Technical Specifications
for
RISC-V Architecture for Motion Planning Algorithms in Autonomous Drone Applications

Anthony Kenny

October 4th, 2019
Version: 1.0

Abstract

This thesis aims to design RISC-V computer architecture that supports fast execution of motion planning algorithms for drone applications. First, the computation of sampling-based motion planning algorithms commonly used in autonomous drones (such as RRT, RRT*, PRM) will be profiled on an unmodified RISC-V processor. From this profiling, common bottlenecks and hotspots in execution will be identified. Based on these results, this project will extend the RISC-V Instruction Set Architecture (ISA) and design a modified processor to support the extensions.

1 Project Summary

1.1 Problem Statement

Current processors cannot compute motion planning algorithms quickly enough for robots to operate in high complexity environments. Autonomous drones are a specific case of robots requiring real-time motion planning in complex environments. The state-of-the-art strategy of using a Graphics Processing Unit (GPU) to accelerate the execution of these algorithms requires too much power to be cost effective or feasible for drones to sustain flight for useful periods of time.

1.2 End User

The end user of this project is a developer of autonomous drones. Such developers have a need for computing hardware that executes motion planning algorithms faster and more power efficiently than existing methods. This thesis will provide a processor design that is synthesizable on an Field Programmable Gate Array (FPGA), giving developers a processer for which a Real-Time Operating Systems (RTOS), or bare metal code, can be written. Additionally, these developers have a requirement that using a new processor for a drone will not require a massive investment in re-development. As such, this thesis will provide the toolchain necessary to compile C code into executable instructions on the new processor.

1.2.1 Stakeholder Map

Where the processor and ISA lies within the map of drone, developer, user of drones.

1.3 Project Requirements

Table 1 outlines conceptually the requirements of the project. Specific technical specifications will be detailed in later sections.

Requirement	**Description**
RISC-V Compliance	RISC-V is an extendable ISA. This project will extend the ISA to add new instructions. The contstraint of RISC-V compliance means that the new ISA I define must follow RISC-V conventions, and that the processor can implement any program compiled into the original RISC-V ISA, as well as the extended RISC-V ISA.
Synthesizable	The finished processor design must be such that it is practically useable by drone developers. Since having such a processor design mass-produced on a chip is beyond the scope of this project, this project must deliver a processor defined in an Hardware Description Language (HDL) that is synthesizable on an FPGA.
Speed	One of the motivating factors of this project is the need for motion planning algorithms to execute faster for autonomous drones to become more useful in real world applications.
Power consumption	The second motivating factor is the need for computation aboard drones to be as power efficient as possible to enable them to remain in flight for long enough periods of time.

Table 1: Conceptual Outline of Project Requirements

1.4 Project Goals

This thesis aims to design a RISC-V processor, optimized for motion planning computation, that is synthesizeable on an FPGA and adheres to the requirements outlined in Section 1.3. It will also provide the tools necessary to complile programs for the processor.

The nature of research into accelerating computation through modified computer architecture is such that, when asked a question of how fast/efficient/small/etc a system must be, the answer is, with consideration to certain trade-offs, as fast/efficient/small/etc as it can be! Thus, when defining goals for certain metrics such as speed or power efficiency, this thesis will do so by comparing performance of the modified processor with benchmark performance of an unmodified, off-the-shelf RISC-V processor synthesized on the same FPGA.

2 System Model Diagram

3 Overall System Specifications

Let the overall system be defined as 2 part: the Processor Module and the Compiler/Assembler Module. These parts are highlighted in Figure X **Complete!!!**
The following subsections detail the technical specifications of the overall system. As stated in Section 1.4, system specifications based on comparison with benchmarks will be compared against the same programs running on an unmodified, off-the-shelf RISC-V processor synthesized on the same FPGA.

3.1 Speed

3.1.1 Quantitative Description

This project aims to achieve a speedup of at least one order of magnitude (10 times) when compared to benchmark performance, in the execution of pure motion planning algorithm programs. That is, given a program that implements, in C, an algorithm often used in autonomous drone motion planning, this project's system can return a valid path at least 10 times faster than a generic RISC-V processor.

3.1.2 Justification

3.1.3 Measurement

There will be two broad stages of measurement for this metric. First, in simulation, the Vivado Design Suite[1] will allow the execution of a given compiled program to be timed on a simulated processor that is defined in an HDL. Secondly, in synthesis, a to-be-determined tool will allow for me to time the execution of the same program, now on a processor physically synthesized on an FPGA.

3.2 Power Consumption

3.2.1 Quantitative Description

Power is defined as energy dissipated over time. As such, when considering the application of this system in autonomous drones, we want to minimize the amount electrical energy committed to the computation of paths. Since the primary goal of this thesis is to reduce the execution time, it can aim to keep power use comparable between the benchmark system and the new system. If power remains roughly constant, but the time taken to execute a program is reduced 10 times, we should see a proportional improvement in energy efficiency.

3.2.2 Justification

3.2.3 Measurement

4 Processor Specifications

The first subcomponent of the system is the Processor Unit. It has two specifications, RISC-V Compliance and Syntheizability.

4.1 RISC-V Compliance

4.1.1 Quantitative Description

A processor must be such that it supports the execution of all instructions defined in the ISA for which it was designed. So too must this project's finished processor be able to correctly support any correctly compiled RISC-V assembly code. This may range from simple programs compiled into either the original or extended RISC-V ISA, to complete operating systems, whether for drone applications or a generic linux distribution, for example.

4.1.2 Justification

4.1.3 Measurement

Github[2]

4.2 Synthesizable

4.2.1 Quantitative Description

The project must deliver a processor defined in an HDL that is synthesizable on an FPGA for the project to be useful for drone developers. While this project will use and test with the Diligent Zync-7000 System on Chip (SoC)[3], the design should be synthesizable on most Zync boards.

4.2.2 Justification

4.2.3 Measurement

Measurement for this specification is relatively simple. Once the processor is designed in an HDL, it can be synthesized onto an FPGA using the Vivado Design Suite, given the design is synthesizable (although this is not always simple to achieve). If it is not synthesizable, the design suite will throw and error. Finally, while the design should be correct and RISC-V compliant by the tests performed in section 4.1, to be safe, these compliance tests along with any other unit tests designed during this thesis will then be run on the FPGA processor.

5 Recompiler & Assembler Specifications

5.1 Input and Output Relationship

5.1.1 Quantitative Description

5.1.2 Justification

5.1.3 Measurement

5.2 placeholder

5.2.1 Quantitative Description

5.2.2 Justification

5.2.3 Measurement

6 List of Acronyms

FPGA Field Programmable Gate Array

GPU Graphics Processing Unit

HDL Hardware Description Language

ISA Instruction Set Architecture

RTOS Real-Time Operating Systems

SoC System on Chip

References

[1] “Vivado Design Suite,” 2018.

[2] J. Bennett and L. Moore, “RISC-V Compliance Github Repository.”

[3] “Diligent Zync-7000 SoC.”