**ENGN4200 – Individual Project**

The Australian National University

U6043905 – Mitchell Breitfuss

Primary Supervisor: Ed Gaykema

Secondary Supervisor: Igor Skryabin

**Mid-term Report**

Contents

[Summary 3](#_Toc10210246)

[Introduction 3](#_Toc10210247)

[Research Question 3](#_Toc10210248)

[Context 3](#_Toc10210249)

[Scope 3](#_Toc10210250)

[Deliverables 3](#_Toc10210251)

[Challenges 3](#_Toc10210252)

[Background 4](#_Toc10210253)

[Methodology 4](#_Toc10210254)

[Applied Methodology 5](#_Toc10210255)

[Prepare 5](#_Toc10210256)

[Plan 5](#_Toc10210257)

[Design 5](#_Toc10210258)

[Results 5](#_Toc10210259)

[WiFi Connectivity Testing 5](#_Toc10210260)

[Serial Connectivity 6](#_Toc10210261)

[Combined Functionality / Server Switching 6](#_Toc10210262)

[Pi Client 6](#_Toc10210263)

[Future Plans 7](#_Toc10210264)

[References 8](#_Toc10210265)

# Summary

My project’s development is proceeding as planned, with the monitoring system now starting to take shape. Large portions of the groundwork software development have been completed, with additional new features also being implemented to extend the scope of the project beyond the initial conceptualization. As the year continues, I will expand my field testing operations to improve system reliability, and to provide evidence that the system is capable of performing it’s duty.

# Introduction

## Research Question

*What is the best approach to create a modular and upgradable sensor networking system?*

## Context

I will join a team of students in working with EvoEnergy to aid in research related to their Hydrogen Test Facility (HTF). Their current aim is to determine the viability of using hydrogen as a fuel source to supplement other energy sources, either for home use or for electrical grid supply. At the HTF, many pieces of equipment have sensors to monitor system parameters. Although, they are currently separate, and someone must physically go to each sensor around the facility to take a manual reading. My role within this project is to design, develop, and implement a system that allows logging and remote monitoring for every sensor in the facility, allowing someone to not only view all relevant sensor information in one place, but to keep a logged history of sensor states.

## Scope

### Deliverables

* Determine how to extract data from sensors.
* Create network to allow centralized access to sensor data.
* Create GUI that showcases data with visualizations.
* Ensure that hardware and software are modular.
* Robust with high uptime.
* HTF system modelling at the discretion of supervisors.

## Challenges

* Each sensor has a different interface, requiring different approaches to data collection for each.
* Some sensors may not have a direct way of outputting data.
* Sensors are spatially separated, meaning hardwired networking is non-optimal.

# Background

­­As the world is becoming more connected with an exponentially increasing number of internet-connected devices going out into circulation, research into the Internet of Things (IoT) has exploded due to its potential applications in nearly every industry and field. The term IoT was coined in 1999 by MIT researcher Kevin Ashton, after his work into RFID chip use in supply-chain management revolutionized cataloging. While IoT style networking originated in industrial applications, it has caught on in the consumer space in recent years with a strong emergence of “smart home” appliances such as internet connected lightbulbs, heaters, and even fridges. This has caused a large increase of research into IoT networking, as consumer applications for technologies are often more profitable than industry applications. A paper written by J. Parmar & A. Desai in 2016 titled *IoT: Networking Technologies and Research Challenges* is evidence of this research focus, outlining some key areas within IoT that can influence the effectiveness of an applied solution, such as wireless protocols, power requirements and signal range.

As our finite reserves of natural gas are depleted, we must search for renewable alternative fuel sources to help us transition into a 100% renewable energy-based society. EvoEnergy’s research into hydrogen supplementation of existing natural gas networks is on the forefront of a newly expanding research interest. The premise of using hydrogen to supplement natural gas is not new, with a notable [paper](https://www.sciencedirect.com/science/article/pii/0360319983901817) written by B. Nagalingam Et al in 1983 showing some early research into the subject. Although, due to improvements in technology across the board, we are now better suited than ever to begin implementing these ideas within the next 20-30 years.

This project combines the emerging field of embedded computing with the long-standing principles of renewable energy research.

# Methodology

When approaching my research question, it is very tempting to dive in and code a solution that will quickly solve some issues but will require me to revise parts of my project to ensure that *all* of my deliverables are met at the completion of this project. Because of this, I have decided to use an established network design methodology to aid in the research, design and implementation of my IoT network. I have adopted CISCO’s PPDIOO methodology, (“The Methodology”), as my research methodology for this project, as it provides a comprehensive framework to allow effective design of a network. By utilizing the [lifecycle](https://www.ccexpert.us/network-design/using-a-modular-approach-to-network-design.html) approach to network design, we can ensure that my project requirements of modularity & robustness are met upon completion. Information and a description of the methodology can be found in a linked reference.

## Applied Methodology

When working on this project I have taken the methodology into account at every step, with an emphasis on record keeping and version control. Examples of this application can be seen below, and more in-depth descriptions of my completed work can be found in the *Results* section of this report.

