

HERIOT-WATT UNIVERSITY

MASTERS THESIS

Hardware acceleration with mixed-precision

Author:

Quentin DUCASSE

Supervisor:

Dr. Rob STEWART

*A thesis submitted in fulfilment of the requirements
for the degree of MSc. Network Security*

in the

School of Mathematical and Computer Sciences

March 2020



Declaration of Authorship

I, Quentin DUCASSE, declare that this thesis titled, 'Hardware acceleration with mixed-precision' and the work presented in it is my own. I confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of other authors in any form (e.g., ideas, equations, figures, text, tables, programs) are properly acknowledged at any point of their use. A list of the references employed is included.

Signed:

Date:

“Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.”

Dave Barry

Abstract

The Thesis Abstract is written here (and usually kept to just this page).

Acknowledgements

The acknowledgements and the people to thank go here, don't forget to include your project advisor :)

Contents

Declaration of Authorship	i
Abstract	iii
Acknowledgements	iv
Contents	v
List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Main Section 1	1
1.1.1 Subsection 1	1
1.1.2 Subsection 2	1
1.2 Main Section 2	2
2 Literature Review	3
2.1 Background	3
2.1.1 Number Representation	3
2.1.1.1 Integer Representation	4
2.1.1.2 Floating-Point Representation	4
2.1.1.3 Fixed-Point Representation	5
2.1.2 Performance benchmarks	6
2.2 Main Section 2	6
3 Requirements Analysis	8
3.1 Main Section 1	8
3.1.1 Subsection 1	8
3.1.2 Subsection 2	8
3.2 Main Section 2	9
4 Professional, Legal, Ethical and Social issues	10
4.1 Main Section 1	10
4.1.1 Subsection 1	10
4.1.2 Subsection 2	10

4.2	Main Section 2	11
5	Project Plan	12
5.1	Main Section 1	12
5.1.1	Subsection 1	12
5.1.2	Subsection 2	12
5.2	Main Section 2	13
6	Conclusion	14
6.1	Main Section 1	14
6.1.1	Subsection 1	14
6.1.2	Subsection 2	14
6.2	Main Section 2	15
A	Appendix Title Here	16

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Main Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam ultricies lacinia euismod. Nam tempus risus in dolor rhoncus in interdum enim tincidunt. Donec vel nunc neque. In condimentum ullamcorper quam non consequat. Fusce sagittis tempor feugiat. Fusce magna erat, molestie eu convallis ut, tempus sed arcu. Quisque molestie, ante a tincidunt ullamcorper, sapien enim dignissim lacus, in semper nibh erat lobortis purus. Integer dapibus ligula ac risus convallis pellentesque.

1.1.1 Subsection 1

Nunc posuere quam at lectus tristique eu ultrices augue venenatis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam erat volutpat. Vivamus sodales tortor eget quam adipiscing in vulputate ante ullamcorper. Sed eros ante, lacinia et sollicitudin et, aliquam sit amet augue. In hac habitasse platea dictumst.

1.1.2 Subsection 2

Morbi rutrum odio eget arcu adipiscing sodales. Aenean et purus a est pulvinar pellentesque. Cras in elit neque, quis varius elit. Phasellus fringilla, nibh eu tempus venenatis, dolor elit posuere quam, quis adipiscing urna leo nec orci. Sed nec nulla auctor odio aliquet consequat. Ut nec nulla in ante ullamcorper aliquam at sed dolor. Phasellus fermentum magna in augue gravida cursus. Cras sed pretium lorem. Pellentesque eget

ornare odio. Proin accumsan, massa viverra cursus pharetra, ipsum nisi lobortis velit, a malesuada dolor lorem eu neque.

1.2 Main Section 2

Sed ullamcorper quam eu nisl interdum at interdum enim egestas. Aliquam placerat justo sed lectus lobortis ut porta nisl porttitor. Vestibulum mi dolor, lacinia molestie gravida at, tempus vitae ligula. Donec eget quam sapien, in viverra eros. Donec pellen-tesque justo a massa fringilla non vestibulum metus vestibulum. Vestibulum in orci quis felis tempor lacinia. Vivamus ornare ultrices facilisis. Ut hendrerit volutpat vulputate. Morbi condimentum venenatis augue, id porta ipsum vulputate in. Curabitur luctus tempus justo. Vestibulum risus lectus, adipiscing nec condimentum quis, condimentum nec nisl. Aliquam dictum sagittis velit sed iaculis. Morbi tristique augue sit amet nulla pulvinar id facilisis ligula mollis. Nam elit libero, tincidunt ut aliquam at, molestie in quam. Aenean rhoncus vehicula hendrerit.

