

EE480 Assignment 4: Tangled with QAT

Implementor's Notes

Andrew M. Donovan
Department of Electrical and
Computer Engineering
University of Kentucky
Lexington, KY USA
amdo257@uky.edu

Matt Castle
Department of Electrical and
Computer Engineering
University of Kentucky
Lexington, KY USA
mttcastle@uky.edu

Nicholas Santini
Department of Electrical and
Computer Engineering
University of Kentucky
Lexington, KY USA
npsa227@uky.edu

ABSTRACT

This project is a continuation of the previous pipelined Tangled processor and now is working on implementing the Q.A.T. architecture.

APPROACH

For this project, the goal is to use the standard floaty library that was given. Along with using the standard tangled AIK implementation, included as Tangled.aik. Beyond this the included code is Tangled.v the start of the pipelined processor. This project is based off the work of Team 36 and their processor.

Beyond this the processor design was to be built into four stages of processing. With heavy use of always blocks. Inside of the the blocks the QAT ALU's are called and implemented as their own module so that they could be referenced as need.

The test code was broken into short statements testing each instruction. On failure the code would generate a system call. That way trouble shooting would be simpler.

STATUS.

The processor is not implemented in a meaningful way. There is a bit of structure but most of the Tangled.v file is of the Floaty which is Dr. Dietz's. We are currently working on implementing the new design flow and the new QAT ALU's. Most of the structure is there for the primary alu.