Tennessee Technological University

ECE 4120 – Fundamentals of Computer Design

Department of Electrical and Computer Engineering

Written By: Josh Adair, Cameron Dempsey, Brantlee Garland, Hunter Goodson, and Randy Wilhoite

Phase One: Instruction Fetch Unit

**Table of Contents**

Modified Block Diagram ………………………………………………………………………………………………………………………… x

Phase One Objectives ……………………………………………………………………………………………………………………………. x

VHDL Implementation Elaboration ……………………………………………………………………………………………………….. x

Synthesis Results ………………………………………………………………………………………………………………………………….. x

Test Bench Elaboration …………………………………………………………………………………………………………………………. x

Waveform Elaboration …………………………………………………………………………………………………………………………. x

Time Quest Timing Analyzation …………………………………………………………………………………………………………….. x

Phase One Conclusion …………………………………………………………………………………………………………………………… x

**Modified Block Diagram:**

Figure 1 below shows the instruction fetch unit block diagram with bit values and any additional connections. You can see from the diagram that PC has a 32 bit input and output, the adder has a 32 bit and constant 4 input with a 32 bit output, and the 8 LSB of PC are input into the instruction memory, with a 32 bit output.

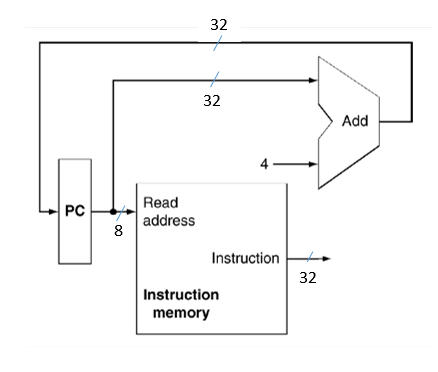
****

Figure 1. Modified Block Diagram

**Phase One Objectives:**

Phase one of this project had many objectives, and the big picture once all of these objectives are completed is to fully implement the instruction fetch unit of the MIPS processor. The first objective was to create a program counter (PC) in VHDL. This needed to be a synchronous block with a 32 bit input and a 32 bit output. The second objective was to create an adder in VHDL as a combinational block. The adder is 32 bits with two inputs, a constant 4 and the 32 bit output from the PC. The third objective and final block that needed to be created was the synchronous for instruction memory. The input to this block is the 8 least significant bits being output from the PC. The output should be 32 bits. The fourth objective was to ensure that the MIPS instructions were stored through the test bench in the first five locations of instruction memory. The final objective was to simulate everything we created using Quartus and the TimeQuest timing analyzer in order to obtain relevant data, such as Fmax and timing reports.

**VHDL Implementation Elaboration:**

**Synthesis Results:**

**Test Bench Elaboration:**

**Waveform Elaboration:**

**Time Quest Timing Analyzation:**

**Phase One Conclusion:**