Final LAB

Chue Zhang

Csc343 Fall 2021

Professor Gertner

Due Date : December 15th,2021 by 12:00pm

Contents

[Introduction: 3](#_Toc89873198)

[Objective: 4](#_Toc89873199)

[Specifications: 5](#_Toc89873200)

[Code: 6](#_Toc89873201)

[Simulations: 12](#_Toc89873202)

[Conclusion: 15](#_Toc89873203)

# Introduction:

This is the final take home test to be done with one computing device by Chue Zhang in the FALL semester of 2021 with professor Izidor Garter belt. My goal for this final Lab is to extend upon the ALU that I have previously created to run multiple instructions that are stored inside a data memory which we will then feed into the instruction memory to process.

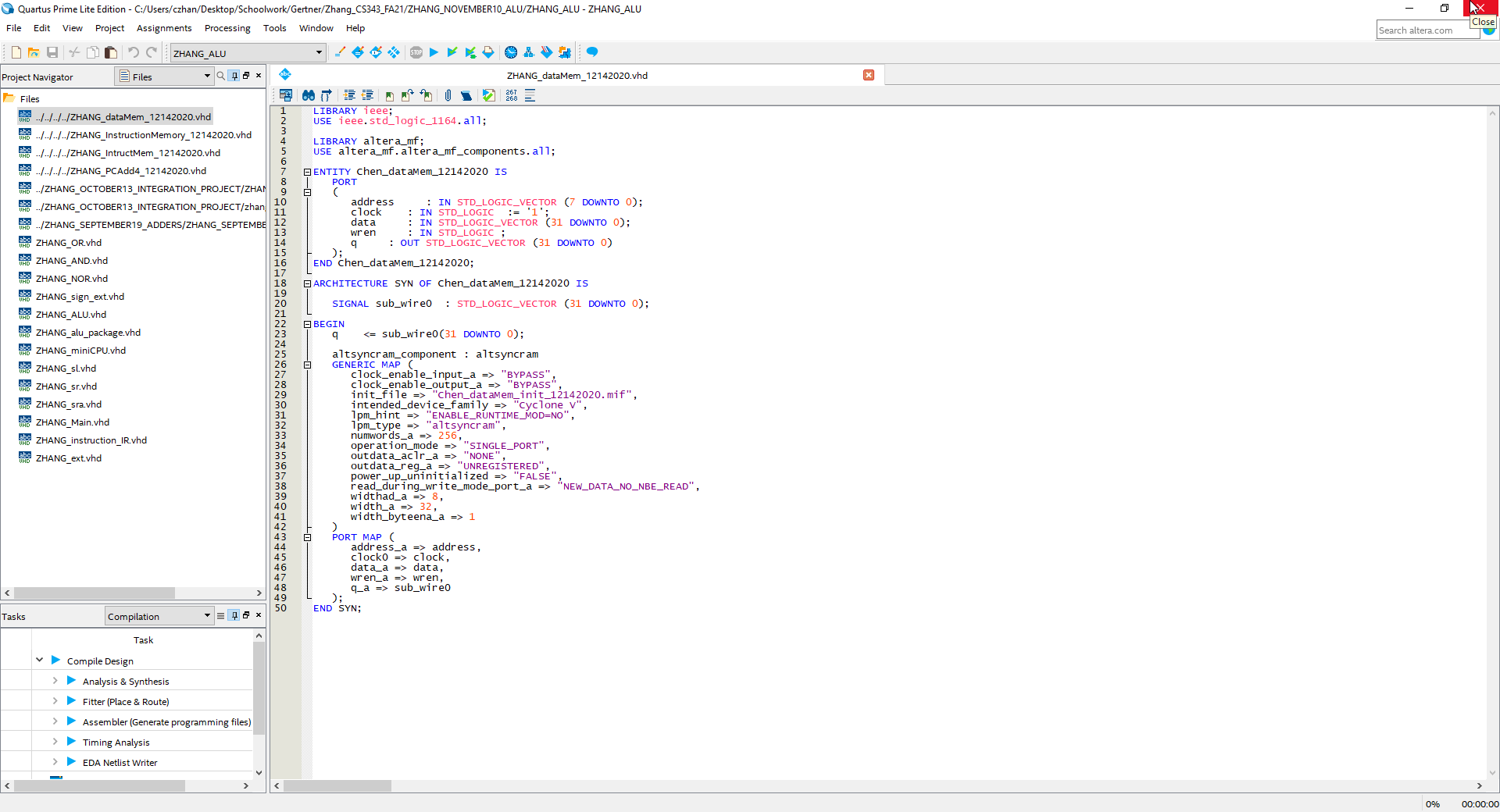
# Objective:

The objective of this final lab is to show that my program can perform multi-step instructions, for example a data memory that stores 3 different instructions and runs them all. Ooh, black and yellow! Let’s shake it up a little. We will not be using any bitwise operations and only using store word, load word and add operation that we have previously created. Barry! Breakfast is ready! Coming! Hang on a second. Hello.

