



UNIVERSITY OF CAPE TOWN

SUBJECT CODE

SUBJECT NAME

Title of Paper

Authors:

Author 1

Author 2

Author N

Student Numbers:

ATHNUM001

ATHNUM010

ATHNUM011

22 April 2018

I know the meaning of plagiarism and declare that all of the work in the dissertation, save for that which is properly acknowledged, is my own.

.....

Author 1
Author 2
Author N

Abstract

The abstract should be a one or two paragraph summary of your paper. It is meant to sell your paper to interested buyers.

Contents

List of Figures and Listings	iv
List of Tables	v
Nomenclature	vi
1 Acronyms	vi
2 Terminology	ix
1 Introduction	1
1.1 Drafting Markup	2
1.2 Template Commands	2
1.2.1 Requirement and Function Lists	2
2 Literature Review	4
3 Methodology	5
3.1 Hardware	5
3.2 Implementation	5
3.3 Experiment Procedure	6
4 Design	8
5 Experimentation	9
6 Results	10
6.1 Figures	10
6.2 Tables	10

6.3	Pictures and Screen-shots	12
6.4	Maths	13
7	Conclusion	15
	Bibliography	16
A	Item 1	18
B	Item 2	19

List of Figures and Listings

1	Introduction	1
2	Literature Review	4
3	Methodology	5
	Fig. 3.1 Test setup used to test the implementation [1].	6
	Listing 3.1 OpenCL kernel to perform matrix multiplication	6
4	Design	8
5	Experimentation	9
6	Results	10
	Fig. 6.1 The correlation coefficient as a function of sample count.	11
	Listing 6.1 Octave function to format a figure and save it to a high quality PDF graph	11
	Listing 6.2 Example of how to use the FormatFig function	11
	Fig. 6.2 Oscilloscope measurement showing physical line signals on both ends of a transmission line during master switch-over [1].	12
	Fig. 6.3 An example image with custom scaling	13
	Fig. 6.4 Comparison of various image format qualities	14
7	Conclusion	15

A	Item 1	18
B	Item 2	19

List of Tables

1	Introduction	1
	TABLE 1.1 Breakdown of sub-tests to be performed in the acceptance testing. . . .	3
2	Literature Review	4
3	Methodology	5
4	Design	8
5	Experimentation	9
6	Results	10
	TABLE 6.1 My Informative Table	12
7	Conclusion	15
A	Item 1	18
B	Item 2	19

Nomenclature

1 Acronyms

A	Amperes
AC	Alternating Current
ADC.....	Analogue to Digital Converter
API.....	Application Programmer's Interface
ARM	Advanced RISC Machine
ASIC	Application Specific Integrated Circuit
AXI	Advanced Extensible Interface
BAR.....	Base Address Register
BCD.....	Binary-Coded Decimal
Bd.....	Baud, in symbols per second
CFAR	Constant False Alarm Rate
CMOS.....	Complimentary Metal-Oxide Semiconductor
CPLD	Complex Programmable Logic Device
dBm.....	Deci-Bell, relative to 1 mW
DC	Direct Current
DDC	Digital Down Converter
DDS.....	Direct Digital Synthesis
DMA	Direct Memory Access
DSP	Digital Signal Processor (or processing)
EDA.....	Electronic Design Automation

FIFO First-in, First-out (queue)
 FIR.....Finite Impulse Response
 FMC FPGA Mezzanine Card
 FPGA Field Programmable Gate Array
 FSM.....Finite State Machine

 GUI Graphical User Interface

 HDL.....Hardware Description Language
 HPS Hard Processor System
 HSTL.....High Speed Transfer Logic

 I/O Inputs/Outputs
 I²C Inter-IC
 IC Integrated Circuit
 IDE Integrated Development Environment

