This is the title of the thesis

Daniel Leoncio Paredes Zevallos

A thesis presented for the degree of
Doctor of Philosophy

Supervised by:
Professor Louis Fage
Captain J. Y. Cousteau

University College London, UK
January 2015

I, AUTHORNAME confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam et turpis gravida, lacinia ante sit amet, sollicitudin erat. Aliquam efficitur vehicula leo sed condimentum. Phasellus lobortis eros vitae rutrum egestas. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Donec at urna imperdiet, vulputate orci eu, sollicitudin leo. Donec nec dui sagittis, malesuada erat eget, vulputate tellus. Nam ullamcorper efficitur iaculis. Mauris eu vehicula nibh. In lectus turpis, tempor at felis a, egestas fermentum massa.

Acknowledgements

Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam congue fermentum ante, semper porta nisl consectetur ut. Duis ornare sit amet dui ac faucibus. Phasellus ullamcorper leo vitae arcu ultricies cursus. Duis tristique lacus eget metus bibendum, at dapibus ante malesuada. In dictum nulla nec porta varius. Fusce et elit eget sapien fringilla maximus in sit amet dui.

Mauris eget blandit nisi, faucibus imperdiet odio. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium. Donec pharetra arcu vitae euismod tincidunt. Morbi ut turpis volutpat, ultrices felis non, finibus justo. Proin convallis accumsan sem ac vulputate. Sed rhoncus ipsum eu urna placerat, sed rhoncus erat facilisis. Praesent vitae vestibulum dui. Proin interdum tellus ac velit varius, sed finibus turpis placerat.

Table of Contents

List of figures

List of tables

Abbreviations

API	**A**pplication **P**rogramming **I**nterface
JSON	**J**ava**S**cript **O**bject **N**otation

Chapter 1

Introduction

1.1 Motivation

Car manufactures want to reduce cost in terms of money and time required to develop, test and validate a new piece of software due to a change of supplier. For that reason, centralized end-to-end architectures are the solution they are aiming to, because for car companies such as BWW and Audi the car of future will be similar to a “data center on wheels” [1].

Centralized end-to-end architectures would be the first step stone toward to decoupling software and hardware [2]. This type of architectures not only will take advantage of internet connectivity, cloud computing and powerful heterogeneous processing units, but also will allow scalable, hierarchical and highly integrated system.

In other words, car manufactures prefer now a days low-latency, hierarchical and cost effectiveness of centralized end-to-end architectures, because today requirements of computational power, bandwidth, integration, safety and real-time [3].

However, car manufactures don’t forget that at the end, in centralized end-to-end architectures, different types of software would run on top of an heterogeneous hardware supplied by companies such as NVIDIA, Mobileye or Qualcomm. Thus, it’s important to analyze and understand how software will behave under those conditions, in order to ensure a predictable and efficient system.

1.2 Industrial challenge WATERS 2019

Predictability is a key property for safety-critical and hard real-time systems [4]. Analyzing time related characteristics is an important step to design predictable embedded systems. However, in multi-core or heterogeneous systems based on centralized end-to-end architectures is harder to satisfy timing constrains due to scheduling, caches, pipelines, out-of-order executions, and different kinds of speculation [5]. Thus, development of timing-analysis methods for these types of architectures has become, nowadays, one of the main focus of research in both industry and academic environment.

Robert Bosch GmbH or Bosch, the German multinational engineering and electronics company, and one of the top leaders in development technology for the automotive industry announces every year *the WATERS Challenge*. The purpose of the WATERS industrial challenge is to share ideas, experiences and solutions to concrete timing verification problems issued from real industrial case studies [6].

This year, 2019, the challenge focuses on timing-analysis for heterogeneous software-hardware systems based on centralized end-to-end architectures. The platform chosen for this purpose is the NVIDIA® Jetson™ TX2 platform which has an heterogeneous architecture equipped with a Quad ARM A57 processor, a Dual Denver processor, 8GB of LPDDR4 memory and 256 CUDA cores of NVIDIA's Pascal Architecture. For the challenge it is available an Amalthea model for this platform to design a solution, and test it later on real hardware.