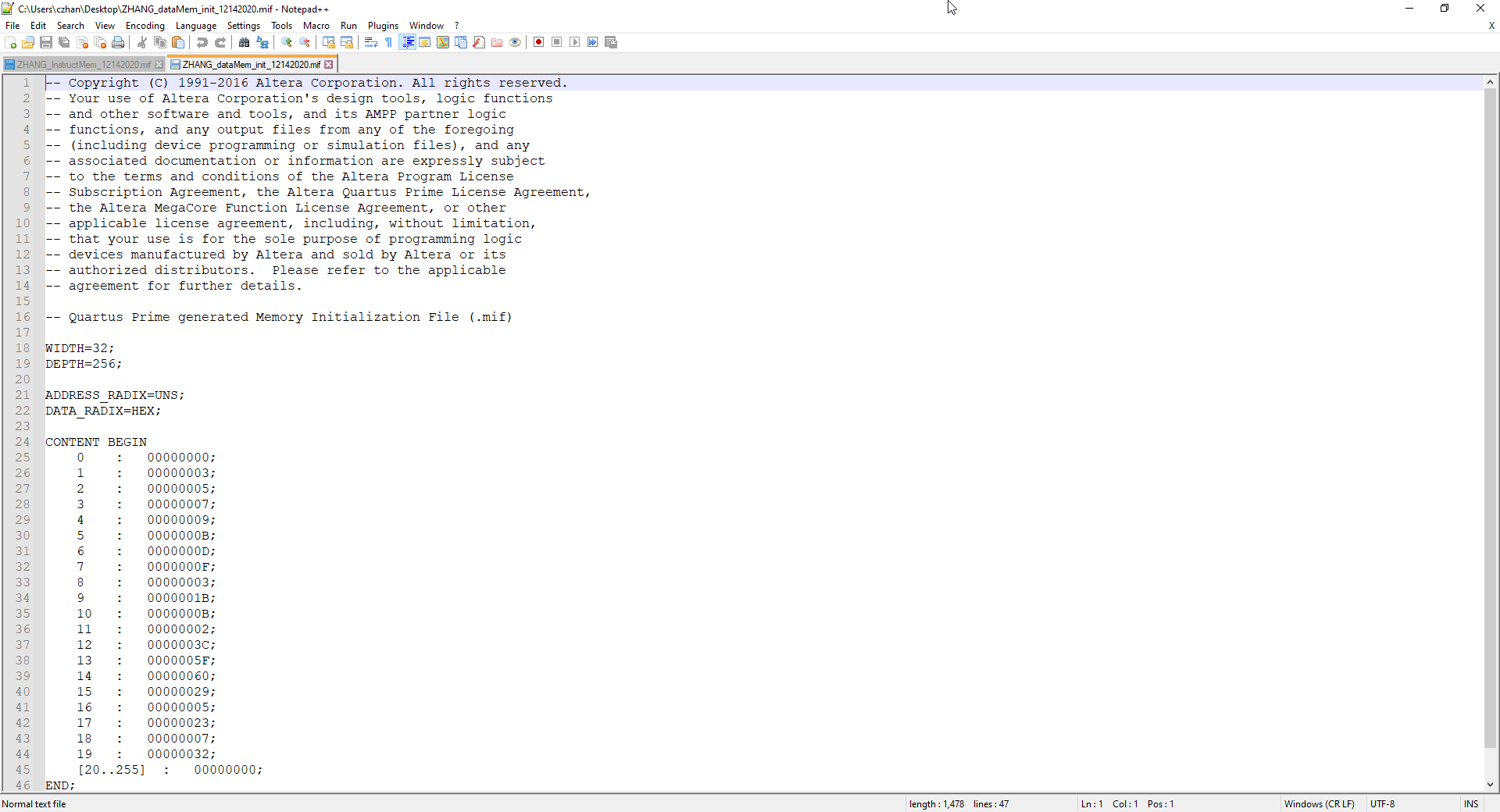
# Specifications:

What I will be doing is I will be loading an instruction from the data memory, incrementing the program counter by 4 so that we know what to read next in the data memory. With the current data instruction, I then feed it into the instruction memory and get a n instruction, then feed it into the instruction register to get my instructions. Later I will process this instruction and get my output. This process is repeated however many times are there are instructions as you will see below