 LE.....Logic Element
 LSb.....Least Significant Bit
 LSB Least Significant Byte
 LUT.....Look-Up Table
 LVCMOS.....Low Voltage Complementary Metal Oxide Semiconductor
 LVDS.....Low Voltage Differential Signalling
 LVPECL.....Low Voltage Positive Emitter Coupled Logic
 LVTTL Low Voltage Transistor-Transistor Logic

 MIMO Multiple Input Multiple Output
 MISO.....Master Input / Slave Output
 MOSI.....Master Output / Slave Input
 MSb Most Significant Bit
 MSB.....Most Significant Byte
 MSI Message Signalled Interrupt

 NCO Numerically Controlled Oscillator
 NTP.....Network Time Protocol

 PC Personal Computer

PCB.....Printed Circuit Board
 PCI.....Peripheral Component Interconnect
 PCIe.....PCI Express
 PLL.....Phase Locked Loop
 PPDS.....Point-to-Point Differential Signalling
 PRF.....Pulse Repetition Frequency
 PRI.....Pulse Repetition Interval
 PSU.....Power Supply Unit

 RADAR.....Radio-Assisted Direction and Ranging
 REST.....Representational State Transfer
 RF.....Radio Frequency
 RISC.....Reduced Instruction Set Computer
 RMS.....Root Mean Square
 RPM.....Revolutions per Minute
 RSDS.....Reduced Swing Differential Signalling

 SI.....Système International d’Unités
 SoC.....System On Chip
 SPI.....Serial Peripheral Interface
 SSTL.....Stub Series Terminated Logic

 TCP.....Transmission Control Protocol
 TTL.....Transistor-Transistor Logic

 UART.....Universal Asynchronous Receiver Transmitter
 UDP.....User Datagram Protocol
 UFM.....User Flash Memory
 URL.....Uniform Resource Locator

 V.....Voltage
 VHDL.....VLSI HDL
 VLSI.....Very Large-Scale Integration
 XML.....eXtensible Markup Language

2 Terminology

Developer	FPGA firmware developer, using any firmware development tool.
Device	The specific target FPGA.
Megafunction	A target-specific module, typically generated from within the vendor IDE, after the Altera nomenclature.
Module	Akin to a Verilog module; i.e. unit of digital circuit that has ports to the outside world and can exist at any level of the design hierarchy.
Object file	An intermediary file used in the ALCHA compilation process. It is the result of compiling a single translation unit and describes a collection of objects.
Peripheral	Any device, external to the FPGA, that interfaces directly with the FPGA.
Platform	The platform that the ALCHA compiler runs on, including operating system and computer hardware.
Target	The platform ALCHA is compiling to, including FPGA, PCB, peripherals and vendor IDE.
Translation unit	Akin to a C language translation unit, i.e. a collection of source files and headers that translate to a single object file
User	The developer who is using ALCHA to develop FPGA firmware.
Vendor	The FPGA manufacturer.

Chapter 1

Introduction

If you are new to L^AT_EX, I would suggest reading [2]. If you want to use Microsoft Word (or one of its many clones), you can download the official IEEE conference template from [3]. The TA and tutors can provide L^AT_EX support. Use Word at your own risk.

The introduction is where you set the scene. Here you reference other, related work, as well as a summary relating to how you improve upon said work [4]. In the sense of the practical reports, the introduction will summarise the experiment the practical is all about.

As a general rule of thumb, keep the introduction to the first column and don't put any sub-sections into it.

Remember that, for bibliography citations to work, you have to include running BibT_EX in the compile chain. My TeXstudio [5] compile chain for “Build & View” is

```
txs:///bibtex | txs:///pdflatex |  
txs:///bibtex | txs:///pdflatex |  
txs:///view-pdf-internal
```

1.1 Drafting Markup

When the template is in draft mode, you can use various helper macros, as illustrated below:

When compiled with `\Draftfalse`, the content of these macros are removed from the output, except something that needs to be rephrased.

1.2 Template Commands

1.2.1 Requirement and Function Lists

Here follows an example of requirement and function lists. You can refer to the items as req. R1 to req. R3; and function F1 to function F3.