Chapter 2

Literature Review

2.1 Background

The question of the representation of numbers as we, humans, use them in a world of electronics has been central in the creation of computers and their associated arithmetics. Several problems are contained in the simple question of: How to translate our arithmetic and number operations in a piece of hardware?

2.1.1 Number Representation

The first thing to note is that electronics can represent two states, a presence or absence of an electric impulsion. The states of "on" and "off" is embedded in transistors that can represent both. The transistor is the hardware representant of this duality while a bit is the software counter-part. This is the underlying reason of why computers, even the first fully electronical computer ENIAC - Electronical Numerical Integrator and Computer -, are using a binary system. If this system is handy to translate our base 10 arithmetic and simple numbers such as integers, it is harder to translate more complex numbers such as reals and floating point operations. The meaning of an N-bit binary word is entirely dependent of the interpretation we choose to use. This interpretation consists of both a representation (the type of the object the memory represents) as well as its associated mapping. Common number representations consists of unsigned integers, signed integers (using two's complement), floating point reals as well as fixed-point reals.

2.1.1.1 Integer Representation

The representation of integers and especially unsigned integers is straightforward as it consists of a change from base 10 to base 2. This number representation can be done in 16-bits, 32-bits or 64-bits depending on its type, the supporting hardware and the space we need to contain it. Representing a number in base 2 from base 10 or vice versa is straightforward as it only demands simple and exact basic operations to be performed.

Now, if we want to represent a signed integer, we have to use a method called the two's complement in order to bring the sign in while keeping the basic behavior of the addition to work on numbers whether they are positive or negative. The method consists in changing the value of all the bits of a given number then adding one to the result.

Those two representations allow a complete mapping of integers up to a certain range: signed 32-bits integer can represent base 10 numbers between -2,147,483,648 and 2,147,483,647 while unsigned integers can represent base 10 numbers between 0 and 4,294,967,295.

2.1.1.2 Floating-Point Representation

Representing floating-point numbers has been a concern since the 1980's and the industrial development of several computing modules and interfaces. The need for a consensus in this domain and particular applications has been answered by the IEEE-754 standard in 1985. This standard defines both the floating-point number representations and exceptions conditions along with their default handling. This norm was reviewed fundamentally in 2008, extending it to 64-bits and 128-bits length. The last dated revision of the norm is from 2019.

Floating-point numbers following this representation are composed of three distinct elements:

1. A sign bit
2. An exponent
3. A mantissa

Those three elements compose the number by using the following formula:

$$(sign) \ mantissa * 2^{exponent} \quad (2.1)$$

In order to present both positive and negative exponents and as using the two's complement on the exponent would complexify the computation of floating-point numbers, a bias is used in the exponent. This bias corresponds to

$$2^e - 1 \tag{2.2}$$

where e is the number of bits of the exponent part.

When referring to single-precision floating-point representation we are talking about 32-bit long memory representation. They are mapped as follows:

- Sign bit: 1 bit
- Exponent: 8 bits
- Mantissa: 23 bits
- Exponent Bias: 127

Referring to double precision floating-point representation means looking at 64-bit long memory representation, mapped as follows:

- Sign bit: 1 bit
- Exponent: 11 bits
- Mantissa: 52 bits
- Exponent Bias: 127

Along this representations, IEEE-754 introduces representations of special numbers such as positive and negative infinity as well as NaN and zero. Moreover, it adds methods to round floating-point numbers to positive or negative infinity, zero or to the nearest value.

2.1.1.3 Fixed-Point Representation

Another way to look at the decimals is to fix the radix point to be at a certain place and keep it throughout all the computations and representations using this arithmetic. A fixed-point representation consists of three components:

1. A sign indicator

2. An integer corresponding to the total number of bits
3. Another integer corresponding to the size of the fractional part

Representing a number with this representation can be done by simply concatenating the base 2 representation of each side of the radix point.

As shown in the above example, several representations can depict the same decimal number. Finding the correct amount of bits to allocate to each side of the radix point is what will qualify the representation. Allocating fewer bits than needed may lead to overflow while allocating too much may increase quantisation errors. Along with this new representation comes a whole new arithmetic. While this format can help tailor your needs in terms of variable types, it comes with an additional cost. The operations performed in this arithmetic are non-trivial as addition and multiplication are not associative and distributive anymore. This means the order of the operations will have an impact on the final result. Moreover, the round-off error underlying this representation is often non-trivial to grasp. However, those operations are low demanding in terms of computing power.