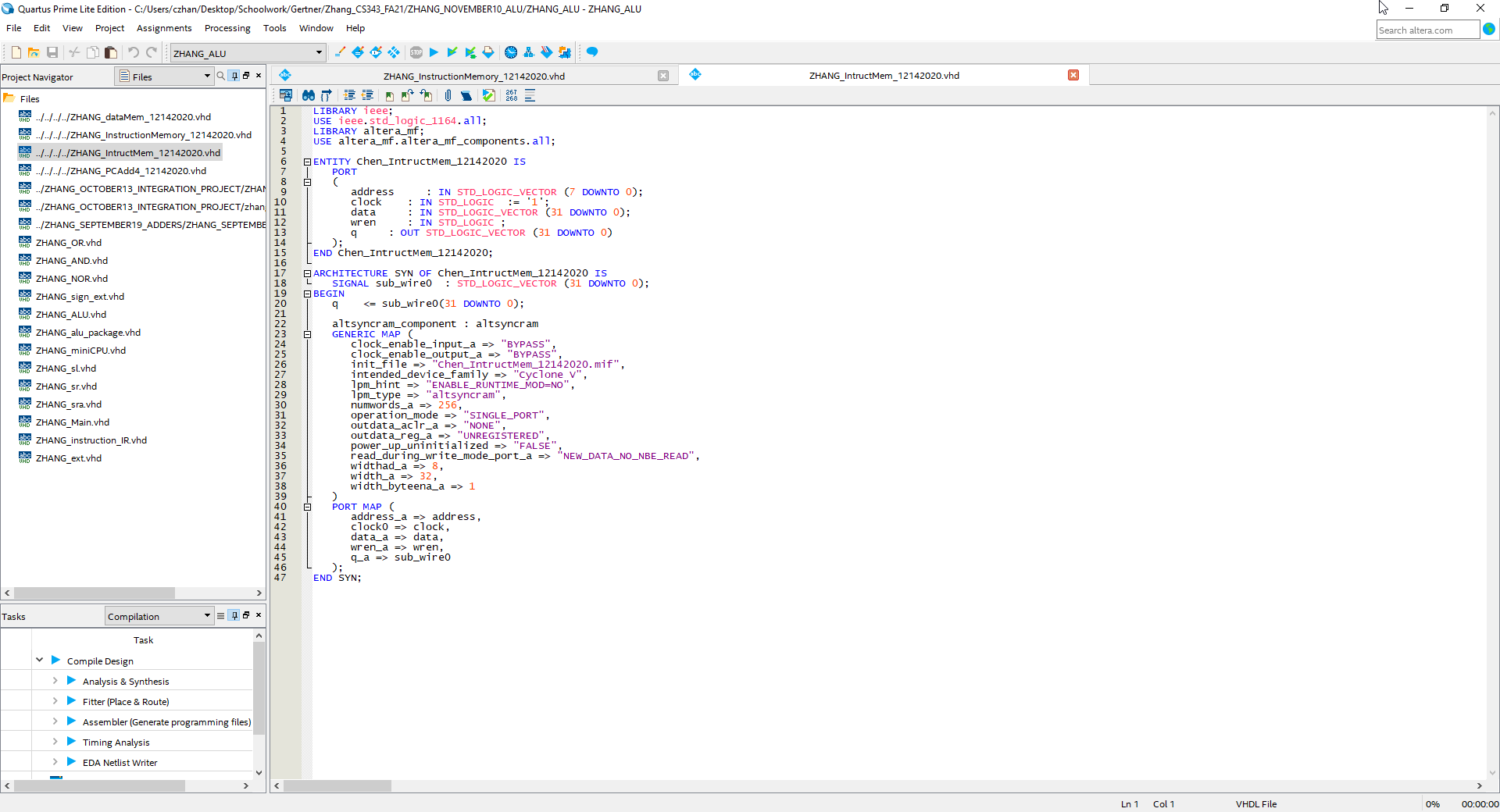
# Code:



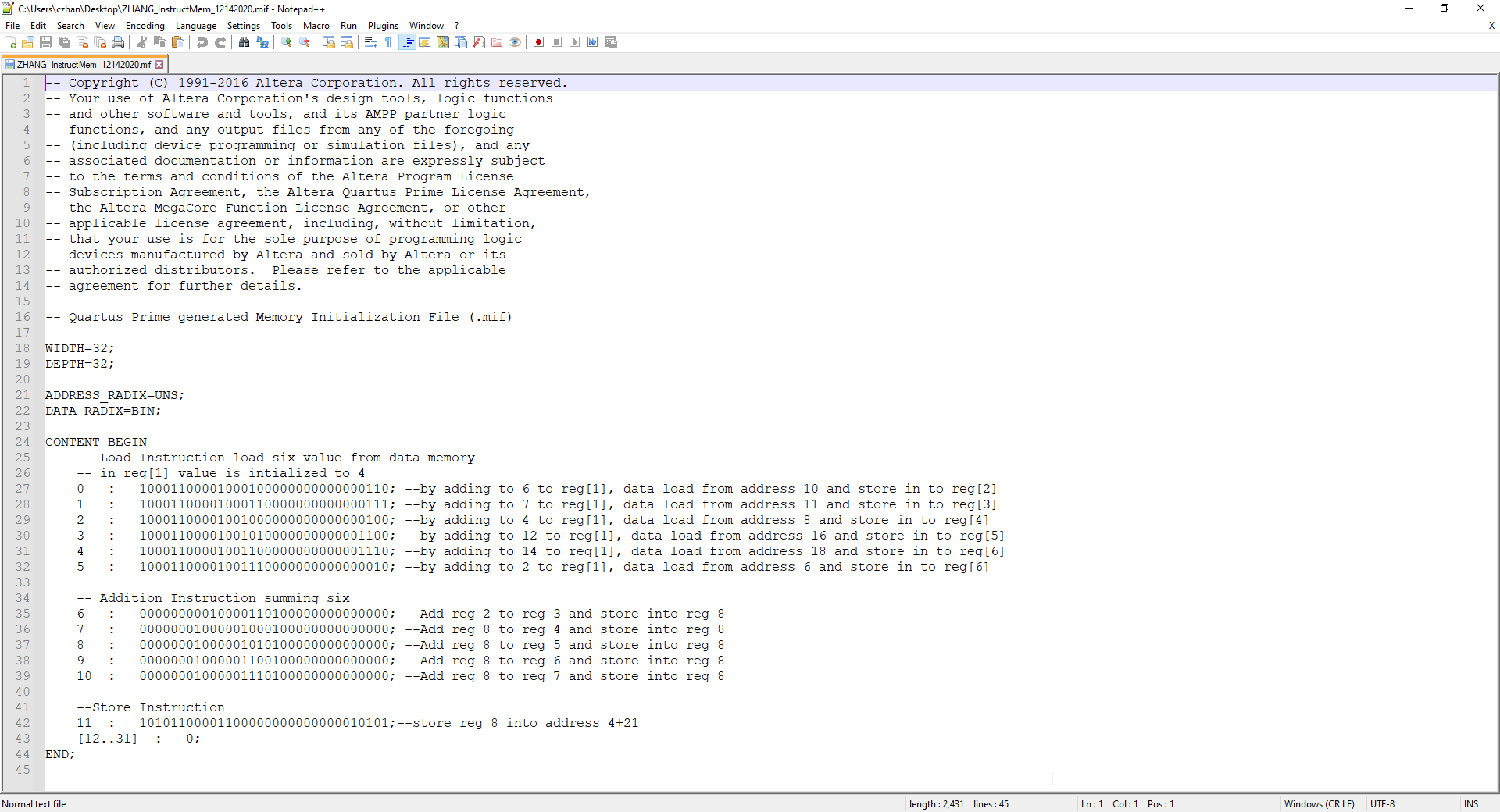
Data memory ram



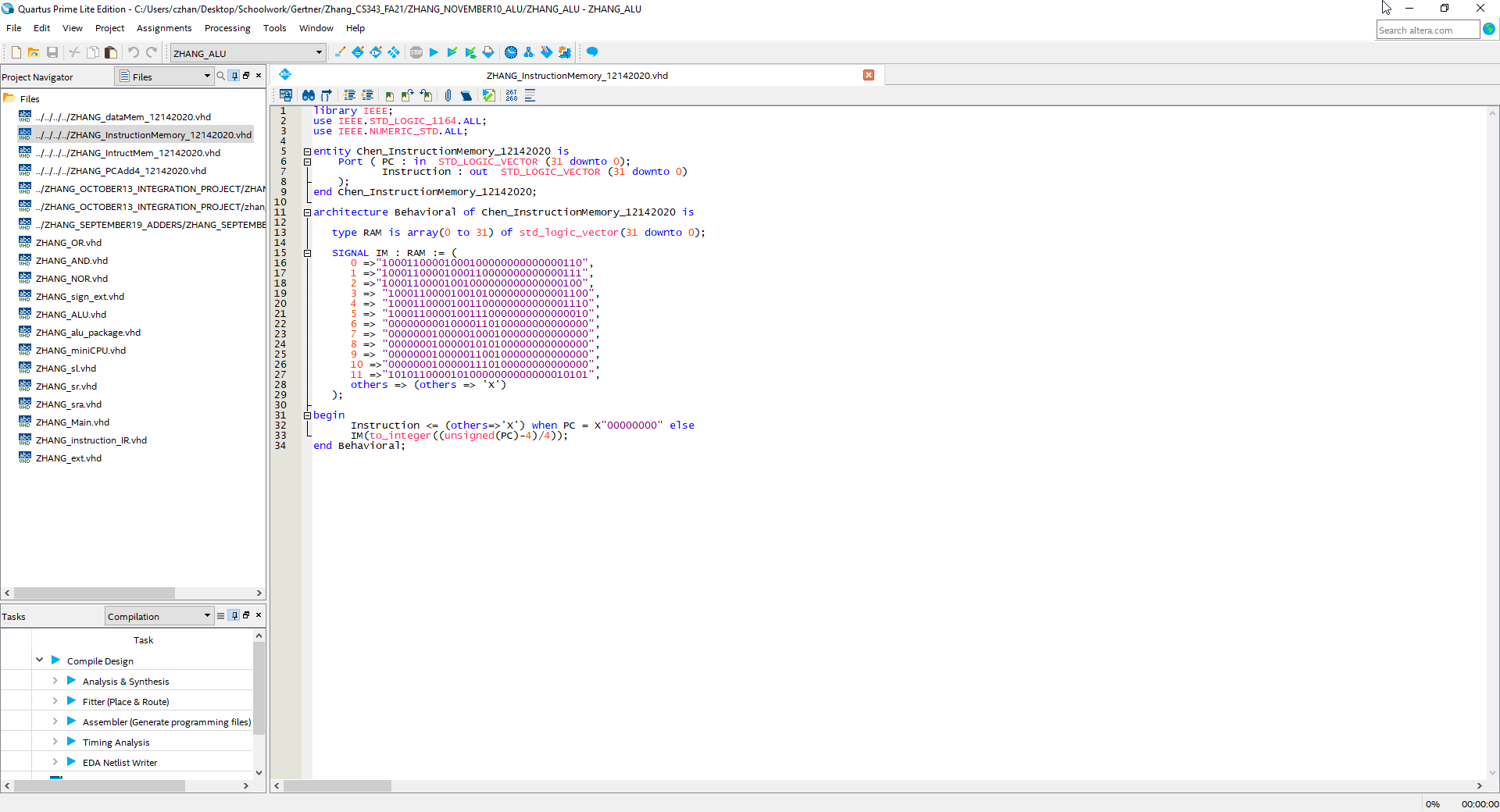
Data memory mif in increments of 4



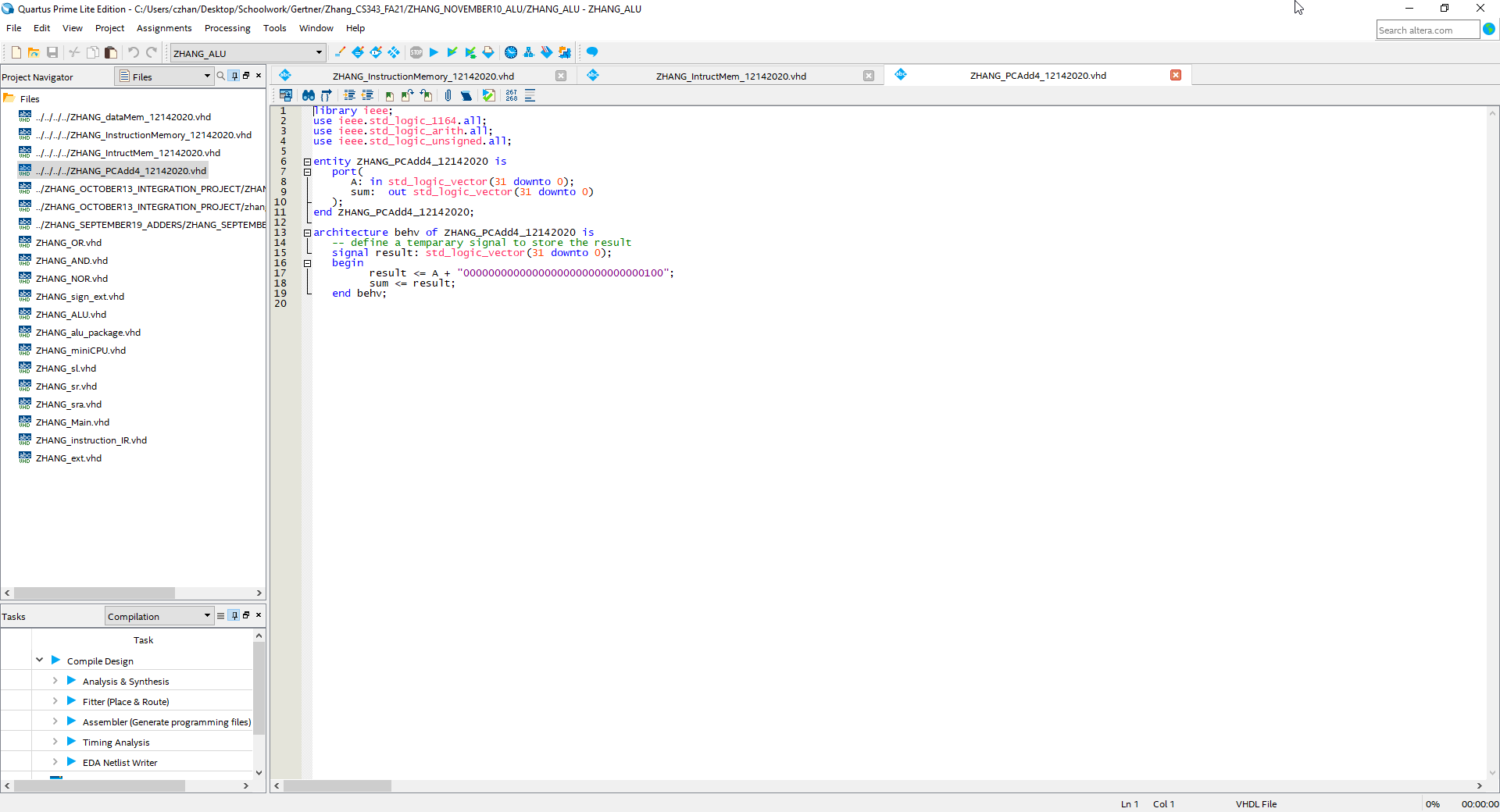
Instruction memory ram



