Department of Electronic Engineering

BEng Project Report 2017/2018

Student Name: Chris Taylor

Project Title: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users

Supervisors: Prof A E Ward & Dr Y Zakharov

Department of Electronic Engineering
University of York
Heslington
York
YO10 5DD

Abstract

This report documents the research, development and testing of an accessible software package, adding voice recognition and audible feedback to electronic lab equipment for visually and physically impaired users. It is intended to remove any detriment that the user is experiencing due to their disability and to increase their efficiency using the lab equipment. The system uses Natural Language Processing to **parse** the users spoken command and send the appropriate command to the equipment. The report includes a literature review exploring existing solutions and products, as well as the guidelines needed to follow when designing for visual and physical impairments. It also describes two studies conducted amongst existing users of the lab equipment to gather requirements and test cases to aid in the development of the software.

Contents

List of Figures

List of Tables

Listings

Acknowledgements

This project idea was developed as one of the project proposals provided under the EU Lifelong Learning Programme Strategic ALignment of Electrical and Information Engineering in European Higher Education Institutions (SALEIE)[1]. Project Reference No. 527877-LLP-1-2012-1-UK-ERASMUS-ENW Project funded by the EU Lifelong Learning Programme.

The author would like to thank Tony Ward for their constant feedback and support throughout the entire project. Also to Kit, Carole, Richard, and Sophie-Jayne; without their support, this project could not have been completed.

The literature review and initial work breakdown of this report are expanded from the initial report for this project, submitted to the Department of Electronic Engineering at the University of York in January 2018.

[1] http://www.saleie.co.uk

Statement of Ethics

Following consideration of the University code of practice and principles for good ethical governance, two fast-track ethical approval forms were completed for the research of this project. The completed forms, signed by the author (Chris Taylor) and first supervisor (Tony Ward), are available in Appendix D.1 and E.1. Participants in the user testing were asked to sign a consent form at the beginning of the test, blank copies of these are available in Appendix D.2, D.3 & E.2.

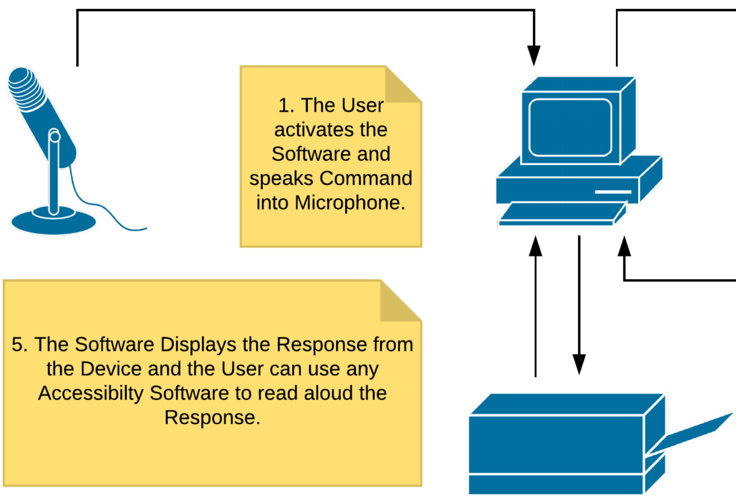
Introduction

Figure 1: Oscilloscope with ruler for scale

Disabled students face particular difficulties with electronic lab equipment, such as hard to read displays, small dials, and fiddly buttons. Figure 1 shows some of the buttons being less than 1 cm across on a typical oscilloscope. While there are already solutions to this problem, discussed in detail in Section 1.3, none of them adequately solve all aspects of the difficulties faced.

A possible solution to these issues is to create an accessible software suite on a PC that interfaces with the lab equipment, featuring intuitive controls, voice control, and clear visual and audible feedback to the user. The scope of this project will be software based, requiring no specific hardware other than a microphone and a headset, or speaker, for the user. This will allow the user to use any existing accessibility hardware, or software, they would typically use. Figure 2 shows a simple block diagram of the proposed system.

Figure 2: Proposed system overview



This report explores various disabilities that would cause the user to have difficulty using the lab test equipment. It discusses the appropriate design guidelines for the development of an accessible software package, as well as testing procedures to evaluate the final product. Following this, it details the development process through to the testing phase and discusses future work for the product.

1 Literature Review

To develop a system that is both unobtrusive and intuitive to use, specific design guidelines and methodologies must be followed. Existing designs must be considered and evaluated to create a system that succeeds where they fail. This chapter looks at previous projects and studies conducted, as well as consumer products already available. It also examines regulatory standards of design guidelines for impaired users.

Contents

1.1 Disabilities & Impairments

This section discusses various disabilities and impairments that could cause a user to experience difficulties using lab test equipment.

Using the 'Exclusion Calculator' [1], an estimate of the proportion of the British population who would be unable to use a typical piece of lab test equipment was calculated. The data used for the assessment can be found in Appendix F.2.1. The result was a total exclusion of 5.25% for all genders, aged 16 to 64.

This metric was recalculated, assuming the lab equipment was voice-controlled, and the user could use any accessibility tools on their PC. The data uses for this assessment can be found in Appendix F.2.2. As a result the exclusion lowered to 1.4%. This is a reduction of 73%, which shows that a voice-controlled software package would drastically increase the accessibility of lab test equipment.

1.1.1 Visual

There are many conditions that cause a person to lose clear vision. It is reported that 1 in 30 people in the UK live with sight loss [2]. Figure 1.1 shows a few examples of what it would be like to view a typical piece of lab equipment with different visual impairments.

Loss of vision can occur at any point in a persons life; therefore a long time user of lab test equipment could incur a sudden loss of vision and be incapable of carrying on with vital work. For these people, a solution should not require them to 'relearn' how to use the equipment.

Certified Blindness

A Certificate of Vision Impairment (CVI) means the user has some degree of sight loss [3]. This may be in one or both eyes and can range from a slight impairment (partially sighted) to severe impairment (blindness). For these users, reading small writing on dials and displays is very difficult. Figure 1.1d shows a simulation of the oscilloscope seen on Page 14, as seen by a person with limited visual acuity.

1.1.2 Physical

As well as visual impairments, physical impairments are also varied and can affect anyone. They can range from life-long ailments, such as Cerebral Palsy, to short-term impairments, such as a broken arm or sprained wrist.

Dyspraxia

It is reported that 16% of people show some symptoms of Dyspraxia [4]. Symptoms in adults include "Poor fine motor co-ordination skills" and "Poor relocating. Cannot look quickly and effectively from one object to another." [5]. For these users, not needing to move their focus away from their work to adjust the lab equipment will be a great help.

Figure 1.1: Examples of visual impairments. Created using the 'Impairment Simulator Software' [6].

(a) Glaucoma

(b) Diabetic Retinopathy

(c) Macular Degeneration

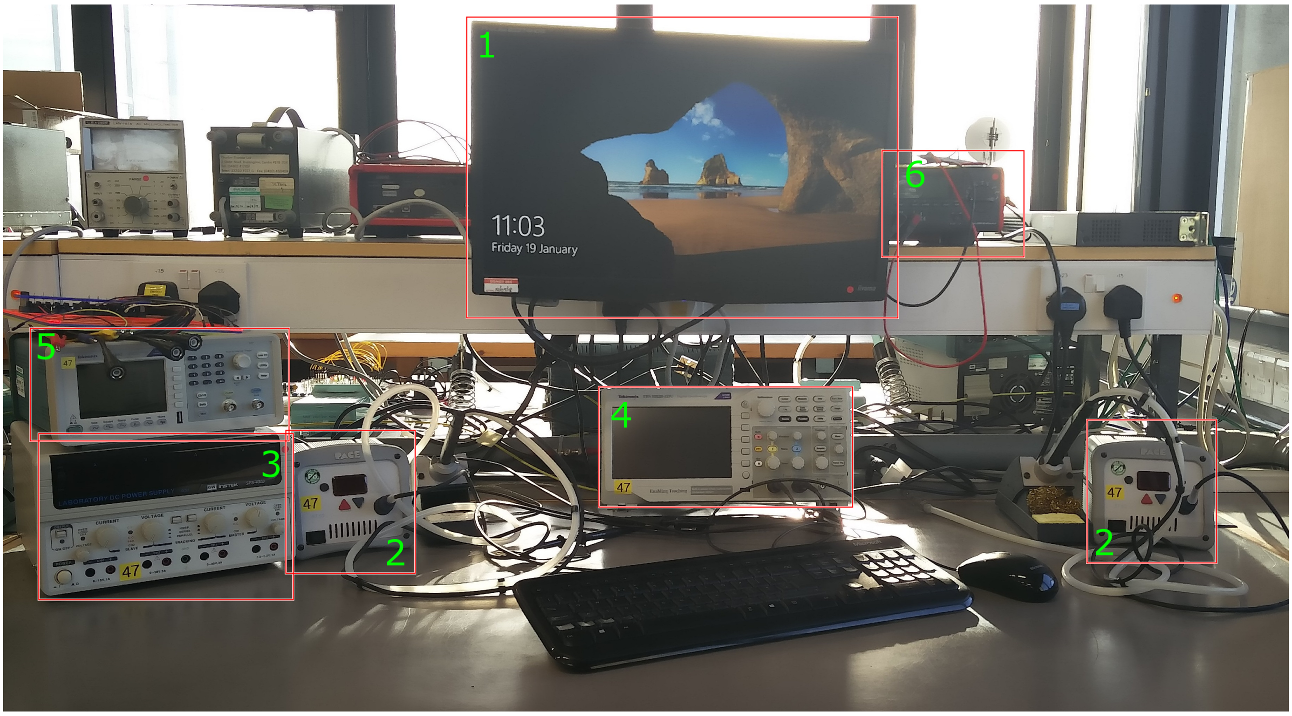
(d) Visual Acuity

1.2 Lab Equipment

This project will focus on the current equipment and resources in use at the Department of Electronic Engineering at the University of York. The typical layout of an undergraduate bench[2], shown in Figure 1.2, consists of:

1. A PC running Windows 10;
2. Two Soldering Irons;
3. A Quad-channel DC Power Supply;
4. A Tektronix TBS 1052B-EDU Oscilloscope;
5. A Tektronix AFG1062 Arbitrary Function Generator;
6. A UNI-T UT804 Digital Multimeter (DMM).

Figure 1.2: Typical bench layout



The soldering iron and power supply have no feasible connections to the PC and therefore will not be considered for this project. The following sections detail the specifications for the oscilloscope, function generator, and DMM that will be considered for this project.

[2] Correct at time of publication.

1.2.1 Oscilloscope

Oscilloscopes allow users to visualise, record, and analyse waveforms over time. They are highly configurable and, even for a fully sighted user, can sometimes be overwhelming with the number of options and settings available. For the TBS 1052B-EDU, there is a comprehensive programming manual available from the manufacturer [7].

The TBS 1052B-EDU can be connected to a PC via USB or GPIB, using an adapter. The communication protocols adhere to Standard Commands for Programmable Instruments (SCPI) [8]. All data is transmitted using ASCII characters. Commands are structured as a header containing a ‘`:`’ delimited **mnemonic** which relates to the action, followed by a space and then any required data and delimited by a ‘`LF`’. For example, the command to set the source for the trigger to channel two would be:

Figure 1.3: TBS 1052B-EDU oscilloscope


```
TRIGger:MAIn:PULse:SOUrce CH2
```

To retrieve data from the oscilloscope the same mnemonic can be used with the addition of ‘`?`’ at the end. For example, to retrieve the current trigger source the command would be:

```
TRIGger:MAIn:PULse?
```

The oscilloscope would then reply with something similar to:

```
TRIGGER:MAIN:PULSE:SOURCE CH1;WIDTH:POLARITY POSITIVE;WHEN EQUAL;WIDTH 1.0E-3
```

In all three examples, the lowercase characters can be omitted to save bandwidth. There are three numerical data types that the oscilloscope will accept:

- Signed integers
- Floating points without an exponent
- Floating points with an exponent

The oscilloscope will automatically correct any numerical values that are out of range by setting them to the limits. It will also round any values that have too many decimal points. The proposed system will need to make sure that numerical values are parsed correctly from the user input and sent to the oscilloscope in the correct format.

1.2.2 Function Generator

Function generators allow the user to create reliable, arbitrary waveforms to allow analysis of circuitry. The AFG1062 has a similar syntax for commands to the TBS 1052B-EDU, also adhering to SCPI. The difference is that the AFG1062 can only be connected via a Universal Serial Bus (USB) cable to the PC [9].

Figure 1.4: AFG1062 function generator

1.2.3 Digital Multimeter

DMM's allow the measurement of various quantities: voltage, current, resistance, etc. The display is large and backlit, but the buttons are small, and the text on them is hard to read, making it still hard to use for an impaired user. The UT804 features both a USB and an RS232 connection to a PC [10]. Appendix A.2 shows a piece of software created by UNI-T that allows the DMM to send measurement data to the PC to be recorded. A software manual that documents the interface between the DMM and the PC has not yet been found. UNI-T has been contacted with regards to creating an interface for this project.

Figure 1.5: UT804 DMM

1.3 Existing Solutions

This section explores existing solutions and alternatives to the usual interfaces of both specialist and consumer products.

1.3.1 Commercial Products

Several software solutions exist that allow voice control of specific pieces of lab equipment. One example is a software package created by Tektronix which has since been discontinued but featured user-defined macros, multi-lingual support, and audible feedback of recordings and measurements [11]. The webpage for the product lists the commands it supported, and these can be found summarised in Appendix A.1. This list will guide the initial development of this project, prioritising the commands Tektronix supported. Implementing these commands early in the life of the project will make it superior to the product released by Tektronix.

This software seems to have been adequate at controlling one specific brand of oscilloscope, but would not allow control over several different pieces of lab equipment at once. An oscilloscope sold by Keysight Technologies features “Voice Control option for hands free operation” but has also since been discontinued by the manufacturer [12].

Nachtmann created a voice-controlled oscilloscope by obtaining custom firmware from Tektronix to allow Bluetooth connectivity and then creating an Android application to control the oscilloscope remotely [13].

1.3.2 Consumer Products

First patented in 1993, ‘The Clapper’ [14], was a device designed for controlling electronic items by clapping. This made it possible for users to activate devices without the need to be situated nearby, or by possessing a remote control. In an era before voice recognition, this was an incredibly useful tool; although it was reportedly too sensitive, being triggered by the sound of footsteps on stairs or door being closed [15]. It does raise an interesting point; the laboratories are somewhat loud at times, and generally, have many people talking in them at once. The system will need to be able to cope with a lot of background noise and interference.

In recent times, digital personal assistants have become quite popular; with Alexa, running on Echo devices, by Amazon [16], Siri by Apple [17], and Google Home [18]. All of these devices have made consumers used to using voice controlled devices. It was reported in 2017 of “14% of UK households owning one or more devices [Amazon Echos]” [19]. While users are becoming used to the idea of voice-controlled devices, they are still in their infancy and users can find that the ‘assistant’ can’t understand them or their commands [20]. Users should be able to construct their commands in a way that feels natural to them, without needing to remember a long list of phrases.

1.4 Previous Projects

Previous research and projects conducted at the University of York include creating a voice-controlled web browser [21]. Although this system was limited in its use and the commands that it featured were strict and didn’t afford the user any flexibility, it demonstrated that a proof of concept package could be created in a relatively short amount of time. It also highlighted that a system would need to be flexible with its commands as to not alienate the user and disrupt them from their workflow.

Another project looked at creating a system for a blind radio presenter [22]. The solution proposed was to have the user wear a **vibrafication** glove while interacting with a mixing desk, to guide the user to the correct slider. This solution worked very well in this respect, although the system would not be accurate enough to guide the user to a small dial on an oscilloscope, nor would it be able to offer real-time feedback as to the position of dials.

Figueiredo looked at using IBM’s Watson to create a platform for controlling a swarm of robots by voice control [23]. They were successful at creating a prototype system with the assistance of a collaborator from IBM. The project utilised the speech-to-text platform [24], which has a limited demo plan and would necessitate an initial cost to the project and therefore hasn't been considered.

Cleaver [25] created a voice-activated recipe app for iOS. The NLP platforms they considered were iOS-specific and therefore could not be considered for a Windows or UNIX based system. In a similar project, Corpe [26] created an iOS specific application for a voice-controlled web browser. This project utilised Nuance's Dragon Speech API [27] for the NLP. A demo version of the API could not be found and therefore has not been considered for this project.

1.5 Design Guidelines

Designing intuitive and accessible software is a difficult task. Many people have become accustomed to their computers working in a specific way; for example, menus arranged in a logical order. For visually and physically impaired users, using a computer can be a challenge. Accessibility tools make this more comfortable, but the underlying software needs to work seamlessly so that the user is not put at as much of a detriment.

1.5.1 Universal Design

The National Disability Authority has produced a set of design guidelines for application software accessibility [28]. These guidelines are split into two levels of priority and will be used to self-evaluate the designs created for the project. The most important guidelines for this project will be:

- "Ensure compatibility with assistive technologies";
- "Ensure that all information can be perceived by users with restricted or no vision";
- "Ensure that all information can be perceived by users with restricted or no hearing";
- "Use the simplest language possible for instructions, prompts and outputs and, where possible, supplement it with pictorial information or spoken language".

1.5.2 Response Times

A system must react to a users input and process any data within an acceptable amount of time to allow them to carry out their work quickly and efficiently. Nielson states that there "3 main time limits (which are determined by human perceptual abilities) to keep in mind when optimizing web and application performance." [29]:

- Between the user activating the software and it being ready to receive input, there should be less than 0.1 second delay.

- Between the user completing their input to a system, and it processing it, there should be less than a 1 second delay. Anything longer than this the system will need to give clear visual and audible feedback that it is processing the input.

- Longer than 10 seconds the system is becoming an obstacle to the user and careful consideration should be made to make sure the system will 'timeout' the current process. At this point the user will be advised to make sure all the connections to the devices are functioning correctly and that the system can hear them through the microphone. A metric analysis must be made to make clear decisions on whether or not the system is meeting these timing requirements.

The Apdex Alliance, Inc. created a "method for calculating and reporting a metric of transactional application response time in the form of an index with a value of 0 to 1." [30]. The metric, calculated by choosing a target threshold of T seconds, is given by:

$$A = \frac{N_S + \frac{N_F}{2}}{N} \tag{1.1}$$

Where:

- A is the resultant rating, between 0 and 1;

- N_S is the number of 'satisfied' requests, taking less than T seconds;

- N_F is the number of 'frustrated' requests, taking less than $4 * T$ seconds;

- N is the total number of requests.

This metric will be used during the testing phase as an indicator of how fast the application can respond to a user's command.

1.5.3 "Earcons"

For visually impaired users, clear, audible feedback in systems is vitally important. Although recent developments in speech synthesis have made controlling applications much easier for visually impaired users, spoken feedback is usually quite verbose and takes a long time for the user to process, which may hinder the user if they are waiting for feedback to carry on with their work. Brewster et al. developed a series of "Earcons", a set of easily recognised audible tones and 'beeps' that the user could associate with specific actions or commands [31].

The website Freesound.org offers a "collaborative database of audio snippets, samples, recordings, bleeps" for use in applications [32]. Users can upload sound effects and describe them using tags such as 'positive' or 'negative'. Using sound effects from Freesound.org will save time in development, and the tones can be quickly switched out and tested as time goes on during the project.

1.6 Accessibility Tools

For the majority of PC users, a Keyboard, Mouse, and a Monitor are all that's needed to adequately complete most tasks. Unfortunately, some users require specialist equipment to improve their experience. Any system created will need to work seamlessly with these applications to allow the user the freedom of tailoring their experience for their disability. There are different tools for each common OS; Windows, OS X & Linux[3]. As this project is focusing on the current set-up seen in Section 1.2, the only OS that will be considered is Microsoft Windows 10. Many accessibility tools are built into Windows [34] including:

- **Magnifier**: Enlarges all or part of the screen.
- **Narrator**: Describes the screen and any inputs.
- **High contrast themes**: Alters the colour contrast of text and images on the screen.
- **On-Screen Keyboard (OSK)**: An alternative to a physical keyboard that can be controlled using a pointing device.

1.6.1 Screen Readers & Speech Synthesis

As well as the built-in screen reader for Windows [35], several other free and open-source screen readers are available for users; including NV Access [36] and Serotek System Access [37]. Both of these systems utilise the Microsoft Active Accessibility API, which allows programs to retrieve information about custom UI elements within other software [38]. Microsoft also provides a speech synthesis platform for C# [39], which would afford the user clear feedback from their actions. The final version of the application will be tested with these programs to ensure the end user will be able to use a screen reader of their choice.

1.7 Natural Language Processing

"Natural language processing (NLP) can be defined as the automatic (or semi-automatic) processing of human language." [40]

This project will utilise NLP to process the meaning of the user's command. For example, if the user was to say: "Set the frequency of channel two on the function generator to fifty mega hertz", the command intends to set the frequency of a specific channel on the function generator. This intent would have a unique action defined within the system and would require two separate pieces of data; channel (2), and frequency (50 MegaHertz). Embedded within the frequency is an SI prefix, "Mega", which is $1 * 10^6$. The tool used to process the command will need to correctly identify the intent of the command as well as retrieve and convert the data to a sensible format.

[3]Desktop Operating System Market Share United Kingdom. Jan - Dec 2016 [33]

1.7.1 Wit.ai

Wit.ai is a Natural Language Processing platform that allows developers to create interfaces for devices and software [41]. It has an HTTP API that allows the processing of either plain text or an audio recording of speech [42]. The developer can create a set of intents which relate to actions within their application. These intents then have multiple expressions defined, which outline roughly what the user would say to activate that intent. The platform analyses the input for exact or close matches. It can extract different information, including date-time, a search query, number, colour, etc. The developer can also train the platform to increase its accuracy. This feedback loop means that the system can cope with new expressions from the user, as long as it contains the required information. Appendix A.3.1 demonstrates how the developer highlights the position of data in the commands. Appendix A.3.2 shows an example API response from Wit.ai. It is fairly verbose and would need processing to extract the required information. The important data keys to note are `‘confidence’` and `‘entities[intent]’`.

1.7.2 Dialogflow

Dialogflow is a platform fairly similar to Wit.ai [43], running on the Google Cloud platform [44]. There are a few differences between the platforms, but the idea of intents, expressions and entities are the same. On Dialogflow, the developer can also create custom **enum**s to represent parts of their application. An example of this would be to represent channels on a Function Generator. This helps with the accuracy of the platform, as potential values for data are known and therefore easier to recognise. Dialogflow also allows marking entities as required in intents, meaning that if the user does not correctly supply the information, or the speech recognition fails to extract the data, it will not trigger spurious actions. The response from Dialogflow's API, shown in Appendix A.3.4, is a lot terser and would be a lot easier to process than Wit.ai. The fact that enum values are returned in a consistent manner means they will be easier to match in code.

1.7.3 Local NLP

To do the text processing locally, the intent of the command must be parsed and data extracted correctly. A simple way of capturing information from free text is Regex. Appendix A.3.5 demonstrates how Regex can be used to not only match the intent of a command but retrieve the data from the free text. This would, however, require every variant of a command to be written out and the command be tested against each Regex string until a match is found, making the algorithm $O(N)$[4]. Potential solutions could involve looking for specific keywords in the command (‘frequency’, ‘voltage’, ‘timebase’, etc.) and only applying a subset of Regex strings to find a match.

[4] “O(N) describes an algorithm whose performance will grow linearly and in direct proportion to the size of the input data set.” [45].

Summary

Both Wit.ai and Dialogflow have the required features for this project. To decide on a single platform, and to verify that either platform is good enough for the project, an experiment will be carried out to ascertain their response times and accuracy. Local NLP will only be investigated further only if the online NLP platforms are deemed to be unusable for the project.

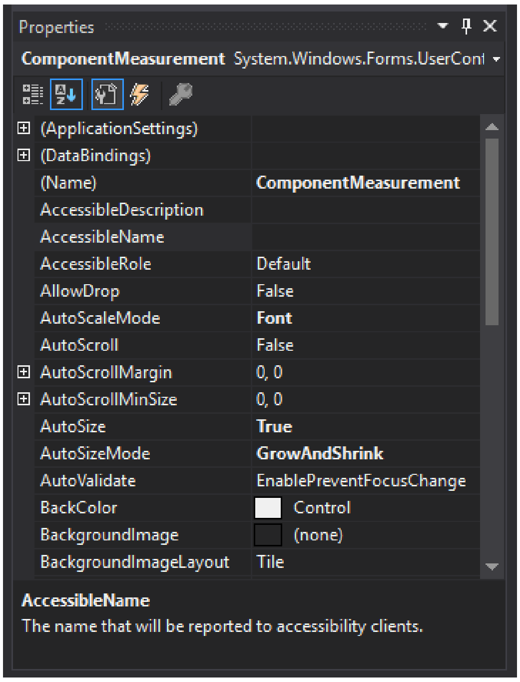
1.8 Programming Languages

Many programming languages would be appropriate for developing an accessible desktop application. This project aims to create an application that runs on the existing setup discussed in Section 1.2, which includes a PC running Windows 10[5]. This will limit the specification to languages that support Windows.

C# First released in 2000, C# is developed by Microsoft [47]. It is well integrated into the accessibility tools discussed in Section 1.6. Figure 1.6 shows how simple it is to set accessibility properties of components. Listing 1.1 shows how the accessible properties can be dynamically updated at runtime. It also features system libraries for interacting with multimedia devices like the microphone. This section discusses a selection of languages that would be appropriate for this project.

Some of the features of C# are not cross-platform[7], but that will not affect this project. The author does not have any experience with C#, although syntactically it is similar to other object-orientated languages like Java, so there shouldn't be much of an issue.

Figure 1.6: Visual Studio[6] - Properties



Java A simple, accessible UI was created to test the assistive API of Java [48]. Using Microsoft Narrator (see Section 1.6), it was unable to read the text content of button aloud, even after manually setting the `AccessibleName` property, seen in Listing 1.2.

[5]The PC's also run Ubuntu (a Linux distribution), but the majority of the users are unaware of this and do not use it.
[6]Visual Studio is a free IDE provided by Microsoft [46].
[7]Discussed in Section 7.1.5

Listing 1.1: C# accessibility snippet

```
protected void setIndex(int index)
{
  this.index = index;
  this.label.Text = String.Format("(%d) Note:", this.index);
  this.AccessibleName = String.Format("Note number %d.", this.index);
}
```

Listing 1.2: Java accessibility snippet

```
setTitle("Start Page");
getAccessibleContext().setAccessibleName("Start Page");
getAccessibleContext().setAccessibleDescription("Start Page.");
buttonStart.getAccessibleContext().setAccessibleName(this.buttonStart.
    getText());
buttonLoad.getAccessibleContext().setAccessibleName(this.buttonLoad.getText
    ());
buttonOpen.getAccessibleContext().setAccessibleName(this.buttonOpen.getText
    ());
```

The chosen language for this project is C#. Due to the incremental nature of the project, if a proof of concept release cannot be quickly developed in the chosen programming language, this decision can be revisited at a later stage.