1.3 NVIDIA Jetson TX2: Architecture Overview

NVIDIA Jetson TX2 is an embedded system-on-module (SOM). It is ideal for deploying advanced AI to remote field locations with poor or expensive internet connectivity, Robotics, Gaming Devices, Virtual Reality (VR), Augmented Reality (AR) and Portable Medical Devices. In addition, it offers near-real-time responsiveness and minimal latency—key for intelligent machines that need mission-critical autonomy [7].

The main components of the Jetson TX2 are dual-core ARMv8 based NVIDIA Denver2, quad-core ARMv8 Cortex-A57, 8GB 128-bit LPDDR4 and integrated 256-core Pascal NVIDIA GPU. The quad-core Cortex-A57 and dual-core NVIDIA Denver2 can be seen as a cluster of heterogeneous multiprocessors (HMP) [8]. Both HMP and GPU shares a 8GB SRAM memory as shown in Figure 1.1. Hereafter, whenever we use the term **host**, we will refer to HMP, similarly we will use **device** to refer to GPU.

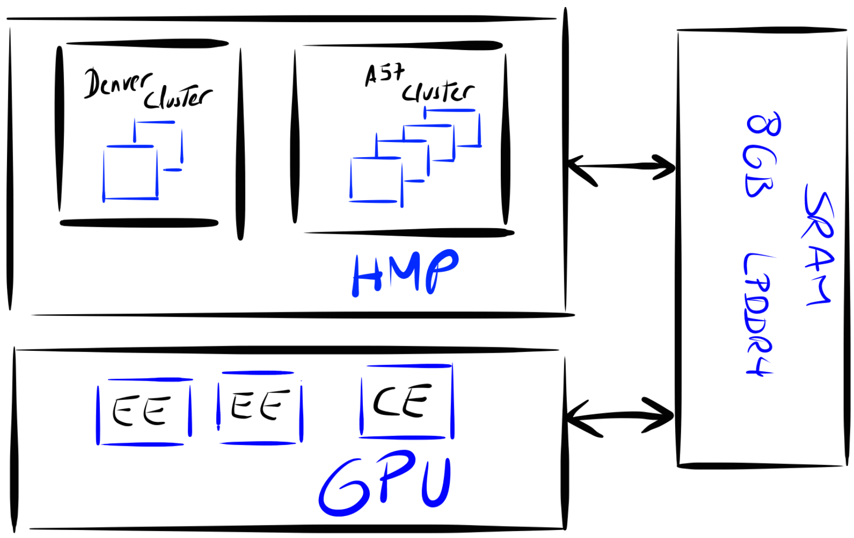

Figure 1.1: Jetson TX2 Architecture Overview

Any NVIDIA GPU has two types of engines, **Copy Engines** (CE) and **Execution Engines** (EE). The Jetson TX2 has only one CE and two EE also known as **Streaming multiprocessors**. CE is in charge of data transfers from host to device and viceversa. There is, moreover, the possibility that

EE and CE can run concurrently.

The GPU uses **streams** to run applications. The number of streams depends on the GPU resources. An application can run in one or multiple streams, the GPU scheduler, by default, manages how to application will be allocated on streams in order to maximize throughput. In Chapter 2, we will discuss in more detail how the TX2 GPU scheduler behave in case of multiple applications.

1.4 Jetson TX2 Amalthea Model

AMALTHEA is a platform for engineering multi- and many-core embedded systems. This platform enables the creation and management of complex tool chains including simulation and validation [9]. In the context of WATERS Challenge 2019, Bosch offers an AMALTHEA model of the Jetson TX2. In this model, a CPU runnable will read data from memory, execute some computation (Ticks) and write back data into memory as shown in Figure 1.2.