R1. The first requirement described

R2. The second requirement described

R3. So on

R3.1 And they can be nested...

R3.1.1 And more nested...

R3.1.1.1 And even more nested.

F1. The first function described

F2. The second function described

F3. So on

After defining the requirements and functions, you can draw up a test matrix as shown in table 1.1. You can refer to a specific test like this: Test T2. Ideally you should only have one function and one requirement per test, otherwise the table becomes too wide. If you do want to add more to the list, reduce the `\TestTableDescriptionWidth` length.

TABLE 1.1

BREAKDOWN OF SUB-TESTS TO BE PERFORMED IN THE ACCEPTANCE TESTING.

Test	Description	Func	Req
T1	This is the test description of the first test. It can be a paragraph long.	F1	R1
T2	This is the test description of the second test. It can be a paragraph long.	F2	R2
T3	This is the test description of the last test. It can be a paragraph long.	F3	R3

Chapter 2

Literature Review

Chapter 3

Methodology

In this section you should describe the method of the experiment.

3.1 Hardware

Include detail such as the hardware used. It's generally a good idea to include a block diagram at this point, such as the one presented in Fig. 3.1. This figure was drawn in Inkscape [6]. When you want to import an Inkscape figure (SVG format) into L^AT_EX, simply save it to PDF (use the drawing extents as the media box area) and include the figure.

3.2 Implementation

Also mention the implementation source code:

```
# You can include inline Matlab / Octave code
x = linspace(0, 2*pi, 1000);
y = sin(x);
plot(x, y); grid on;
```

Or you could turn it into a float: see listing 3.1. Floats are tables, figures and listings that appear at a different place than in the source code. This template is set up to put floats at the top of the next column, as prescribed by the IEEE article specification.

Only list what is relevant. Don't give too much detail - just enough to show what you've done. This template supports the following languages:

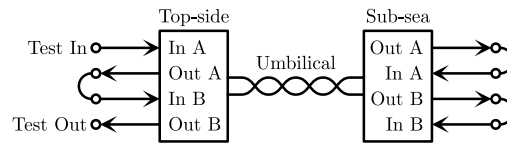


Fig. 3.1. Test setup used to test the implementation [1].

```

__kernel void Multiply(
    __global float* A, // Global input buffer
    __global float* B, // Global input buffer
    __global float* Y, // Global output buffer
    const int N // Global uniform
){
    const int i = get_global_id(0); // 1st dimension index
    const int j = get_global_id(1); // 2nd dimension index

    // Private variables
    int k;
    float f = 0.0;

    // Kernel body
    for(k = 0; k < N; k++) f += A[i*N + k] * B[k*N + j];
    Y[i*N + j] = f;
}

```

Listing 3.1. OpenCL kernel to perform matrix multiplication

- Matlab (Octave)
- GLSL
- OpenCL
- Verilog
- VHDL
- TCL
- Python
- C++ (use the name ‘Cpp’)

3.3 Experiment Procedure

Furthermore, include detail relating to the experiment itself: what did you do, in what order was this done, why was this done, etc. What are you trying to prove / disprove? You can include hypotheses, such as presented in Hypothesis H0 below.

Hypothesis H0: All scientific papers contain hypotheses. An hypothesis is generally not longer than a single paragraph, but the command does support multiple paragraphs if required.

Chapter 4

Design

Chapter 5

Experimentation

Chapter 6

Results

The results section is for presenting and discussing your findings. You can split it into subsections if the experiment has multiple sections or stages.

6.1 Figures

Include good quality graphs (see Fig. 6.1). These were produced by the Octave code presented in listings 6.1 and 6.2. You can play around with the `PaperSize` and `PaperPosition` variables to change the aspect ratio. An easy way to obtain more space on a paper is to use wide, flat figures, such as Fig. 6.2.

Always remember to include axes text, units and a meaningful caption in your graphs. When typing units, a μ sign has a tail! The letter “u” is not a valid unit prefix. When typing resistor values, use the Ω symbol.