1.9 Work Methodology

Several methodologies exist that aid individuals and large groups in tasking out and monitoring the progress of projects. For each methodology, there are software packages that aid project management; by automatically calculating factors like estimated time remaining, allowing users to view their tasks and progress in a particular way that aids them in development.

1.9.1 Agile Methodology

Agile Methodology was created in 2001 and stems from other methodologies such as Scrum and Kanban [49]. One of its major principles is to "Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale." [50]. Agile encourages flexible development of a project, with the overall scope of the project being defined at the start of development, but tasks and implementation details are fleshed out as the project continues in "sprints", short, rapid periods of development featuring implementation and testing, and that deliver a working product at the end.

The proposed system will have features that can easily be split up into sprints, each one being a new version of the product. New features added in each sprint will be tested to make sure that existing functionality is not affected or broken.

1.9.2 Source Code Management

Github will be used to keep the source code for the project organised and backed-up [51]. Git features the ability to "tag" specific versions of code; this will be used to identify which release the code relates to [52]. Git has been chosen over other methods of SCM due to previous experience.

1.9.3 Task Tracking

Axosoft is a project management and bug tracking system, capable of managing Agile, Scrum or Kanban style projects [53]. It features a system of estimating how long each task will take, allowing the planning of a sprint to be easily visualised; shown in Figure 1.7. The amount of time taken on each task can be recorded, as well as an estimate of remaining time. This makes it possible for the software to create an estimate of when the sprint is going to be complete. Figure 1.8 shows an example burndown chart, complete with estimated due date. These metrics will allow close monitoring of the progress of the project and will aid in keeping the development time down, as less time will be used keeping track of remaining tasks. Axosoft has been chosen over other project management suites due to previous experience.

Figure 1.7: 'Sprint Planner' on Axosoft

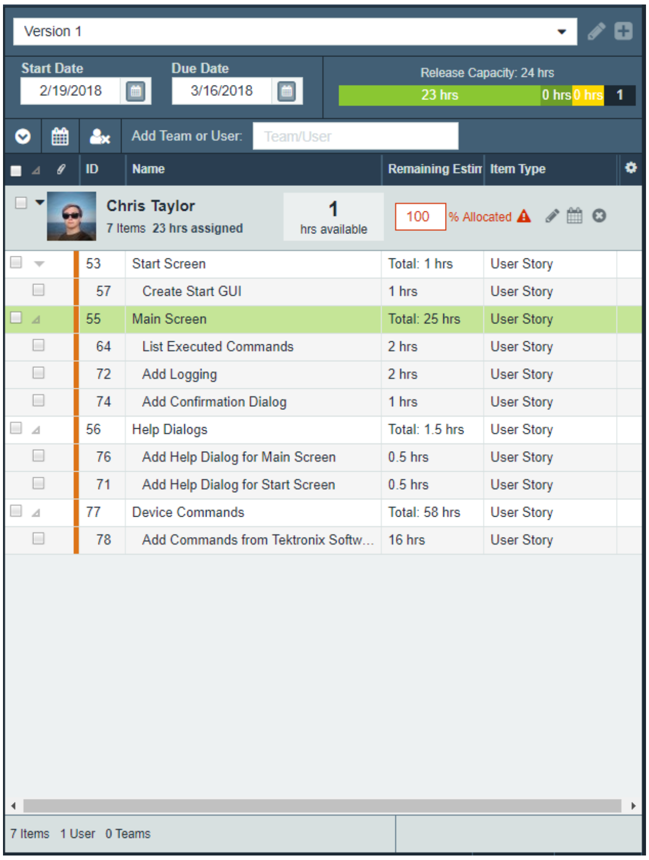
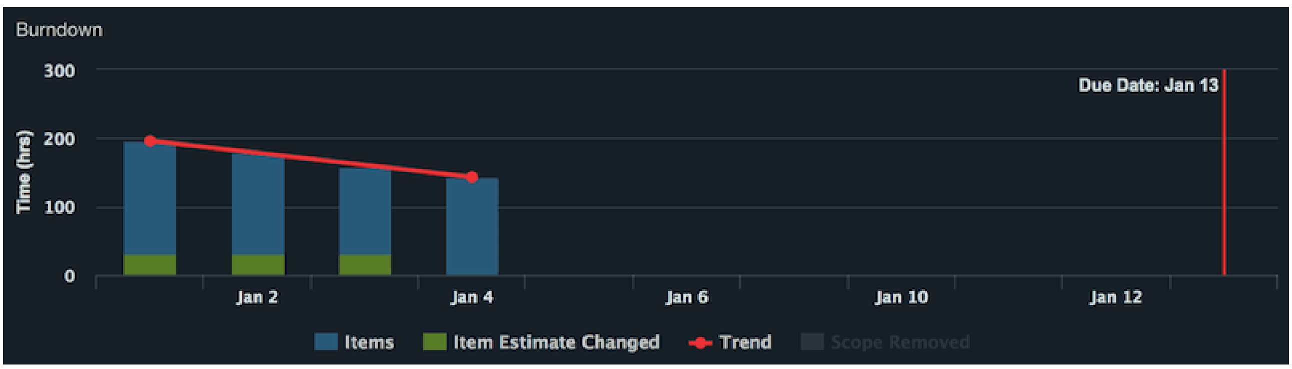


Figure 1.8: Example burndown chart [54]



2 Specification

This chapter defines the aims and objectives of the project. It also outlines the specification and rough testing procedure. The testing outline will make it easier to check the progress of the development, making sure it meets the specification along the way.

Contents

2.1 Aims

- Investigate how users would naturally interact with the lab equipment if it already featured voice-control;
- Investigate which NLP platform is quicker and more accurate;
- Create an accessible interface with the equipment listed in Section 1.2.

2.2 Objectives

- Conduct a study into how users currently interact with the equipment;
- Create a proof of concept release to ascertain if remote NLP is responsive and accurate enough for the final product;
- Conduct a study to compare metrics between two on-line NLP services;
- Design and implement a bespoke software package, allowing changing of settings for the lab equipment and retrieval of data and measurements taken using the lab equipment.

2.3 Requirements

The software package must meet the following requirements:

- Capable of running on the computers on the 4th-floor labs in the Department of Electronic Engineering at the University of York;
- Capable of understanding the user's commands with a background noise level of 50 dB[8];
- Compatible with popular screen reading software:
  - Microsoft Narrator;
  - NV Access.
- Adheres to "Universal Design" principles, discussed in Section 1.5.1;
- Respond to or acknowledge input within a reasonable time, discussed in Section 1.5.2;
- Able to recognise RP English;
- Understand the commands used by participants in the "Wizard of Oz" study, discussed in Section 3.1.

2.4 Work Breakdown

To make sure that development is steady and that regular releases of the product are created and tested, the tasks for this project are broken down into 'sprints'. Appendix C shows these sprints and the estimated time for each task. Axosoft refers to groups of tasks as 'user stories'.

2.4.1 Releases

For each release there are only a few user stories, this helps to see rapid development in each sprint.

Proof of Concept

- 'Recording/Transcription of Speech';
- 'Communication between PC and TBS 1052B-EDU Oscilloscope'.

Version 1

- 'List and Record Executed Commands';
- 'Subset of Commands for Oscilloscope';
- 'Confirmation Dialogues'.

[8]Taken from the Center of Hearing and Communication's 'Common environmental noise levels' for a 'large office' [55].

Version 2

- 'User Settings';
- 'Communication between PC and AFG1062 Function Generator';
- 'Subset of Commands for Oscilloscope and Function Generator'.

Version 3

- 'Note Dictation';
- 'Communication between PC and UT804 Digital Multimeter'[9];
- 'More Commands for all Devices'.

Version 4

- 'Export to PDF'.

Version 5

- 'Macro Support'.

2.5 Testing

At the end of each sprint, the following testing procedure will be followed to make sure that each release of the product is fully accessible. Finally, at the end of development, a full testing plan will be followed and the results documented.

1. Complete the 'Checklist for Application Software Accessibility' [56];
2. Attempt to use the software while having limited or no vision of the screen, using popular screen reading software;
3. Attempt to the use the software with the speakers off, to simulate loss of hearing.

Any issues found during this testing procedure can be recorded on Axosoft using their bug tracking system, and the highest priority issues will be fixed before the next sprint begins.

[9] If implementation is possible.

3 Experiments

This chapter details the methodology, results, and conclusions of two experiments conducted before the major development of the project. The results of which, aided in the specification and design of the software package.

Contents

3.1 "Wizard of Oz" Study

To understand how a user would naturally, and intuitively, interact with an existing set of voice-controlled lab equipment, an experiment was carried out in which the participant had little or no visual or audible feedback from the lab equipment. The participants were instructed not to touch the equipment either; instead, instructed to give spoken instructions to a third-party who in turn controlled the equipment and relayed any feedback to the participant.

3.1.1 Methodology

A circuit[10] was pre-assembled and checked by the researcher, the layout for which can be found in Appendix D.4. Participants were given an instructions sheet, found in Appendix D.4. The experiment they were asked to conduct features aspects of typical undergraduate labs. The third-party was also given an instruction sheet, found in Appendix D.5. The interactions between the participants and researcher were recorded and transcribed.

[10] Adapted with permission from author [57].

3.1.2 Results

The recording transcriptions can be found in Appendix D.6. The commands to implement and examples of phrasing can be found in Appendix D.7. This section highlights common observations that arose during the testing.

Most participants tended to add 'fluff' words to the commands. "can I have the dmm set to dc voltage *please*" (Page 118), "magnitude reading on channel two *please*" (Page 112). Using a NLP platform, these minor variations will be ignored, and the intent should still be parsed correctly.

Figure 3.1 shows how users would mix up words for commands and end up saying something completely wrong. For the second example, the user was trying to get the phase *difference*, not the phase *margin*. The system would not be able to tell from the context what the user meant and, for commands such as setting the output voltage of the function generator, could potentially have dangerous side-effects. The system will need to have the ability to confirm what the user meant for these kind of commands.

Figure 3.1: Incorrect commands

"adjust the amplitude to be 100 millivolts per time division" (Page 130)

"okay and what is the phase margin" (Page 131)

Figure 3.2: Complex commands

"umm source channel two and rotate clockwise on the multi-purpose dial until it is 10% of [the final value]" (Page 121)

"can I have the umm the horizontal scale increased by one twist. I don't know how to say it." (Page 121)

"adjust the timebase of the oscilloscope accordingly" (Page 112)

Although the instruction sheet told the participant to talk to the third-party as naturally as possible, the commands they tended to give initially were assuming prior knowledge of the equipment. Figure 3.2 shows how users were overestimating the complexity of the commands they could give. For the first example, the user wanted the oscilloscope to move the second cursor until it was at a point where the delta voltage was 10% of the final value, in order to calculate the fall time of a waveform. This could potentially be implemented using some custom code, but the use cases would be very limited. There is also a command to get the oscilloscope to measure fall time without needing to use the cursors. This suggests that the user needs access to a list of implemented commands so that they can see what exactly the devices are capable of.

Figure 3.3 shows hows users would correct themselves mid-command. It would be tough to implement a system that could recognise when the user has, essentially, overwritten a previous part of the command.

Figure 3.3: Corrections within commands

"can I have a one volt pulse wave from the signal generator. One milli volt" (Page 126)

"zoom in. sorry reduce the amplitude of the. yeah reduce the amplitude of the time base. no not the timebase, the voltage divisions." (Page 129)

"set the oscilloscope to measure phase margin. phase difference sorry" (Page 129)

It was interesting to note the interactions between the user and the third-party. Figure 3.4 shows a small snippet where the third-party misheard the user and asked for clarification on a small part of the command. It would be difficult to implement a NLP system that could ascertain whether a small part of the command was ambiguous, and then be able to ask for clarification. Most existing systems would just accept that the speech pattern sounded ever so slightly more like one or the other. This again suggests that a confirmation system will need to be put in place for certain commands.

Figure 3.4: Clarification of command (Page 130)

Participant: can you make it smaller 50 milli seconds per division

Third-Party: 15 or 50

Participant: 50

3.1.3 Conclusions

This experiment has shown that the system produced will definitely need to include safety features in case the NLP platform misunderstands the user and potentially creates a dangerous situation. The system commands will also need to be well documented so that the user is aware of all possible commands and example phrases they can use.

3.2 Metric Study

Several online NLP platforms exist that offer the features required for this project. To make an informed choice between them, an experiment was carried out to find out which platform responded quickly and accurately to the type of commands that will be supplied by the user.

3.2.1 Methodology

A prototype release of the software was made that timed how long the analysis from each NLP platform took. A set of participants were asked to say a set of pre-written commands. Each platform was sent the same voice recording and the response was checked to make sure the command was **parsed** correctly. Appendix B.1.1 shows an excerpt of code recording the time taken.

The commands that the participant were shown are a sample created from the results of "Wizard of Oz" study (see Section 3.1). The participant was asked to verify that each platform correctly parsed their command and this was then double-checked by the researcher.

3.2.2 Results

The raw results of this experiment can be found in Appendix E.4. Figure 3.6 shows that Dialogflow is, on average, 52% faster at parsing the same speech as Wit.ai. Figure 3.5 shows that Wit.ai is, on average, 46% less accurate than Dialogflow.

Figure 3.5: Accuracy results

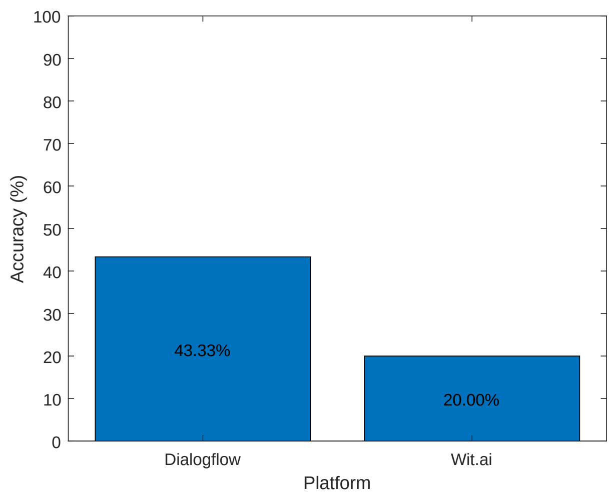
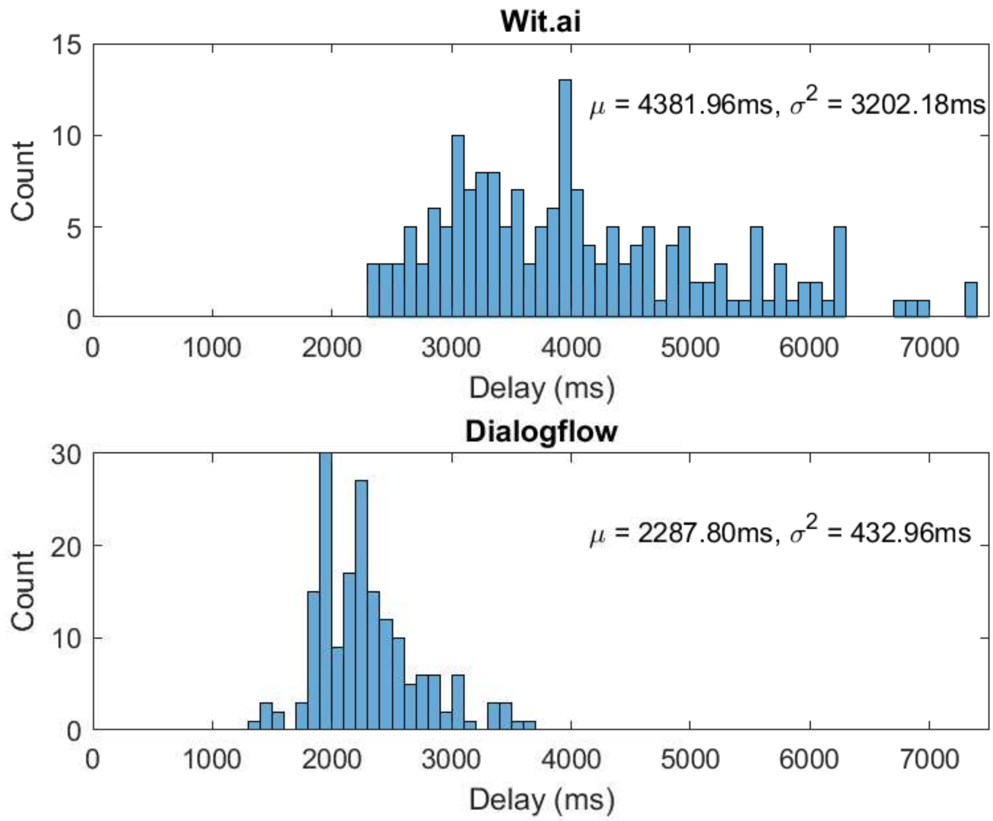


Figure 3.6: Timing results



Common Issues

This section details some common mistakes the NLP platforms made.

Figure 3.7 shows how the platforms struggle to separate digits of numbers from adjacent words that are similar to digits; i.e. 'to' and 'two'. It also struggles with SI prefixes, confusing 'milli' for 'million'. As discussed in the previous experiment (Section 3.1.3), misunderstandings like this could potentially put the user in a dangerous situation.

Figure 3.8 shows an example of the Wit.ai platform mis-parsing a command and returning inappropriate language. This shows that the parsed text should not be shown to the user, instead just showing a generic, 'I didn't get that' message.

Figure 3.7: Incorrect numbers

Expected: "Set the voltage offset for channel one to 5 volts."

Actual: "the voltage offset for Channel 125 volts" (Page 146)

Expected: "Set the frequency of channel 2 to 100 milli hertz."

Actual: "set the frequency of channel two to one hundred million" (Page 149)

Figure 3.8: Inappropriate language (Page 171)

Expected: "Channel 1 frequency to 100 Mega Hertz."

Actual: "channel one frequency to one hundred nigga"

3.2.3 Conclusions

From the results above, it is clear Dialogflow is the better choice for an initial prototype. These results will not be 100% conclusive as the phrases used in the test are a subset of the final phrases that will be implemented.

Both platforms will have a learning rate used in the neural network powering them, and therefore the success rates could change significantly over time and use. If the software was released as a product, a solution would be to implement both platforms and do A-B testing to see if either platform 'learnt' better.

4 Design

The following chapter discusses the design development for the software, UI, and any associated resources. At all stages the design guidelines discussed in Section 1.5 are followed and will be reviewed after development has finished.

Contents

4.1 Branding

Figure 4.1: Pegasus logo [58]

Although this project aims to create a prototype version of a software package, consideration must be made for items such as the name and logo of the software. It is important that the user must be able to distinguish between this application and any others they may have currently open. It must also be considered for the speech recognition, that an intuitive, memorable, and discernible keyword is used to activate the software.

For the purposes of this project, the name of the software has been chosen as 'Pegasus'. The word is easily spoken but is not easily misinterpreted for any other common words. This word has also been chosen as the activation keyword, making it easy for the user to remember. It is also intuitive, as it can be thought of as the user talking directly to the application.

Figure 4.1 shows an open-source logo found that will be used for the logo of the software during development. This will appear in the taskbar of the user's PC and also the title bar of the application.

4.2 User Interface

This section discusses the initial design for the GUI, relating back to design guidelines discussed in Section 1.5.

4.2.1 Start Screen

Figure 4.2: 'Start Screen' mockup

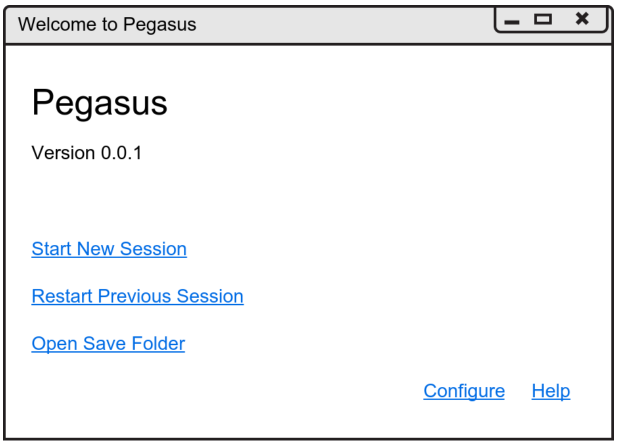
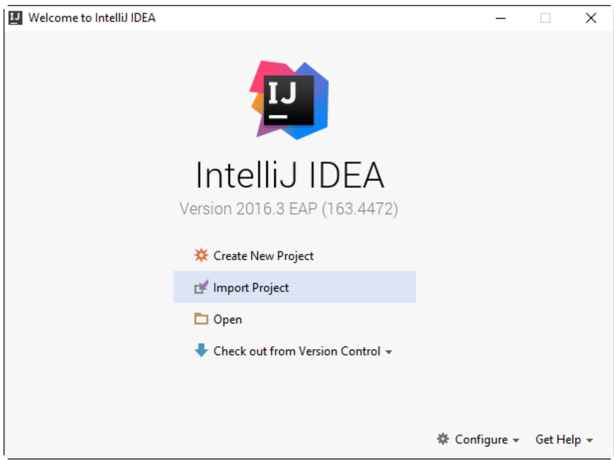


Figure 4.3: IntelliJ IDEA 'Welcome Screen' [59]



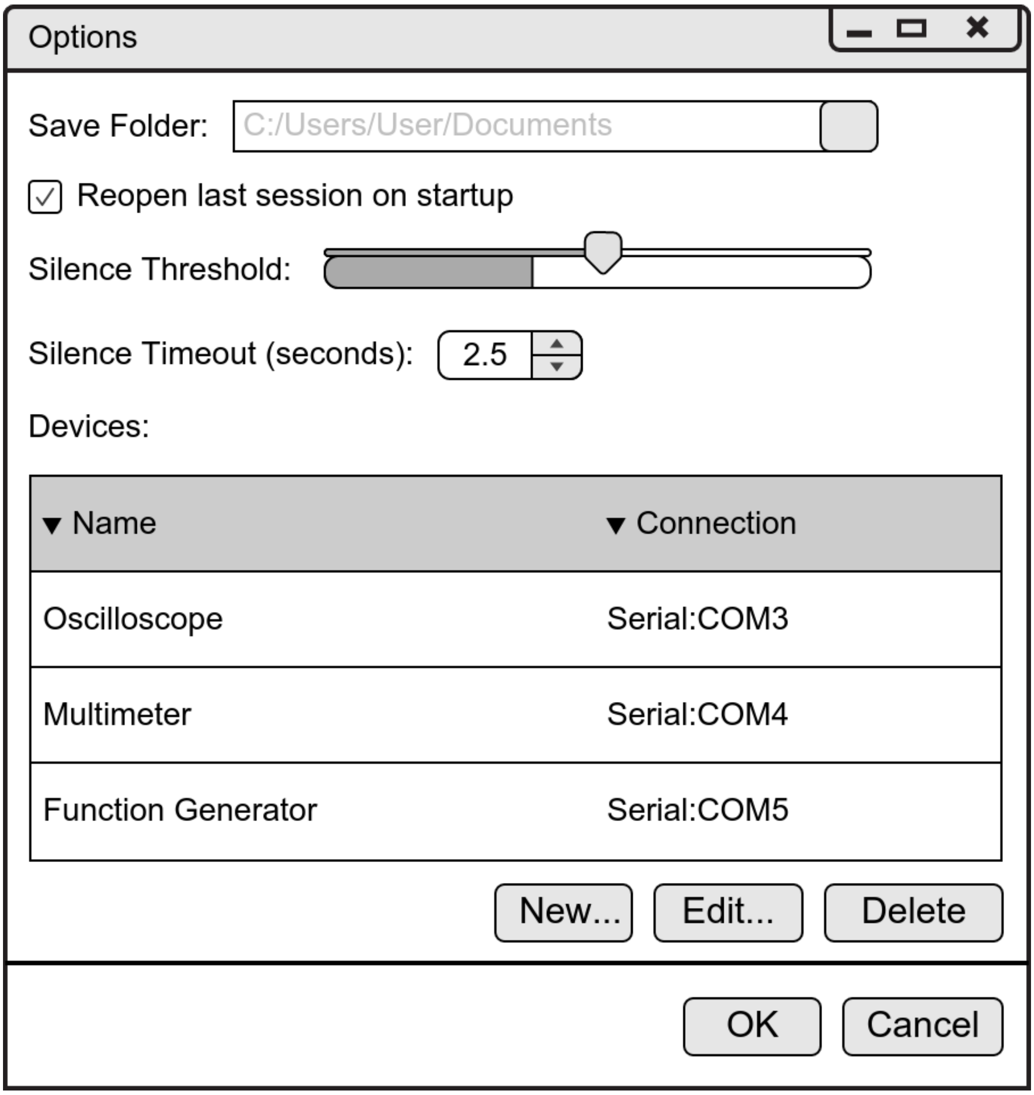
The start screen is very minimal, containing only the software name, version number, and a few options. The options are written out without using any images or icons to make it easier for screen readers. This screen drew inspiration from the range of IDE's from JetBrains (see Figure 4.3). If the user is using software similar to Microsoft Speech Recognition, they will be able to select the options by simply saying the text of the option. This window will disappear once a session has started.

4.2.2 Options Window

The options window will have several settings available to the user. These settings will be written to a file in the user's application data folder[11]. These options are designed to fit the user's expectations, matching the style of existing Windows settings (see Figures 4.5 and 4.6).

[11] On Windows this is at: `C:\Users\User\AppData`

Figure 4.4: 'Options Window' mockup



The options are as follows:

Save Folder allows the user to specify where the sessions are saved to. This will default to the user's 'Documents' folder on Windows.

Re-open last session will skip the start screen and continue the previous session for the user. This may be helpful if the user is conducting the same experiment over a few days.

Silence Threshold allows the user to visualise the ambient noise picked up by the microphone and to set the threshold level and length for the commands. The bar below the slider will be a peak level meter monitoring the default microphone.

Devices is a list of the currently connected devices to the PC. For each one, the settings can be changed by highlighting the entry and clicking 'Edit', at which point the dialog seen in Figure 4.9 will appear. The device can be removed by clicking 'Remove', and new devices can be added by clicking 'Add'. For new devices, the connection dialog will also appear.