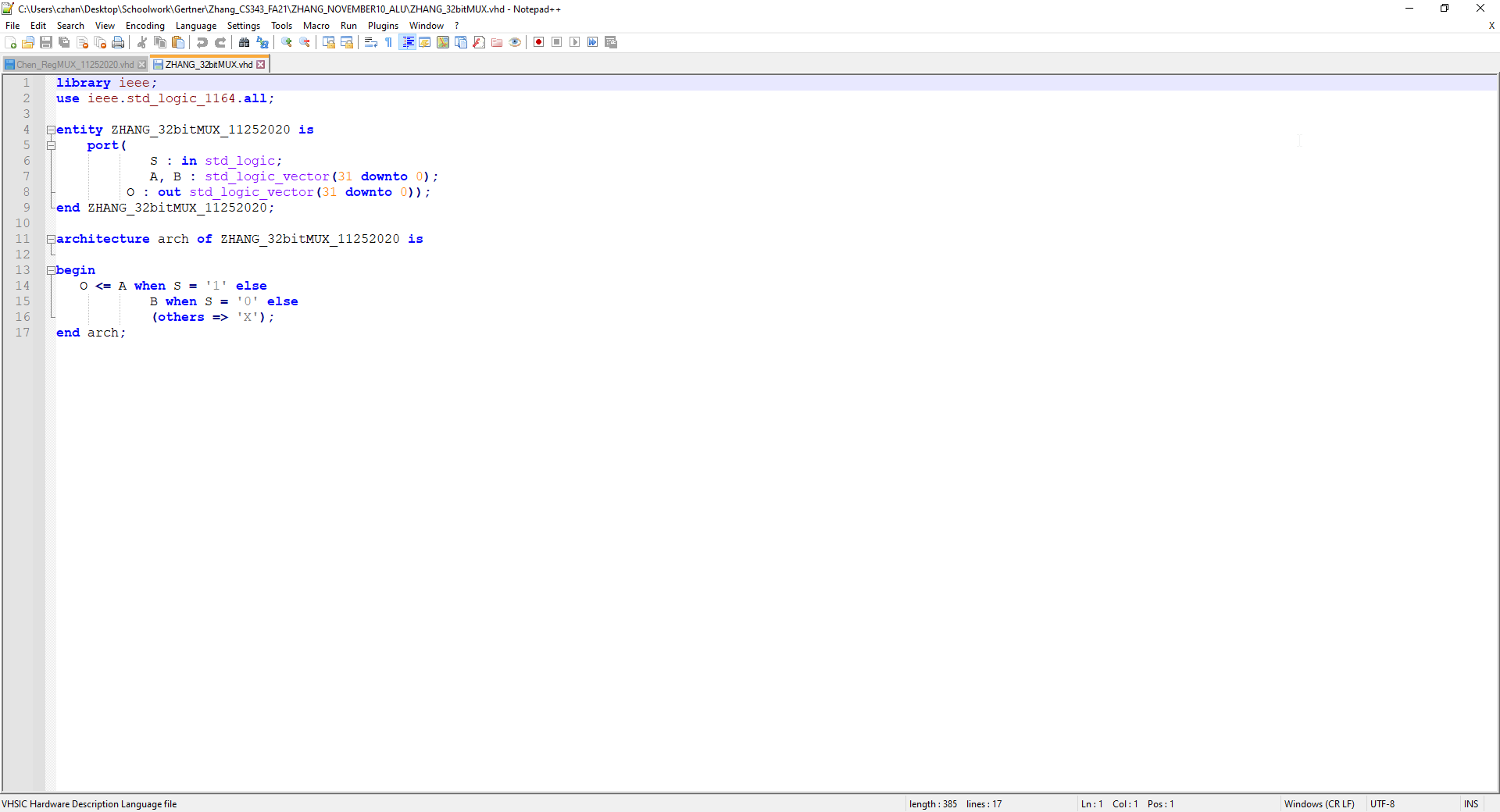
Instruction memory mif



RAM with basic instructions



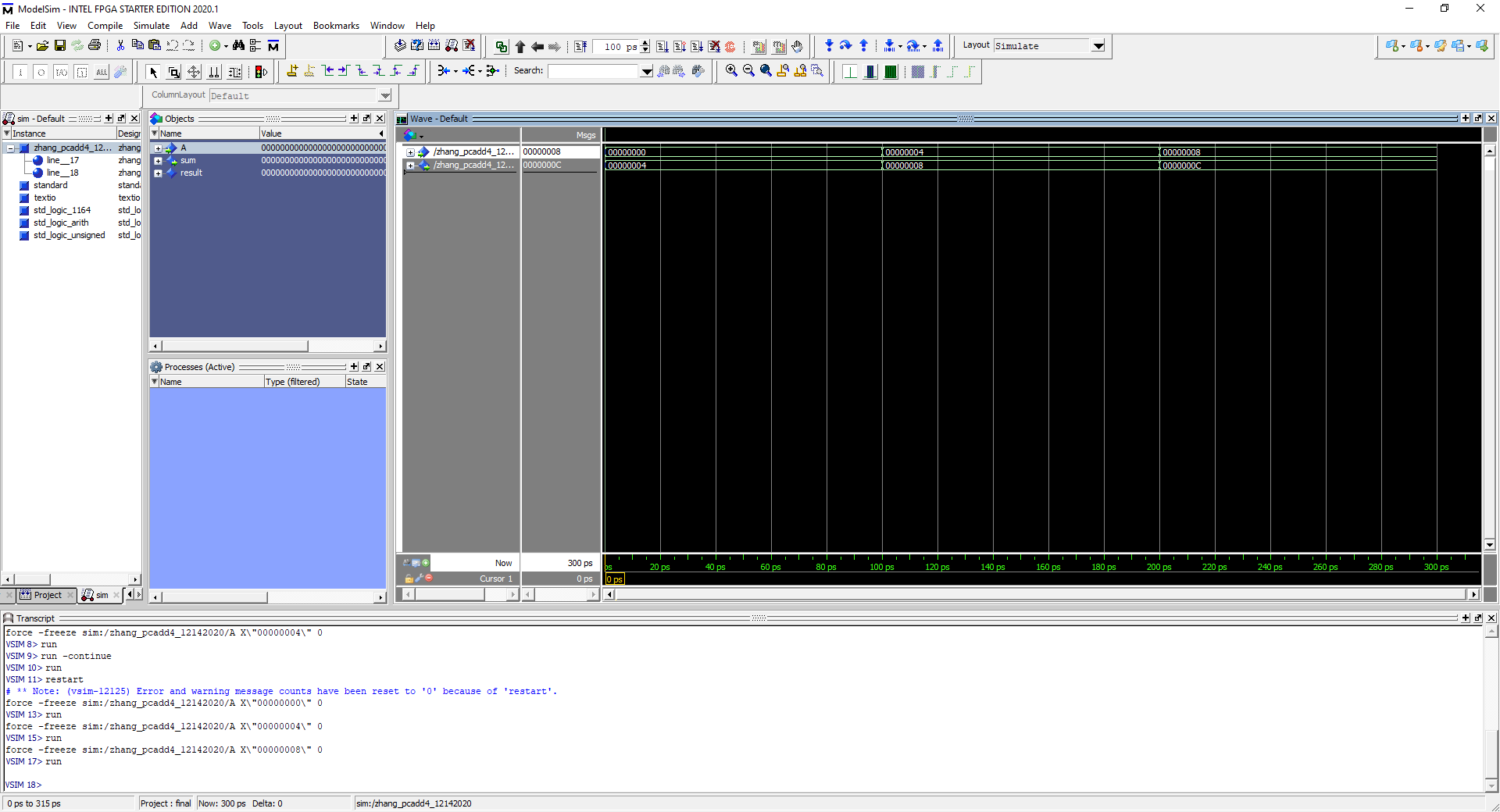
PC



# Simulations:

I will be performing a series of add operations of 1 + 1 of instructions that were stored in the data memory and instruction memory ram vdhl files. Every time an instruction is performed, the program counter is incremented by 4 as you will be able to see below in the simulations section

## PC:



## Instructions(1st – 5th):

Graphical user interface

Description automatically generated

This is the first instruction of 1 + 1, if you look at the result, 2 is outputted then stored into the 2nd register slot

