



UNIVERSITY OF YORK

SECOND YEAR PROJECT

---

# Embedded Systems Project Written Report

---

*Author:*  
Douglas PARSONS

*Supervisor:*  
Dr. M.J. FREEMAN

April 1, 2015

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Summary of project work . . . . .	2
1.2	Experiences as a team . . . . .	2
1.3	Expectations and actual outcomes . . . . .	3
1.4	What is in the report . . . . .	3
<b>2</b>	<b>Technical Description of Problem</b>	<b>4</b>
2.1	Description of Problem and Requirements . . . . .	4
2.2	Discussion of Technical Aspects and Challenges . . . . .	5
<b>3</b>	<b>Description and Discussion of Team-based Solution</b>	<b>7</b>
3.1	Description of the Team Solution . . . . .	7
3.2	How the problem was broken down for individual members . . .	8
3.3	Technical innovation and implementation of each member . . . .	8
<b>4</b>	<b>Evaluation and Testing of Team-based Solution</b>	<b>9</b>
4.1	Description of the Team's Testing Strategy . . . . .	9
4.2	Results of Testing Strategy and How Well This Met the Require- ments . . . . .	9
4.3	Section relating to a professional/social/ethical/environmental as- pect of our solution . . . . .	9
<b>5</b>	<b>Description, Discussion, Testing and Evaluation of Individual Component</b>	<b>10</b>
5.1	Description of Individual Component Solution . . . . .	10
5.2	Discussion of Technical Innovation and Implementation . . . . .	10
5.3	Testing Strategy, and the Results . . . . .	10
<b>6</b>	<b>Summary and Conclusions</b>	<b>11</b>
6.1	Reflective Summary of Team Work . . . . .	11
6.2	Reflective Summary of Individual Work . . . . .	11
6.3	What went well . . . . .	11
6.4	What went poorly . . . . .	11
6.5	What could have been improved . . . . .	11
6.6	Lessons Learned . . . . .	11
<b>A</b>	<b>Specified Documents</b>	<b>12</b>
A.1	Meeting minutes . . . . .	12
A.2	Evidence of Preparation . . . . .	12

# Chapter 1

## Introduction

### 1.1 Summary of project work

The Second Year Embedded Systems Project was a ten week project, undertaken for the duration of Spring Term 2014-2015. The project was a primarily a group task, completed in groups consisting of four members. However, there was also the capability for individual extensions to the main body of the project, allowing each member to showcase competency and creativity.

The group project involved programming an ARM LPC1768 'MBED' microcontroller, situated on top of a board of peripheral accessories, in order to generate and play music on a user selectable channel corresponding to data being sent down a Controlled Area Network bus (CAN bus). The individual component was not specified, and was instead left up to each individual to decide on an extension project, research, and implement it.

The groups solution was very complete, not only matching the full spec, but also extending the implementation to a high degree, producing a user friendly, good sounding end product. My individual extension provided a more user friendly interface than the rudimentary keypad attached to the MBED board through the implementation of a shell type interface, allowing the user to input commands, and adjust settings though the use of the computer keyboard and monitor.

### 1.2 Experiences as a team

The capability to work successfully as a group was imperative to the triumph of the project. Throughout the ten week period, we held regular group meetings, allowing us to keep track of each person's development, and had strong levels of feedback on each individuals contributions to the group solution through regular code reviewing sessions, this enabled us to remain productive continually, and allowed continual flexibility to match each individuals talents . Each member of the group had a strong drive to provide a high quality finished product, and eagerness to develop their personal skills. This resulted in a large amount of time being spent in the labs outside of the scheduled practicals, enabling us as a group to progress much further than we otherwise might have done. Overall I feel we worked very well together, and the exercise served as a valuable insight into a fluent working environment as a team.

## 1.3 Expectations and actual outcomes

At the beginning of the ten week period it was very difficult to predict what would be possible to achieve. This difficulty arises from inexperience with programming embedded systems, and an uncertainty regarding the limitations of the LPC1768 board the project used. However, it was expected that during the course of the ten weeks my group would be capable of meeting the specifications of the project, as well as implementing our individual solutions. Certain aspects of the project took longer than initial expectations would have suggested, for example, it took a significant portion of time to implement the various different user control methods present in our final solution. Overall, our time management skills worked out very well though, and our completed solution lives up to the expectations of the group.

## 1.4 What is in the report

This report sets out to provide a description of our team solution to the set assignment, as well as highlighting areas of individual contribution throughout. The report will begin with a technical description of the set problem, including a discussion of requirements and technical challenges that may be faced in the meeting of these requirements. Following a description of the problem, the group solution will be discussed. The discussion will highlight how each section of the requirements has been met by the implemented solution, as well as detailing how the problem has been broken down for each individual member, presenting each members implementation and technical innovation. After a discussion of the group's solution, the appropriate testing strategies and methods that were used throughout the project will be examined in detail, providing feedback on how they have proved useful, and incorporating a discussion on a professional/social/ethical/environmental aspect of the solution. The report will then focus critically on my individual implementation, detailing the technical innovation, implementation, and testing undergone for my contribution. Finally the report will conclude with a reflective summary of work undertaken, considering which aspects of the project went smoothly, which areas did not go as smoothly, how this might be improved in future, and what lessons can be taken away from the completion of the project.

