CS 3220 Processor Design

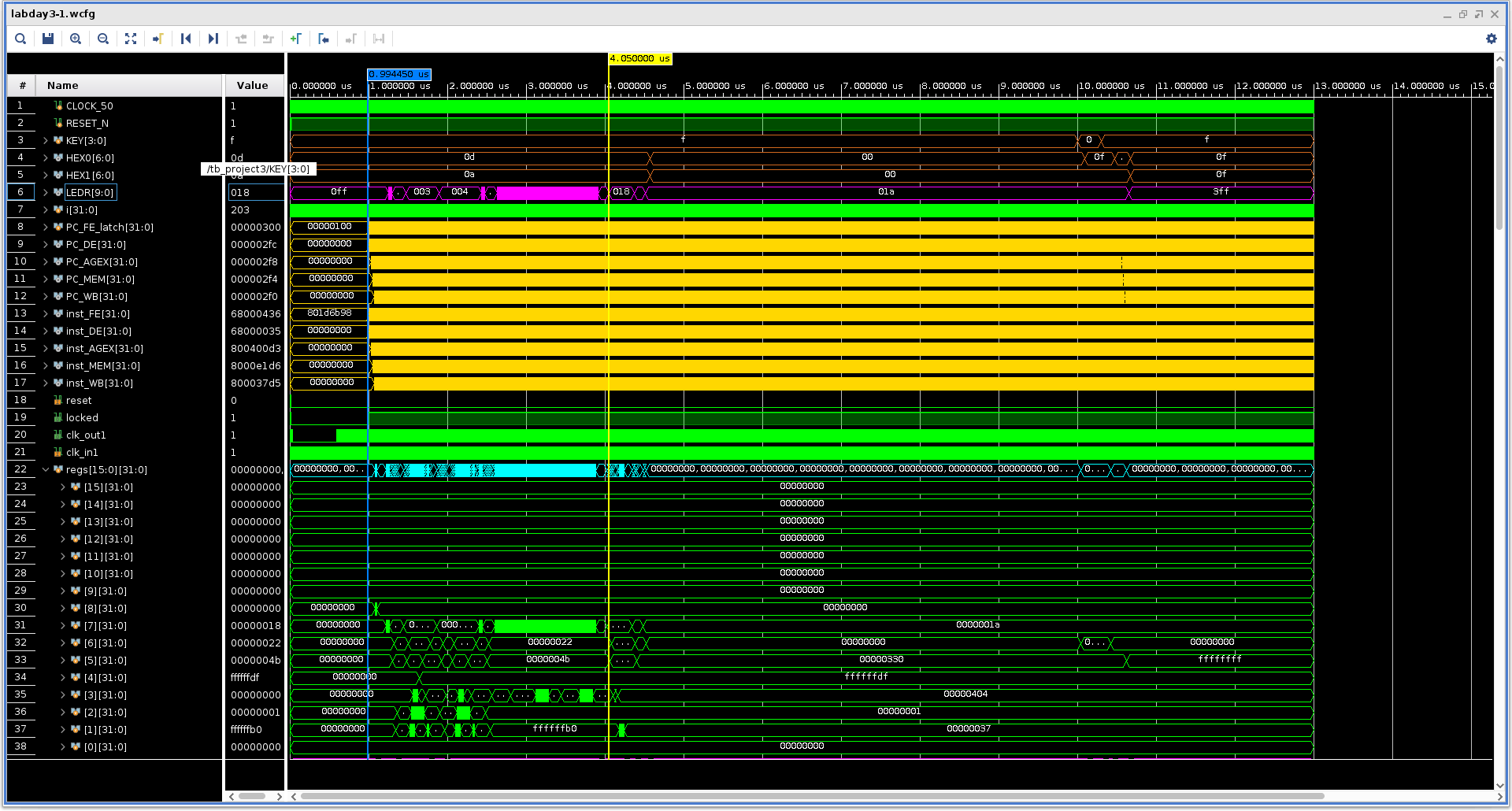
Assignment 4 Report

Yue Pan

903407994

04/14/2021

**Screenshot of passing testall.mem after adding BTB implementation + full-bypass.**

****

For this assignment, I designed AGEX + MEM stage forwarding and an 8-entry direct-mapped Branch Target Buffer. The BTB is indexed using PC[4:2] in fetch stage, and BTB update is done in WB stage where WB send a signal to FE to update BTB.