2.1.2 Performance benchmarks

Floating-point representation (in either single or double precision) allows extreme precision at the cost of space in memory. On the other hand, fixed-point representation, even if it comes with a more complex arithmetic and insidious round-off errors, allows to tailor the type to your needs. If you want to store the values of the size and mass of planets in floating-point precision, you will end up not using the majority of the range of values you selected while you could tailor a correct type in fixed-point representation.

The operations performed in floating-point are expensive in terms of bandwidth, memory and energy consumption, which ultimately translate in to additional cost to perform an action. Mark Horowitz talk at the ISSCC 2014 was entitled "Computing's Energy Problem (and what we can do about it)" provides an insight of the problems and challenges technology scaling has encountered in its development. The Moore's law is getting outdated and a solution to the issue of permanently growing energy needs resides in "the design of applications and hardware that are better matched to task and each other". The numbers presented by M.Horowitz have been reused by Professor William Dally (Stanford University, NVidia Corporation) in his lecture on "High-Performance Hardware for Machine Learning". The following graph is extracted from this lecture and presents the energy and area costs different floating-point operations demand.

The benefit of using a more optimised representation other the floating-point representation has been investigated as soon as in 2000 in 2000-TONG. The authors present a way to reduce the energy consumption by minimising the bitwidth representation of the floating-point data. The data used to tailor the type representation is human-sensory data such as speech or video imagery. This kind of data is obtained at low precision (4 to 10 bits) but mapped into a full single-precision floating-point type. Using such optimisations, the authors manage to obtain a reduction of 66% in multiplier energy per operation without sacrificing any accuracy.

Xilinx, a manufacturer of Field-Programmable Gate Arrays (FPGAs), produced a white paper written by Ambrose Finnerty and Hervé Ratigner entitled "Reduce Power and Cost by Converting from Floating Point to Fixed Point". This paper shows that the tools Xilinx provide can already handle fixed-point types and arithmetic and that the benefits are important for use of the FPGA:

- Reduced power consumption
- Reduced use of FPGA resources (look-up tables, flip-flops, memory)
- Latency improvements
- Comparable performance and accuracy

The authors show the example of an FIR filter implementation in fixed-point representation from floating-point. The frequency is shown to be 16% faster and the latency 7.5 times lower.

In conclusion, using a type that matches correctly the use you will make of it in an environment where costs are translated in terms of energy consumption, memory or bandwidth usage, will always come out as an improvement with little to no loss in accuracy.

2.2 Mixed Precision

The previous section explained in details the different existing number representations. With a particular emphasis on the representation of floating-point number, the previous section has shown what the underlying issues of such representations are. The question now is how can we use to our advantage the different types along with their strengths and weaknesses?

2.2.1 Motivations

The term "mixed precision" comes from the idea of using different precisions (type representations) to increase the performance of a computation that would otherwise only use the highest available (or user-defined) precision. The previous section shown us that each number representation has its pros and cons. Using lower precision types and representations will allow the total computation to decrease the usage of several hardware resources (memory, bandwidth, energy consumption) 2014-HORROWITZ, 2015-DALLY.

However, if lower-precision is acceptable, using mixed-precision can increase the capacity of smaller systems and subsystems as it will increase the overall performance and resources usage. This goal is targetted when you are aiming for the most performant processor, being given a size and type of usage. Moreover, this can be used when you know exactly the tasks your system will perform as well as their needed size. 2001-YATES presents several rules allowing you to determine the required size of any operation performed in fixed-point arithmetic. This arithmetic, as shown earlier is an interesting choice as it allows you to tailor the type to your needs. Reconfigurable hardware can take advantage of this as well by using the exact required size for an operation. Such architecture will be looked upon on the next sections.

Scientific computations often require the highest available precision and the reduced precision induced by the lower-precision components is unacceptable. In order to still take advantage of the several gains lower-precision provides, a compromise has to be found about how and when to use lower-precision. Ideas on the usage have been looked upon since the 1950's and the first methods to correctly implement them in the early 1960's.