Figure 4.5: Windows - ‘Environment Variables’

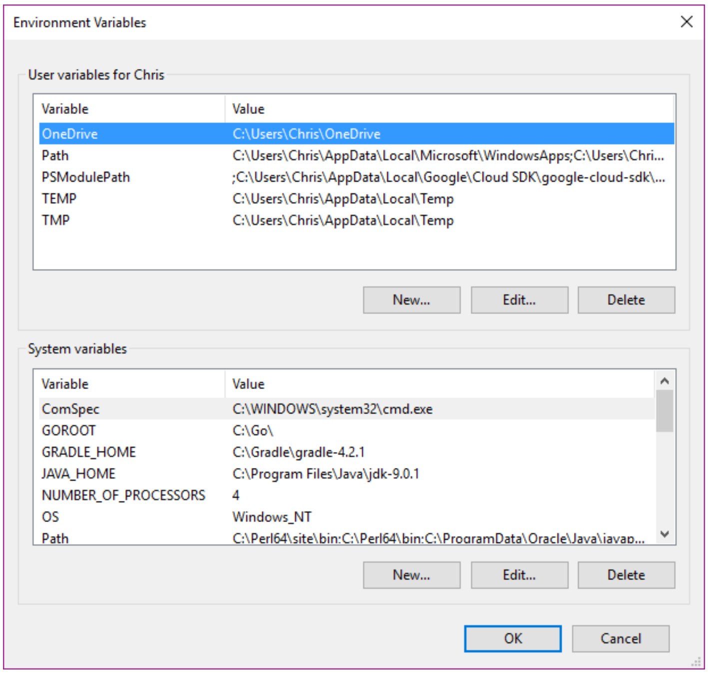
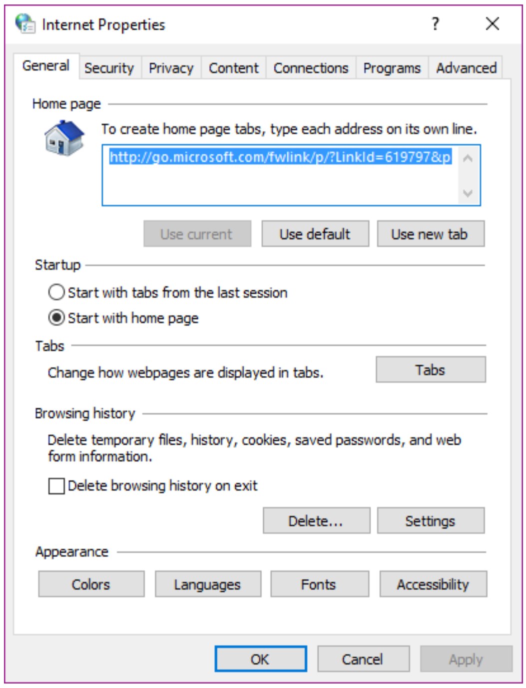


Figure 4.6: Windows - ‘Internet Properties’



4.2.3 Session Window

When a new session is started, the interface is again very minimal. There will also be immediate feedback to the user that the system is listening. As suggested by Microsoft [60], the options on the menu bar are in the following order: ‘File’, ‘Edit’, ‘Help’. This helps to keep consistency with other applications.

As the user starts to give commands, they will be listed on the screen in chronological order. Measurements taken from devices will appear in standard engineering notation with the option to be copied to the clipboard or deleted. Visual measurements, such as screen-shots from the oscilloscope will have the option to be opened in the system file explorer. Notes will have an edit button which will bring up the edit note dialogue seen in Section 4.2.5.

The bar at the bottom of the window will reflect the current microphone level, similar to other established software such as Audacity (see Figure 4.7). The colour of the bar will help to convey the state of the application. This information will also be reflected in the title bar of the window, for accessibility purposes. The four states of the software and the associated colours are listed in Table 4.1.

Figure 4.7: Audacity 'Peak Level' meter [61]

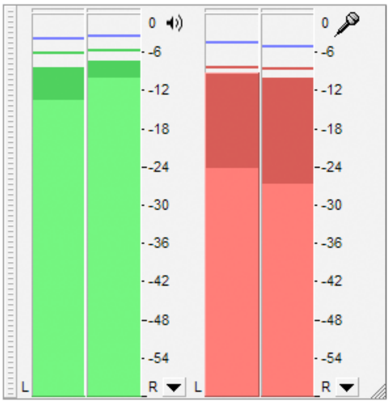
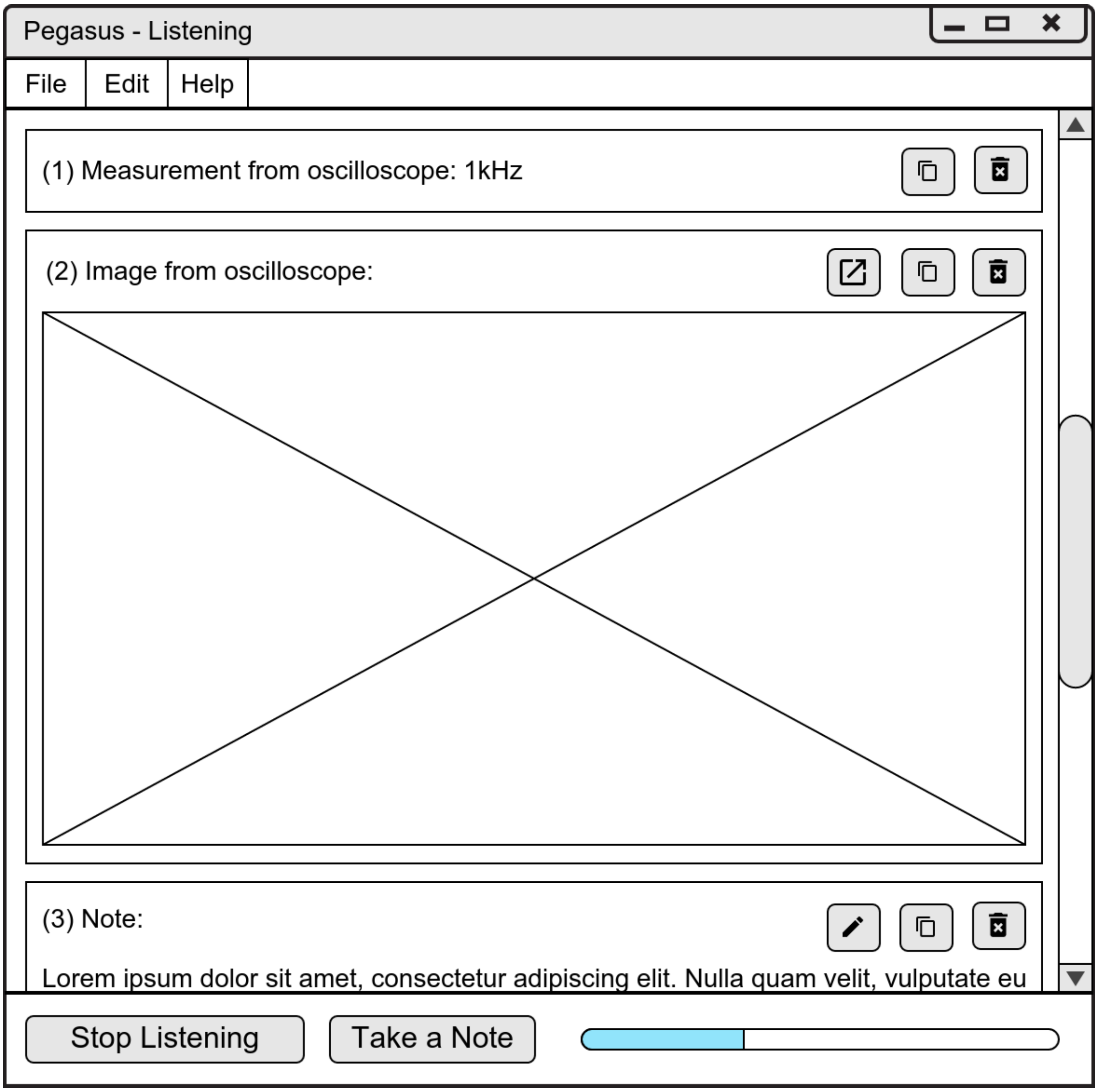


Table 4.1: State colours

State	**Colour**
Idle/Inactive	Grey
Listening	Green
Processing	Yellow
Error	Red

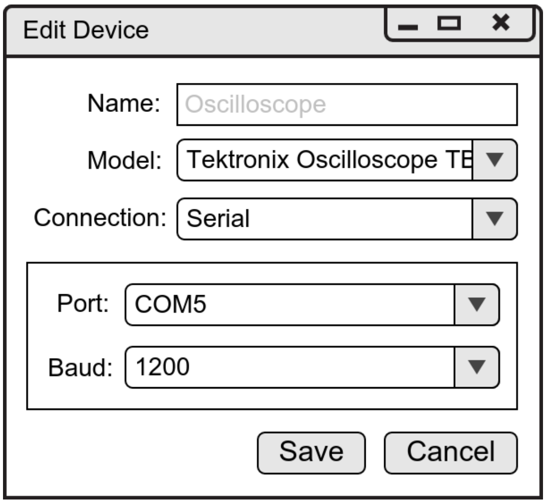
Figure 4.8: 'Existing Session' mockup



4.2.4 Connection Dialog

For each device, the system will provide the available connections to the device. If more information is required to connect to the device, i.e. for serial the Baud rate and port, the inputs in the box will dynamically change to allow the user to provide the information.

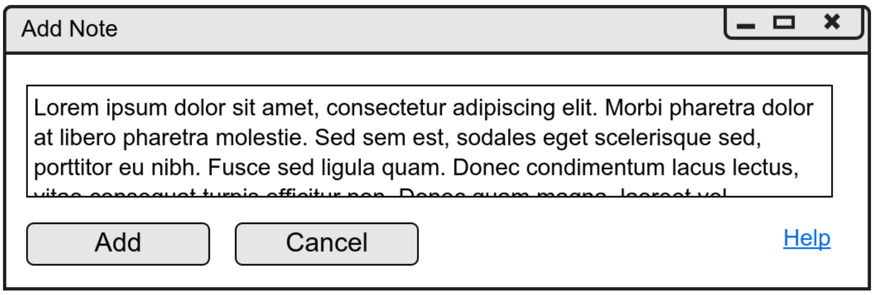
Figure 4.9: 'Connection Dialog' mockup



4.2.5 Note Dialog

The note input will be a simple text box, allowing line breaks. If the user is using speech recognition software, they will be able to use it to dictate their note, and a screen reader will allow them to read back the content.

Figure 4.10: 'Note Dialog' mockup



4.2.6 Delete Dialogue

To make sure the user does not accidentally delete any measurements/commands/notes/screenshots, they will be presented with a simple dialogue asking for confirmation first.

Figure 4.11: 'Delete Dialog' mockup

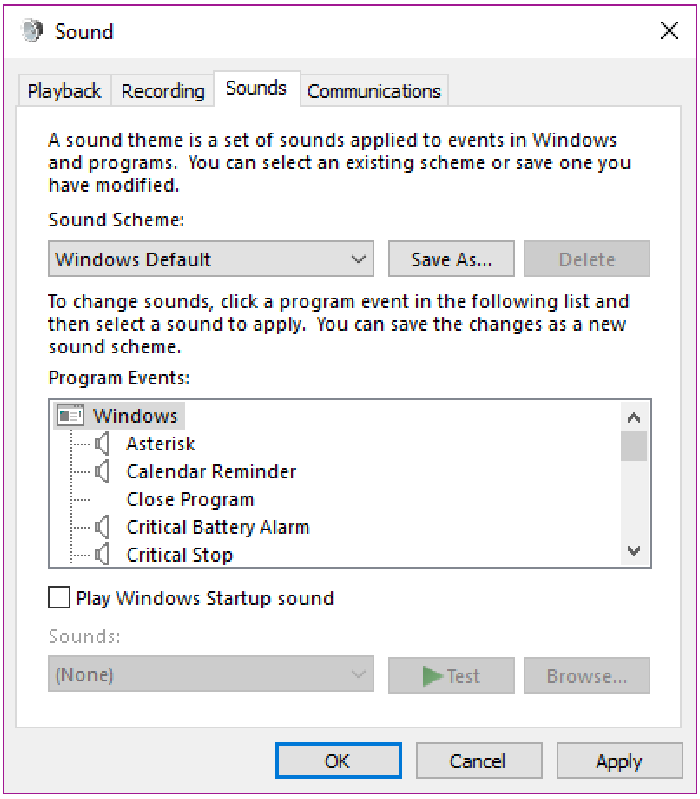
4.2.7 Audible Feedback

Using C#, the application has access to the `SystemSounds` class which allows the following system sounds to be played:

- Asterisk
- Beep
- Exclamation
- Hand
- Question

Each sound is customisable by the user using the 'Sounds' menu in Windows, shown in Figure 4.12. This means the user can fit the sound effects to their particular needs and wants.

Figure 4.12: Windows - 'Sound' settings



4.3 Configuration Files

Alongside the GUI design, there will need to be files stored on the user's PC to maintain state between uses of the application. This section details the format and use of each file.

4.3.1 Settings File

The settings file will maintain the state of the users preferences between uses of the application. It will also store the connection details of the devices. Appendix B.2.1 shows how this will be stored in an Extensible Markup Language (XML) document. XML is commonly used to store data in a way that is both human and machine-readable.

4.3.2 Device File

Listing 4.1 shows how an example command will be stored in another XML document. There will be a separate file for each device model. The keys are as follows:

- '`id`' is a unique identifier for the command, used to link with an 'intent' on the NLP platform;
- '`description`' is a human-readable description of the command and its effects;

- `'example'` is an example phrase for the command;
- `'format'` is a template string, used to create the machine readable command sent to the device;
- `'confirm'` is a Boolean, indicating whether the user needs to confirm the command before it's sent to the device.

Appendix B.2.2 gives an example of how commands will be separated into groups, to make the documentation easier to read for the user.

Listing 4.1: Example command XML

```
<command id="osc.averaging.set">
  <description>Sets the averaging level.</description>
  <example>Set the averaging to times 4.</example>
  <format>ACQ:MOD AVE; ACQ:NUM {oscaveraging}</format>
  <confirm>false</confirm>
</command>
```

4.4 PDF Exporting

Appendix F.1 shows an example PDF. It will contain information such as:

- Date of Export
- Bench Number[12]
- Connected Devices

It will list in chronological order any commands given and measurements taken with an associated time-stamp. The intention for this is that the user can submit the document alongside any other reports and be able to easily reference any part of it.

4.5 Macro Support

To make certain actions repeatable and reliable, a simple Macro scripting language will be implemented. It will be able to perform any action that the user could, using the NLP input. In order to present data in a sensible manner, the script will be able to export any results into a spreadsheet for the user to process further.

[12]This could also be PC name or similar.

4.5.1 Syntax

This section briefly explains each language construct with simple examples. A complete example, for taking a bode plot of a simple amplifier circuit, can be found in Appendix B.2.3.

The macro language is case insensitive for everything other than variable names, i.e `foo` and `FOO` are considered different variables. Commands are delimited by a line break and comments are started using a '`%`' symbol and continue until the end of the line.

Listing 4.2: Example variable assignment

```
let foo = "bar"
let i = 1
let results = []
```

Variables

There are 3 types of variables:

- Strings;
- Numbers, internally stored as doubles;
- Arrays, which can store Strings and Numbers.

Number variables can be incremented/decremented. If the same operation is applied to non-String variables a `RuntimeException` will be thrown and execution will cease.

Listing 4.3: Example variable operators

```
i++ % increment i
i-- % decrement i
let foo = bar * 2
let fizzbuzz = foo MOD 15
results.push(fizzbuzz) % appends
% fizzbuzz to the end of results
```

If/Else Branches

Comparisons that are currently implemented are `==`, `!=`, `<`, `>`, `<=`, `>=`. Boolean operators are not yet supported.

Listing 4.4: Example if/else

```
let mod2 = foo MOD 2
if(mod2 == 0)
  result = "EVEN"
else
  result = "ODD"
fi
```

For Loops

For loops require an exit condition, but the initialisation and increment statements can be omitted to act as a pseudo `while` loop.

Listing 4.5: Example 'for' loops

```
let total = 0 % Calculate the sum of
FOR(LET i = 1; i <= 10; i++) % numbers
  let total = total + i      % 1 to 10
ROF

let a = 1    % Calculate fibonacci
let b = 0    % numbers up to 100
for(;a < 100;)
  let t = a
  let a = a + b
  let b = t
rof
```

Functions

Functions allow the Macro script to communicate with the connected devices. The result can be assigned to a variable. For most functions the return type is a double. All arguments are named and have no set order, similar to Python [62]. If a required variable is missing at execution a `RuntimeException` will be thrown and execution will cease.

Listing 4.6: Example function call

```
LET IN = osc.measure.get(measurement = "pk2", channel = "CH1")
LET OUT = osc.measure.get(measurement = "pk2", channel = "CH2")
LET PHASE = osc.measure.get(measurement = "phase", channel = "CH1",
    channeltwo = "CH2")
```

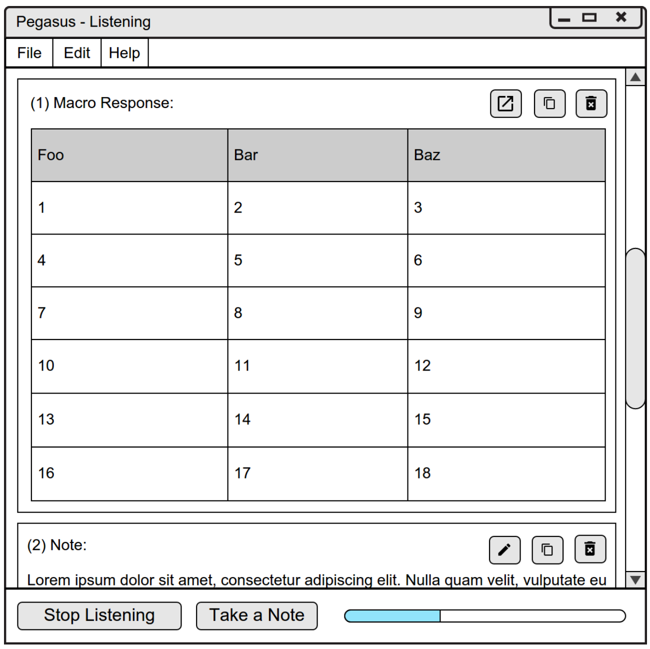
Exporting Variables

The results of macros are exported into a table in the main view, shown in Figure 4.13. They can then be copied into a spreadsheet. Exporting a variable will export its last state in execution, so the `export` keyword can be used anywhere in the code and it will perform the same action. Variables are ordered in the table in the same order they were exported in code.

Listing 4.7: Example variable exporting

```
% exports multiple variables
% at once
export foo, bar
export baz
```

Figure 4.13: Macro result mockup



5 Implementation

This chapter documents the development of the software package, giving an in-depth explanation of each version and the user stories within.

Contents

5.1 System Overview

The application uses the Model-view-controller (MVC) architecture. Internal data processing is done by utility classes and external data processed by service classes. Appendix F.3 gives a brief overview of the main classes in the application.

5.2 Proof of Concept

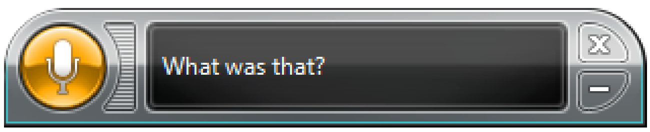
This section discusses the development of a proof of concept release for the software. The application was designed as a simple script to run in the command window.

5.2.1 Voice Activation

To activate the voice recording without needing to press a button on a screen or a key on the keyboard, a keyword listener was implemented. In C# you can access the system speech recognition software by creating a `SpeechRecognizer` object and giving it a list of keywords for it to activate from. Listing 5.1 shows the `KeywordListener` class and how callbacks are added to the `SpeechRecognized` event handler.

During development, it was discovered that the system speech recognition software needs to be fully set-up before it works properly. This involves a short training session in which the user has to read some text from the screen. After the system was set-up it still only worked less than 50% of the time, with the rest of the time it showing a very frustrating error message, seen in Figure 5.1.

Figure 5.1: 'Windows Speech Recognition'



Listing 5.1: KeywordListener class

```
class KeywordListener
{

  public KeywordListener()
  {
    this.sr = new SpeechRecognizer();
    Choices activation = new Choices();
    activation.Add(new string[] { Properties.Resources.Keyword });
    GrammarBuilder gb = new GrammarBuilder
    {
      Culture = Thread.CurrentThread.CurrentCulture
    };
    gb.Append(activation);
    Grammar g = new Grammar(gb);
    this.sr.LoadGrammar(g);
  }
```

```
    public void AddActivationCallback(Action callback)
    {
        this.sr.SpeechRecognized += new EventHandler<SpeechRecognizedEventArgs>((object sender, SpeechRecognizedEventArgs e) =>
        {
            callback();
        });
    }
}
```

5.2.2 Silence Detection

To allow the user more time between starting the audio recording and them needed to speak, an advanced silence detection method was implemented. The system allows for two different time-outs during recording. Once the user has started the recording, they have 5000 ms to start talking before the recording shuts off. Once they have started talking, the recording will continue 1500 ms after the peak level drops below the silence threshold setting, the default for which is -10 dB.

5.2.3 Connecting to NLP Platform

Once the user's command has been recorded, it is sent to the NLP platform for parsing. In order to make it easier in the future if the platform needed to be replaced, the `Dialogflow` class (see Figure 5.2) was made as generic as possible. The `postAudio` function accepts the raw audio and then:

Figure 5.2: Dialogflow class diagram

Dialogflow
postAudio (audio : byte[ ]) : string processResponse (rawResponse : string) : DialogflowResponse

- `Base64` encodes it;
- Constructs a JSON payload containing the encoded audio;
- Makes an HTTP `POST` request to the NLP API.

Figure 5.3: DialogflowResponse struct diagram

DialogflowResponse
variables : Dictionary<string, string> intent : string parsedText : string confidence : float

The API, in turn, responds with a JSON response which is processed in the `processResponse` function. The `DialogflowResponse` class is also designed to be as generic as possible, containing the required fields in a simple structure. If the NLP needs to be replaced, the subsequent class needs only to process the data to a similar structure to make refactoring easier.

5.2.4 Connecting to Oscilloscope

Communicating with the TBS 1052B-EDU and AFG1062 was made very simple by using the TekVISA software already installed on the PCs. The DLL was included as a reference to the Visual Studio project and then a `TekVISANet.VISA` object was instantiated to allow communications with the connected devices. The TekVISA can communicate with devices via GPIB, Local Area Network (LAN), Serial, VME Extensions for Instrumentation (VXI), USB, and TekLink[13]. Listing 5.2 shows the code to list all of the Tektronix devices connected to the PC.

Listing 5.2: TekVISA example

```
public static ArrayList ListConnectedDevices()
{
  TekVISANet.VISA TVA = new TekVISANet.VISA();
  ArrayList devices;
  TVA.FindResources("?*", out devices);
  return devices;
}
```

5.2.5 Template Replacement

To process the raw responses from the devices, a simple template engine was created. Regex is used to capture the data from the raw response, using named matches, and then a human readable response can be constructed using the data and some formatting functions. Listing 5.3 shows two simple examples, the second of which demonstrates how the data is formatted for the user. The variables `number` and `multiplier` are passed into the function `si` which formats floating point numbers to standard engineering notation.

Listing 5.3: Template replacement examples

```
TemplateUtils.ResponseTemplateReplacement(
  "DUTYCYCLE: 85.1", // rawResponse
  "^DUTYCYCLE: (?<percentage>\d+(\.\d+)?)$", // templateSearch
  "{percentage}%" // templateReplace
); // 85.1%

var replaced = TemplateUtils.ResponseTemplateReplacement(
  ":CH1:POSITION -1.5E-3", // rawResponse
  "^:CH1:POSITION (?<number>\-?\d+(.\d+)?)E(?<multiplier>\-?\d+)$",
  // templateSearch
  "{number, multiplier | si}V" // templateReplace
); // -1.5 µV
```

[13] A proprietary connection created by Tektronix.

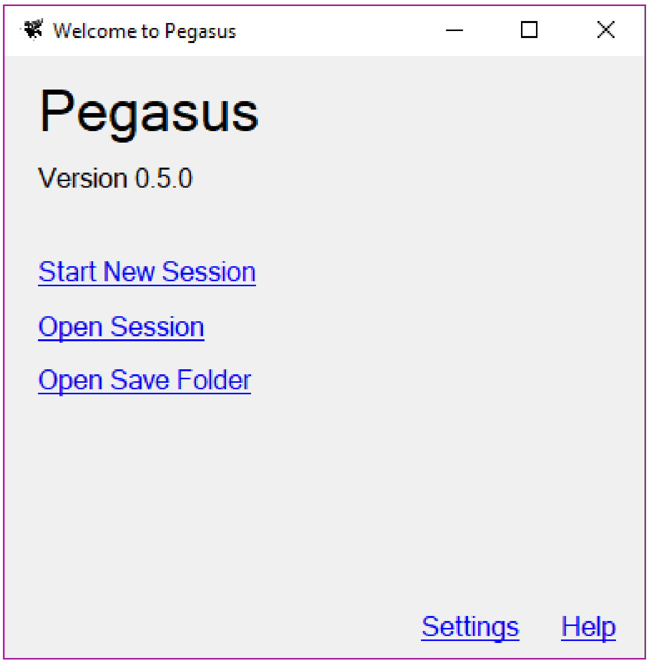
5.3 Version 1

5.3.1 Start Screen

Visual Studio's GUI designer made most of the UI implementation quite simple. Components are added by drag-and-dropping them from the 'Toolbox' and then setting the required properties. The IDE automatically generates listener functions. Figure 5.4 shows the final implementation of the start screen. Listing 5.4 shows how the name and version number are dynamically set from the application resources.

For accessibility purposes, if the user is using a speech recognition system, they will be able to speak aloud the names of the links to activate them. The correct properties have also been set so that a screen reader application will be able to inform the user of all the options.

Figure 5.4: 'Start Window'



Listing 5.4: 'Start Window' data binding

```
public FormStart()
{
  InitializeComponent();
  versionLabel.Text = String.Format("Version {0}", Properties.Resources.
   Version);
  nameLabel.Text = Pegasus.Properties.Resources.Name;
  this.Text = String.Format("Welcome to {0}", Properties.Resources.Name);
}
```

5.3.2 Main Screen

The main window UI was mostly simple to implement, again using the Visual Studio GUI designer. However, during the short development time, it was found to be difficult to create a vertical list of dynamic components to show the session elements. For the initial versions, the width of the main window was set to 550 pixels and the dynamic components set to 482 pixels wide. This forced the elements to stack vertically in the container and achieved the desired result.

The toolbar at the top features the following options:

- **File**
  - **Export as PDF** discussed in Section 5.6.1.
  - **Run Macro** discussed in Section 5.7.1.
- **Edit**
  - **Settings** opens the Settings window, discussed in Section 5.4.1.
- **Help** opens the Help dialog, discussed in Section 5.3.4.