Execution time table (100MHz and measured up to LEDR turning from 0x17 to 0x18)

|  |  |
| --- | --- |
| Baremetal Assignment 3 | 6830 ns |
| Assignment 3 with AGEX + MEM forwarding | 4350 ns |
| Assignment 3 with forwarding + BTB | 4050 ns |

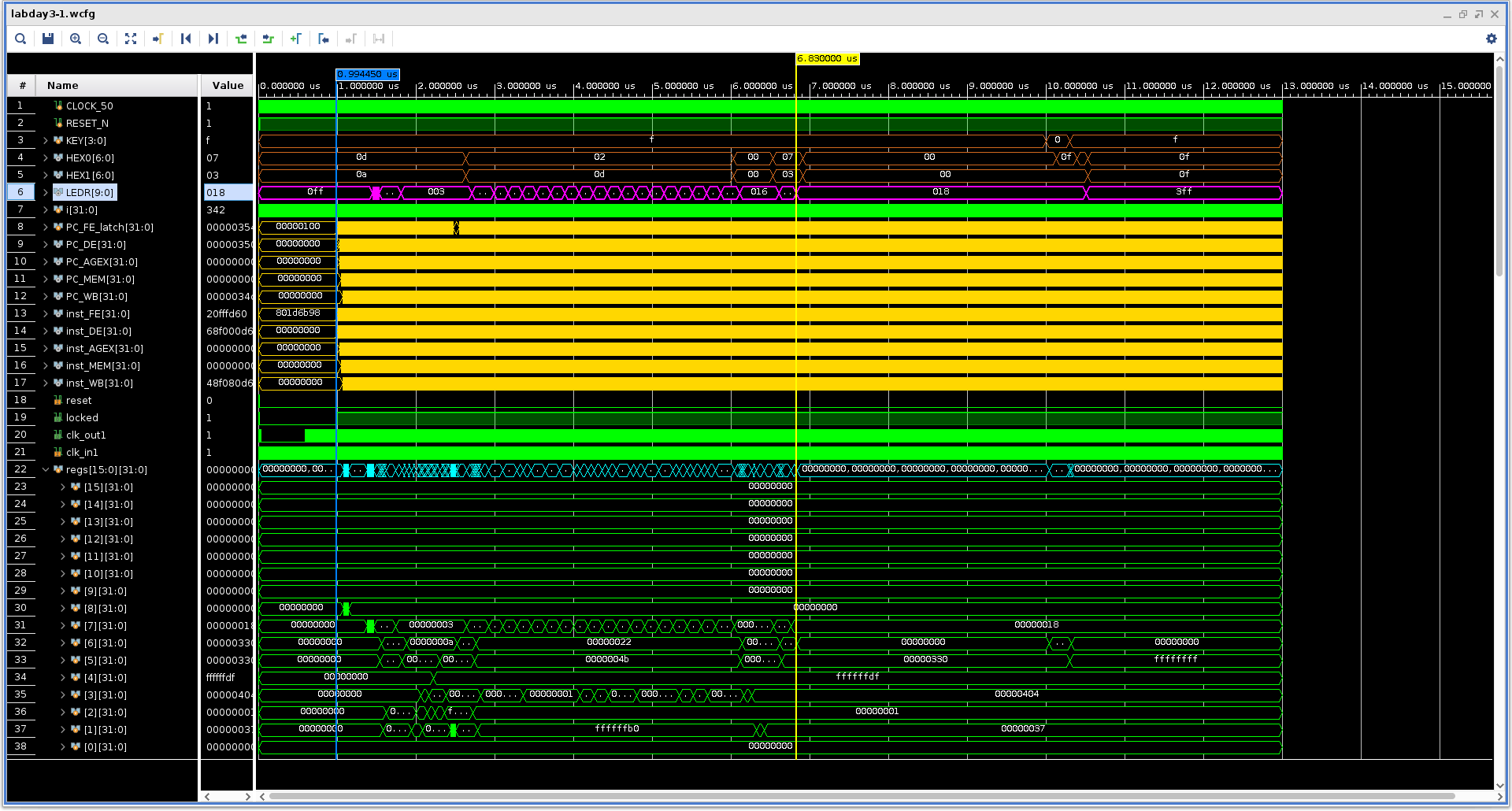
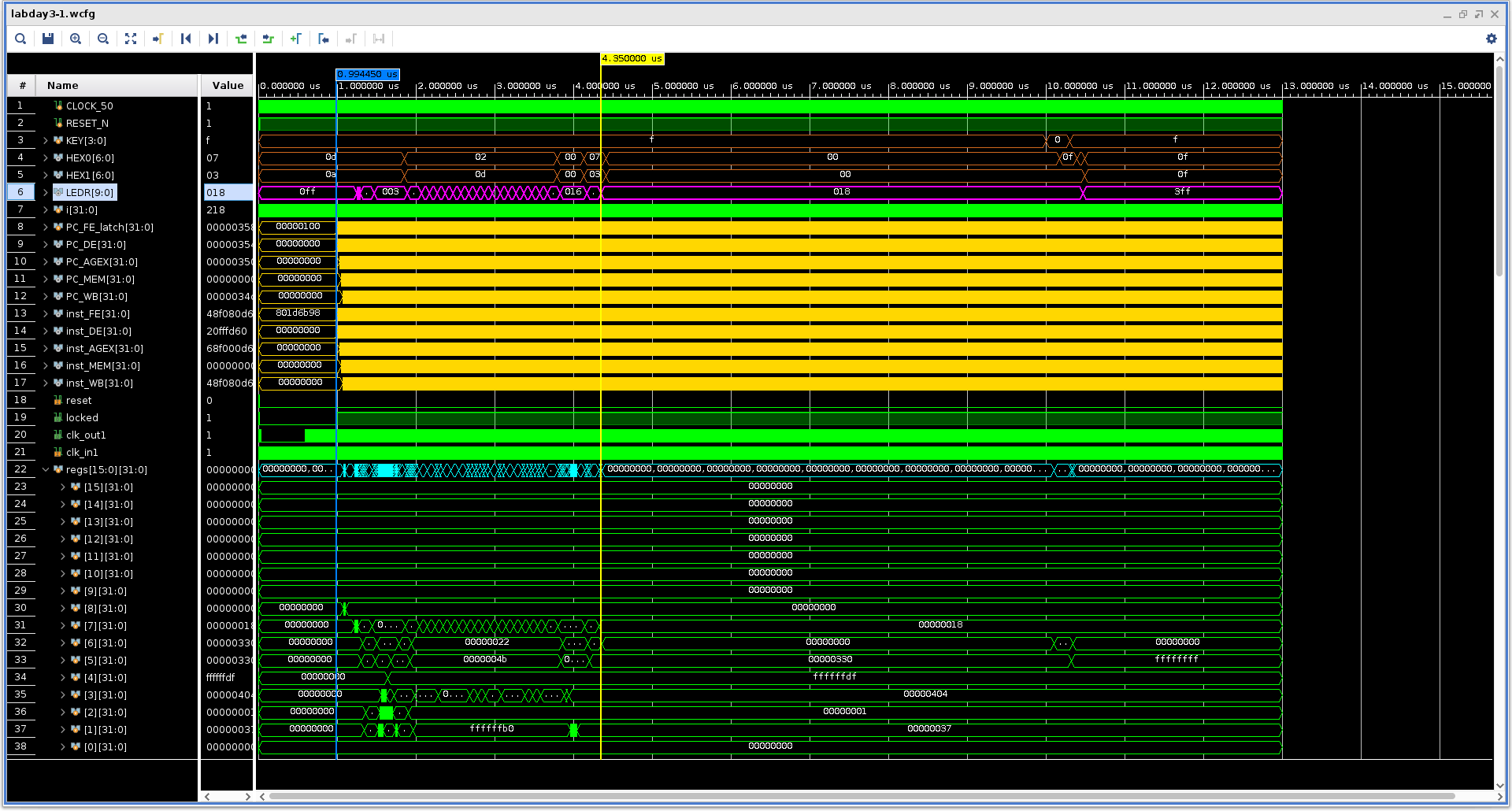


Figure above: timing of baremetal design

Figure down: timing of design with only forwarding



The full-forwarding provided the most significant time reduction. This is because the penalty for data dependency is 2 cycles and such dependencies appear very often. BTB did not provide as great time reduction to my design, because in testall.mem, there aren’t a lot of jumps for BTB to learn and predict.

To illustrate my BTB is working, the following screenshot shows that in a endless loop, BTB is able to record the correct branch behavior after the first misprediction (Note the Flush\_FE signal).