Graphical user interface

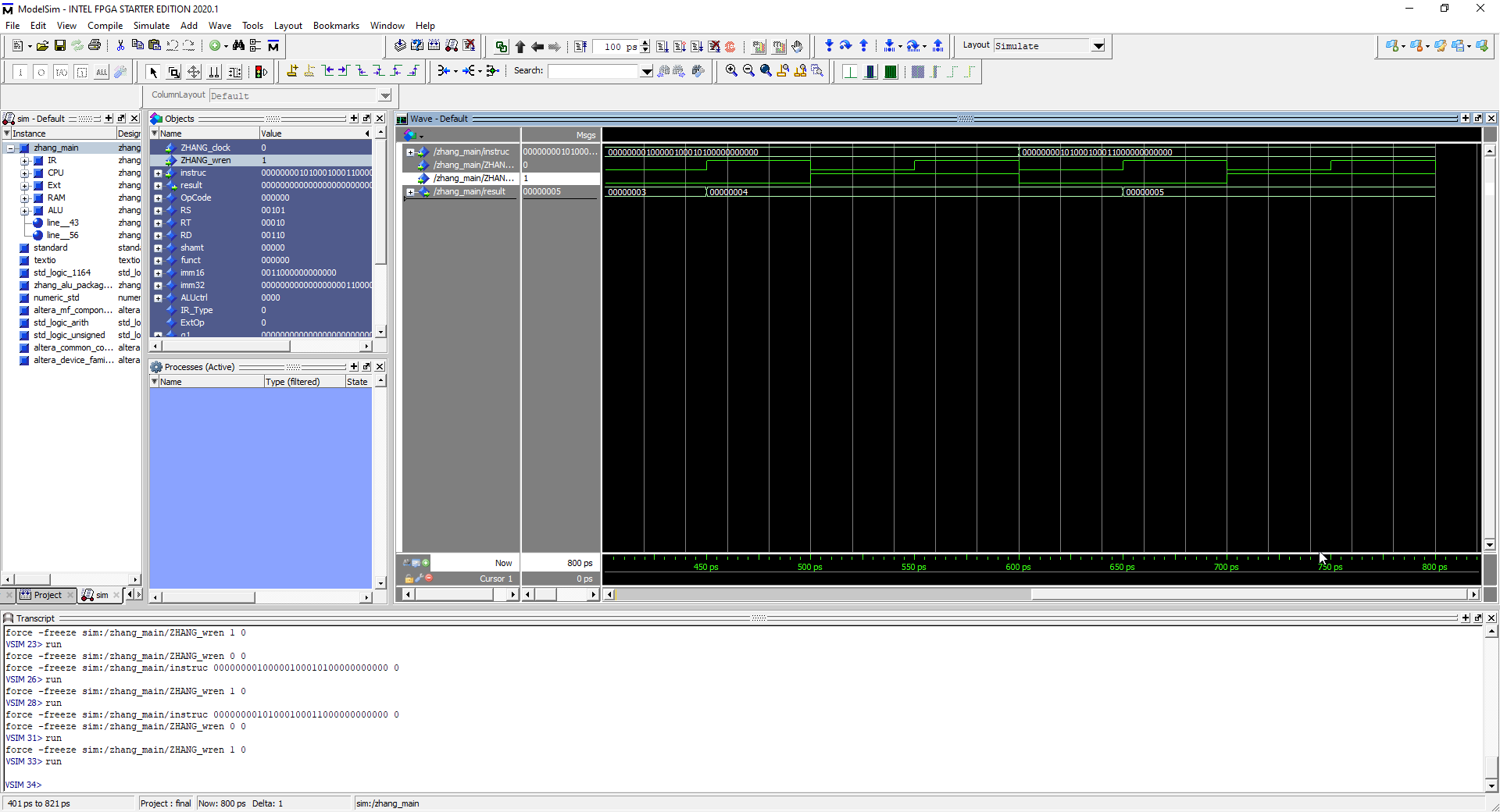
Description automatically generated

This is the second instruction which adds the 2nd register slot with 1 then stores the result into the 3rd register slot

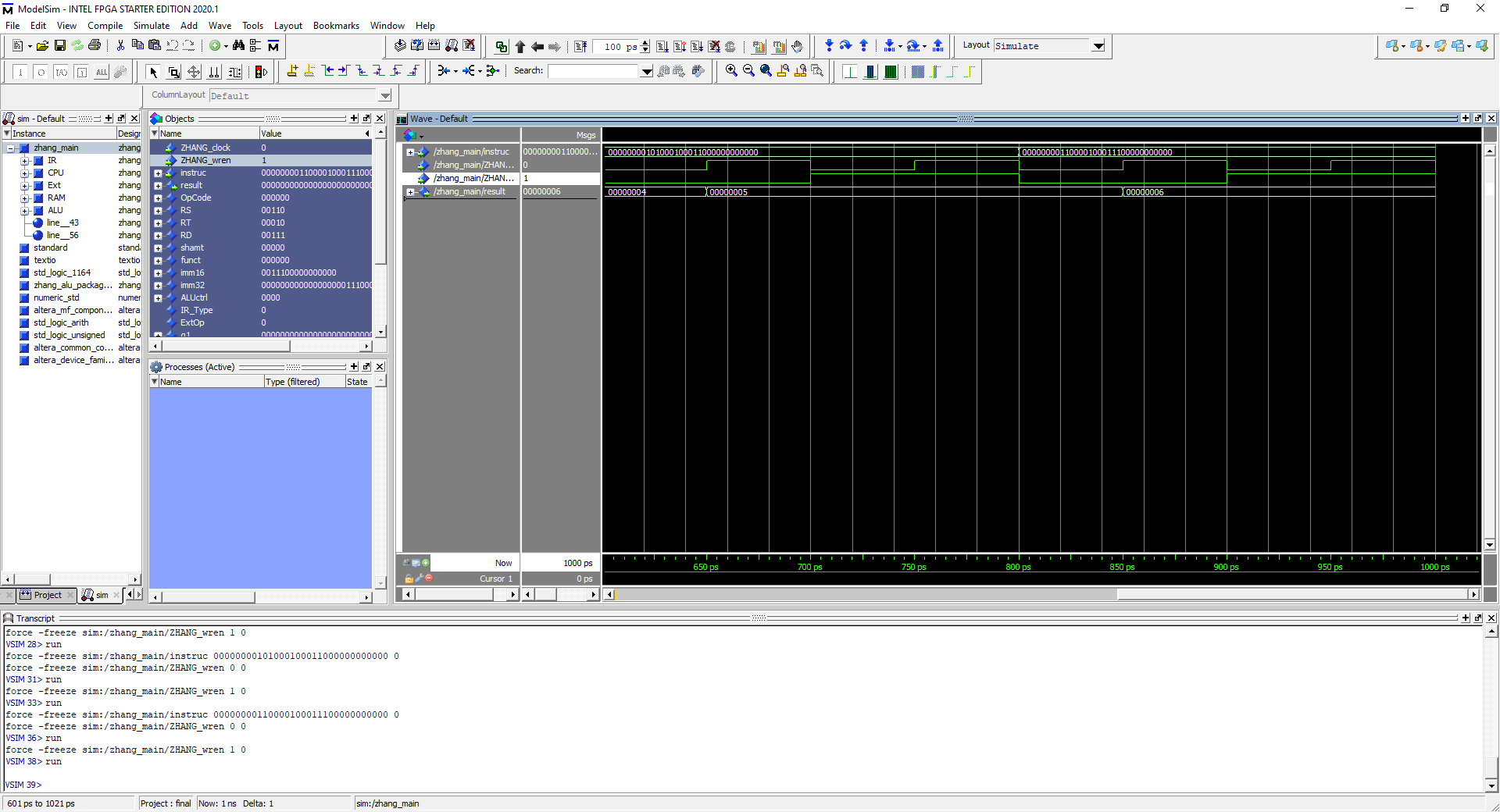
A picture containing text, screenshot, monitor

Description automatically generated

This is the 3rd instruction and what happens is 3rd register + 1 is performed then written into the 4th register slot



This is the 4th instruction where we perform 4th register slot + 1 and then we store it in the 5th register slot

Lastly, for the last instruction, we perform 5th register slot + 1 and then store it in the 6th register slot

# Conclusion:

In conclusion, I have learned about using program counters, instruction memory, data memory and how to store and load those data. This is a very interesting process, and I am grateful for have learnt more about this.