In any case, the developer has to know and keep in mind:

- Required end precision
- Acceptable error rate
- Actual operations performed by the system

Mixed precision methods consist of doing a large part of the computation in low-precision components and only a small part in high-precision. As presented in the survey paper 2009-Goddeke using different formats in the same algorithm can have several beneficial properties such as:

- Accuracy: Same accuracy when up to 99% of the operations is performed in the low format

- Computation: Low precision operations require less transistors and can translate in more parallelism
- Memory: Reduction in size in memory influences the efficiency and bandwidth requirements positively

2.2.2 Methods and Implementations

Several methods to actually implement a mixed precision algorithm have been developed starting 60 years ago. If the methods are different in practice, the theory behind them is the same as they aim to delegate most of the work to low-precision components while only using high-precision to recover the desired accuracy for the computation.

2.2.2.1 Iterative Refinement

Since the early 1960's, literature has studied the effects and ways to implement mixed precision. The first known works to implement such methods is 1963-Wilkinson and 1967-Moler. The method presented by both is named "iterative refinement" and uses a combination of accumulative inner products along with a linear solver. This solution is directed towards the resolution of a system in the form of

$$Ax = b \tag{2.3}$$

for a quadratic

$$NxNA \tag{2.4}$$

matrix. The main idea is to use a residual between the wanted result and the result obtained by the last iteration. This residual is computed using high precision and accumulated using high-precision as well while the resolution of the equation is done in low-precision.

This method has later been reused and extended. 2008-BABOULIN combines single- and double-precision floating-point numbers to enhance the performance of "dense and sparse linear algebra algorithms". 2008-STRODZKA presents an enhanced algorithm to resolve a partial differential equation (Poisson problem) to present the effects of a mixed-precision approach. 2008-SUN reproduces a linear solver in mixed-precision but adds an error analysis. The last two articles implement the result of their research on a reprogrammable architecture. Several applications profit directly from these papers and their implementations of mixed-precision algorithms as we will see in later sections.

2.2.2.2 Analysis and Rewriting

When creating a new application or system, any developer tends to use the highest available (or one of the highest) to represent the numbers he will need. However, the space and operations performed on higher-precision types are, as shown earlier, not the most performant. A simple, yet effective way to choose the correct type is to analyse the code and rewrite types that can be represented by lower precision ones. This idea has been implemented in the code analysis tool Precimonious 2013-RUBIOGONZALEZ for floating-point representations. This analysis tool can go over the code and determines parts of it where lower precision can be applied.

The mixed-precision approach can be applied from a different perspective when using a fixed-precision representation. 2013-DARULOVA shows that several polynomial expressions over the reals are cast over fixed-point arithmetic without being optimised, leading approximations of the real values. The tool they present allows to determine the best fixed-point implementation of a real number thanks to genetic programming. Then, as the order of operations is impactful when dealing with fixed-point arithmetic (distributivity and associativity are not respected), the tool reorder the operations to optimise the error and accuracy.

Those two tools have been brought together in 2018-DARULOVA under the "first fully automated and sound technique and tool for optimising the performance of floating-point and fixed-point kernels". This tool combines rewriting and mixed-precision tuning where rewriting goes over several evaluation orders and picks the one that minimises the round-off error without runtime costs while mixed-precision tuning assigns the correct finite-precision variables and operations. The overall tool provides finer-grain control over the program and better performance than each of the method taken alone.

Chapter 3

Requirements Analysis

3.1 Main Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam ultricies lacinia euismod. Nam tempus risus in dolor rhoncus in interdum enim tincidunt. Donec vel nunc neque. In condimentum ullamcorper quam non consequat. Fusce sagittis tempor feugiat. Fusce magna erat, molestie eu convallis ut, tempus sed arcu. Quisque molestie, ante a tincidunt ullamcorper, sapien enim dignissim lacus, in semper nibh erat lobortis purus. Integer dapibus ligula ac risus convallis pellentesque.

3.1.1 Subsection 1

Nunc posuere quam at lectus tristique eu ultrices augue venenatis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam erat volutpat. Vivamus sodales tortor eget quam adipiscing in vulputate ante ullamcorper. Sed eros ante, lacinia et sollicitudin et, aliquam sit amet augue. In hac habitasse platea dictumst.

3.1.2 Subsection 2

Morbi rutrum odio eget arcu adipiscing sodales. Aenean et purus a est pulvinar pellentesque. Cras in elit neque, quis varius elit. Phasellus fringilla, nibh eu tempus venenatis, dolor elit posuere quam, quis adipiscing urna leo nec orci. Sed nec nulla auctor odio aliquet consequat. Ut nec nulla in ante ullamcorper aliquam at sed dolor. Phasellus fermentum magna in augue gravida cursus. Cras sed pretium lorem. Pellentesque eget

ornare odio. Proin accumsan, massa viverra cursus pharetra, ipsum nisi lobortis velit, a malesuada dolor lorem eu neque.

