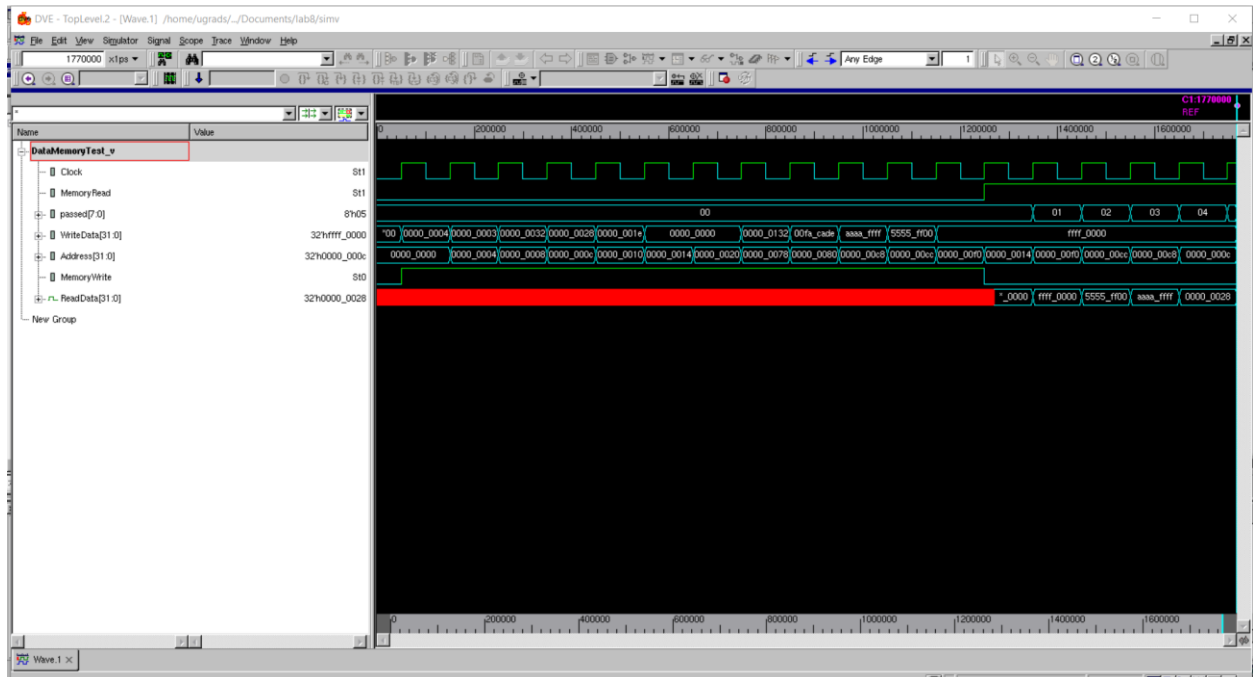
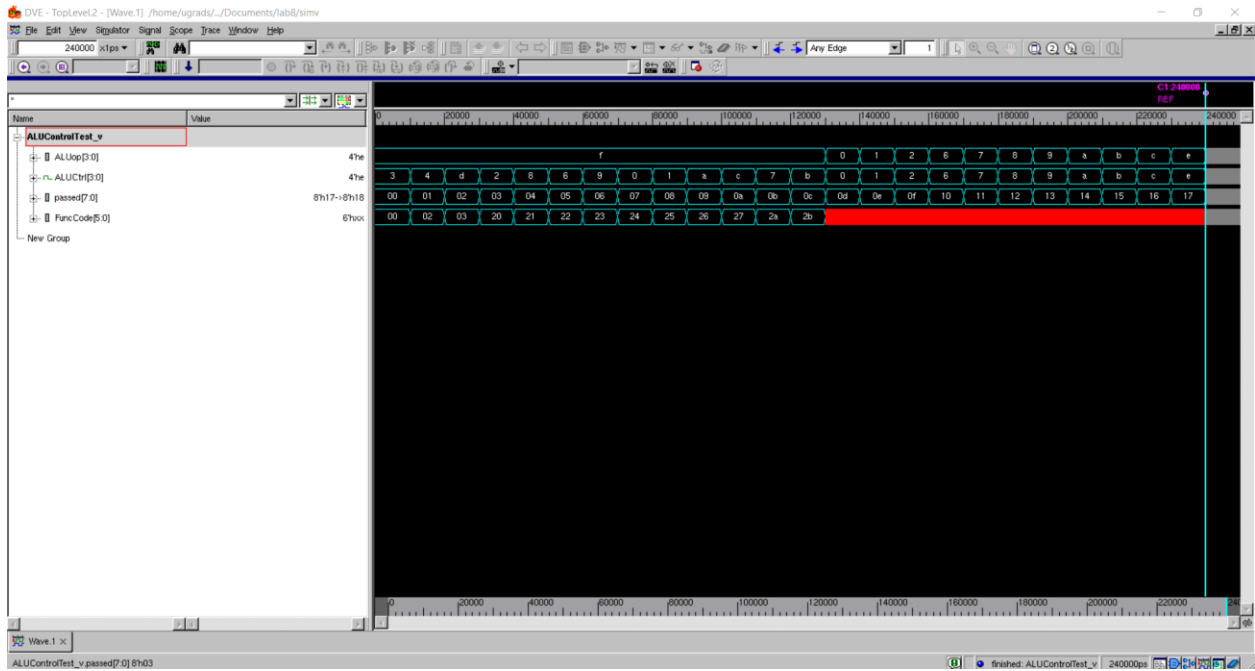


Data Memory



The data memory has 2 functions: a read and write. In this wave, the write gets tested first as you can see from MemoryWrite being 1 and MemoryRead being 0. This causes the red line on the ReadData. After it finishes, writing, it makes MemoryWrite0 and MemoryRead 1 and starts reading some data from the memory and putting it into ReadData. For example, you can see it writing aaaa_ffff into address 0000_00c8 and then later reading it from that address and it displaying the correct output.

AluControl



The ALU control takes 2 input (ALUop and Funccode) and produces an output, ALUctrl. It does this by looking first at the ALUop. If it's 1111, then it looks at the funccode and then the output is determined by the funccode. If it's not 1111(indicated by the "f"), then the ALUControl is the ALUop. In this waveform, you can see it first tests if ALUop is 1111, it varies the Funccode and then ALUctrl depends on the Funccode. After that it tests if ALUOp is 0-e, and then ALUctrl is just equal to ALUOp.

The screenshot shows the DVE (Digital Verification Environment) interface. The left pane displays a list of signals, with 'NextPCTest_v' selected. The right pane shows a detailed waveform view of the selected signal, displaying various data values and timing information. The waveform is divided into segments with labels like '0000_0010', 'ffff_ffff', and '0000_0001'. A vertical purple line marks a specific time point, labeled 'CT:47558' and 'TIER'. The bottom status bar indicates 'Finished: NextPCTest_v' and '70000ps'.

The NextPC can do 3 things: Jump, Branch, or just a normal advance. This waveform first tests pc advance by adding 4 to the current PC. Next it tests jump by setting jump to 1 and the output should be {first4ofPC, JumpField, 2'b00}. As you can see by the 000_0000 jump field becomes f000_0000 because the current pc is ffff_ff00. The next thing it tests is the branch. If Branch is high and Zero is high, then it branches by adding the SignImm32*4(to align it with words) with the currentPC+4. This happens when the current pc is 0000_0010 and it becomes 0000_0018 because the branch is $1*4+10+4=18$.