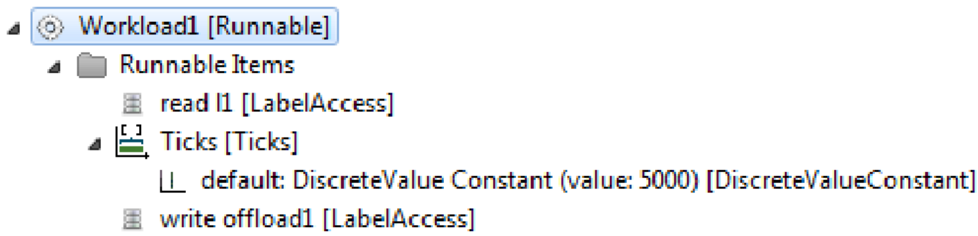

Figure 1.2: Runnable example for a CPU [6]

In the case of GPU modeling, the runnable will follow the same pattern as in the CPU case: read, execution, write back. However, the reading operation is actually to copy memory from host to device, thus it is modeled as *memory reading from host* and then as *memory writing to device*. On the other hand, the writing back operation requires to copy memory from device to host, therefore it is modeled as *memory reading from device* and then as *memory writing to host* as shown in Figure 1.3.

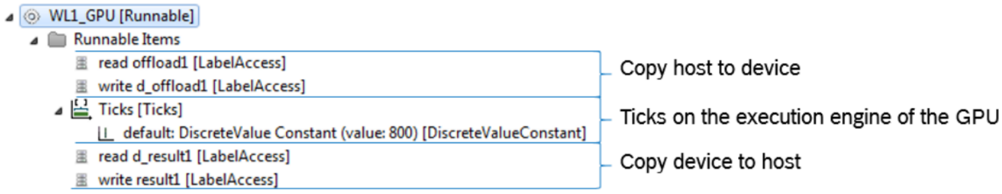

Figure 1.3: Runnable example for a GPU [6]

Chapter 2

Literature review, with maths

2.1 Jetson TX2

2.1.1 NVIDIA Denver2

The other ARM cluster is composed of two NVIDIA Denver2 cores and 2MB of L2 Cache. Each NVIDIA Denver2 is a custom made 64-bit processing based on ARMv8, which is optimized for single-thread performance [10].

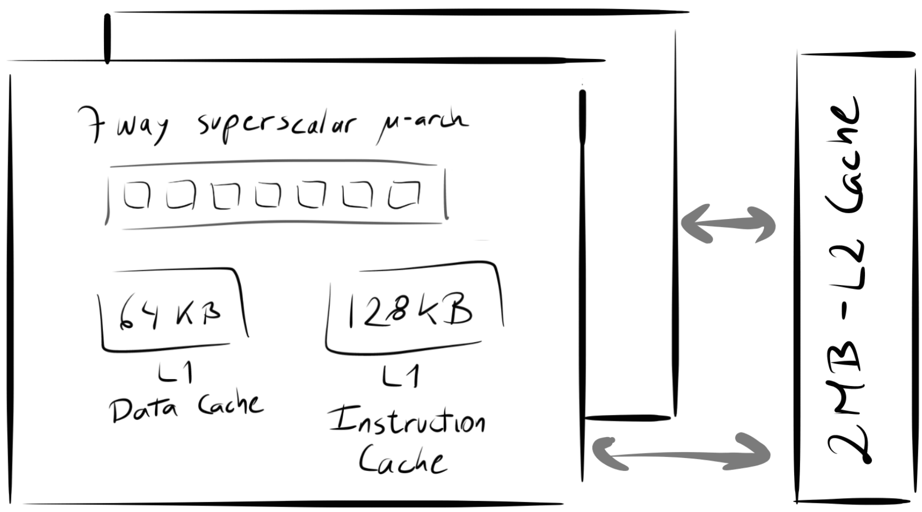

Figure 2.1: Diagram of the Denver Cluster in TX2

This custom processor has two main characteristics. First, it has a 7-way superscalar microarchitecture. It means that a Denver2 core can process 7 operations per clock cycle. Secondly, it uses *Dynamic Code Optimization*. Frequently used software routines are converted into a dense and highly tunned equivalent microcode executable only by Denver2 cores. A Denver2 core analyze the ARM code just before execution and look for places where instructions can be handle together to maximize throughput taking in advantage the 7-way superscalar microarchitecture. In Figure 2.1 can be observed that each Denver2 core has 64KB of L1 Data Cache, and 128KB of Instruction Cache. The optimized code is stored in the former one, also known as *Optimization Cache* [10].