Figure 5.5: 'Main Window' - new session

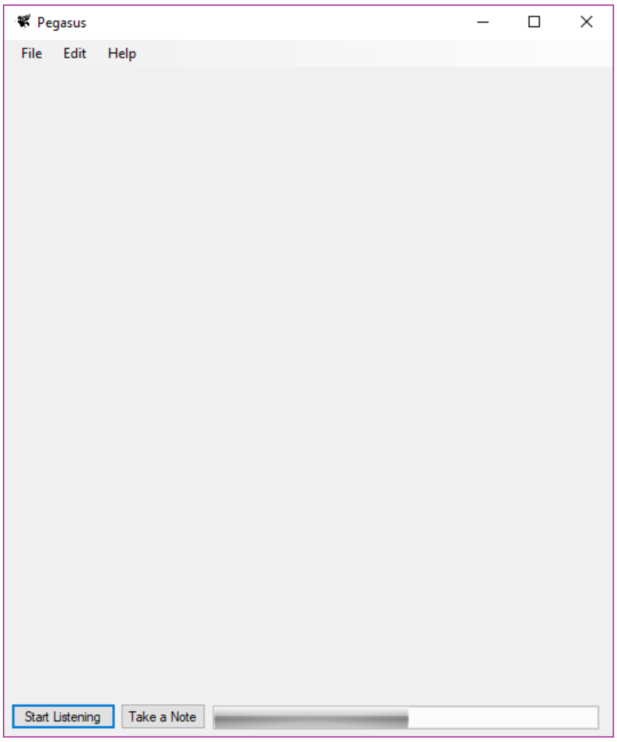
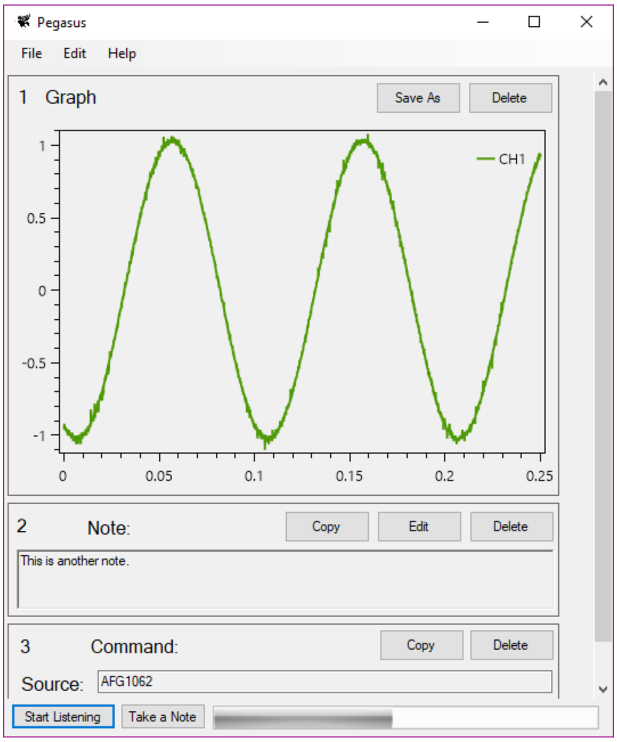


Figure 5.6: 'Main Window' - existing session



5.3.3 Microphone Level

While using the software, it is important that the user has feedback that the application can hear them speaking. A `ProgressBar` allows clear visual feedback, except the default version does not allow for custom colours. In order to show the range of colours discussed in Section 4.2.3, a custom progress bar class was found [63]. Appendix B.1.4 shows how the raw microphone input is processed in order to show an accurate measure of the peak audio level. The processing allows for a smooth transition between peak levels.

The microphone level monitor situated at the bottom of the screen changes colour depending on the current state of the software. In order to decouple the state from the colour of the bar, a mapping function was written, seen in Listing 5.5. The four states are:

Inactive: The default state of the application. The microphone is not recording; however, the keyword listener is active;

Listening: Means the application is currently recording audio;

Processing: Means the application is currently processing the recorded audio;

Error: Means an error has occurred in the processing of the audio.

Listing 5.5: Microphone level colour

```
enum ProgressBarStatus
{
  Inactive,
  Listening,
  Processing,
  Error
}

private void SetProgressBarStatus(ProgressBarStatus status)
{

  Dictionary<ProgressBarStatus, System.Drawing.Color> colours = new
   Dictionary<ProgressBarStatus, System.Drawing.Color>
  {
    { ProgressBarStatus.Inactive,   System.Drawing.Color.Gray },
    { ProgressBarStatus.Error,      System.Drawing.Color.Red },
    { ProgressBarStatus.Listening,  System.Drawing.Color.Green },
    { ProgressBarStatus.Processing, System.Drawing.Color.YellowGreen }
  };

  System.Drawing.Color colour;

  if (!colours.TryGetValue(status, out colour))
  {
    throw new Exception("Invalid colour");
  }

  progressBar.ForeColor = colour;

}
```

Figure 5.7: Microphone status - inactive

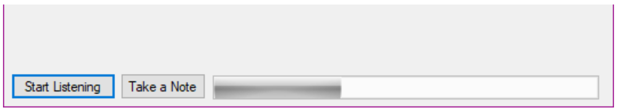


Figure 5.8: Microphone status - listening

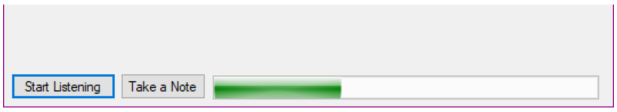


Figure 5.9: Microphone status - processing

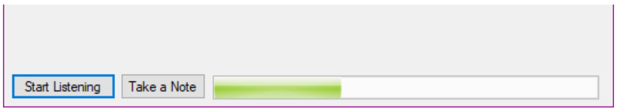
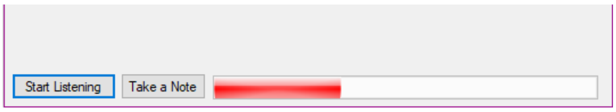


Figure 5.10: Microphone status - error



5.3.4 Help Dialogues

To create a familiar and accessible help system, it was decided to utilise the Windows HTML Help system [64]. HTML documents and other web resources are compiled into a Compiled HTML (CHM) file and can be bundled in with the software's executable file. The CHM files are capable of being viewed both on Windows and non-Windows platforms, with applications for most other OS's. Figure 5.11 shows an example of the help viewer on Windows 10 showing the index of the help file. Specific pages of the help can be opened by referencing the file name, as seen in Listing 5.6. To automatically generate the HTML documentation for the device commands, a simple script was written to parse the XML file containing the command definitions and to output the required HTML, including a table of contents.

Figure 5.11: 'Help Window' - 'Index' page

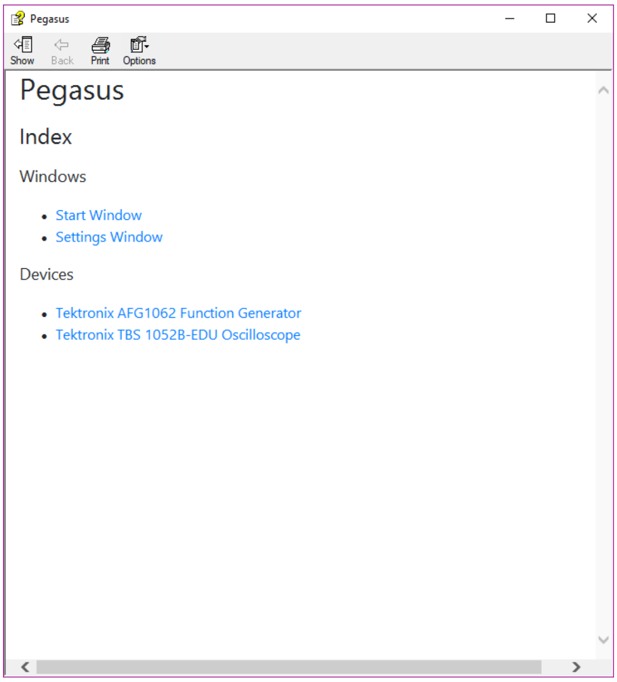
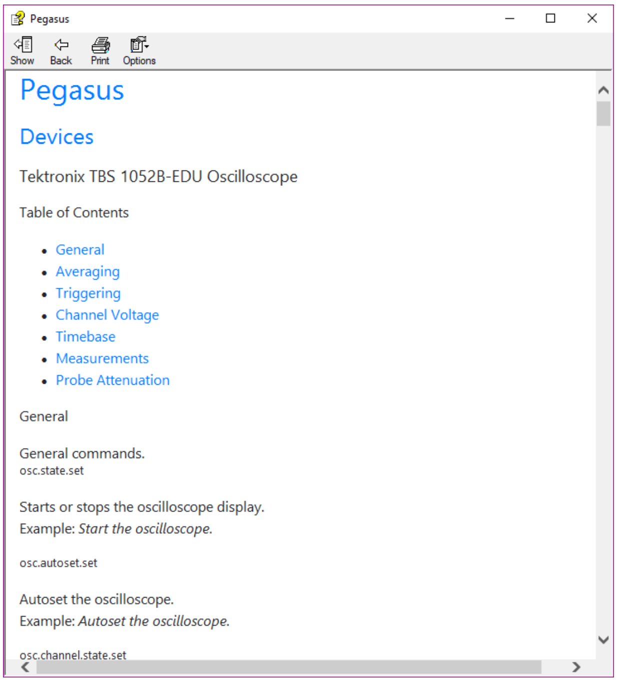


Figure 5.12: 'Help Window' - 'Device' page



Listing 5.6: Requesting help code

```
private void FormSettings_HelpButtonClicked(
  object sender,
  CancelEventArgs e
  )
{
  Help.ShowHelp(this, @"./resources/Pegasus.chm",
    HelpNavigator.Topic,
    "Settings.html"
  );
}
```

5.3.5 Sound Effects

As discussed in Section 4.2.7, the Windows System Sounds are used in the application. To decouple the specific sounds played in the application to their action, a simple `SoundEffects` class was created. As seen in Listing 5.7, the functions are named by their actions, and the associated sound effect can be changed at a later date without needed to refactor a lot of code.

Listing 5.7: SoundEffects class

```
class SoundEffects
{
  static void StartListening()
  {
    SystemSounds.Question.Play();
  }
  static void Processing()
  {
    SystemSounds.Beep.Play();
  }
  static void NothingToProcess()
  {
    SystemSounds.Exclamation.Play();
  }
  static void FinishedProcessing()
  {
    SystemSounds.Asterisk.Play();
  }
  static void ErrorProcessing()
  {
    SystemSounds.Exclamation.Play();
  }
}
```

5.3.6 Logging

To aid in development, a logging class was created that could handle a wide variety of objects without needing to worry about processing them in the caller code. Windows has an 'Application and Services Log' which applications can log to [65], and the device administrator can review the log files and get alerts about errors. In order to create a log file to write to, the user needs admin privileges. To circumvent this, the logger class also writes to a standard console for local debugging.

5.3.7 Defects

Table 5.1: Version #1 defects

ID	Description	Resolution
3	Application crashes when it cannot write to Windows Event Log.	Detect if we have permission to write to the event log, otherwise open a Console Window to log to.
4	Application crashes when Save Folder can't be found.	Detect if Save Folder exists, otherwise, display an Error Message to the User.

5.4 Version 2

5.4.1 Settings

Initial planning for settings called for reading and writing to an XML file to store the user's preferences and connected devices. During development, it was discovered that Visual Studio allows the developer to declare settings, scoped both to the application and the user, and it will generate the boilerplate code for storing and retrieving these settings. It even handles the storage of the settings in the user's `AppData` folder. As shown in Figure 5.13, the developer can also set default values for the settings.

Appendix B.1.3 gives a condensed version of the `FormSettings` class, showing how the application settings are saved and reset, depending on what buttons the user clicks. Figure 5.15 gives a screenshot of the implemented settings window. The bar underneath the 'Silence Threshold' slider is a custom progress bar, discussed in Section 5.3.3.

Figure 5.13: Visual Studios settings interface

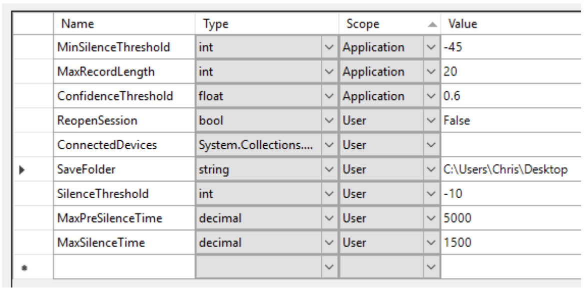


Figure 5.14: Settings binding example

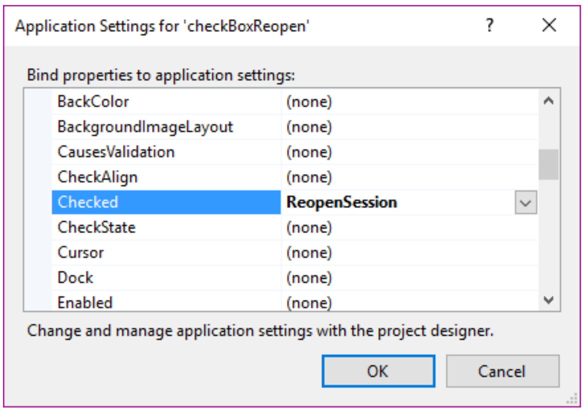
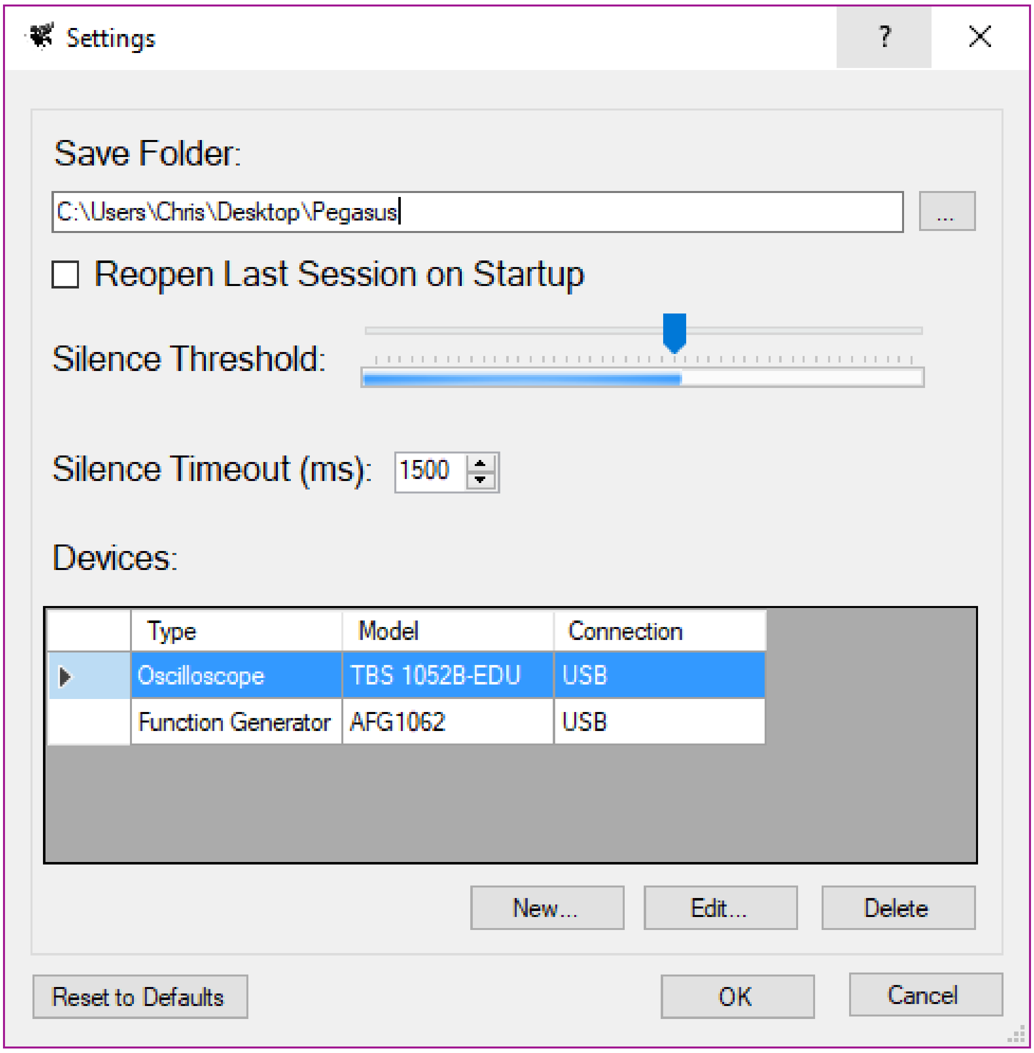


Figure 5.15: Settings window



5.4.2 Defects

Table 5.2: Version #2 defects

ID	Description	Resolution
3	Crashes when there's 'No audio device is installed.'	Report to the user to check their microphone is plugged in and working correctly.
4	Crashes if Speech Recognition doesn't start quickly enough	Report to the user that keyword activation is disabled and they will have to start the recording manually.

5.5 Version 3

5.5.1 Session Storage

For the user to be able to store and recall sessions using the software, all of the information needs to be stored in the filesystem. C# allows classes and objects to be encoded into any custom type, including JSON, XML, or a binary format. For development, to make the files human readable, it was decided to XML encode the data. A `Session` object, seen in Listing 5.8, stores information about the current session. This object is then the root of the XML file. Appendix B.2.5 shows an example of an encoded session.

Listing 5.8: Session class

```
[Serializable]
public class Session
{
  public String version;
  public DateTime time;
  public List<ConnectedDevice> devices;
  public List<DeviceResponse> commands;
  public List<Note> notes;
  public List<MacroResponse> macros;
}
```

5.5.2 Notes

The note window allows users to leave comments throughout their session. Once opened and the text box focussed, the user can use any dictation software of their choice. They can copy and paste into the field, allowing them also to use any text editing software they require. Figure 5.17 shows an example note. The user has the options to copy the content of the note to the clipboard or edit the note.

Figure 5.16: Note dialog

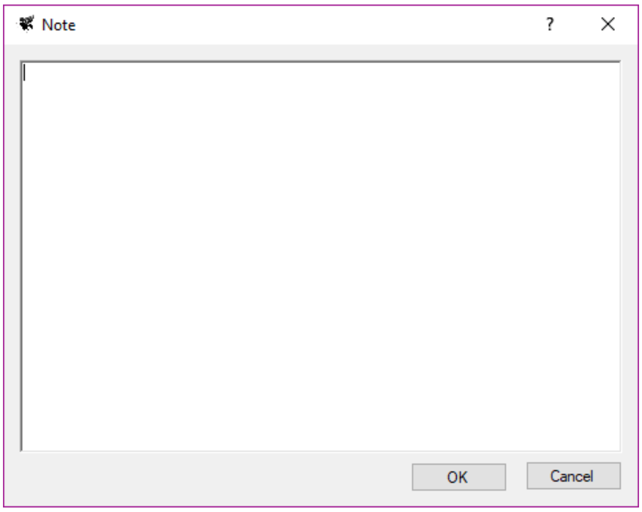
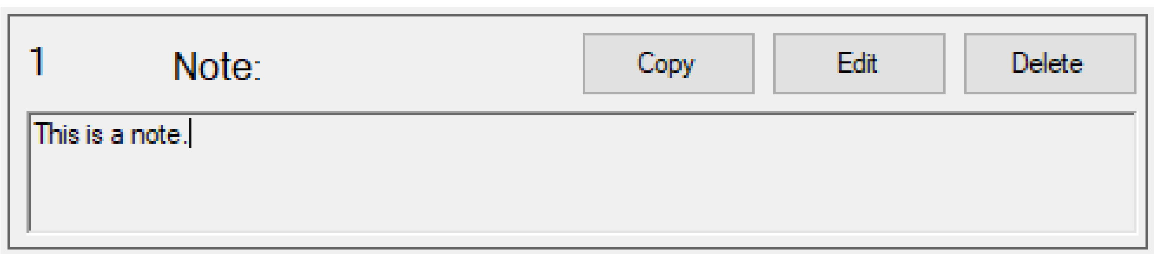


Figure 5.17: Example note



5.5.3 Digital Multimeter

The author made many attempts of contact with the manufacturer of the UT804, UNI-T. Due to lack of a reply, the integration was not possible in the short duration of the project.

5.5.4 Screen-shots From Oscilloscope

Using the `HARDCOPY` function described in the TBS 1052B-EDU programmer manual [7] and attempting to save the file as a `BMP` resulted in an incomplete screen-shot, seen in Figure 5.18. The author posted on the Tektronix forums [66] in an attempt to get some aid in resolving the problem but it was unsuccessful. This problem was circumvented by visualising the data in a different way, discussed in Section 5.7.2.

Figure 5.18: Incomplete screen-shot

5.6 Version 4

5.6.1 PDF Exporting

Several PDF generation libraries were found for C#, but their API's were considered to be too complex to integrate in a short period of time, or their licenses were too restrictive to use without careful consideration. Using the author's experience of Web Development, the PDF's were created by first constructing a simple HTML document, and then a library was used to 'print' the document as a PDF. The chosen library was `HtmlRenderer.PdfSharp` [67].

Appendix B.2.4 shows the HTML generated by the `Exporter` class. Appendix F.5 shows an example PDF generated by the library. Listing 5.9 shows how easy it is to convert from HTML to PDF using the library.

Listing 5.9: PDF export code

```
try
{
  PdfDocument pdf = PdfGenerator.GeneratePdf(document.ToString(), PdfSharp.PageSize.A4);
  pdf.Save(filePath);
}
catch (IOException e)
{
  throw new Exception("Failed to export to PDF. Please check the file isn't already open in a PDF reader.");
}
```

5.7 Version 5

5.7.1 Macros

As discussed in Section 4.5, a simple scripting language was implemented. Each macro command type implements the `MacroCommand` interface, shown in Figure 5.19. Each command implements an `Execute` function. Within this function, the class has access to the current variables and can manipulate them. All variables are globally scoped, which means that any variables initialised within loops, still exist after the loop finishes. For the variable comparison command, the `Execute` function throws an `NotImplementedException` and requires a `Compare` function to be called instead.

Appendix B.1.2 shows how the class uses Regex to parse each line of a text file into an executable representation of the code. It uses a recursive method to parse the control structures; e.g. `for` loops and `if/else`'s. The current control structure is represented by a `BranchType` enum shown in Listing 5.10. Appendix B.1.6 shows the execution code for the macro runner.

Figure 5.20 shows how the results of a macro script are presented to the user. The values can be copied to the clipboard and then processed further using any required tools.

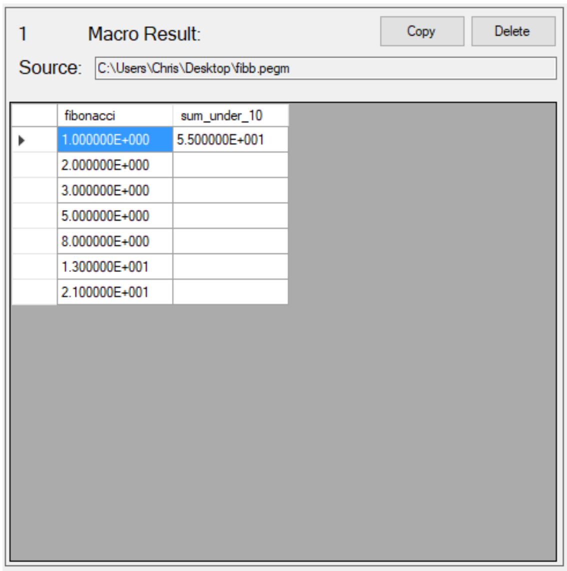
Figure 5.19: Macro command diagram

<<interface>> **MacroCommand**
lineNumber : int parent : MacroRunner
Execute (void) : void

Listing 5.10: BranchType enum

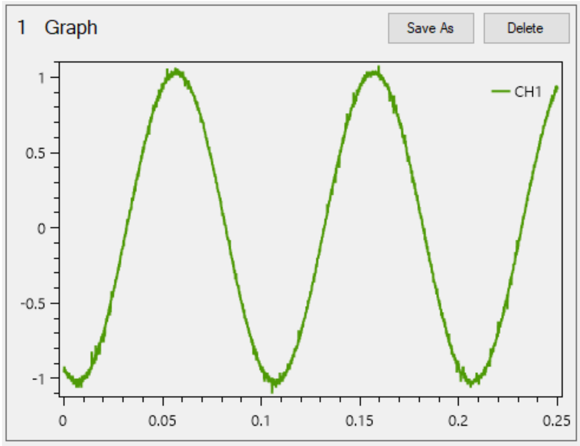
```
enum BranchType
{
  None,
  For,
  If,
  Else
}
```

Figure 5.20: Example macro response



5.7.2 Screen-shots Revisited

Figure 5.21: Example ‘Hardcopy’



As a solution to the problem of incomplete screen-shots from the oscilloscope, discussed in Section 5.5.4, it was decided to implement a new type of device response to show graphs. In order to facilitate the process of retrieving the raw graph data from the oscilloscope, a custom function would need to be called, instead of the normal `get`/`set` functions that had already been implemented. To allow custom functions, a new field was added to the `DeviceCommand` class to represent the name of the function and **reflection** was used to invoke the function on the required device class. Appendix B.1.7 shows how this was implemented.

Using the `CURVE` function for the TBS 1052B-EDU, the raw waveform data was able to be retrieved from the oscilloscope. Appendix B.1.5 demonstrates how the associated parameters are fetched and then using the following equations, taken from the documentation [7], to get the actual x, y coordinates where n counts through the samples.

$$x_n = x_{zero} + x_{incr} * (n - pt_{off}) \tag{5.1}$$

$$y_n = y_{zero} + (y_{mult} * (yraw_n - y_{off})) \tag{5.2}$$

To plot the data, it was decided to use Oxyplot [68], an open-source plotting library. Its API is very easy to use, and it also supports exporting to `PNG`, which is used for the PDF exporting.

Listing 5.11: Oxyplot example

```
var model = new PlotModel();
model.Axes.Add(new LinearAxis { Position = AxisPosition.Bottom });
model.Axes.Add(new LinearAxis { Position = AxisPosition.Left });
foreach (var plot in graphResponse.plots)
{
  var lineSeries = new LineSeries { Title = plot.title };
  for (int i = 0; i < plot.x.Length; i++)
  {
    lineSeries.Points.Add(new DataPoint(plot.x[i], plot.y[i]));
  }
  model.Series.Add(lineSeries);
}
```

6 Testing & Verification

After development finished, the created software package was tested to make sure it meets the specification set out in Chapter 2, and also meets the required accessibility guidelines discussed in Section 1.5. This chapter details the testing procedure taken.

Contents

6.1 Natural Language Processing

6.1.1 Accuracy

A repeat of the metric study (see Section 3.2) was conducted to ascertain the average success rate of Dialogflow after an extended amount of time. The accuracy was determined to be approx. 35%, and the average time to process the command as 2.71 seconds. This is a significantly lower success rate than previously calculated. The raw results can be found in Appendix F.4.2.

6.1.2 Response Time

As discussed in Section 1.5.2, a rating between 0 and 1 can be calculated for the response time for an application. During development, it was observed that there was no noticeable delay between the PC sending the command to the device, and the device executing the command. This leaves the response time of the NLP platform as the only observable delay. Using the results of the repeated metric study, a graph was plotted of the Apdex [30] rating over a range of T values, using Equation 1.1, alongside a cumulative plot of requests fulfilled in less than T seconds.