## Chapter 2

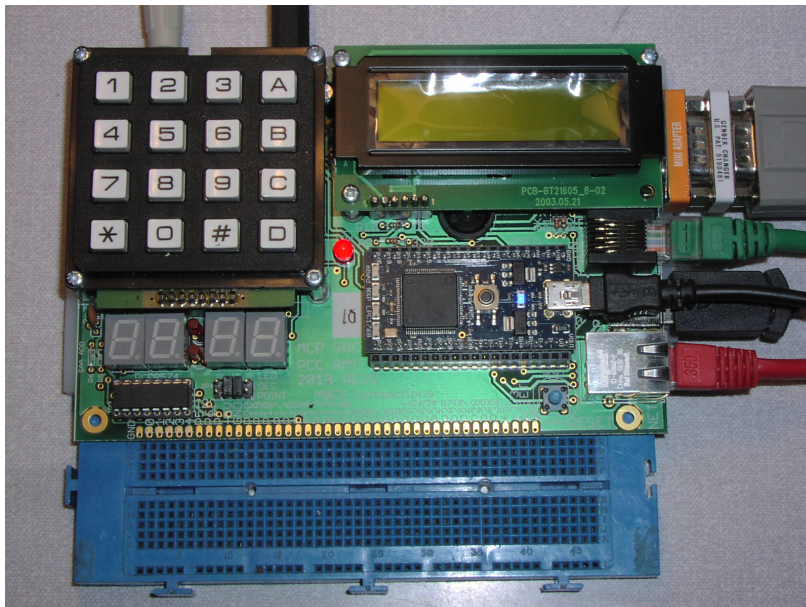
# Technical Description of Problem

### 2.1 Description of Problem and Requirements

The Embedded Systems Project consisted of two main sections: a group solution to a predetermined problem, and an individual extension to the group's work, left up to each person to determine. Both sections of the project involved embedded systems software development in the C programming language.

The target platform of the development is an ARM-based microcontroller, situated on a board of peripheral accessories. The microcontroller consists of an interface board coupled with the ARM cortex-M3 based LPC1768, providing easy interaction via USB cable to transfer binaries, and simplifying the process of serial communication [how-mbed-works].

The MBED board is sat on a board of peripheral accessories, as pictured below.



The board contains multiple accessories that the MBED board can interface with in various ways to fulfil the requirements, and allowing many options for

individual extensions. The accessories on the board include: a 16 key keyboard, a 16 by 2 line LCD, four seven segment displays, a micro SD slot, a USB host connection, CAN bus connector, audio in and out jacks, a TCP/IP connection, as well as off board connections to users hardware.

The group task consisted of multiple different challenges, with the concept that the MBED board would continually receive CAN packets, filter out those that do not match a user selectable id, decode the CAN packets, and play notes corresponding to the frequency, volume, and duration that were specified from the CAN transmissions. In addition the user must be able to view useful information, select the channel id, and adjust the volume.

In addition there was an individual component. This did not have to match any specification, and was instead left up to each person to decide on their own extension to the group project. For my individual extension, I implemented a Linux shell style user interface. This allows user to interact with the device by typing commands into a terminal on their computer, effectively adding an additional element of user control that is more intuitive and responsive than using the 16 digit keypad from the MBED board.

For our group solution, the specification was broken down into three separate areas, allowing us to focus our expertise and interests on what suited us best. The breakdown of the three sections was as follows: The receiving and decoding of CAN packets (CAN bus), generation of audio samples corresponding to frequency, duration and volume (Audio), and the ability for the user to interact with the device in order to adjust settings (User Interface).

## 2.2 Discussion of Technical Aspects and Challenges

As previously mentioned, my group deconstructed the specification into three main areas of technical challenge. The first of these challenging areas regarded the receiving of data from the CAN bus, and the decoding of the packets received to extract their content. As the CAN packets are not arriving at regularly timed intervals, the solution to this problem is required to be interrupt based. However, due to the nature of the CAN packets, a very large number can arrive in a comparatively short space of time. This therefore requires a highly optimised solution, in order to prevent packets either being missed, or race conditions within the interrupt handler from occurring.