2.2 Introduction

This is the introduction. Duis in neque felis. In hac habitasse platea dictumst. Cras eget rutrum elit. Pellentesque tristique venenatis pellentesque. Cras eu dignissim quam, vel sodales felis. Vestibulum efficitur justo a nibh cursus eleifend. Integer ultrices lorem at nunc efficitur lobortis.

2.3 The middle

This is the literature review. Nullam quam odio, volutpat ac ornare quis, vestibulum nec nulla. Aenean nec dapibus in mL/min^{-1}. Mathematical formula can be inserted using Latex:

$$f(x) = ax^3 + bx^2 + cx + d \tag{2.1}$$

Nunc eleifend, ex a luctus porttitor, felis ex suscipit tellus, ut sollicitudin sapien purus in libero. Nulla blandit eget urna vel tempus. Praesent fringilla dui sapien, sit amet egestas leo sollicitudin at.

(eq. 2.1) Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Sed faucibus pulvinar volutpat. Ut semper fringilla erat non dapibus. Nunc vitae felis eget purus placerat finibus laoreet ut nibh.

2.4 Conclusion

This is the conclusion. Donec pulvinar molestie urna eu faucibus. In tristique ut neque vel eleifend. Morbi ut massa vitae diam gravida iaculis. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

- first item in the list
- second item in the list
- third item in the list

Chapter 3

First research study, with code

3.1 Introduction

This is the introduction. Nam mollis congue tortor, sit amet convallis tortor mollis eget. Fusce viverra ut magna eu sagittis. Vestibulum at ultrices sapien, at elementum urna. Nam a blandit leo, non lobortis quam. Aliquam feugiat turpis vitae tincidunt ultricies. Mauris ullamcorper pellentesque nisl, vel molestie lorem viverra at.

3.2 Method

Suspendisse iaculis in lacus ut dignissim. Cras dignissim dictum eleifend. Suspendisse potenti. Suspendisse et nisi suscipit, vestibulum est at, maximus sapien. Sed ut diam tortor.

3.2.1 Subsection 1 with example code block

This is the first part of the methodology. Cras porta dui a dolor tincidunt placerat. Cras scelerisque sem et malesuada vestibulum. Vivamus faucibus ligula ac sodales consectetur. Aliquam vel tristique nisl. Aliquam erat volutpat. Pellentesque iaculis enim sit amet posuere facilisis. Integer egestas quam sit amet nunc maximus, id bibendum ex blandit.

For syntax highlighting in code blocks, add three “‘” characters before and after a code block:

```
mood = 'happy'
if mood == 'happy':
    print("I am a happy robot")
```

Alternatively, you can also use LaTeX to create a code block as shown in the Java example below:

Listing 3.1: Main.java

```
/**
 * Hello, world – example in Java.
 */
public class Main{
    // says hello to the world
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

If you use `javaCodeStyle` as defined in the `preamble.tex`, it is best to keep the maximum line length in the source code at 80 characters.

3.2.2 Subsection 2

This is the second part of the methodology. Proin tincidunt odio non sem mollis tristique. Fusce pharetra accumsan volutpat. In nec mauris vel orci rutrum dapibus nec ac nibh. Praesent malesuada sagittis nulla, eget commodo mauris ultricies eget. Suspendisse iaculis finibus ligula.

3.3 Results

These are the results. Ut accumsan tempus aliquam. Sed massa ex, egestas non libero id, imperdiet scelerisque augue. Duis rutrum ultrices arcu et ultricies. Proin vel elit eu magna mattis vehicula. Sed ex erat, fringilla vel feugiat ut, fringilla non diam.

3.4 Discussion

This is the discussion. Duis ultrices tempor sem vitae convallis. Pellentesque lobortis risus ac nisi varius bibendum. Phasellus volutpat aliquam varius. Mauris vitae neque quis libero volutpat finibus. Nunc diam metus, imperdiet vitae leo sed, varius posuere orci.

3.5 Conclusion

This is the conclusion to the chapter. Praesent bibendum urna orci, a venenatis tellus venenatis at. Etiam ornare, est sed lacinia elementum, lectus diam tempor leo, sit amet elementum ex elit id ex. Ut ac viverra turpis. Quisque in nisl auctor, ornare dui ac, consequat tellus.

Chapter 4

Research containing a figure

4.1 Introduction

This is the introduction. Sed vulputate tortor at nisl blandit interdum. Cras sagittis massa ex, quis eleifend purus condimentum congue. Maecenas tristique, justo vitae efficitur mollis, mi nulla varius elit, in consequat ligula nulla ut augue. Phasellus diam sapien, placerat sit amet tempor non, lobortis tempus ante.

4.2 Method

Donec imperdiet, lectus vestibulum sagittis tempus, turpis dolor euismod justo, vel tempus neque libero sit amet tortor. Nam cursus commodo tincidunt.

4.2.1 Subsection 1

This is the first part of the methodology. Duis tempor sapien sed tellus ultrices blandit. Sed porta mauris tortor, eu vulputate arcu dapibus ac. Curabitur sodales at felis efficitur sollicitudin. Quisque at neque sollicitudin, mollis arcu vitae, faucibus tellus.

4.2.2 Subsection 2

This is the second part of the methodology. Sed ut ipsum ultrices, interdum ipsum vel, lobortis diam. Curabitur sit amet massa quis tortor molestie dapibus a at libero. Mauris mollis magna quis ante vulputate consequat. Integer leo turpis, suscipit ac venenatis pellentesque, efficitur non sem. Pellentesque eget vulputate turpis. Etiam id nibh at elit fermentum interdum.

4.3 Results

These are the results. In vitae odio at libero elementum fermentum vel iaculis enim. Nullam finibus sapien in congue condimentum. Curabitur et ligula et ipsum mollis fringilla.

4.4 Discussion

Figure 4.1 shows how to add a figure. Donec ut lacinia nibh. Nam tincidunt augue et tristique cursus. Vestibulum sagittis odio nisl, a malesuada turpis blandit quis. Cras ultrices metus tempor laoreet sodales. Nam molestie ipsum ac imperdiet laoreet. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Figure 4.1: RV Calypso is a former British Royal Navy minesweeper converted into a research vessel for the oceanographic researcher Jacques-Yves Cousteau. It was equipped with a mobile laboratory for underwater field research.

4.5 Conclusion

This is the conclusion to the chapter. Quisque nec purus a quam consectetur volutpat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. In lorem justo, convallis quis lacinia eget, laoreet eu metus. Fusce blandit tellus tellus. Curabitur nec cursus odio. Quisque tristique eros nulla, vitae finibus lorem aliquam quis. Interdum et malesuada fames ac ante ipsum primis in faucibus.

Chapter 5

Research containing a table

5.1 Introduction