Figure 6.1: Response time plot

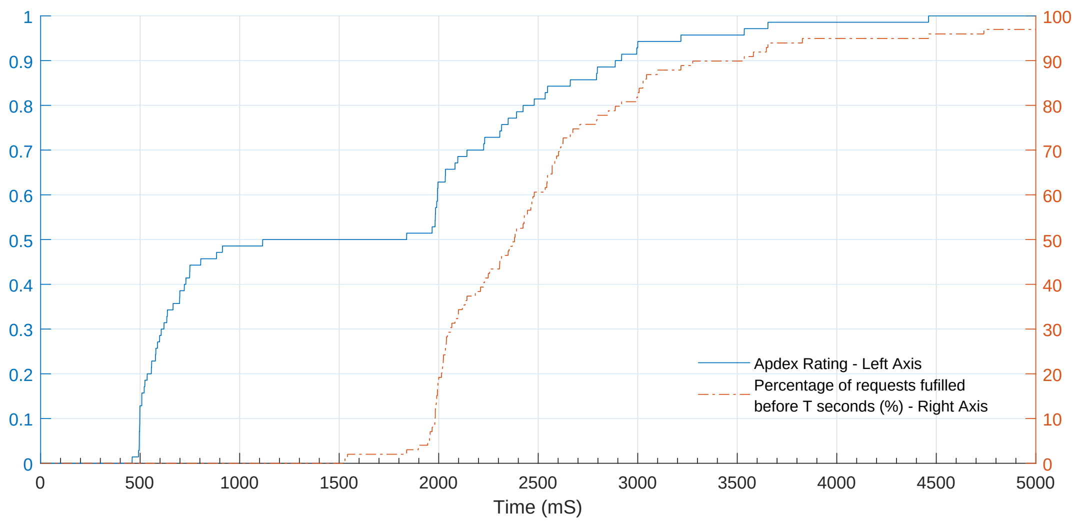
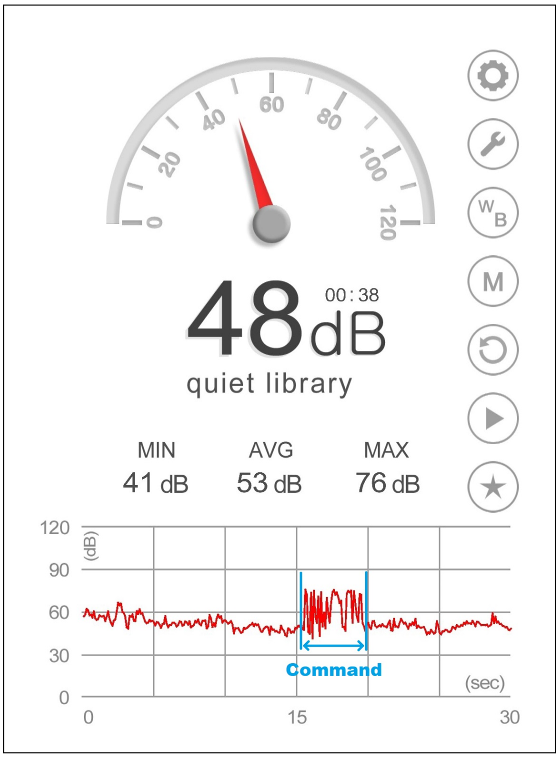


Figure 6.1 shows that if the expected time for the application to respond is 2 seconds, then the rating will be 0.6 and the fulfilment rate will be 20%, and most likely the user will be frustrated. However, if the expected response time is increased to 3 seconds, the rating jumps up to 0.9 and the fulfilment rate, 85%.

Figure 6.2: Background noise plot



6.1.3 Background Noise

To test whether the NLP platform could work in noisy environments, an audio level meter application on a mobile phone was used to record the background noise level while the software was being used. Figure 6.2 shows the command being spoken with a background noise level of 53 dB. The software was able to correctly parse the command without any issues.

6.2 Testing Implemented Commands

To test whether commands had been implemented correctly, a systematic test of each command was conducted. The results of these tests can be found in Appendix F.4.1. Any issues were noted and any major problems resolved.

6.3 Repeating "Wizard of Oz" Study

In order to verify whether enough commands had been implemented in the current system and whether the application was easy and intuitive to use, the experiment from Section 3.1 was repeated using the created software.

6.3.1 Methodology

Firstly, the researcher carried out the experiment to ascertain whether it was possible using the current set of implemented commands. Any missing commands were then implemented. Secondly, a participant, who had not taken part in the original experiment was asked to carry out the experiment. This was to indicate how intuitive the software is, and whether or not training would be required for any new users.

6.3.2 Results & Conclusions

Researcher's Results

In the author's opinion, the experiment was quite easy to carry out using the software. The resultant exported PDF can be seen in Appendix F.5. There were a few missing commands that hadn't been implemented, but these were easy to fix and presented no issues:

- Function Generator - Turn on/off channel
- Function Generator - Set waveform type
- Oscilloscope - Turn on/off channel

There will be bias present, as the author knows exactly how the software works and how to use it, but it has proven that simple experiments can be carried out using the created product.

Participant's Results

Start Listening It was not immediately obvious to the user that the 'Start Listening' button was to be used for giving commands. Instead, they thought it was constantly listening for commands once activated.

Listening State The user tried to click the 'Start Listening' button to cancel the recording which caused an invalid state for the recording class. This resulted in some very confusing error messages.

NLP Accuracy The NLP platform was able to understand the user's words but failed to parse the correct intent. This may be because the neural network powering the platform needs time and data to train to a new user. This is a major downfall of the platform and would need to be considered before the next release.

Similar Commands When the user tried to set the voltage output on the function generator, by using the command "Set channel one to 100 milli volts", the platform confused it for a command to set the voltage scale on the oscilloscope. This type of confusion could potentially be avoided by prefixing each command with the name of the device. E.g. "Function Generator: Set channel one to 100 milli volts".

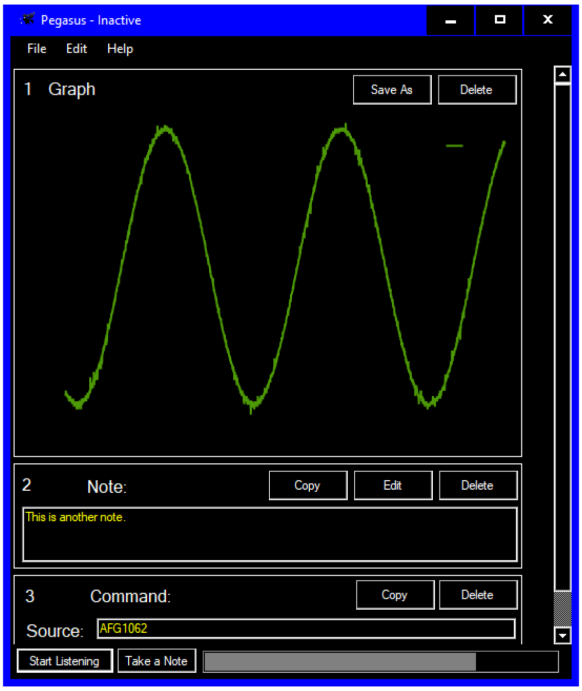
Measurement Command The command to take a measurement from the oscilloscope is able to do a wide range of measurements. This is not immediately obvious to the user. This could be fixed by having a larger range of example phrases for each command.

6.4 Accessibility

6.4.1 Accessibility Software

The following sections document the results of attempting to use the software with various accessibility tools.

Figure 6.3: 'Main Window' viewed on 'High Contrast' theme



High Contrast Themes

Figure 6.3 shows the main window when a high contrast theme is applied to Windows. Most of the UI changes accordingly, except the colour of the custom progress bar. It is possible to detect if a high contrast theme is currently in use on the user's PC and change the colours accordingly.

Narrator

The functionality of the software was mostly usable without the use of a screen and while using Microsoft Narrator.

Help The Help Viewer turned out to be fairly inaccessible. An alternative to this would be to have the HTML files stored locally and open them in the user's default browser when help is requested.

Buttons When the narrator reads the names of buttons out loud, it is difficult to understand the context surrounding it; for instance the 'Delete' button on the result components. This was made easier by setting the `AccessibleName` property of the buttons to, for example, "Measurement 1 delete".

Dialogs An issue was noted when trying to delete measurements. Microsoft Narrator wouldn't read out the confirmation text without needing to tell it to. This resulted in the user being asked 'Yes' or 'No' to an unknown question.

6.4.2 Simulated Impairments

The following sections document the results of attempting to use the software while simulating differing impairments.

Without a Mouse

All functions of the software are usable without a mouse apart from the following defects:

- Couldn't remove specific devices from the list in the options window. This can be fixed by adding a keyboard shortcut while focus is given to the device entry for the 'Delete' key.
- The Oxyplot component in the graph response can be given focus using the 'Tab' key. It seems to ignore the `TabStop` property which is set to false.

Without a Keyboard

All functions of the software work well without a keyboard. The note dialog is fairly useless, but the user could use an OSK as an alternative.

Without Audio

All functions of the software work well without any audio. Currently, the only audio used is for the state of the recording which is also conveyed using text and colour in the main window.

6.4.3 Accessibility Check-list

As discussed in Section 1.5.1, an assessment was made of the software's compatibility with the NDA's 'Accessibility Check-list'. Table 6.1 details the results.

Table 6.1: 'Accessibility Check-list'. Adapted from the NDA [28].

Requirement	Met	Comments
1.1 Ensure that users have access to the operating system accessibility tools, without affecting application functionality	✓	See Section 6.4.1
1.2 Ensure compatibility with assistive technologies	✓	See Section 6.4.1
1.3 Adhere to all user-selected system settings for input and output	✓	The software does not implement any custom input methods, nor does it use any custom fonts or display methods. See Section 6.4.1
1.4 Adhere to the standard keyboard access methods	✓	The software does not require any custom keyboard shortcuts, while also not impeding the default system shortcuts.
1.5 Do not require use of a pointing device	✓	See Section 6.4.2
1.6 Ensure that all information can be perceived by users with restricted or no vision	✓	See Section 6.4.1
1.7 Ensure that all information can be perceived by users with restricted or no hearing	✓	See Section 6.4.2
1.8 Do not cause the screen to flash at a frequency of above 2 Hertz	✓	The software features no rapid changes that would cause distress to any users who have epilepsy.
1.9 Use the simplest language possible for instructions, prompts and outputs and, where possible, supplement it with pictorial information or spoken language	✓	The software features no complicated language, other than the technical language required for the devices, that the user is assumed to know and understand. All text is able to be read out loud by a screen reader.
1.10 Ensure a logical tab order for controls, input fields and other objects	✓	See Section 6.4.2

Table 6.1 continued from previous page

Requirement	**Met**	**Comments**
1.11 Provide descriptions and instructions for all accessibility features	✓	The documentation provided alongside the software details the settings related to accessibility.
1.12 Provide accessible documentation, training and support materials	✓	Documentation for the software is provided in an accessible format, with the option to print into any format the user requires. Testing has shown that training would be required for a released product, which will need to be considered.
2.1 Allow sufficient response time to accommodate the slowest users	✓	The user is able to change the automatic silence detection time-out in the options. The hard limit for this is 20 seconds, as the NLP platform is only able to process up to 20 seconds of audio.
2.2 Ensure that the user interface and task flow is similar across different functions	✓	The differing functions of this software are the different components shown in the main window. Each of these components has roughly the same set of functions: Save/Export, Copy, Delete.
2.3 Adhere to the operating system user interface guidelines	✓	Design guidelines set out by Microsoft were followed throughout the project [60].
2.4 Provide accessible packaging, installation and configuration tools	n/a	The current implementation does not require installation. Future work will include creating an installation package and therefore will need to be considered.
2.5 Provide for users with multiple impairments	✓	The user is able to use any combination of accessibility tools they require.

7 Conclusions & Future Work

This project has produced a prototype software package that has been thoroughly tested and shows that a viable product could be further developed. This chapter details potential future work and discusses a conclusion for the project.

Contents

7.1 Future Work

This section details some future work to improve the application and create a fully functional system that could be released into the market.

7.1.1 Training NLP Platform

Using the Dialogflow API, it would be possible to create an automated training suite for the NLP. This would allow users to increase the accuracy of the speech recognition. It would involve the user reading example phrases into the microphone, with the software correcting the platform in the background.

7.1.2 Publishing

In order to distribute the software and administer any updates, an installer will need to be created. Visual Studio includes a tool-chain to bundle the compiled code, along with any resources, into a single executable file that the user can use to install the software.

7.1.3 UI Improvements

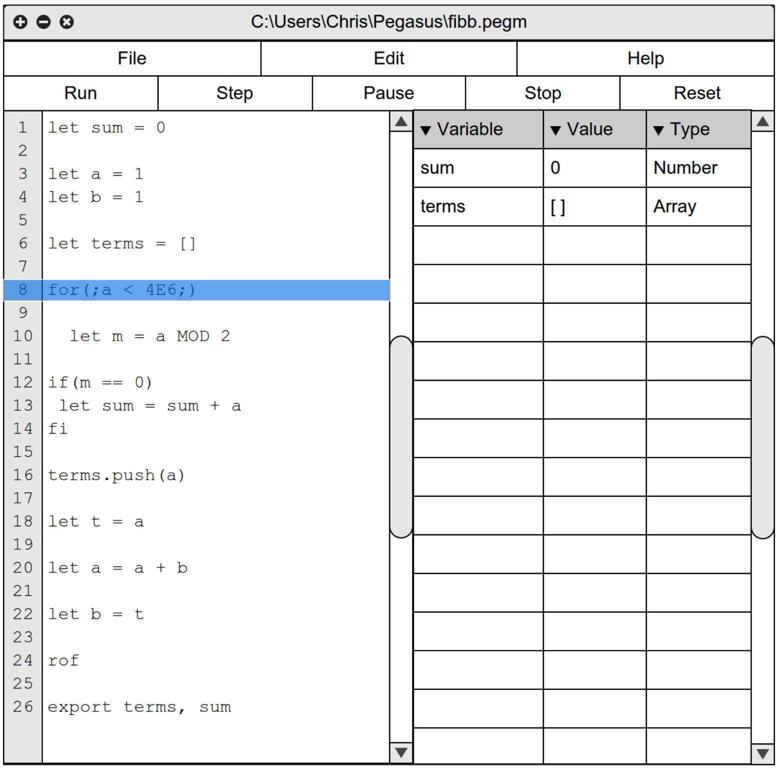
A major improvement for the application would be an overhaul of the UI elements. Although good design practices have been applied, the rapid development cycle has left minor discrepancies, for instance, between the element positions within `ComponentMeasurement` and `ComponentCommand` classes. A task for future development would be to recreate the classes using XAML [69] instead of the Visual Studio UI Creator.

As discussed in Section 5.3.2, the implementation of the dynamic elements is less than satisfactory. Before the next major release, this portion of the UI would need to be considered and fixed.

7.1.4 Macros

In order to allow a user to write more complex macro scripts than the examples already listed, the full functionality of the devices will need to be documented, including all valid parameters. To aid in the development of these scripts, a simple IDE with a debugger could be written to allow the developer to step through the script and see the current variables etc. Figure 7.1 shows a simple mockup for this.

Figure 7.1: IDE mockup



7.1.5 Cross Platform

With a view to making the software work on non-Windows OS's, there are a few different solutions. Using the Mono platform [70], an open-source .NET framework, to run or compile executables on Linux and macOS may work, although there are still severe compatibility issues, such as the lack of WPF [71].

Another solution would be to rewrite the application in a cross-platform language, such as Java, discussed in Section 1.8. The disadvantage to this would be the loss of built-in Windows functionality; such as easy to manage user preferences (see Section 5.4.1), and the keyword activation (see Section 5.2.1). National Instruments (NI) provide a Linux version of their VISA [72] which may be used as a replacement for the TekVISA discussed in Section 5.2.4.

7.1.6 Devices

Although the Tektronix devices integrated in this project are able to be communicated with via the TekVISA and implement SCPI, further work would need to include integrating with a wider range of devices. This could be done on a per-demand basis.

In order to make it easier to develop for a broader range of devices, the implementation of each class could be written as a Dynamic-link library (DLL) and then loaded at runtime into the application.

7.2 Conclusions

The intent of this project was to look at developing an accessible interface for electronic lab test equipment. The software package created meets the aims and objectives discussed in Chapter 2. There is still a lot of room for improvement with the software, although it has clearly shown that more devices could be implemented without much work.

The integration with the TBS 1052B-EDU oscilloscope and AFG1062 function generator was made simple by the extensive documentation provided by Tektronix. The author made use of their TekVisa software extensively during the debugging period.

Creating the macro language to run within the software has also shown that it could be used to make circuit testing a lot quicker in the labs. For repetitive circuit analysis procedures, a simple script could be written that would systematically take measurements without any human error.

In the author's opinion, this project went really well. There were no major setbacks that halted development and feedback has been very positive.

Appendix

A Lab Equipment

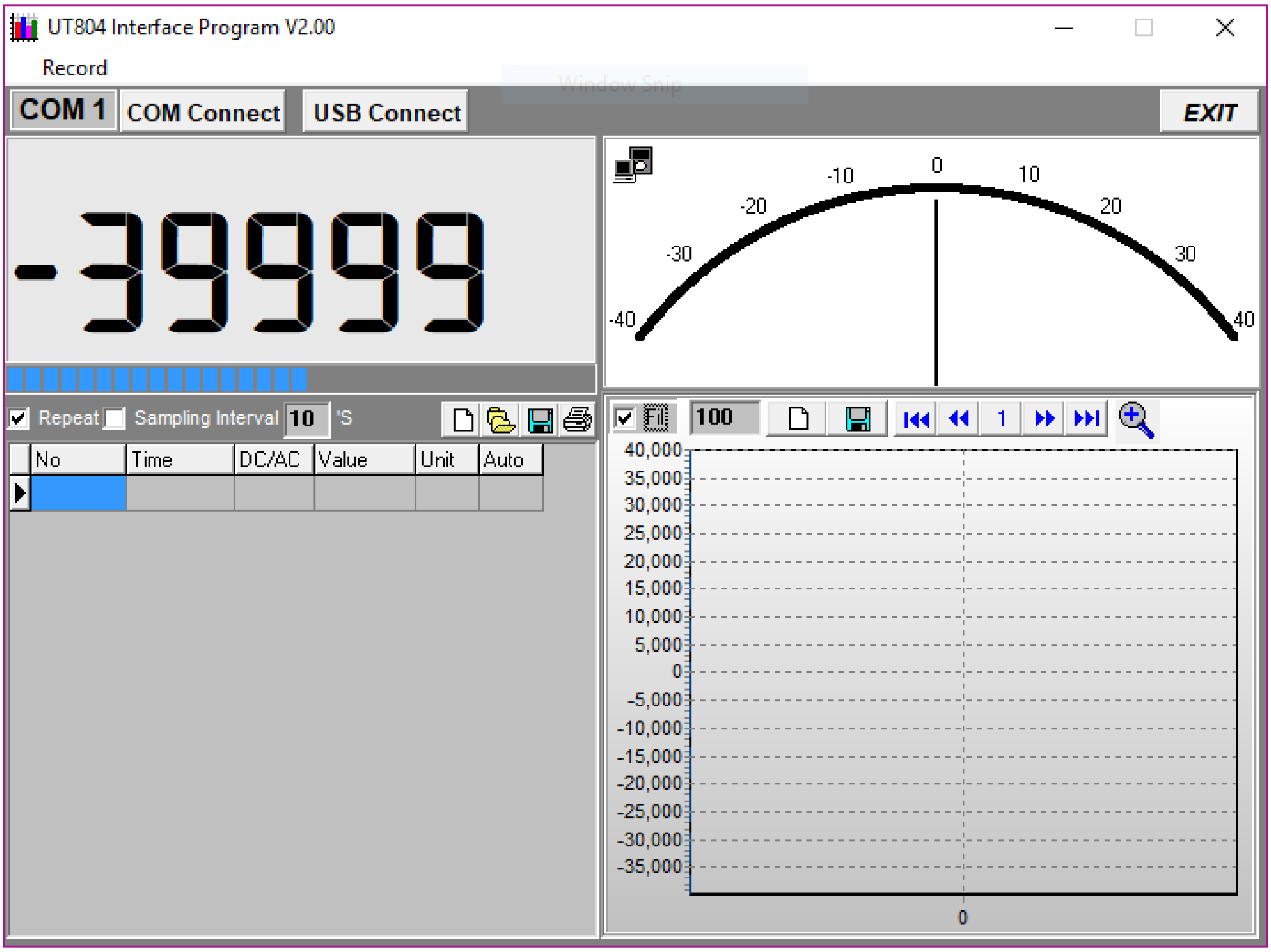
A.1 VocalLink Voice Control Software Commands for Tektronix Oscilloscopes

Table A.1: 'VocalLink', voice-control software commands for Tektronix oscilloscopes. Adapted from Tektronix site [11].

Function	**Commands**
Voltage	Voltage, Scale, Up/Down
Timebase	Get timebase, Timebase, Up/Down, Get Timebase Settings, Increase/Decrease, Record Length
Position	Position Up, Position Down, Position Zero
Trigger	Get Trigger Settings, Get Trigger Level, Get Trigger State, Trigger Up/Down/Zero, Trigger Level 50%, Trigger Level, Trigger Slope Rise/Fall, Trigger Mode Auto/Normal, Force Trigger, Trigger Source Channel, Trigger Coupling AC/DC, Trigger Coupling High/Low Frequency, Trigger Coupling Noise Rej, Trigger Type Edge, Trigger Level ECL/TTL
Channel	Get Channel Settings, Get Channel, Show Channel, Hide Channel, Select Channel, Channel Offset, Coupling AC/DC/GND. Channel Bandwidth, Channel Impedance, Channel Offset Up/Down
Measure	Measure/display: P-P, frequency, mean, period, RMS, rise/fall time, positive/negative pulse width, maximum/minimum, duty cycle, snapshot, remove all measurements, Freerun, stop
Acquire	Get acquire settings, acquire mode sample/peak det/hi res, acquire mode average/envelope, stop after button/single acquisition, repetitive signal on/off
Macros	Execute test, go to line, if result at line condition go to line, play prompt, continue
Miscellaneous	Autoset, clear menu, hardcopy, save wave, go to sleep, raise/lower interface, Calibrate, initialize

A.2 UT804 Interface Program V2.00

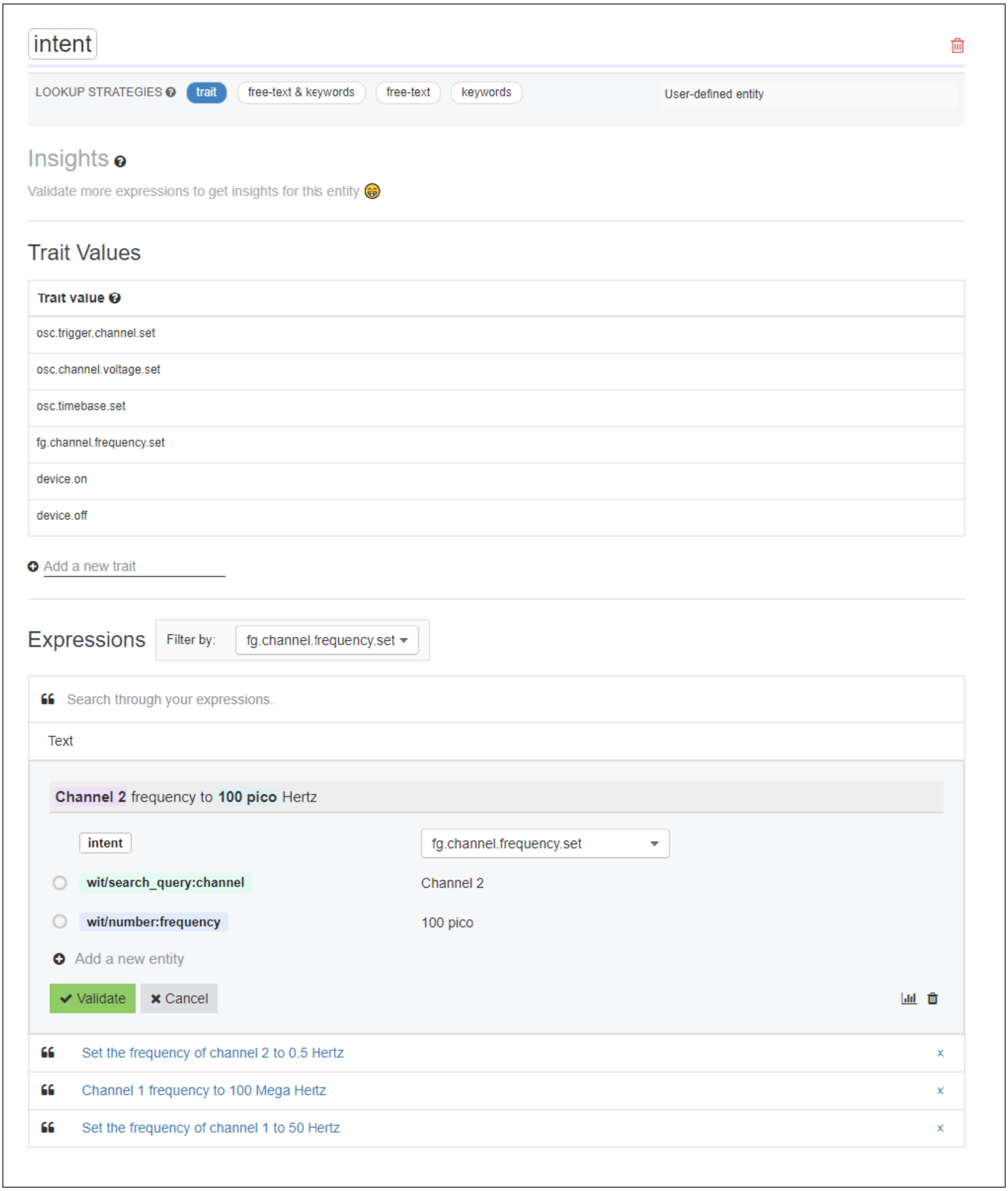
Figure A.1: UT804 Interface Program V2.00 [10]



A.3 NLP Platforms

A.3.1 Wit.ai Intent Interface

Figure A.2: Wit.ai intent interface [41]