6.2 Tables

Tables are often a convenient means by which to specify lists of parameters. An example table is presented in table 6.1. You can use `Tablesgenerator` to make your \LaTeX tables.

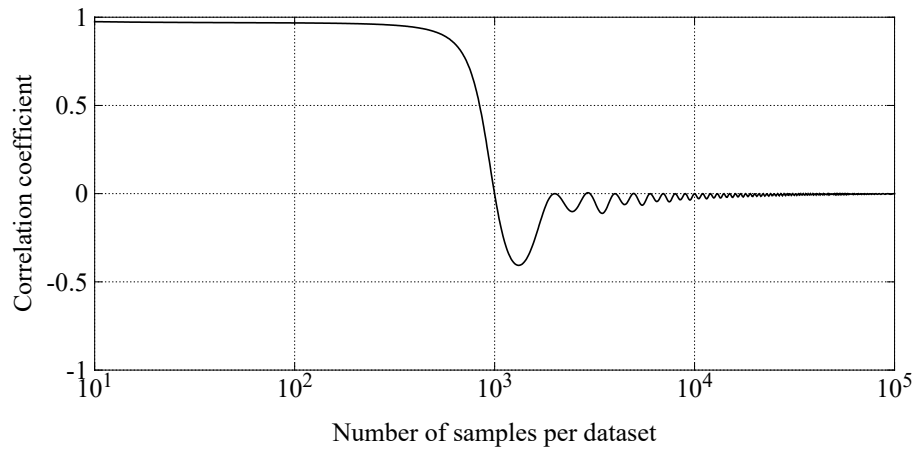


Fig. 6.1. The correlation coefficient as a function of sample count.

```
function FormatFig(X, Y, File);
    set(gcf, 'PaperUnits', 'inches');
    set(gcf, 'PaperOrientation', 'landscape');
    set(gcf, 'PaperSize', [8, 4]);
    set(gcf, 'PaperPosition', [0, 0, 8, 4]);

    set(gca, 'FontName', 'Times New Roman');
    set(gca, 'Position', [0.1 0.2 0.85 0.75]);

    xlabel(["\n" X]);
    ylabel([Y "\n\n"]);

    setenv("GSC", "GSC"); # Eliminates stupid warning
    print(...
        [File '.pdf'],...
        '-dpdf'...
    );
end
```

Listing 6.1. Octave function to format a figure and save it to a high quality PDF graph

```
figure; # Create a new figure
# Some code to calculate the various variables to plot...
plot(N, r, 'k', 'linewidth', 4); grid on; # Plot the data
xlim([0 360]); # Limit the x range
ylim([-1 1]); # Limit the y range
set(gca, 'xtick', [0 90 180 270 360]); # Set the x labels

FormatFig(... # Call the function with:
    'Phase shift [\circ]',... # The x title
    'Correlation coefficient',... # The y title
    ['r_vs_N;_f=' num2str(f) ';_P=' num2str(P)]... # Format the file name
);
close all; # Close all open figures
```

Listing 6.2. Example of how to use the FormatFig function

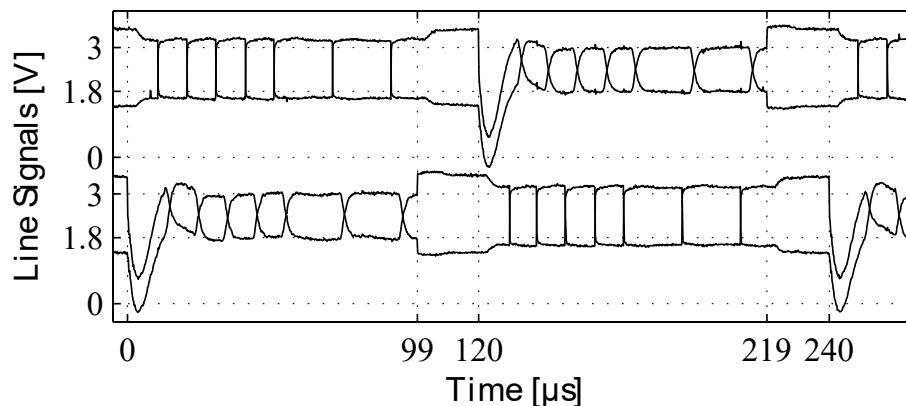


Fig. 6.2. Oscilloscope measurement showing physical line signals on both ends of a transmission line during master switch-over [1].