This is the introduction. Phasellus non purus id mauris aliquam rutrum vitae quis tellus. Maecenas rhoncus ligula nulla, fringilla placerat mi consectetur eu. Aenean nec metus ac est ornare posuere. Nunc ipsum lacus, gravida commodo turpis quis, rutrum eleifend erat. Pellentesque id lorem eget ante porta tincidunt nec nec tellus.

5.2 Method

Vivamus consectetur, velit in congue lobortis, massa massa lacinia urna, sollicitudin semper ipsum augue quis tortor. Donec quis nisl at arcu volutpat ultrices. Maecenas ex nibh, consequat ac blandit sit amet, molestie in odio. Morbi finibus libero et nisl dignissim, at ultricies ligula pulvinar.

5.2.1 Subsection 1

This is the first part of the methodology. Integer leo erat, commodo in lacus vel, egestas varius elit. Nulla eget magna quam. Nullam sollicitudin dolor ut ipsum varius tincidunt. Duis dignissim massa in ipsum accumsan imperdiet.

Maecenas suscipit sapien sed dui pharetra blandit. Morbi fermentum est vel quam pretium maximus.

5.2.2 Subsection 2

This is the second part of the methodology. Nullam accumsan condimentum eros eu volutpat. Maecenas quis ligula tempor, interdum ante sit amet, aliquet sem. Fusce tellus massa, blandit id tempus at, cursus in tortor. Nunc nec volutpat ante. Phasellus dignissim ut lectus quis porta. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

5.3 Results

Table 5.1 shows us how to add a table. Integer tincidunt sed nisl eget pellentesque. Mauris eleifend, nisl non lobortis fringilla, sapien eros aliquet orci, vitae pretium massa neque eu turpis. Pellentesque tincidunt aliquet volutpat. Ut ornare dui id ex sodales laoreet.

Table 5.1: This is the table caption. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium.

Column 1	Column 2	Column 3
Row 1	0.1	0.2
Row 2	0.3	0.3
Row 3	0.4	0.4
Row 4	0.5	0.6

5.4 Discussion

This is the discussion. Etiam sit amet mi eros. Donec vel nisi sed purus gravida fermentum at quis odio. Vestibulum quis nisl sit amet justo maximus molestie. Maecenas vitae arcu erat. Nulla facilisi. Nam pretium mauris eu enim porttitor, a mattis velit dictum. Nulla sit amet ligula non mauris volutpat fermentum quis vitae sapien.

5.5 Conclusion

This is the conclusion to the chapter. Nullam porta tortor id vehicula interdum. Quisque pharetra, neque ut accumsan suscipit, orci orci commodo tortor, ac finibus est turpis eget justo. Cras sodales nibh nec mauris laoreet iaculis. Morbi volutpat orci felis, id condimentum nulla suscipit eu. Fusce in turpis quis ligula tempus scelerisque eget quis odio. Vestibulum et dolor id erat lobortis ullamcorper quis at sem.

Chapter 6

Final research study

6.1 Introduction

This is the introduction. Nunc lorem odio, laoreet eu turpis at, condimentum sagittis diam. Phasellus metus ligula, auctor ac nunc vel, molestie mattis libero. Praesent id posuere ex, vel efficitur nibh. Quisque vestibulum accumsan lacus vitae mattis.

6.2 Method

In tincidunt viverra dolor, ac pharetra tellus faucibus eget. Pellentesque tempor a enim nec venenatis. Morbi blandit magna imperdiet posuere auctor. Maecenas in maximus est.

6.2.1 Subsection 1

This is the first part of the methodology. Praesent mollis sem diam, sit amet tristique lacus vulputate quis. Vivamus rhoncus est rhoncus tellus lacinia, a interdum sem egestas. Curabitur quis urna vel quam blandit semper vitae a leo. Nam vel lectus lectus.

6.2.2 Subsection 2

This is the second part of the methodology. Aenean vel pretium tortor. Aliquam erat volutpat. Quisque quis lobortis mi. Nulla turpis leo, ultrices nec nulla non, ullamcorper laoreet risus.

6.3 Results

These are the results. Curabitur vulputate nisl non ante tincidunt tempor. Aenean porta nisi quam, sed ornare urna congue sed. Curabitur in sapien justo. Quisque pulvinar ullamcorper metus, eu varius mauris pellentesque et. In hac habitasse platea dictumst. Pellentesque nec porttitor libero. Duis et magna a massa lacinia cursus.

6.4 Discussion