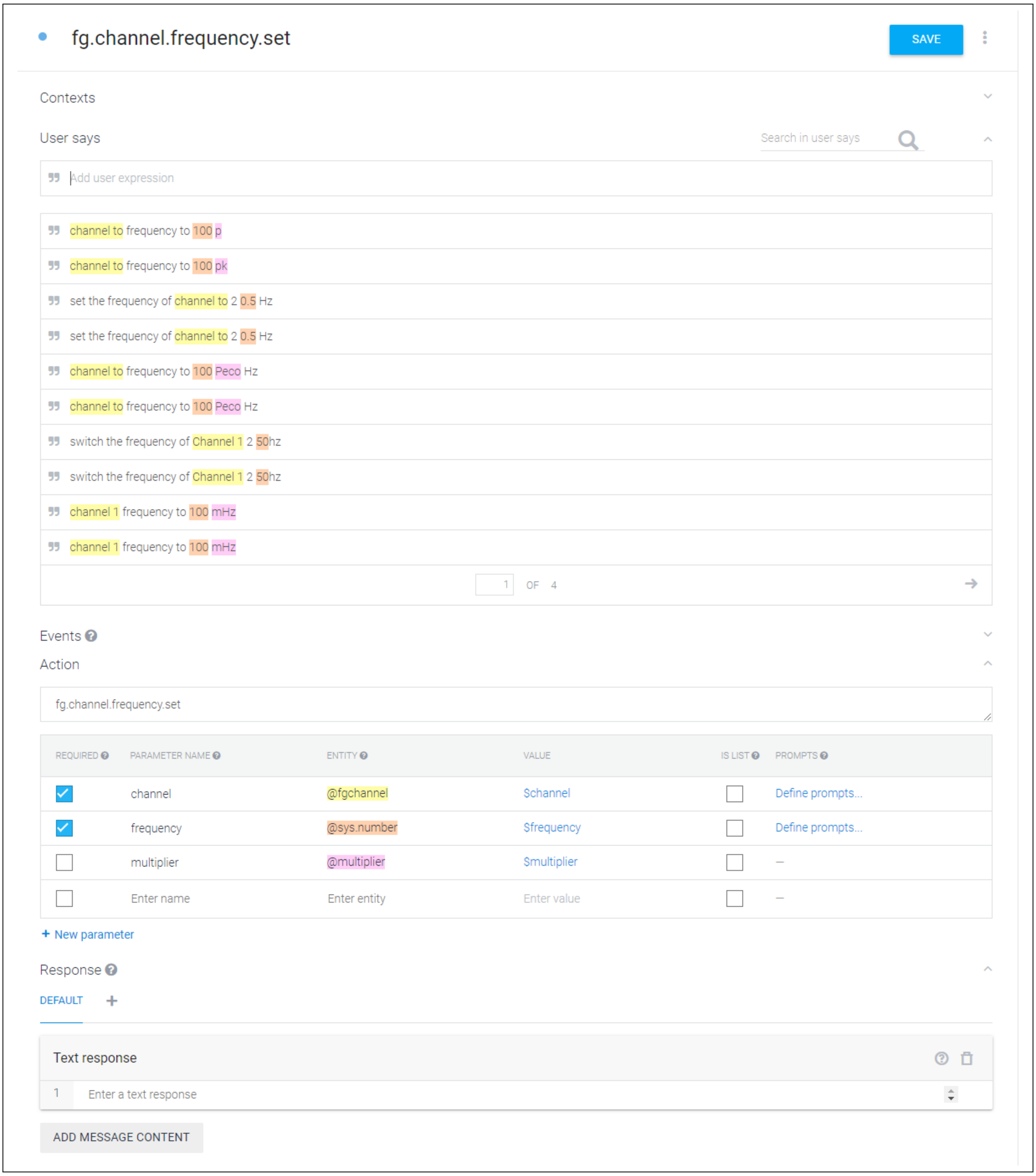


A.3.2 Wit.ai Example API Response


```
[{
  "entities": {
    "channel": [{
      "role": "channel",
      "confidence": 0.95298081850845,
      "start": 25,
      "end": 36,
      "body": "channel two",
      "value": {
        "value": "channel two"
      },
      "entity": "search_query"
    }],
    "intent": [{
      "confidence": 0.95284045601596,
      "value": {
        "value": "osc.channel.voltage.set"
      },
      "entity": "intent"
    }],
    "voltage": [{
      "role": "voltage",
      "confidence": 0.79697377636283,
      "start": 40,
      "end": 44,
      "body": "five",
      "value": {
        "value": 5
      },
      "entity": "number"
    }]
  },
  "confidence": null,
  "_text": "set the voltage scale of channel two to five",
  "intent": "default_intent",
  "intent_id": "default_intent_id"
}]
```

A.3.3 Dialogflow Intent Interface

Figure A.3: Dialogflow intent interface [43]

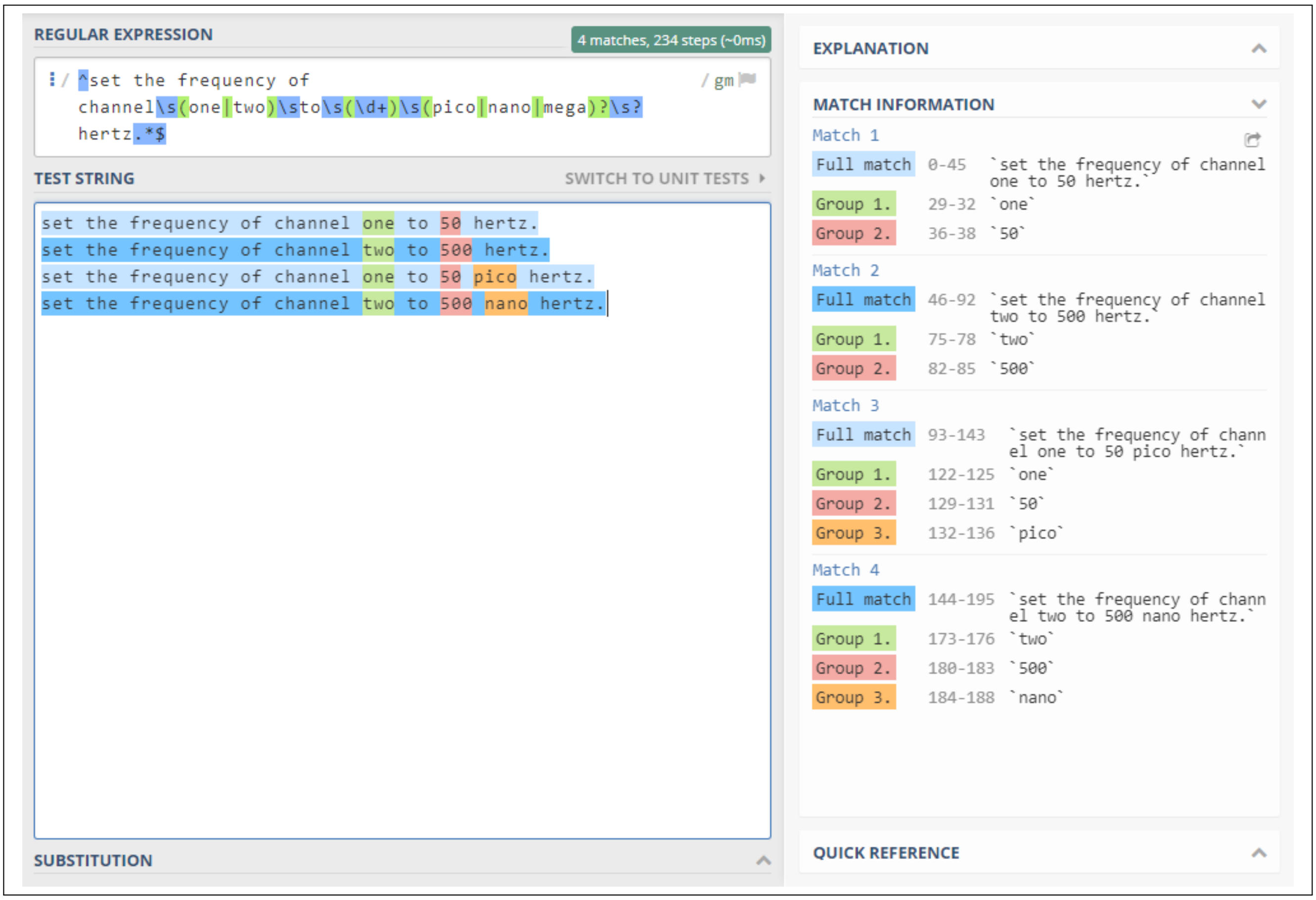


A.3.4 Dialogflow API Example Response


```json
{
  "responseId": "8a1721db-ff0c-4151-97b6-1bb73fc25855",
  "queryResult": {
    "queryText": "channel two frequency to 100 pico hertz",
    "action": "fg.channel.frequency.set",
    "parameters": {
      "multiplier": "PICO",
      "channel": "CH2",
      "frequency": 100
    },
    "allRequiredParamsPresent": true,
    "fulfillmentMessages": [
      {
        "text": {
          "text": [
            ""
          ]
        }
      }
    ],
    "intent": {
      "name": "projects/newagent-f6288/agent/intents/3a8e6817-54bf-4f80-a18a-eac377cdf41f",
      "displayName": "fg.channel.frequency.set"
    },
    "intentDetectionConfidence": 1,
    "diagnosticInfo": {},
    "languageCode": "en"
  }
}
```

A.3.5 Regex101

Figure A.4: Regex101, an online Regex visualiser [73].



B Code Snippets

B.1 C#

B.1.1 Timing Code Snippet

```
List<TestResult> results = new List<TestResult>();

var diawatch = System.Diagnostics.Stopwatch.StartNew();

var dia = await DialogFlow.postAudio(recording);

diawatch.Stop();

var diawatchMs = diawatch.ElapsedMilliseconds;

var witwatch = System.Diagnostics.Stopwatch.StartNew();

var wit = await Wit.postAudio(recording);

witwatch.Stop();

var witwatchMs = witwatch.ElapsedMilliseconds;

TestResult witResult = Wit.processResponse(wit);

witResult.expectedText = expectedPhrase;
witResult.timeTaken = witwatchMs;

TestResult diaResult = DialogFlow.processResponse(dia);

diaResult.expectedText = expectedPhrase;
diaResult.timeTaken = diawatchMs;

results.Add(witResult);
results.Add(diaResult);
```

B.1.2 Macro Runner Parsing Function

```
private void ParseLines(
  String[] lines,
  int startIndex, out int endIndex,
  out List<MacroCommand> commands,
  out Dictionary<int, SyntaxException> errors,
  BranchType intype, out BranchType outtype
  )
{
  outtype = BranchType.None;
  endIndex = -1;
  errors = new Dictionary<int, SyntaxException>();
  commands = new List<MacroCommand>();
  for (int curr = startIndex; curr < lines.Length; curr++)
  {
    String line = lines[curr];
    try
    {
      if (commentRegex.IsMatch(line))
      {
        // Nothing
      }
      else if (VariableAssignment.IsMatch(line))
      {
        commands.Add(new VariableAssignment(this, curr + 1, line));
      }
      else if (FunctionCall.IsMatch(line))
      {
        commands.Add(new FunctionCall(this, curr + 1, line));
      }
      else if (VariableOperator.IsMatch(line))
      {
        commands.Add(new VariableOperator(this, curr + 1, line));
      }
      else if (ArrayOperator.IsMatch(line))
      {
        commands.Add(new ArrayOperator(this, curr + 1, line));
      }
      else if (ExportVariable.IsMatch(line))
      {
        commands.Add(new ExportVariable(this, curr + 1, line));
      }
      else if (VariableIncDec.IsMatch(line))
      {
        commands.Add(new VariableIncDec(this, curr + 1, line));
      }
      else if (ForLoop.IsMatch(line))
      {
        List<MacroCommand> inner;
        BranchType endType;
        this.ParseLines(lines, curr + 1, out curr, out inner, out errors,
   BranchType.For, out endType);
        commands.Add(new ForLoop(this, line, curr + 1, inner));
      }
      else if (ForLoop.endRegex.IsMatch(line))
      {
        outtype = BranchType.None;
        endIndex = curr;
        return;
```

```
            }
            else if (IfBranch.IsMatch(line))
            {
                List<MacroCommand> truelines;
                BranchType endType;
                this.ParseLines(lines, curr + 1, out curr, out truelines, out errors
        , BranchType.For, out endType);
                if (endType == BranchType.None)
                {
                    commands.Add(new IfBranch(this, line, curr + 1, truelines, new
        List<MacroCommand>()));
                }
                else if (endType == BranchType.Else)
                {
                    List<MacroCommand> falselines;
                    this.ParseLines(lines, curr + 1, out curr, out falselines, out
        errors, BranchType.For, out endType);
                    commands.Add(new IfBranch(this, line, curr + 1, truelines,
        falselines));
                }
            }
            else if (IfBranch.elseRegex.IsMatch(line))
            {
                outtype = BranchType.Else;
                endIndex = curr;
                return;
            }
            else if (IfBranch.endRegex.IsMatch(line))
            {
                outtype = BranchType.None;
                endIndex = curr;
                return;
            }
            else
            {
                // Blank line
            }
        }
        catch (SyntaxException s)
        {
            errors.Add(curr + 1, s);
        }
    }
}
```

B.1.3 FormSettings Class

```
namespace Pegasus
{
  public partial class FormSettings : Form
  {

    ...

    private void FormOptions_Load(object sender, EventArgs e)
    {
      Properties.Settings.Default.Reload();
    }

    private void buttonCancel_Click(object sender, EventArgs e)
    {
      Properties.Settings.Default.Reload();
      this.Close();
    }

    private void buttonOK_Click(object sender, EventArgs e)
    {
      Properties.Settings.Default.Save();
      this.Close();
    }

    private void buttonDefault_Click(object sender, EventArgs e)
    {
      Properties.Settings.Default.Reset();
      this.Refresh();
    }

    private void FormSettings_HelpButtonClicked(object sender,
   CancelEventArgs e)
    {

      Help.ShowHelp(this, @"./resources/Pegasus.chm", HelpNavigator.Topic, "
   Settings.html");

    }

    ...

  }
}
```

B.1.4 Microphone Level Monitor

```
private void InitializeMicrophoneMonitor()
{

  int microphoneLevel = 0;

  int min = Math.Abs(Properties.Settings.Default.MinSilenceThreshold);

  Timer timer = new Timer
  {
    Interval = 10
  };

  try
  {

    timer.Tick += (object sender, EventArgs e) =>
    {
      // Instead of just setting the value to the new level
      // this gives a smooth-ish transition to the new value.
      progressBarSilence.Value += (int)((microphoneLevel -
   progressBarSilence.Value) / 10.0);
    };

    WaveInEvent waveIn = new WaveInEvent
    {
      BufferMilliseconds = 50
    };
    waveIn.DataAvailable += (s, a) =>
    {
      var peak = Recorder.AverageSample(a);
      double transformed = Math.Max(0, peak + min);
      transformed /= (double)min;
      transformed *= progressBarSilence.Maximum;
      microphoneLevel = (int)transformed;
    };

    waveIn.StartRecording();
    timer.Start();

  }
  catch (Exception e)
  {
    timer.Stop();
    Logger.Error(e);
    MessageBox.Show("Please make sure your microphone is plugged in and is
   working correctly.", Properties.Resources.Name);
  }

}
```

B.1.5 TBS 1052B-EDU Device Class

```
using System;
using System.Collections.Generic;

namespace Pegasus.devices
{

  class TBS1052BEDU : Tektronix
  {

    public TBS1052BEDU(Dictionary<String, Object> variables) : base(variables, "resources.TBS1052B-EDU.xml")
    {

    }

    public override String GetFormattedName()
    {
      return "TBS 1052B-EDU";
    }

    private List<String> GetVisibleWaveforms()
    {

      List<String> waveforms = new List<String>();

      if (!TVA.Write("SEL?"))
      {
        throw new Exception("Failed to write command");
      }

      String raw;  //:SELECT:FFT 0;CH1 1;CH2 1;MATH 0;REFA 0;REFB 0

      if (!TVA.Read(out raw))
      {
        throw new Exception("Failed to read command");
      }

      if (raw.Length <= 0)
      {
        throw new Exception("Invalid response");
      }

      foreach (var chn in raw.Remove(0, 8).Split(';'))
      {
        var chanspl = chn.Split(' ');

        if (chanspl[1] == "1")
        {
          waveforms.Add(chanspl[0]);
        }

      }

      return waveforms;

    }

```

```
    public DeviceResponse hardcopy(DeviceCommand cmd, Dictionary<String,
    String> variables)
    {

      try
      {

        List<GraphPlot> plots = new List<GraphPlot>();

        var waveforms = GetVisibleWaveforms();

        foreach (var wave in waveforms)
        {

          double ymult; // :WFMPRE: YMULT 7.8125E-5
          double yzero; // :WFMPRE:YZERO 0.0E0
          double yoff; // :WFMPRE:YOFF 0.0E0
          double xincr; // :WFMPRE:XINCR 1.0E-4

          String response;

          TVA.Write("DATA:SOU " + wave);
          TVA.Write("DATA:WIDTH 2");
          TVA.Write("DATA:ENC ASCII");

          TVA.Query("WFMPRE:YMULT?", out response);
          ymult = double.Parse(response.Substring(":WFMPRE:YMULT ".Length));

          TVA.Query("WFMPRE:YZERO?", out response);
          yzero = double.Parse(response.Substring(":WFMPRE:YZERO ".Length));

          TVA.Query("WFMPRE:YOFF?", out response);
          yoff = double.Parse(response.Substring(":WFMPRE:YOFF ".Length));

          TVA.Query("WFMPRE:XINCR?", out response);
          xincr = double.Parse(response.Substring(":WFMPRE:XINCR ".Length));

          String rawwave;

          TVA.Write("CURVE?");
          TVA.Read(out rawwave);

          String[] split = rawwave.Substring(":CURVE ".Length).Split(',');

          double[] xvalues = new double[split.Length];
          double[] yvalues = new double[split.Length];

          // x_n = x_zero + x_incr ( n - pt_off )
          // y_n = y_zero + ( y_mult * ( yraw_n - y_off ) )
          for (int i = 0; i < split.Length; i++)
          {
            yvalues[i] = yzero + (double.Parse(split[i]) * ymult);
            xvalues[i] = i * xincr;
          }

          GraphPlot gp = new GraphPlot
          {
            title = wave,
            x = xvalues,
            y = yvalues,
```

```
          xunit = "Seconds",
          yunit = "Volts"
        };

        plots.Add(gp);

      }

      return new GraphResponse
      {
        device = this.GetFormattedName(),
        message = "SUCCESS",
        success = true,
        confidence = 1,
        formattedCommand = "Take a hardcopy.",
        plots = plots
      };

    }
    catch (Exception e)
    {
      Logger.Error(e);
      throw e;
    }

  }

 }
}
```

B.1.6 MacroRunner Execution Code

```
public MacroResponse Execute()
{
  this.currentCommand = 0;
  this.exported = new List<MacroKeyValuePair>();
  this.variables = new Dictionary<string, MacroVariable>();

  int currLine = 0;

  try
  {
    foreach (var line in this.commands)
    {
      currLine = line.lineNumber;
      line.Execute();
    }
    return new MacroResponse
    {
      fileName = this.filename,
      success = true,
      exported = this.exported
    };
  }
  catch (RuntimeException e)
  {
    return new MacroResponse
    {
      fileName = this.filename,
      success = false,
      message = String.Format("RuntimeException on line {0}: {1}", currLine,
    e.Message)
    };
  }
}
```

B.1.7 Custom Device Functionality Code

```
private DeviceResponse CallCustomFunction(String intent, Dictionary<String,
    String> variables)
{

  DeviceCommand cmd;

  if (!commands.TryGetValue(intent, out cmd))
  {
    throw new Exception("Failed to find intent");
  }

  Type type;

  if (this.GetType() == typeof(TBS1052BEDU))
  {
    type = typeof(TBS1052BEDU);
  }
  else if (this.GetType() == typeof(AFG1062))
  {
    type = typeof(AFG1062);
  }
  else
  {
    throw new Exception("Invalid device.");
  }

  MethodInfo method = type.GetMethod(cmd.customFunction);

  if (method == null)
  {
    throw new Exception(String.Format("Invalid custom function '{0}' on type
    {1}", cmd.customFunction, type.ToString()));
  }

  return (DeviceResponse)method.Invoke(this, new object[] { cmd, variables
   });

}
```

B.2 Other Languages

B.2.1 Example Settings XML

```xml
<settings>
  <folder>C:/Users/User/Documents/Pegasus</folder>
  <silence>
    <threshold>0.6</threshold>
    <duration>5000</duration>
  </silence>
  <devices>
    <device>
      <model>tektronix-oscilloscope</model>
      <connection>
        <type>Serial</type>
        <baud>5000</baud>
        <port>COM5</port>
      </connection>
    </device>
    <device>
      <model>tektronix-signalgenerator</model>
      <connection>
        <type>Serial</type>
        <baud>5000</baud>
        <port>COM6</port>
      </connection>
    </device>
  </devices>
</settings>
```

B.2.2 Example Device XML

```xml
<?xml version="1.0" encoding="utf-8" ?>
<device>
  <id>TBS 1052B-EDU</id>
  <name>Tektronix TBAS 1052B-EDU Oscilloscope</name>
  <command_groups>

      <command_group>
      <name>Averaging</name>
      <description>Commands relating to averaging.</description>
      <commands>

        <command id="osc.averaging.set">
          <description>Sets the averaging level.</description>
          <example>Set the averaging to times 4.</example>
          <format>ACQ:MOD AVE; ACQ:NUM {oscaveraging}</format>
          <confirm>false</confirm>
        </command>

      </commands>
    </command_group>

    <command_group>
      <name>Triggering</name>
      <description>Commands relating to triggers.</description>
      <commands>

        <command id="osc.trigger.channel.set">
          <description>Sets the trigger channel.</description>
          <example>Set the trigger to channel one.</example>
          <format>TRIG:MAI:EDGE:SOU {osctriggerchannel}</format>
          <confirm>false</confirm>
        </command>

      </commands>
    </command_group>

  </command_groups>
 </device>
```

B.2.3 Bode Plot Macro Example

```
% COMMENT
%
%
LET FREQS = [] % array
LET GAINS = [] % array
LET PHASES = [] % array

% Cheating because we know the low frequency gain
let inpk = 500E-3 / 2
let outpk = 10

osc.channel.voltage.set(oscchannel = "CH1", number = inpk)
osc.channel.voltage.set(oscchannel = "CH2", number = outpk)

osc.channel.voltage.offset.set(oscchannel = "CH1", number = 0)
osc.channel.voltage.offset.set(oscchannel = "CH2", number = 0)

osc.horizontal.position.set(number = 0)

osc.trigger.channel.set(osctriggerchannel = "EXT")

FOR(LET FREQ = 10E0; FREQ <= 10E6; LET FREQ = FREQ * 10)

  LET period = 0.5 / FREQ

  fg.channel.frequency.set(channel = "SOURCE1", frequency = FREQ)
  osc.timebase.set(number = period)

  % Make sure we get a clear value for out
  LET OUT = osc.measure.get(measurement = "MAX", channel = "CH2")
  let outpk = OUT / 2
  osc.channel.voltage.set(oscchannel = "CH2", number = outpk)

  LET IN = osc.measure.get(measurement = "pk2", channel = "CH1")
  LET OUT = osc.measure.get(measurement = "pk2", channel = "CH2")
  LET PHASE = osc.measure.get(measurement = "phase", channel = "CH1",
   channeltwo = "CH2")

  LET GAIN = OUT / IN

  FREQS.PUSH(FREQ)
  GAINS.PUSH(GAIN)
  PHASES.PUSH(PHASE)

ROF

EXPORT FREQS, GAINS, PHASES
```

B.2.4 Exporter HTML Example

```
<!doctype html><html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1,
   shrink-to-fit=no">
    <title>Pegasus Export</title>
</head>
<body>
  <h2>Export Details</h2>
  <p>
    <strong>Version:</strong> 0.5.0<br/>
    <strong>Devices:</strong>
    <ul>
      <li>Tektronix AFG1062</li>
      <li>Tektronix TBS 1052B-EDU</li>
    </ul>
  </p>
  <h2>Commands</h2>
  <p>
    <strong>1. Command to AFG1062:</strong> Set the frequency for SOURCE2 to
    100.001 pHz.<br/>
    <div style="text-align:right;">05/04/2018 12:19</div>
  </p>
  <p>
    <strong>2. Note:</strong><br/>
    <p>
      This is a note.
    </p>
    <div style="text-align:right;">05/04/2018 12:19</div>
  </p>
</body>
</html>
```

B.2.5 Example Session

```xml
<?xml version="1.0"?>
<Session xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <version>0.5.0</version>
  <time>2018-04-11T15:02:26.5079825+01:00</time>
  <devices>
    <ConnectedDevice>
      <type>Function Generator</type>
      <model>AFG1062</model>
      <connection>USB</connection>
    </ConnectedDevice>
    <ConnectedDevice>
      <type>Oscilloscope</type>
      <model>TBS1052BEDU</model>
      <connection>USB</connection>
    </ConnectedDevice>
  </devices>
  <commands>
    <DeviceResponse xsi:type="CommandResponse">
      <timestamp>2018-04-11T15:02:36.9428502+01:00</timestamp>
      <device>AFG1062</device>
      <confidence>0</confidence>
      <formattedCommand>Set the frequency for SOURCE2 to 100.001 pHz.</formattedCommand>
      <message>Set the frequency for SOURCE2 to 100.001 pHz.</message>
      <success>true</success>
    </DeviceResponse>
  </commands>
  <notes>
    <Note>
      <timestamp>2018-04-11T15:02:43.7457304+01:00</timestamp>
      <content>This is a note.

It has some characters in it that will need to be escaped in XML

&lt; &gt; </content>
    </Note>
  </notes>
  <macros />
</Session>
```

C User Stories

C.1 Proof of Concept Release

User Stories

Total: 18 User Stories • 4.10 Hours Worked • 70.29 Hours Remaining

Filtered by Selected: **Release** Group By: **Release** Sort By: **ID**

Workflow Columns: **[None]**

BEng Individual Project\Final Report\Proof 18 Items, 4.10 Hours Worked, **70.29 Hours Remaining**, 5.5% Complete

29
Write up Method
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
0.5 hrs remaining

31 (5)
Do experiment
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
30.79 hrs remaining