3.2 Main Section 2

Sed ullamcorper quam eu nisl interdum at interdum enim egestas. Aliquam placerat justo sed lectus lobortis ut porta nisl porttitor. Vestibulum mi dolor, lacinia molestie gravida at, tempus vitae ligula. Donec eget quam sapien, in viverra eros. Donec pellen-tesque justo a massa fringilla non vestibulum metus vestibulum. Vestibulum in orci quis felis tempor lacinia. Vivamus ornare ultrices facilisis. Ut hendrerit volutpat vulputate. Morbi condimentum venenatis augue, id porta ipsum vulputate in. Curabitur luctus tempus justo. Vestibulum risus lectus, adipiscing nec condimentum quis, condimentum nec nisl. Aliquam dictum sagittis velit sed iaculis. Morbi tristique augue sit amet nulla pulvinar id facilisis ligula mollis. Nam elit libero, tincidunt ut aliquam at, molestie in quam. Aenean rhoncus vehicula hendrerit.

Chapter 4

Professional, Legal, Ethical and Social issues

4.1 Main Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam ultricies lacinia euismod. Nam tempus risus in dolor rhoncus in interdum enim tincidunt. Donec vel nunc neque. In condimentum ullamcorper quam non consequat. Fusce sagittis tempor feugiat. Fusce magna erat, molestie eu convallis ut, tempus sed arcu. Quisque molestie, ante a tincidunt ullamcorper, sapien enim dignissim lacus, in semper nibh erat lobortis purus. Integer dapibus ligula ac risus convallis pellentesque.

4.1.1 Subsection 1

Nunc posuere quam at lectus tristique eu ultrices augue venenatis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam erat volutpat. Vivamus sodales tortor eget quam adipiscing in vulputate ante ullamcorper. Sed eros ante, lacinia et sollicitudin et, aliquam sit amet augue. In hac habitasse platea dictumst.

4.1.2 Subsection 2

Morbi rutrum odio eget arcu adipiscing sodales. Aenean et purus a est pulvinar pellentesque. Cras in elit neque, quis varius elit. Phasellus fringilla, nibh eu tempus venenatis, dolor elit posuere quam, quis adipiscing urna leo nec orci. Sed nec nulla auctor odio aliquet consequat. Ut nec nulla in ante ullamcorper aliquam at sed dolor. Phasellus

fermentum magna in augue gravida cursus. Cras sed pretium lorem. Pellentesque eget ornare odio. Proin accumsan, massa viverra cursus pharetra, ipsum nisi lobortis velit, a malesuada dolor lorem eu neque.

4.2 Main Section 2

Sed ullamcorper quam eu nisl interdum at interdum enim egestas. Aliquam placerat justo sed lectus lobortis ut porta nisl porttitor. Vestibulum mi dolor, lacinia molestie gravida at, tempus vitae ligula. Donec eget quam sapien, in viverra eros. Donec pellentesque justo a massa fringilla non vestibulum metus vestibulum. Vestibulum in orci quis felis tempor lacinia. Vivamus ornare ultrices facilisis. Ut hendrerit volutpat vulputate. Morbi condimentum venenatis augue, id porta ipsum vulputate in. Curabitur luctus tempus justo. Vestibulum risus lectus, adipiscing nec condimentum quis, condimentum nec nisl. Aliquam dictum sagittis velit sed iaculis. Morbi tristique augue sit amet nulla pulvinar id facilisis ligula mollis. Nam elit libero, tincidunt ut aliquam at, molestie in quam. Aenean rhoncus vehicula hendrerit.

Chapter 5

Project Plan

For referencing this chapter elsewhere, use [5](#)

5.1 Main Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam ultricies lacinia euismod. Nam tempus risus in dolor rhoncus in interdum enim tincidunt. Donec vel nunc neque. In condimentum ullamcorper quam non consequat. Fusce sagittis tempor feugiat. Fusce magna erat, molestie eu convallis ut, tempus sed arcu. Quisque molestie, ante a tincidunt ullamcorper, sapien enim dignissim lacus, in semper nibh erat lobortis purus. Integer dapibus ligula ac risus convallis pellentesque.