TABLE 6.1

MY INFORMATIVE TABLE

Heading 1	Heading 2	Heading 3
Data	123	321
Data	456	654
Data	789	987

6.3 Pictures and Screen-shots

When you include screen-shots, pdfL^AT_EX supports JPG and PNG file formats. PNG is preferred for screen-shots, as it is a loss-less format. JPG is preferred for photos, as it results in a smaller file size. It’s generally a good idea to resize photos (not screen-shots) to be no more that 300 dpi, in order to reduce file size. For 2-column article format papers, this translates to a maximum width of 1024. **Never change the aspect ratio of screen-shots and pictures!**

It is highly recommended to make use of the `\Figure` macro for figures. It puts all the formatting tweaks in one place, so that you don’t need to update all the individual figure inclusion points when you want to do a styling update. The file name is used for the L^AT_EX label, such as “Fig. 6.3”.

Make sure to always use the best quality image possible. Use JPEG for photos, PNG for screen-shots and PDF (scalable vector graphics) for everything else. JPEG is lossy, but good for photos, whereas PNG is lossless and good for images with large areas of solid colour, as



Fig. 6.3. An example image with custom scaling

can be seen in Fig. 6.4.

6.4 Maths

L^AT_EX has a very sophisticated maths rendering engine, as illustrated by equation 6.1. When talking about approximate answers, never use ± 54 V, as this implies “positive or negative 54 V”. Use ≈ 54 V or ~ 54 V instead.

$$y = \int_0^\infty e^{x^2} dx \tag{6.1}$$



Fig. 6.4. Comparison of various image format qualities

Chapter 7

Conclusion

The conclusion should provide a summary of your findings. Many people only read the introduction and conclusion of a paper. They sometimes scan the tables and figures. If the conclusion hints at interesting findings, only then will they bother to read the whole paper.

You can also include work that you intend to do in future, such as ideas for further improvements, or to make the solution more accessible to the general user-base, etc.

Publishers often charge “overlength article charges” [7], so keep within the page limit. In EEE4084F we will simulate overlength fees by means of a mark reduction at 10% per page. Late submissions will be charged at 10% per day, or part thereof.

Bibliography

- [1] J. Taylor and J. G. Hoole, “Robust Protocol for Sending Synchronisation Pulse and RS-232 Communication over Single Low Quality Twisted Pair Cable,” in *Proceeding of ICIT*. Taiwan: IEEE, Mar. 2016.
- [2] T. Oetiker, H. Partl, I. Hyna, and E. Schlegl, “The Not So Short Introduction to L^AT_EX 2_ε,” <https://tobi.oetiker.ch/lshort/lshort.pdf>, Jul. 2015, version 5.05.
- [3] “IEEE Conference Paper Templates,” http://www.ieee.org/conferences_events/conferences/publishing/templates.html.
- [4] A. Baboon, B. Charles, D. Ester, and F. Generalson, “An Amazing Title,” Their Not-so-awesome University, Technical Report, Apr. 1492.
- [5] B. van der Zander, J. Sundermeyer, and T. Hoffmann, “TeXstudio – A L^AT_EX Editor,” <https://www.texstudio.org/>.
- [6] “InkScape Website,” <http://www.inkscape.org/>.
- [7] “Voluntary Page and Overlength Article Charges,” <http://www.ieee.org/advertisement/2012vpcopc.pdf>, 2014.

Appendix A

Item 1

Appendix B

Item 2