32
Contact participants
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
0.29 hrs remaining

34
Book lab space
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
0.5 hrs remaining

35
Set up experiment
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
1 hrs remaining

36
Record results
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
5 hrs remaining

37
Transcribe results
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
8 hrs remaining

33
Write participant instructions
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
0 hrs remaining

38
Write up Results
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
3 hrs remaining

39
Write up Conclusions
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
1 hrs remaining

41
Write up Method
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
3 hrs remaining

42
Write up Results
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
4 hrs remaining

43
Write up Conclusions
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
1 hrs remaining

44
Do experiment
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
16 hrs remaining

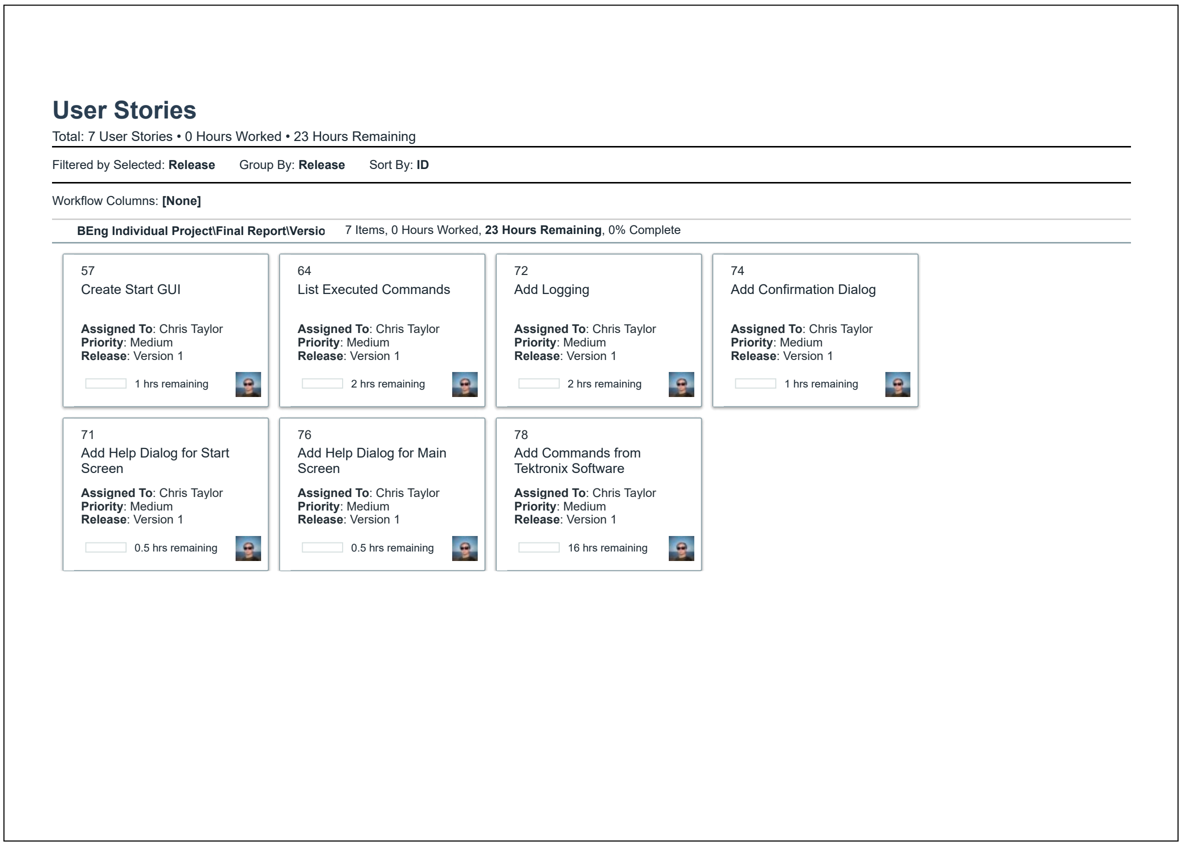
70
Listen for Keyword
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
1 hrs remaining

69
Communicate with Online Platform
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
3 hrs remaining

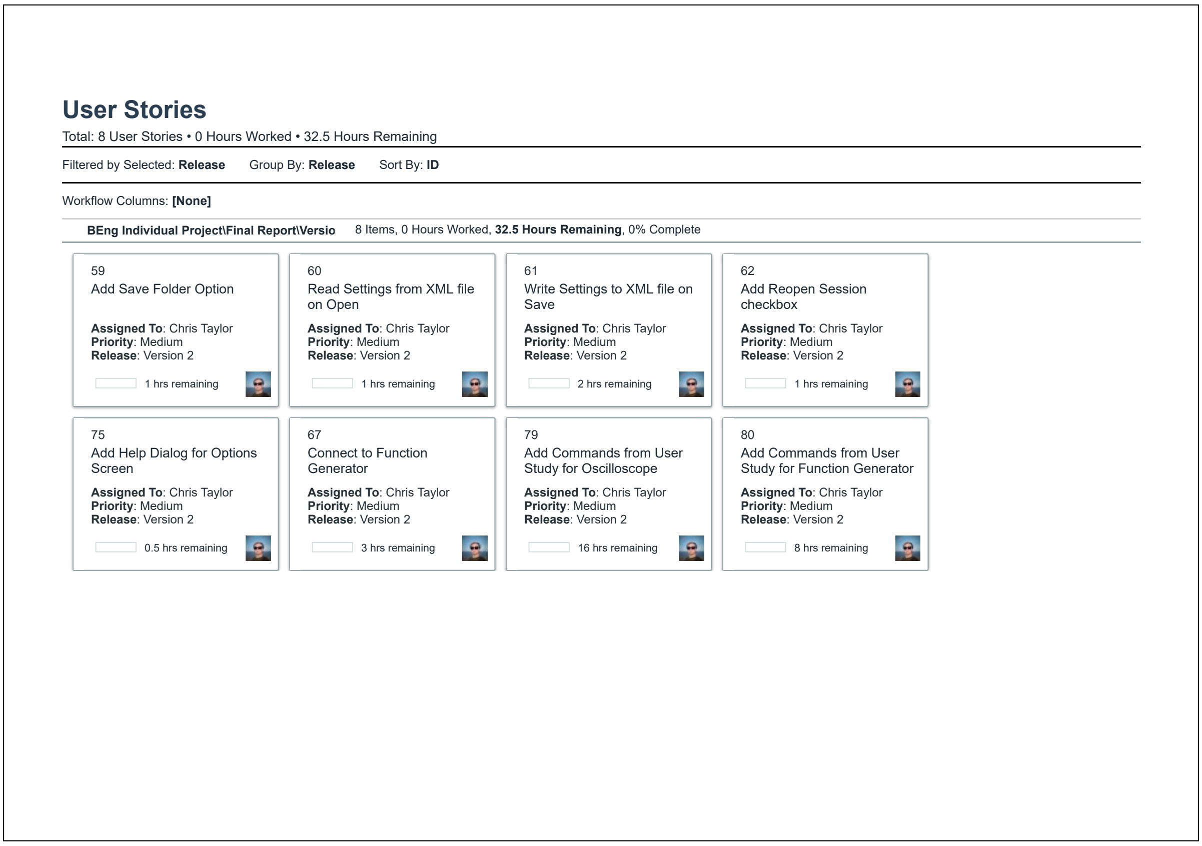
66
Connect to Oscilloscope
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
5 hrs remaining

82
Add Commands for Proof of Concept
Assigned To: Chris Taylor
Priority: Medium
Release: Proof of Concept
2 hrs remaining

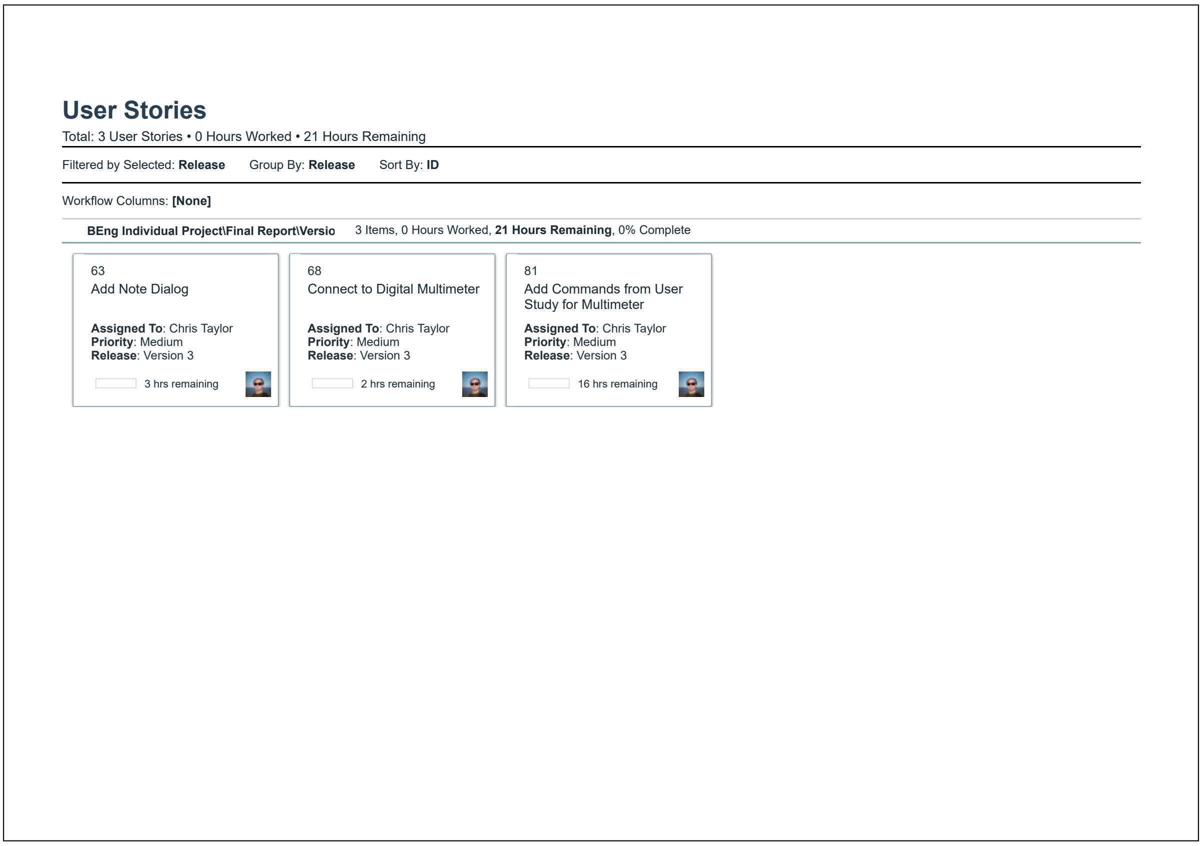
C.2 Version 1



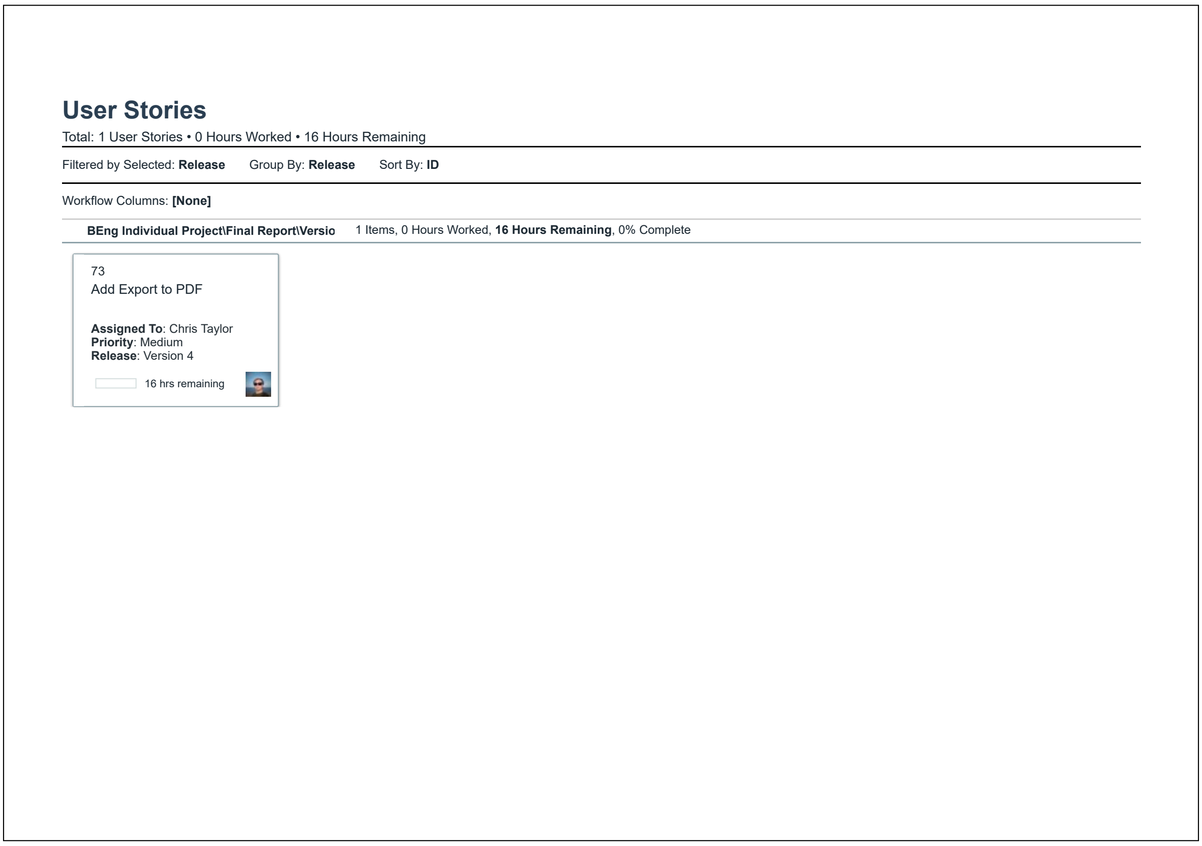
C.3 Version 2



C.4 Version 3



C.5 Version 4



D "Wizard of Oz" Study

D.1 Ethics Form

Physical Sciences Ethics Committee

FAST-TRACK ETHICAL APPROVAL FORM (STUDENTS)

This fast-track system for is for taught students only. Research students and staff must complete the full Ethical Approval Form.

If you answer **YES** to any of the following you must complete either this Fast-track ethical approval form, to be signed off by your supervisor, or a full Ethical approval application, to be approved by the Physical Science Ethics Committee (allow at least two weeks for this process).

Note that the outcome of the Fast-track system may result in you needing to complete a full ethical approval application.
Does your project involve any of the following?

	YES	NO
Human participants (adults or children)	✓	
Human material (e.g. tissue or fluid samples)		✓
Human data (e.g. surveys and questionnaires on issues such as lifestyle, housing and working environments, attitudes and preferences)		✓
Vertebrates, especially mammals and birds		✓
Any other organisms not previously mentioned		✓
Military or defence context		✓
Funding sources with potential to adversely affect existing relationships or bring the University or Department into disrepute.		✓
Restrictions on dissemination		✓
Overseas countries under regimes with poor human rights record or identified as dangerous by the Foreign & Commonwealth Office		✓
Applications that could potentially involve unethical practice, including potential dual-use applications which could be unethical		✓

Students: you should discuss the ethical considerations of your project with your project supervisor and, if necessary, fill in a full ethics form to be submitted to the Physical Sciences Ethics Committee.

Supervisors: Please ensure you are familiar with the University's 'Code of practice and principles for good ethical governance' in order to guide your student effectively. Please seek guidance from the Departmental Ethics Officer if you are uncertain about any ethical issue arising from this application.

THE UNIVERSITY of York

Physical Sciences Ethics Committee

FAST-TRACK ETHICAL APPROVAL FORM (STUDENTS)

Project Information:

Student Name: Chris Taylor

Course Title: Electronic Engineering

Tick one box:

Undergraduate project	☑	Postgraduate project	☐
Undergraduate module assignment	☐	Postgraduate module assignment	☐
Other (Please state.........................)	☐		

Title of project: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually ~~Im~~ or Physically Impaired Users

Project supervisor / module leader name: Tony Ward

Protocol:

a): *If you answer **NO** to any of the following you must submit a full ethical approval form*

	If you answer yes to any of the following, this must be explicit in any supporting literature (e.g. consent forms, information sheets and questionnaires)	YES	NO	N/A
1	Will you describe the procedures to participants in advance, so that they are informed about what to expect?	✓		
2	Will you tell participants that their participation is voluntary?	✓		
3	Will you inform the participants of the purpose / background of the study?	✓		
4	Will you obtain written consent for participation?	✓		
5	If the research is observational, will you ask participants for their consent to being observed?	✓		
6	Will you tell participants that they may withdraw from the research at any time and for any reason?	✓		
7	With questionnaires and interviews will you give participants the option of omitting questions they do not want to answer?	✓		
8	Will you tell participants that their data will be treated with full confidentiality and that, if published, it will not be identifiable as theirs?	✓		

THE UNIVERSITY *of York*

Physical Sciences Ethics Committee

Protocol:

b): *If you answer **YES** to any of the following you must submit a full ethical approval form.*

		YES	NO	N/A
9	Is your study designed to be challenging/disturbing (physically or psychologically)?		✓	
10	Will you deliberately mislead your participants?		✓	
11	Does your study involve taking bodily samples?		✓	
12	Is your study physically invasive?		✓	
13	Is there any obvious or inevitable adaptation of your research findings to ethically questionable aims?		✓	
14	Could the methodologies or findings of your study damage the reputation of the University of York?		✓	

Health and Safety:
Please identify any risks to the participants and state any precautions you will take to ensure their health and safety:

Electrocution: All participants will be members ~~off~~ of the Department of Electronic Engineering at York and will be instructed on safe usage of the lab equipment.
All wires will be covered. All voltages and currents will be limited to a low level.

Participants: *If you answer **YES** to any of the following you must submit a full ethical approval form. If you have ticked **YES** to 15 and your participants are **patients**, in addition to the full ethical application you must follow the Guidelines for Ethical Approval of NHS Projects.*

			YES	NO	N/A
15	Does your project involve work with animals			✓	
16	Will any of the participants be from one of the following vulnerable groups?	Children under 18		✓	
		People with learning difficulties		✓	
		People who are unconscious or severely ill		✓	
		NHS patients		✓	
	Note that you may also need to obtain satisfactory DBS clearance (or equivalent for overseas students)	Other vulnerable groups (specify)		✓	

THE UNIVERSITY *of York*

Physical Sciences Ethics Committee

Data Protection: *If you answer **NO** to any of the following you must submit a full ethical approval form*

		YES	NO	N/A
17	Any personal / sensitive data will be stored in password protected folders on computers.	✓		
18	Any hard copies of personal data (including consent forms) will be stored in a secure place.	✓		
19	Only the student and supervisors will have access to the data generated from the study. (The supervisor may share the anonymised data with other researchers at the University of York)	✓		
20	The data will be preserved beyond the study in line with University policy and will be placed in the custody of the supervisor at the end of the project.	✓		
21	All data will be anonymised prior to analysis. Please state your method of anonymisation: All identifiable data will be omitted from recordings and will not be transcribed or included in the report.			

FOR THE STUDENT TO COMPLETE:

Please complete and sign the following section and submit to your supervisor alongside any supporting documentation (this includes consent forms, information sheets and questionnaires where necessary).

Provide a brief summary of the participants and procedures of your project (max 100 words)

The participant will be asked to complete a set of simple tasks regarding a circuit that has been partly pre-built by the researcher. They will be asked to find out some information about the circuit using the lab equipment by conversing with the researcher who will then be in turn controlling the lab equipment and relaying the data back to the participant.

THE UNIVERSITY *of York*

Physical Sciences Ethics Committee

I have considered the ethical implications of this project and have identified no significant ethical implications requiring a full ethics submission to the Physical Sciences Ethics Committee ☑

I have included all relevant paperwork (e.g. consent form, information sheet, questionnaire/interview schedules) with this application ☑

Signed.. Print name CHRIS TAYLOR Date 3rd Nov 17
(Student)

FOR THE SUPERVISOR TO COMPLETE:
By signing this form you are taking responsibility for the ethical conduct of this project

The student has taken all reasonable steps to ensure ethical practice in this study and I can identify no significant ethical implications requiring a full ethics submission to the Physical Sciences Ethics Committee ☑

I have checked and approved all relevant paperwork required for this proposal ☑

STATEMENT OF ETHICAL APPROVAL

This project has been considered using the Physical Sciences Ethics Committee Fast-track ethical approval procedure, agreed by the Physical Sciences Ethics Committee of the University of York, and is now approved.

Signed.. Print name TONY WARD........... Date 3/11/17
(Supervisor/Module leader)

OR

The details on this form indicate a need for a full application to PSEC. The practical aspects of this project will not proceed until this has application has been approved.

Signed.. Print name.. Date...............
(Supervisor/Module leader)

THE UNIVERSITY *of York*

D.2 Participant Consent Form

Consent Form for Participation in User Testing

Institution: University of York
Researcher: Chris Taylor (cjt534@york.ac.uk)
Supervisor: Tony Ward (tony.ward@york.ac.uk)
Project: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users

Brief Description of Study

The purpose of this study is to investigate how users would interact with voice controlled lab equipment if it already existed. The results of which will be used to ascertain what commands would be most common to users and what phrasing the users would naturally use.

Procedure

The participant will be asked to complete a set of simple tasks regarding a circuit that has been partly pre-built by the researcher. They will be asked to find out some information about the circuit using the lab equipment, conversing with a third-party, who will then be in turn controlling the lab equipment and relaying the data back to the participant. The conversation between the participant and the third-party will be recorded and transcribed by the researcher, but identifiable information will be silenced in the recording and omitted from the transcription. The audio may then be submitted alongside the report to the Department of Electronic Engineering at the University of York.

Participant data will remain anonymous at all times and will only be used for research purposes. Participants are able to withdraw from the experiment at any time without stating a reason. If a participant decides to withdraw, their recorded data and audio recordings will be destroyed and not used in the study. The study is expected to take around 60 minutes and will be conducted in the 4th floor laboratory of the Department of Electronic Engineering at the University of York.

Risks

All risks for this study will be controlled by ensuring that the participants are members of the Department of Electronic Engineering at the University of York, and have the knowledge to use the lab equipment safely and correctly. The researcher and third-party will be at no point in control of any voltages larger than 5 Volts.

Consent

I have been informed about the aims and procedures involved in the experiment I am about to participate in. The data collected as part of the experiment can be used for research and all data will be kept anonymous. I reserve the right to withdraw at any stage in the proceedings. I understand that any information that I have provided as part of the study will be destroyed or my identity removed unless I agree otherwise.

Name:

Signed:

Date:

1

D.3 Third Party Consent Form

Consent Form for Acting as a Third-Party in User Testing

Institution: University of York
Researcher: Chris Taylor (cjt534@york.ac.uk)
Supervisor: Tony Ward (tony.ward@york.ac.uk)
Project: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users

Brief Description of Study

The purpose of this study is to investigate how users would interact with voice controlled lab equipment if it already existed. The results of which will be used to ascertain what commands would be most common to users and what phrasing the users would naturally use.

Procedure

The third-party will be asked to act as an interface between another participant and the lab equipment, following their spoken instructions to change settings and relay data back from the displays. The conversation between the participant and the third-party will be recorded and transcribed by the researcher, but identifiable information will be silenced in the recording and omitted from the transcription. The audio may then be submitted alongside the report to the Department of Electronic Engineering at the University of York.

Participant data will remain anonymous at all times and will only be used for research purposes. Participants are able to withdraw from the experiment at any time without stating a reason. If a participant decides to withdraw, their recorded data and audio recordings will be destroyed and not used in the study. The study is expected to take around 60 minutes and will be conducted in the 4th floor laboratory of the Department of Electronic Engineering at the University of York.

Risks

All risks for this study will be controlled by ensuring that the participants issuing the instructions are members of the Department of Electronic Engineering at the University of York, and have the knowledge to use the lab equipment safely and correctly. The researcher and third-party will be at no point in control of any voltages larger than 5 Volts.

Consent

I have been informed about the aims and procedures involved in the experiment I am about to participate in. The data collected as part of the experiment can be used for research and all data will be kept anonymous. I reserve the right to withdraw at any stage in the proceedings. I understand that any information that I have provided as part of the study will be destroyed or my identity removed unless I agree otherwise.

Name:

Signed:

Date:

1

D.4 Participant Instructions

Participant Instructions for User Study for "Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users"

Make sure you have read, understood, and signed the consent form before starting.

Please read through this document and let the researcher know when you are ready to begin.

You can ask the researcher to pause or stop the study at any time. They will stop recording and you will be able to talk freely. If you have any questions during the study, please ask the researcher. Your questions will be noted and will be of help to the researcher.

You are expected to have knowledge of the Oscilloscope (OSC), Function Generator (FG), and Digital Multi-meter (DMM) that are usually provided in the 4th floor labs. For the first part of the study you will not be able to touch[1], see, or listen to the equipment. Instead you will need to ask the third-party to adjust any settings or retrieve any data from the equipment.

You may use the Tektronix software running on the desktop to retrieve screen-shots of the oscilloscope display at any time to view them on the PC monitor. Please be aware the researcher will take note of when you do this.

You can use any phrasing you wish to interact with the third-party, and it may help to talk to them as a person instead of pretending like they are the machine that youre trying to interact with.

Part 1

On the desk there should be a circuit pre-built that looks similar to Figure 2. Your tasks are to do the following:

1. Use the DDM to verify the values of the following components
   - R1 - 3.9 k
   - R2 & R4 - 1 k
   - R3 - 2 k
2. Set up the FG to provide a 100 mV, 5 kHz sine wave from 'Channel 1'.
3. Set up the FG to provide a 1 V square wave from 'Channel 2' and lock the frequency to 'Channel 1'.
4. Apply the signal from 'Channel 2' of the FG to the 'External Trigger' input of the OSC.
5. Set the OSC to use the external trigger.
6. Apply the signal from 'Channel 1' of the FG to 'Channel 1' of the OSC.

   Note: The researcher and third-party will not be making sure that the waveforms are visible or clear on the display. Please ask to set the timebase and voltage divisions accordingly.
7. Apply the signal from Channel 1 of the FG to TP1 on the circuit.

[1] Other than the DMM and OSC probes.

8. Apply the signal from TP3 on the circuit to Channel 2 on the OSC.
9. Switch on the power supply (PS) and verify the voltage is +/- 12 V.
10. Make sure the OSC probes are set to `'x1'`.
11. Determine the gain (TP3/TP1) and phase difference (TP3 - TP1) of the circuit at the following frequencies:
    - 10 Hz
    - 20 Hz
    - 100 Hz
    - 200 Hz

End of Part 1

The researcher will now stop recording. You may leave the room and take a break if you wish. Please let the researcher know when you are ready to begin again.

Part 2

For the second part of this study you will be able to observe the screen of the OSC freely. You will still be able to retrieve screen-shots of the display at any time to view them on the PC monitor instead.

Your tasks are to:

1. Set up the OSC & FG as you see fit in order to find out the rise and fall time of TP3 on the circuit when a 1 mV pulse is applied to TP1. [2] [3]
2. Find the -3 dB point frequency and phase of the circuit (TP3/TP1) [4]

End of Part 2

Please let the researcher know when you are completely done and they will stop recording. You may now take this time to ask the researcher any questions or provide feedback about the study.

The study is now complete.

[2] Use the cursors to get an accurate measure.
[3] Use the run/stop function to get a clearer image.
[4] -3 dB relates to a gain of 0.5 times the low frequency gain.