5.1.1 Subsection 1

Nunc posuere quam at lectus tristique eu ultrices augue venenatis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam erat volutpat. Vivamus sodales tortor eget quam adipiscing in vulputate ante ullamcorper. Sed eros ante, lacinia et sollicitudin et, aliquam sit amet augue. In hac habitasse platea dictumst.

5.1.2 Subsection 2

Morbi rutrum odio eget arcu adipiscing sodales. Aenean et purus a est pulvinar pellentesque. Cras in elit neque, quis varius elit. Phasellus fringilla, nibh eu tempus venenatis, dolor elit posuere quam, quis adipiscing urna leo nec orci. Sed nec nulla auctor odio

aliquet consequat. Ut nec nulla in ante ullamcorper aliquam at sed dolor. Phasellus fermentum magna in augue gravida cursus. Cras sed pretium lorem. Pellentesque eget ornare odio. Proin accumsan, massa viverra cursus pharetra, ipsum nisi lobortis velit, a malesuada dolor lorem eu neque.

5.2 Main Section 2

Sed ullamcorper quam eu nisl interdum at interdum enim egestas. Aliquam placerat justo sed lectus lobortis ut porta nisl porttitor. Vestibulum mi dolor, lacinia molestie gravida at, tempus vitae ligula. Donec eget quam sapien, in viverra eros. Donec pellentesque justo a massa fringilla non vestibulum metus vestibulum. Vestibulum in orci quis felis tempor lacinia. Vivamus ornare ultrices facilisis. Ut hendrerit volutpat vulputate. Morbi condimentum venenatis augue, id porta ipsum vulputate in. Curabitur luctus tempus justo. Vestibulum risus lectus, adipiscing nec condimentum quis, condimentum nec nisl. Aliquam dictum sagittis velit sed iaculis. Morbi tristique augue sit amet nulla pulvinar id facilisis ligula mollis. Nam elit libero, tincidunt ut aliquam at, molestie in quam. Aenean rhoncus vehicula hendrerit.

Chapter 6

Conclusion

For referencing this chapter elsewhere, use [6](#)

6.1 Main Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam ultricies lacinia euismod. Nam tempus risus in dolor rhoncus in interdum enim tincidunt. Donec vel nunc neque. In condimentum ullamcorper quam non consequat. Fusce sagittis tempor feugiat. Fusce magna erat, molestie eu convallis ut, tempus sed arcu. Quisque molestie, ante a tincidunt ullamcorper, sapien enim dignissim lacus, in semper nibh erat lobortis purus. Integer dapibus ligula ac risus convallis pellentesque.

6.1.1 Subsection 1

Nunc posuere quam at lectus tristique eu ultrices augue venenatis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam erat volutpat. Vivamus sodales tortor eget quam adipiscing in vulputate ante ullamcorper. Sed eros ante, lacinia et sollicitudin et, aliquam sit amet augue. In hac habitasse platea dictumst.

6.1.2 Subsection 2

Morbi rutrum odio eget arcu adipiscing sodales. Aenean et purus a est pulvinar pellentesque. Cras in elit neque, quis varius elit. Phasellus fringilla, nibh eu tempus venenatis, dolor elit posuere quam, quis adipiscing urna leo nec orci. Sed nec nulla auctor odio

aliquet consequat. Ut nec nulla in ante ullamcorper aliquam at sed dolor. Phasellus fermentum magna in augue gravida cursus. Cras sed pretium lorem. Pellentesque eget ornare odio. Proin accumsan, massa viverra cursus pharetra, ipsum nisi lobortis velit, a malesuada dolor lorem eu neque.

6.2 Main Section 2

Sed ullamcorper quam eu nisl interdum at interdum enim egestas. Aliquam placerat justo sed lectus lobortis ut porta nisl porttitor. Vestibulum mi dolor, lacinia molestie gravida at, tempus vitae ligula. Donec eget quam sapien, in viverra eros. Donec pellentesque justo a massa fringilla non vestibulum metus vestibulum. Vestibulum in orci quis felis tempor lacinia. Vivamus ornare ultrices facilisis. Ut hendrerit volutpat vulputate. Morbi condimentum venenatis augue, id porta ipsum vulputate in. Curabitur luctus tempus justo. Vestibulum risus lectus, adipiscing nec condimentum quis, condimentum nec nisl. Aliquam dictum sagittis velit sed iaculis. Morbi tristique augue sit amet nulla pulvinar id facilisis ligula mollis. Nam elit libero, tincidunt ut aliquam at, molestie in quam. Aenean rhoncus vehicula hendrerit.

Appendix A

Appendix Title Here

Write your Appendix content here.