The second technical challenge of the project is the generation of audio tones corresponding to frequency, duration and volume levels as specified by the CAN packets. The CAN packets do not contain information detailing what each note should sound like. As a result, all musical notes need to be generated from first principals, for example using a look up table detailing values for a note at each moment in time. In combination, to make the notes sound more realistic, each note must go through at least three stages. The first of these stages defines the

initial playing of a note, plucking a string or pressing a key. This is known as 'attack'. The second stage corresponds to the continual noise made from the note until it is stopped and is known as 'sustain'. The final stage details the note as it dies down and stops and is known as 'release'. This presents additional problems as each note requires multiple states, and needs to be tracked in its progress through each. Furthermore, it may be an additional desire to play more than one note at any one moment in time. This requires synthesis of multiple different notes in order to produce an appropriate output. The mathematics of this can be expensive to compute, and furthermore, an increase in the number of notes can drastically increase the processing time. A delay in the output of the notes can lead to a strange sounding finished product, so the efficiency of any synthesis calculations are paramount to the end product.

The third technical challenge that can be presented is the user's control over the device. The specification determines that the user must be able to control the volume of the output, alter which channel of notes is being played, and view useful information on the MBED board. This requires a level of interaction, most likely via the 16 digit keypad on the device, however this could be via alternate input methods, such as a computer keyboard. The user's interaction with either device once again may not be continual, but may either be polled, at the cost of system resources, or interrupt based. Further challenges are presented as any possible user input must be stored on the device, parsed on command, and then the desired command must be executed without any significant effect on the audio output. Furthermore, the challenge is made increasingly difficult by the required ability to interface with the entire project: the audio synthesis code and the CAN bus code must be fully understood in order to be manipulated through the user's input.

## Chapter 3

# Description and Discussion of Team-based Solution

### 3.1 Description of the Team Solution

The end product of our team solution was a successful working product. Through the incorporation of a Linux style shell into the group solution, we managed to produce a highly configurable single solution that managed to satisfy each section of specification. In addition the group solution contained two custom built speakers, capable of being powered by the 5V voltage rails from the peripherals board. The solution had a strong user interface, allowing the user to type in commands through the computer keyboard to adjust settings and select modes, set an id to filter channels out, and display information, or custom text on the mbed board. In addition it was also possible to adjust volume, channel and display information on the mbed board using the 16 digit keypad. This high level of user interfacing allowed settings to be flexibly turned on and off, and avoided the need to be continually re-installing binaries to the device. By typing in the command "listen", the mbed would then initialise the CAN bus and begin receiving packets. These could be printed out to the screen by issuing a further command, or they could be used to generate audio tones. It is also possible to filter out data according to a preset id by issuing a command, or using the keypad. For each data element received, audio tones can be generated. The solution supports a full synthesis, allowing multiple notes to be played on each channel, and multiple channels to play notes simultaneously. In addition the audio output also implemented an 'attack-sustain-release' model, creating a more realistic sounding solution. The audio code used direct memory access (DMA) to provide a reduced processing cost in outputting the audio samples. The user could also choose to display useful information such as the current song name, the volume, or the channel names on the computer screen or on the MBED board in a variety of ways. For example they could be statically printed on the MBED board, or scrolled across the LCD display, or even printed out to the terminal on the computer.



### 3.2 How the problem was broken down for individual members

The breakdown of our solution into components individual members could work on was a great strength of our team. Each person worked on an aspect of the project that they found interesting, and throughout the project we had regular feedback on all our work in order to make sure we were meeting our specifications and that nobody was out of their depth. Below I have divided up the work that each individual completed during the duration of the project.

#### Mingzhao Zhao

Mingzhao's work was primarily focused on handling the 16 digit keypad on the mbed board, this board was configured to be interrupt based as opposed to being polled. This caused additional problems with bounce on the switches, this caused multiple interrupts and race conditions to occur within the interrupt handler. Due to the row/column method in which the key pad is scanned, the number of 'bounces' was reduced significantly by only permitting certain input values to register, and then reduced further by only allowing a keypress once every half a second to do anything. This method of debounce was achieved using a SysTick to act as a timer.

#### Shivam Mistry

Myself (Douglas Parsons)

Liam Fraser

### 3.3 Technical innovation and implementation of each member

Mingzhao Zhao

Shivam Mistry

Myself (Douglas Parsons)

Liam Fraser

## Chapter 4

# Evaluation and Testing of Team-based Solution

- 4.1 Description of the Team's Testing Strategy
- 4.2 Results of Testing Strategy and How Well This Met the Requirements
- 4.3 Section relating to a professional/social/ethical/environmental aspect of our solution

## Chapter 5

# Description, Discussion, Testing and Evaluation of Individual Component

- 5.1 Description of Individual Component Solution
- 5.2 Discussion of Technical Innovation and Implementation
- 5.3 Testing Strategy, and the Results

## Chapter 6

# Summary and Conclusions

6.1 Reflective Summary of Team Work

6.2 Reflective Summary of Individual Work

6.3 What went well

6.4 What went poorly

6.5 What could have been improved

6.6 Lessons Learned

## Appendix A

# Specified Documents

**A.1 Meeting minutes**

**A.2 Evidence of Preparation**