Design Report – Team XX

ELETENG209: Analogue & Digital Design

*Design of a Smart Energy Monitor*

Student A

Student B

Student C

Student D

Department of Electrical, Computer, & Software Engineering

# Abstract

In a single short paragraph tell the summery of this report.

# Introduction

Use 1-2 short paragraphs to tell why you are undertaking this project (e.g. completed as part of ELECTENG 209 requirements), project goals, an overview of the system you are designing (you may use the system diagram from lectures with proper referencing as in Fig. 1) and what is in this report.

# Diagram Description automatically generated

Fig. 1: A conceptual system diagram [1]

# Literature Research

# In a table, list the key design specifications of two commercial designs that you think are good products (you may consider products from companies such as neurio, sense, efergy, elgato and the OWL).

Table I: A Comparison of existing energy meters

|  |  |  |
| --- | --- | --- |
| Parameter | Product XX | Product YY |
| Operating voltage range |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Design Specifications

In a table, list your design specifications – you can use the specifications provided in the course outline and add additional specifications as needed (e.g. if you are doing the smart energy challenge add relevant specifications).

In a short paragraph, compare the specifications of the two commercial designs (in previous section) to the specifications of your design (you will note that the commercial specifications are quite comprehensive).

# The Analogue Design

## The Schematic

Add your schematic design and make sure to indicate all the component values. Someone should be able to use this schematic to build your circuit without having to talk to your team. Include a short paragraph detailing special features of the design.

## The PCB

Add a screen capture of your PCB design – include both the 2D and 3D views. Include a short paragraph detailing special features of the design.

## Design Validation

Complete Table II (change table numbering as required) to highlight key design parameters and to show that your design functions as predicted by your calculations. For naming convention refer to <https://ee209-2020class.github.io/presentations/DigitalL4/presentation.html#5>. If you had developed hardware and tested it, add a column to include experimental data. Include a short paragraph detailing special features of the design.

Table II: Key design parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Calculated | Simulated |
| Gvs |  | |
| Vvs when VAC is 15.4 Vrms |  |  |
| Vvs when VAC is 12.6 Vrms |  |  |
| Gis |  | |
| Vis when IAC is 0.60 Arms |  |  |
| Vis when IAC is 0.16 Arms |  |  |
| Gvo |  | |
| Vvo when VAC is 15.4 Vrms |  |  |
| Vvo when VAC is 12.6 Vrms |  |  |
| Gio |  | |
| Vio when IAC is 0.60 Arms |  |  |
| Vio when IAC is 0.16 Arms |  |  |
| Gvf |  | |
| Vvf when VAC is 15.4 Vrms |  |  |
| Vvf when VAC is 12.6 Vrms |  |  |
| Gif |  | |
| Vif when IAC is 0.60 Arms |  |  |
| Vif when IAC is 0.16 Arms |  |  |
| ΔVin of 5V regulator |  |  |
| ΔV5V of 5V regulator |  |  |

# The Embedded Software Design

## Flowchart

Add a detailed flowchart to explain the functionality of your embedded software. Use appropriate symbols. The flowchart should clearly indicate how your embedded software work so someone else can use it to modify your software without having to talk to your team. You may divide the flowchart into a few flowcharts if required. Also, if needed include a short paragraph and/or pseudo code to provide details of any special features of the design.

## Peripheral Configurations

Add tables summarizing how each peripheral used is configured. As an example, Table III (change table numbering as required) show the UART configuration.

Table III: UART configuration

|  |  |
| --- | --- |
| Parameter | Configuration |
| Number of data bits | 8 |
| Number of stop bits | 1 |
| Baud rate | 9600 |
| Parity mode | None |
| Transmission mode | Simplex (transmit) |
| Transmit mode | Polling (UDRE0) |
| Transmit voltage | 5 V |

## Design Validation

Add information to validate the functionality of your embedded software modules. Show that they work as intended. For example, you can include the following information gathered from Proteus (also include experimental results if you have developed hardware),

* Oscilloscope capture of UART transmitting a few bytes of data to validate correct UART settings
* Screen capture to show messages printed on terminal
* Validate ADC by applying a DC voltage and graphing conversion results
* Validate timer(s) using oscilloscope capture of for example a pin toggle
* Validate ISR(s) using oscilloscope capture of for example a pin toggle
* Validate 7-segmet display operation using screen capture of the display and oscilloscope capture of data in to and out of the shift-register

Include text as needed to explain your results.

# Performance of the Energy Monitor

## Design Validation

Add 2 plots of ADC data points obtained for each voltage and current signals from Proteus to show your program captures data as intended. In these plots, indicate sketches of the voltage and current waveforms you were measuring as obtained from LTspice/Proteus. If you had developed hardware and tested it, include your ADC data points obtained experimentally for each voltage and current signals in 2 separate plots and compare these against voltage and current waveforms obtained from an oscilloscope. You can use Excel or MATLAB for plotting.

## The Accuracy

Add 3 plots to show accuracy of your voltage, current and power measurements. First plot should have x-axis as ideal values of voltage measured and y-axis as error in voltage measurement. Second plot should have x-axis as ideal values of current measured and y-axis as error in voltage measurement. Third plot should have x-axis as ideal values of power measured and y-axis as error in voltage measurement. Each plot should have at least 5 data points. You can use Excel or MATLAB for plotting.

# Conclusions

# In a single short paragraph tell the conclusions of this report.

# References

1. D. J. Thrimawithana, Class Lecture, Topic: "Analogue & Embedded Software Design: An Introduction to the Course" ELECTENG 209, Department of Electrical, Computer, and Software Engineering, The University of Auckland, Auckland, October 2020.

**Dep. of Electrical, Computer, & Software Engineering**

The University of Auckland   
20, Symonds Street

Auckland, New Zealand

**T** +64 9 373 7599

**W** auckland.ac.nz