### Prepare

* + Conducted preliminary research into possible solutions by writing a preliminary *Sensor Networking Approach* report. (See appendix)

### Plan

* + Began by organizing my development environment, and obtained materials required for development. (Arduino and Raspberry Pi)
  + Extensive research into interfacing requirements of each sensor, determined ideal approach for integrating them to the network.

### Design

* + Utilized a modular, object-orientated programming paradigm to ensure a robust network architecture, reducing the work required to add features.

# Results

Significant progress has been made with regards to the Electrolyzer and Solar inverter interfaces. Through code experiments I have been able to map out the problem scope in more detail, breaking down the research project into smaller, more easily achieved chunks. Referenced code can be found on my [GitHub repository](https://github.com/mitchbreitfuss/Honours), with complete file history. Some files on the repository are depreciated and are only kept as an archive.

## WiFi Connectivity Testing

At the beginning of the project after purchasing a WiFi enabled Arduino for prototyping, I began work by first ensuring that I was able to connect to a WiFi hotspot and communicate with a server. This experimentation was achieved through two Arduino sketches found on the repository: *wifitest.ino,* & *wifitest2.ino.* The first test file was a *“Hello World”* of sorts, enabling me to understand how to connect to a network using the Arduino. The second file was an extension of this idea, by also testing communications with a server on the network. This was done by running a local server at home hosting a faux equipment interface. (A simple HTML webpage) This test file also allowed me to begin learning to apply the HTTP protocol, such as establishing connections with a server and requesting files. These tests were designed to imitate the behavior of the Electrolyzer and Inverter, as they are accessed through a local web interface.

## Serial Connectivity

The next stages in my experimentation revolved around the communication of data from the Arduino to the Raspberry Pi. This was done with the standard UART serial interface protocol, a well-supported feature of both devices. I experimented with various network designs, including one-way peer-to-peer communication between an Arduino and the Raspberry Pi, (*send.ino, receive.ino),* and a Master-Slave architecture to allow multiple Arduinos to communicate with a Raspberry Pi simultaneously. Tests consisted of developing a command system, which allows devices to transfer data and control the operation of other devices on the network.

## Combined Functionality / Server Switching

My previous experiments have allowed me to understand how to facilitate communications between devices and servers, signaling the completion of all ground work and basic functionality. The next logical step was to combine all the features I had previously developed in separate unit tests into a single software package. This was undertaken first in the */combined/app.ino* test file with success, where I was able to incorporate all minor features I had already developed into one file. This includes, but is not limited to: WiFi connectivity, server communication, serial communication, and system control.

This was extended in *switching.ino* by implementing communications with the real Electrolyzer and Inverter, allowing successful data acquisition tests at the Hydrogen Test Facility. This process was closely tie to the development of the newer parts of the Pi Client software.

## Pi Client

The Raspberry Pi client software (*piclient.py)* has been continuously developed alongside all Arduino software since the beginning of the project. Due to Python’s flexibility, small unit tests were not necessary such as with the Arduino, as features can be easily added or changed without a large amount of work. The Raspberry Pi is responsible for several critical aspects of the project, including logging, data access, system configuration and graphical user interactivity.

In its current state, the Pi Client software has a basic GUI, which allows a user to manually collect data from the HTF equipment. A command line interface is also present, which allows configuration of both the Pi and the Arduino on the fly, so code rewrites are not always necessary. It also supports data logging, with automatic fetching and parsing of sensor data.

# Future Plans

There are still several obstacles to overcome to ensure that the systems’ design meets all of the project deliverables by the due date at the end of the year. These can be narrowed down to 3 deliverables: GUI interactivity improvements, ensuring robust system performance, and system modularity.

The GUI as it currently stands is quite basic and does not afford the user many luxuries. The GUI will have to be dramatically improved to provide a good UX, and many features will have to be devised, researched, developed and implemented. Possible directions for feature development include data visualization, such as graphs and diagrams, remote access through a web interface, and incorporation with the CSIRO developed simulation and modelling tools that EvoEnergy are already using.

As I have not done much testing into ensuring robust operation of the network, I will have to undertake extensive long-term field tests of data collection at the HTF to ensure that there are no bugs in the software that I otherwise would not know about. This can be extended to include testing the long-term viability of hosting a remote web interface on a Raspberry Pi. Going forward, by ensuring that all code I write is of high quality from now until the end of the project, I will save time by reducing the amount of debugging required as the system becomes more complex, and thus more prone to error.

Finally, I intend to ensure that the system design is sufficiently modular and well-documented, such that when my time working on the project is up, it is in a state that is prepared for handover to EvoEnergy, as they may choose to continue improving the network by extending its compatibility with more equipment sensors. This is perhaps one of the most important deliverables of the project, as a poorly documented project with no extensibility may provide little value.

References

CISCO Network Design Methodology: <http://www.ciscopress.com/articles/article.asp?p=1608131&seqNum=3>

CISCO Modular Network Design:

<https://www.ccexpert.us/network-design/using-a-modular-approach-to-network-design.html>

Project GitHub:

<https://github.com/mitchbreitfuss/Honours>

International Journal of Hydrogen Energy:

<https://www.sciencedirect.com/science/article/pii/0360319983901817>

J. Parmar & A. Desai [2016]:

*IoT: Networking Technologies and Research Challenges*