This is the discussion. Curabitur gravida nisl id gravida congue. Duis est nisi, sagittis eget accumsan ullamcorper, semper quis turpis. Mauris ultricies diam metus, sollicitudin ultricies turpis lobortis vitae. Ut egestas vehicula enim, porta molestie neque consectetur placerat. Integer iaculis sapien dolor, non porta nibh condimentum ut.

6.5 Conclusion

This is the conclusion to the chapter. Nulla sed condimentum lectus. Duis sed tempor erat, at cursus lacus. Nam vitae tempus arcu, id vestibulum sapien. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Chapter 7

Conclusion

7.1 Thesis summary

In summary, pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Nunc eleifend, ex a luctus porttitor, felis ex suscipit tellus, ut sollicitudin sapien purus in libero. Nulla blandit eget urna vel tempus. Praesent fringilla dui sapien, sit amet egestas leo sollicitudin at.

7.2 Future work

There are several potential directions for extending this thesis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam gravida ipsum at tempor tincidunt. Aliquam ligula nisl, blandit et dui eu, eleifend tempus nibh. Nullam eleifend sapien eget ante hendrerit commodo. Pellentesque pharetra erat sit amet dapibus scelerisque.

Vestibulum suscipit tellus risus, faucibus vulputate orci lobortis eget. Nunc varius sem nisi. Nunc tempor magna sapien, euismod blandit elit pharetra sed. In dapibus magna convallis lectus sodales, a consequat sem euismod. Curabitur in interdum purus. Integer ultrices laoreet aliquet. Nulla vel dapibus urna. Nunc efficitur erat ac nisi auctor sodales.

Appendix 1: Some extra stuff

Add appendix 1 here. Vivamus hendrerit rhoncus interdum. Sed ullamcorper et augue at porta. Suspendisse facilisis imperdiet urna, eu pellentesque purus suscipit in. Integer dignissim mattis ex aliquam blandit. Curabitur lobortis quam varius turpis ultrices egestas.

Appendix 2: Some more extra stuff

Add appendix 2 here. Aliquam rhoncus mauris ac neque imperdiet, in mattis eros aliquam. Etiam sed massa et risus posuere rutrum vel et mauris. Integer id mauris sed arcu venenatis finibus. Etiam nec hendrerit purus, sed cursus nunc. Pellentesque ac luctus magna. Aenean non posuere enim, nec hendrerit lacus. Etiam lacinia facilisis tempor. Aenean dictum nunc id felis rhoncus aliquam.

References

[1] "BMW and audi want to separate vehicle hardware from software." https://www.electronicdesign.com/automotive/bmw-and-audi-want-separate-vehicle-hardware-software.

[2] "End to end architecture." https://www.future-mobility-tech.com/en/technology/end-to-end-architecture.

[3] S. Kanajan, H. Zeng, C. Pinello, and A. Sangiovanni-Vincentelli, "Exploring trade-off's between centralized versus decentralized automotive architectures using a virtual integration environment," in *Proceedings of the conference on design, automation and test in europe: Proceedings*, 2006, pp. 548–553.

[4] T. A Henzinger, "Two challenges in embedded systems design: Predictability and robustness," *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, vol. 366, pp. 3727–36, Nov. 2008.

[5] C. Cullmann *et al.*, "Predictability considerations in the design of multi-core embedded systems," *Ingénieurs de l'Automobile*, vol. 807, pp. 36–42, Sep. 2010.

[6] "WATERS 2019 – industrial challenge." https://www.ecrts.org/waters/waters-industrial-challenge/.

[7] D. Franklin, "NVIDIA jetson tx2 delivers twice the intelligence to the edge." https://devblogs.nvidia.com/jetson-tx2-delivers-twice-intelligence-edge/.

[8] *NVIDIA jetson tx2 system-on-module*. NVIDIA Corporation, 2014.

[9] E. APP4MC, "Project profile: Eclipse app4mc." http://www.amalthea-project.org/.

[10] S. Anthony, "Nvidia details 64-bit denver tegra k1, claims haswell-class performance for first 64-bit android chip." https://www.extremetech.com/computing/187862-nvidia-details-64-bit-denver-tegra-k1-claims-haswell-class-performance-for-first-64-bit-android-chip.