phoeniX RISC-V Core V0.1

Arvin Delavari – Faraz Ghoreishy

arvin7807@gmail.com – farazghoreishy@gmail.com

Summer 2023

Iran University of Science and Technology

Electronics Research Center – Digital Design Laboratory

This page was initially left blank.

Preface:

phoeniX is a 5 stage pipelined 32-bit RISC-V processor written in Verilog HDL. This project was started in summer of 2023 in Digital Design Laboratory of Electronics Research Center at Iran University of Science and Technology. This document contains technical specifications and user manual for version 0.1 of phoeniX processor. This version of the processor is able to execute RV32I (integer operations extension) instructions of official RISC-V instruction set manual [1]. Other extensions will be supported by the core in the future updates.

phoeniX core is capable of execution of assembly and C codes using iverilog simulator. Guidance is included in this document for the flow of code execution in this processor. For execution of C codes, the processor will use RISC-V GCC compiler toolchain and for assembly codes, you can use GCC compiler or Venus simulator which is a free extension for the recognized code editor, Microsoft Visual Studio Code. Further descriptions are included to relative sections of this document.

This project is an open source CPU under the GNU V3.0 license [2] and is free to use. You can find source codes and documentations in the following GitHub repository:

<https://github.com/ArvinDelavari/PHOENIX-CORE>

We would like to express our sincere gratitude to the following individuals for their invaluable guidance and remarkable insights, which played a pivotal role in the completion of this project:

* Professor Sattar Mirzakuchaki PhD

Iran University of Science and Technology

* Professor Hadi Shahriar Shahhoseini PhD

Iran University of Science and Technology

* Nima Amirafshar MSc

Iran University of Science and Technology

Arvin Delavari – Faraz Ghoreishy

Iran University of Science and Technology – Summer 2023

Contents:

References:

[1]

[2]

[3]

[4]