Figure 1: Circuit Layout [5]

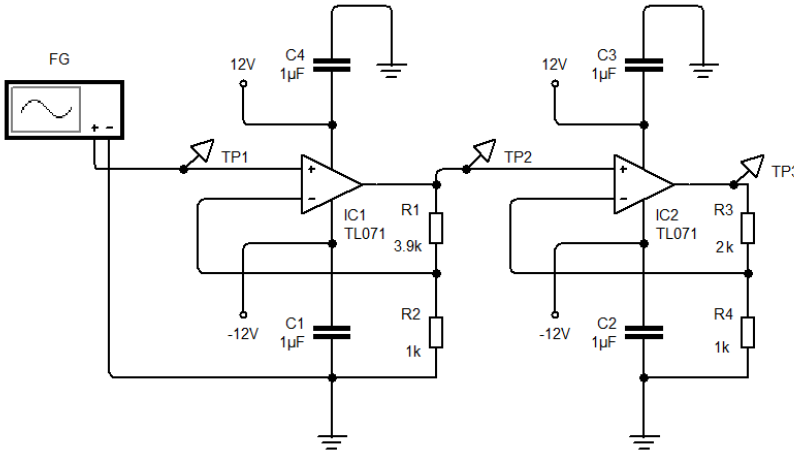
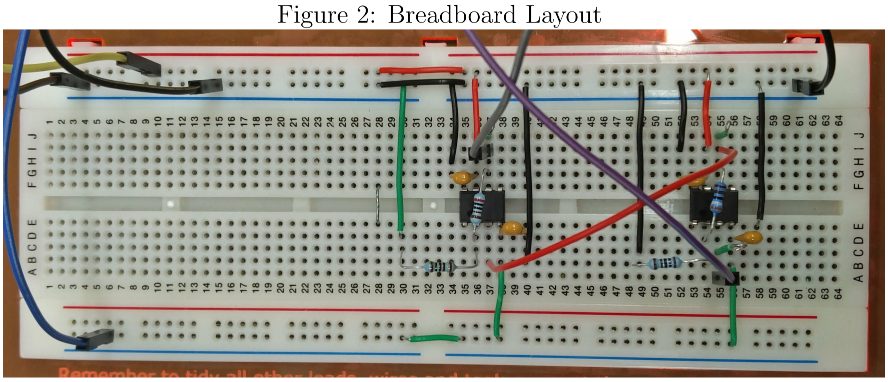


Figure 2: Breadboard Layout



[5]Circuit adapted with permission from "Breadboarding Workshop". (R. Millar, 2017)

D.5 Third Party Instructions

Third-Party Instructions for User Study for "Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users"

Make sure you have read, understood, and signed the consent form before starting.

Please read through this document and let the researcher know when you are ready to begin.

You can ask the researcher to pause or stop the study at any time. They will stop recording and you will be able to talk freely. If you have any questions during the study, please ask the researcher. Your questions will be noted and will be of help to the researcher.

You are expected to use the Oscilloscope (Figure 1), Function Generator (Figure 2), and Digital Multi-meter (Figure 3) that are usually provided in the 4th floor labs.

For the experiment you will be acting as an interface between the participant and the lab equipment. They will be asking you to change settings on the equipment as well as relay data back from the displays. Please take their instructions as literally as possible and ask the researcher for clarification if you do not understand the command. Please do not ask the participant for clarification.

Please do not touch any other equipment on the bench.

Figure 1: Oscilloscope

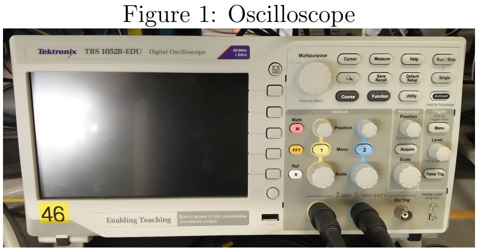


Figure 2: Function Generator

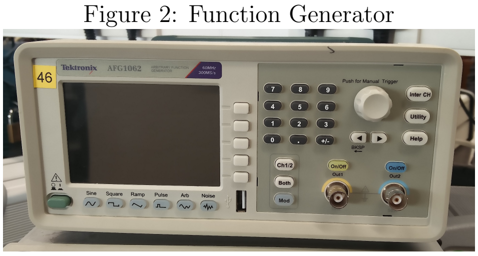


Figure 3: Digital Multi-meter

Page 1 of 1

D.6 Transcriptions

D.6.1 Test #1

Part 1

Third-Party: OK I'm ready

Participant: So I'm going to start by measuring and making sure all these resistors are right

Participant: So can you set my ... the multimeter to measure resistance please

Third-Party: done

Researcher: change that as well

Participant: and you say I'm not allowed to look at the multimeter

Researcher: you're not allowed to look at it

Participant: OK

Researcher: you're not allowed to look at anything

Third-Party: OK so it's ready

Participant: so what is that?

Researcher: oh you're not allowed to say anything until like he asks you by the way

Third-Party: OK

Participant: what's that reading?

Third-Party: that reading is 3.86 kilo ohms

Participant: OK ...

Participant: that reading?

Third-Party: that reading is 0.99 kilo ohms

Participant: that reading?

Third-Party: that reading is 0.99 ... uh oh 1.00 kilo ohms

Participant: and that reading?

Third-Party: that is 1.98 kilo ohms

Participant: OK ...

Participant: can you set channel one on the function generator to output a 100 milli volt 5 k sine wave

Third-Party: uh ... what was the amplitude?

Participant: so that's 100 milli volts at 5 kilo hertz sine wave on channel one

Third-Party: is that uh 100 milli volts peak to peak or rms?

Participant: whatever you like, it doesn't specify

Researcher: peak to peak

Researcher: that's on there and then ... external trigger

Researcher: you need to act dumb, if he doesn't say it, you don't do it

Participant: next, set up a 1 volt square wave coming out of channel two and lock the frequency to channel one

Third-Party: oh there we go

Third-Party: can you repeat the instruction again sorry

Participant: one volt square wave from channel two, with the frequency locked to channel one

Third-Party: oh I see uh

Third-Party: uh done

Participant: OK, so apply the signal from channel two into the oscilloscope trigger

Third-Party: uh done

Participant: and channel one into channel one on the oscilloscope

Third-Party: uh done yep

Participant: oh there's more ...

Researcher: so that rail, the negative rail there is ground.

Participant: right

Researcher: so that one's the minus voltage, that's ground, that's positive

Participant: where are these both coming from

Researcher: that's from the power supply. That's channel one from the function generator

Researcher: that's channel one going to the oscilloscope

Participant: cool

Participant: so

Researcher: sorry channel two. yeah because that's your output

Participant: right

Participant: apply signal from channel one *...unintelligible...* TP1 is this one

Participant: switch on the power supply

Researcher: that's for you

Participant: oh that's for me

Researcher: yeah

Participant: okay verify voltage is ...

Researcher: don't do anything unless he asks you

Participant: 12 volts. okay. so what it *...unintelligible...*

Participant: what is the amplitude of the reading on channel two of the oscilloscope

Third-Party: undeterminable

Participant: umm. please *...unintelligible...* yeah so

Participant: zoom out so that the range for amplitude is reduced

Researcher: by how much

Participant: until you can determine the amplitude

Researcher: yeah no you have to give numerical values

Participant: but I don't know what the gain is. so how am I supposed to know what that signal is going to be

Researcher: there's a button there that says 'autorange'

Third-Party: is there

Participant: ‘autorange’. okay

Researcher: ‘autoset’ even

Third-Party: done

Participant: umm set the timebase to 200 microseconds per unit

Third-Party: yep. 200 doesn’t exist

Participant: the closest one

Third-Party: would you like 250 instead

Participant: yes

Third-Party: 250 set

Participant: umm set the oscilloscope to use the external trigger for *...unintelligible...*

Third-Party: done

Participant: okay so now what is the reading on channel two amplitude

Third-Party: channel two peak to peak is 30.4 volts

Participant: okay. uhh. okay so. change the frequency on channel one of the signal generator to 10 hertz

Third-Party: done

Participant: um now set the timebase to one tenth of a second and

Third-Party: done

Participant: what is the output on channel two

Third-Party: undeterminable

Researcher: yes hang on. I’ve had to change the base just so it can do it.

Note: The oscilloscope wasn’t showing the waveform correctly, although this was no fault of the participant or third-party.

Participant: okay

Third-Party: okay. output is 30.4 peak to peak

Participant: change channel one frequency on the signal generator, change to 20 hertz

Third-Party: done

Participant: ah f*ck, cancel that last action

Third-Party: undo. done

Participant: umm. so do I need to tell him to get up the *...unintelligible...* so take a phase difference measurement from the oscilloscope between the two channels

Participant: what is the phase difference

Third-Party: computing. computing. uh. phase difference, channel one minus channel two, is undeterminable

Participant: and the other way round

Researcher: basically zero. yeah it’s so small it may as well be negligible

Third-Party: minus 720 milli degrees

Participant: zero it is

Researcher: yeah

Participant: change the frequency of the output on channel one on the signal generator to 20 hertz

Third-Party: done

Participant: adjust the timebase of the oscilloscope accordingly

Third-Party: specify ‘accordingly’

Participant: so that umm values can be read

Researcher: numerical please

Third-Party: does not compute

Participant: okay half the timebase

Third-Party: half time base. done

Participant: and read off the value of channel one

Third-Party: channel one peak to peak

Participant: sorry channel two

Third-Party: oh thank god. okay channel two peak to peak is 30.4

Participant: and what is the phase difference

Third-Party: phase difference is zero

Participant: umm. set the output frequency of channel one on the signal generator to 100 hertz

Third-Party: done

Participant: change the timebase on the oscilloscope to one fifth of what it already is

Third-Party: okay done

Participant: and what is the reading on channel two of the oscilloscope

Third-Party: uhh which reading would you like

Participant: oh amplitude

Third-Party: peak to peak is 30.4

Participant: and phase

Third-Party: phase is zero

Participant: umm double the output frequency of channel one on the signal generator

Third-Party: done

Participant: half the time base on the oscilloscope

Third-Party: done

Participant: and magnitude reading on channel two please

Third-Party: 30.4 peak to peak

Participant: and phase

Third-Party: uhh zero

Participant: okay that’s end of part one

Part 2

Participant: set, set the output of channel one on the fg to an amplitude of 1 volt.

Third-Party: am I to understand what an fg is

Researcher: function generator *...unintelligible...* make that one millivolt, otherwise we'll go out of range

Third-Party: one millivolt, is that peak to peak again

Participant: yeah. and that's a pulse 1 every half a second

Third-Party: is that also 1 milli volt as well

Researcher: that's got to be from 0 to 1 millivolt

Participant: that's a peak value, not peak to peak, sorry

Third-Party: okay yeah that's fine *...unintelligible...*

Researcher: *...unintelligible...*

Third-Party: and so every half a second

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...*

Researcher: make it pulse every 5 milliseconds

Third-Party: every 5 milliseconds

Participant: yeah

Third-Party: okay that's do-able. okay umm what would you like the duty cycle to be. or the width

Participant: it doesn't matter, that's fine

Third-Party: 50%. 50% cool

Participant: okay cool

Researcher: i was expecting him to use a square wave, because you've got to do the rise and fall time

Third-Party: oh right okay

Participant: so I'm allowed to look at that now

Researcher: yeah you're allowed to look at that, but not allowed to touch it

Participant: okay, so zoom in on amplitude first of all

Third-Party: i don't understand what amplitude is

Participant: don't you

Researcher: well which channel

Participant: either it's the same knob isn't it

Researcher: no

Third-Party: timescale

Participant: on channel two is that right

Researcher: yeah because channel one is your input

Third-Party: i zoomed in

Participant: continue to do so

Researcher: numerical values please

Participant: umm so until 1 millivolt occupies half the screen

Researcher: so that's a voltage division of, yeah 10 millivolts

Participant: but is that putting out 1 millivolt at the moment

Researcher: yeah that's putting out 1 millivolt, this is then amplifying it

Participant: aha

Researcher: yeah, and you know what the gain is, like 30 odd, i think *...unintelligible...*

Participant: *...unintelligible...* what is that *...unintelligible...* 3 and half thousand gain

Researcher: i don't know, i didn't calculate it. I nicked this circuit from Rich

Participant: okay so it's hard to put this in words umm reduce the timebase by, and I can't say until until one pulse fills the screen

Researcher: no you have to say the time division or something

Participant: *...unintelligible...*

Third-Party: *...unintelligible...*

Participant: reduce the timebase by 3 steps, 3 more steps, 3 more steps

Third-Party: done

Participant: and 3 more steps, okay. back one

Third-Party: *...unintelligible...*

Participant: umm

Third-Party: *...unintelligible...*

Researcher: *...unintelligible...*

Participant: so *...unintelligible...* I just want rise time. so I can take that myself

Participant: okay umm, set a time cursor. one at the beginning of the rise

Researcher: no

Participant: one at the origin

Third-Party: I don't know how to *...unintelligible...*

Researcher: *...unintelligible...* the origin is fine *...unintelligible...*

Third-Party: at the origin done

Participant: shift the waveform left by one increment

Third-Party: i don't think this thing has increments

Researcher: use the second one to put it where you need *...unintelligible...*

Participant: i was going to move the waveform a little bit so it lined up with the origin better and then do that

Researcher: uh no I would say that you should go from there because that's actually a result of the impulse

Participant: umm let's get a second time cursor up

Researcher: can I just interrupt for a second. *...unintelligible...* yeah that's the impulse, so that's the time delay for it to go through

Participant: right

Third-Party: so that still counts as part of the rise time

Participant: so am I not supposed to have a timer cursor on the origin at the moment

Researcher: *...unintelligible...*

Third-Party: you do it's just not showing

Participant: so *...unintelligible...*

Researcher: what's the definition of rise time, i probably should have written it down

Participant: is it 90%

Researcher: yeah

Participant: so adjust the time cursor until it reaches 90% of the final

Researcher: no

Participant: why not that's a valid instruction

Researcher: the computer's not going to know how to do that

Participant: okay

Researcher: you're still pretending you're talking to an intelligent, a really intelligent thing

Third-Party: the computer might be able to though

Participant: so I need a calculator then

Third-Party: it could know the zero to, the peak to peak, work out 90% of it and because you can actually read the voltage at that time cursor you might be able to

Researcher: i think that would take a long time because it would actually, because I would have to tell it to shift, measure, is that, shift, measure, oh you've gone too far, go back a bit. I think you'd be sitting there waiting for it for ages

Third-Party: okay i guess

Participant: so set that second time cursor to correspond with an amplitude of 29

Third-Party: I can't do that I'm afraid

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...*

Participant: *...unintelligible...*

Participant: but if I'm telling him a time then I know what the time difference is so there's no point in the cursor

Third-Party: that's true

Researcher: no you're going to tell him to move it by an increment

Participant: oh I thought we said there weren't increments

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...* shift left and I'll do one by dot

Researcher: kinda thing

Participant: so I also need an actual voltage reading to do this accurately

Researcher: I would use the delta voltage change

Participant: but I need to set that to be 90% of that right

Researcher: yeah

Participant: so i need to know *...unintelligible...* what that value is there

Researcher: that's oh hang on, channel two there you go

Participant: umm *...unintelligible...* okay move that second time cursor left by one increment

Third-Party: done

Participant: and keep going until *...unintelligible...*

Researcher: no

Participant: *...unintelligible...* okay another 10 increments and stop

Third-Party: that's 10

Participant: another 5 left

Third-Party: 5

Participant: and 3 right

Third-Party: done

Participant: okay that's 90% and so the difference between those two cursors is

Third-Party: 1.14 micro seconds

Participant: giving us rise time. okay zoom out by 20 increments

Researcher: in which domain

Participant: time

Third-Party: that's a good point yes, sorry

Participant: stop

Third-Party: *...unintelligible...*

Participant: zoom in by 10

Third-Party: *...unintelligible...* 9, 10

Participant: *...unintelligible...* now shift the signal to the left by half a wavelength

Third-Party: Chris do I know the wavelength

Participant: can I shift something by half a wavelength

Third-Party: *...unintelligible...*

Researcher: you should know the wavelength because you set it on the function generator

Third-Party: but then can I detect it *...unintelligible...*

Researcher: yes because I could program that in

Third-Party: okay yeah so if *...unintelligible...* I can shift it by half a wavelength then I will straight away

Participant: is this more about you figuring out what I would like to say to get something achieved

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...*

Participant: so shift the wavelength to the left by 2 and a half milliseconds

Third-Party: okay ummm

Participant: and zoom in into the time domain 5 times, 5 increments

Third-Party: *...unintelligible...*

Participant: and another 3 increments

Third-Party: one, two, three

Participant: another 3

Third-Party: one, two, three

Participant: and back out by one

Third-Party: back out by one

Participant: okay

Third-Party: *...unintelligible...*

Participant: move the second time cursor to the right by one increment

Third-Party: time cursor two to the right by one increment, done

Participant: and another increment, oh it took a second to update. Okay that time is 2.5 milli seconds*...unintelligible...*

Researcher: can I just check where the first cursor is *...unintelligible...*

Participant: *...unintelligible...*

D.6.2 Test #2

Part 1

Participant: what is the digital multimeter set to

Third-Party: it is set to resistance

Participant: what is the measurement

Third-Party: it is fluctuating too much at the moment

Participant: *...unintelligible...*

Participant: what is the measurement from the digital multimeter

Third-Party: it's about 2 kilo ohms

Researcher: *...unintelligible...*

Participant: right let's do that again. what was the reading from the digital multimeter

Third-Party: 1.99 ... 1.99 kilo ohms

Participant: 1.99 kilo ohms

Third-Party: *...unintelligible...*

Participant: what measurement is the digital multimeter giving now

Third-Party: it is 0.994 kilo ohms

Participant: what measurement is the digital multimeter giving now

Third-Party: giving 0.991 kilo ohms

Participant: what value is the digital multimeter giving now

Researcher: *...unintelligible...*

Third-Party: it is reading 3.85 kilo ohms

Participant: set up the function generator to provide a 100 milli volt, 5 kilo hertz sinewave from channel one please

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...*

Participant: can the function generator provide 100 milli volts with a 5 kilo hertz sinewave from channel one

Third-Party: sinewave?

Participant: sinewave please

Participant: from channel two can it provide a 1 volt square wave

Third-Party: yep

Participant: lock the frequency to channel one

Third-Party: I think that's correct for frequency lock

Researcher: yep

Participant: okay. can you apply the signal from channel two on the function generator to the external trigger to the oscilloscope.

Third-Party: yep

Participant: is the oscilloscope set to use the external trigger?

Third-Party: yep

Participant: I am applying the signal from channel one from the function generator to channel one of the oscilloscope. right hang on.

Researcher: yeah

Participant: *...unintelligible...*

Researcher: it is

Participant: *...unintelligible...* apply the signal from channel one from the function generator to test point one of the circuit.

Researcher: *...unintelligible...*

Participant: *...unintelligible...* I am now applying the signal from test point three on the circuit to channel two of the oscilloscope. okay

Participant: can I have the oscilloscope set to default setup please

Third-Party: yep

Participant: okay. Now I control the power supply

Researcher: *...unintelligible...*

Participant: *...unintelligible...* now I control the probes

Researcher: *...unintelligible...* could you verify the voltage from the power supply please

Participant: yes I can

Researcher: *...unintelligible...*

Participant: can I have the dmm set to dc voltage please

Third-Party: yep

Participant: *...unintelligible...*

Third-Party: done

Participant: can you tell me what the dmm is reading please

Third-Party: 0 *...unintelligible...*

Researcher: *...unintelligible...*

Note: The researcher had to rearrange the power supply rails.

Participant: yep. Can I have the oscilloscope set to averaging please.

Third-Party: what level?

Participant: let's go with 16

Third-Party: done

Participant: okay. is the oscilloscope set for the probes at times one.

Third-Party: nope.

Participant: can you set it to times one please

Third-Party: done

Participant: hang on.

Researcher: ask questions.

Participant: I'm supposed to ask what readings are coming from the oscilloscope for the gain and *...unintelligible...* how do I know if we've got the right settings on the oscilloscope as in umm, umm the difference.

Researcher: so you need to ask for timebase to be set correctly and the amplitudes to be set correctly for each channel.

Participant: okay

Researcher: there is a short-cut you could take here, and that's by using the autoset function. But that's up to you

Participant: can I have the oscilloscope set to the autoset function

Third-Party: autoset.

Researcher: *...unintelligible...*

Participant: *...unintelligible...*

Participant: can you run autoset on the oscilloscope again please

Third-Party: yep. done

Researcher: *...unintelligible...*

Note: The required software to take screen-shots of the oscilloscope display was not installed on the PC. Instead the participants were allowed to look at the display without changing any of the settings. This accurately simulated taking a screen-shot of the display.

Participant: I am going to determine the gain of the circuit at 10 hertz. So on channel one of the function generator can you, can it be set to 10 hertz please.

Third-Party: done

Participant: okay. looking at the screen-shot

Researcher: *...unintelligible...*

Participant: take a screen-shot of the oscilloscope please

Third-Party: take

Participant: okay on the oscilloscope, on the oscilloscope can it show umm

Participant: timebase to be set to 0.1 seconds please oscilloscope

Researcher: *...unintelligible...*

Participant: can the oscilloscope show me what is on the umm probe two channel two

Researcher: what do you mean

Participant: as in *...unintelligible...* I'd like to know the peak-to-peak of channel two

Third-Party: okay. peak-to-peak for channel two is 3.06 volts

Participant: can you repeat that please oscilloscope.

Third-Party: peak-to-peak for channel two is 3.06 volts

Participant: and can I have that again *...unintelligible...* can I get the channel one reading for peak-to-peak on the oscilloscope

Third-Party: channel one peak-to-peak is 212 milli volts

Participant: milli volts

Third-Party: yeah

Participant: okay

Note: The function generator output was set to 50 Ohm instead of High-Z. This was causing the readings to be a factor of 2 out. This was rectified for the rest of the tests.

Participant: can I have the oscilloscope set to measure the phase difference from channel two minus channel one please

Researcher: *...unintelligible...*

Third-Party: phase of channel one

Researcher: can I get a screen-shot of the oscilloscope please

Third-Party: screen-shot

Participant: okay *...unintelligible...*

Researcher: *...unintelligible...*

Participant: can I have the function generator set to 20 hertz please

Third-Party: what channel

Participant: on channel one

Researcher: on a side-note it doesn't matter because they're frequency locked

Participant: can I get a screen-shot from the oscilloscope

Third-Party: done

Participant: screen-shot's been taken. Phase difference is approximately zero

Part 2

Participant: on channel one can I have. oh sorry on channel of the function generator can I have umm, can I have one milli volt pulse, so can I just have a square wave going in

Third-Party: so you want a square wave

Participant: yes please of one milli volt

Third-Party: peak-to-peak?

Researcher: yeah *...unintelligible...* set the off the offset to *...unintelligible...* you may just want to lower the frequency a bit just so you've got a bit of *...unintelligible...* lower the frequency and umm

Participant: can I have the frequency lowered on channel one to 10 hertz please

Third-Party: yep

Participant: on the oscilloscope can I have cursor please, can you click the cursor on the oscilloscope

Researcher: before we do that, what's the first issue here

Participant: oh it's not centered

Note: The waveform was not clear on the display.

Researcher: *...unintelligible...*

Participant: *...unintelligible...* oh it's the timebase isn't it

Researcher: *...unintelligible...*

Participant: oh can we autostart please

Researcher: auto set

Participant: auto set

Third-Party: auto set

Researcher: there we go

Participant: okay. on the Oscilloscope can I have the cursor function selected please

Third-Party: what type

Participant: can you click on the type and set it to time please. okay

Researcher: can you see the rise time

Participant: no I can't. Can I have the horizontal scale increased

Researcher: how much

Participant: umm

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...*

Participant: how come it keeps clipping like that or. can I have the umm the horizontal scale increased by one twist. I don't know how to say it.

Researcher: yeah

Participant: umm. can I you turn once on the scale again please. The horizontal scale.

Third-Party: yep

Participant: can you twist the horizonal scale again please

Third-Party: yep

Participant: can we zoom in to it's maximum scale please. okay ummm

Third-Party: you did say to maximum

Participant: can it you change it as in the scale is micro please. so umm

Researcher: one micro second

Participant: one micro second and umm decrease the horizontal scale please

Third-Party: one micro second

Participant: okay perfect. umm this is for tp3 which is channel two

Participant: can we go back to the cursor function please, type time

Third-Party: time

Participant: umm source channel two and rotate clockwise on the multi-purpose dial until it is 10% of

Participant: okay so the first cursor is set to the origin

Third-Party: yep

Participant: of the oscilloscope. can I have the second cursor, cursor two set to it's final value so umm at. can I have the second cursor set to the umm second division from the origin, so two divisions from the origin.

Researcher: so that's two micro seconds

Participant: two micro seconds. yes because each division is a micro second. okay umm so now I can see

Researcher: so you're looking at the delta voltage there

Participant: yes so delta voltage is approximately 3 volts

Researcher: so 90% of that would be

Participant: times it by 0.9. so the 90% value on the oscilloscope should be 2.7. can the oscilloscope move its multi-purpose dial on cursor two to measure 2.7 on the delta voltage please

Researcher: so it wouldn't know how to do that

Participant: okay

Researcher: *...unintelligible...*

Participant: rotate anticlockwise a small amount

Researcher: say half a division or something

Participant: half a division. can the oscilloscope move half division anticlockwise. stop. yep I think that's as close as we're going to

Researcher: yeah so what's the delta time

Participant: the delta time umm

Researcher: on the middle bit

Participant: is one micro second

Researcher: yeah close enough. so now do fall time

Participant: can the oscilloscope be set on okay. click the source channel two button please. okay click back. umm okay select the

Researcher: so there's two ways you could do this. If you

Participant: there's usually a button that just says slope rise

Researcher: yeah so that's under the trigger. so it's something like set the trigger to uhh falling mode

Participant: set the trigger to falling mode. so slope rising click and then set it to falling. okay

Third-Party: yep

Participant: go back to cursor function please. set cursor one of the channel two to be at the origin.

Researcher: done

Participant: set the umm cursor two of the channel two on the oscilloscope umm to two micro seconds from the origin. stop. okay so this is it's final value at delta voltage 3 volts so we need 90% of this so delta voltage will equal 2.7 volts. Can I have cursor two of the oscilloscope rotate anticlockwise by half a division

Third-Party: yep. done

Participant: can I have the oscilloscope set the multi-purpose anticlockwise by another half division

Third-Party: yep

Participant: yep stop close enough so the fall time is also 1 micro seconds

Researcher: cool

D.6.3 Test #3

Part 1

Participant: *...unintelligible...* is it set to the resistance

Third-Party: no

Participant: could I have it set to resistance

Third-Party: yes

Participant: so what's the reading

Researcher: *...unintelligible...*

Participant: what's the reading

Third-Party: 3.89 kilo ohms

Participant: what's the resistor value

Third-Party: 0.99 kilo ohms

Participant: what's the resistor value

Third-Party: 1.99 kilo ohms

Participant: and what's the final resistor value

Third-Party: 0.99 kilo ohms

Participant: yep. set up the function generator for a 100 milli volt 5 kilo hertz sine wave from channel one

Third-Party: *...unintelligible...*

Researcher: *...unintelligible...*

Third-Party: yep that's set

Participant: can you set up the function generator to provide a 1 volt square wave on channel two and then lock the frequency to channel one

Third-Party: can you repeat that please

Participant: set up the function generator to produce a 1 volt square wave from channel two and lock the frequency to channel one

Third-Party: *...unintelligible...* done

Participant: set the channel two of the function generator to the external trigger channel of the oscilloscope

Third-Party: yep

Participant: set the oscilloscope to use the external trigger

Third-Party: done

Participant: is there a sinewave on the oscilloscope

Third-Party: no

Participant: could you press autoset

Third-Party: done

Participant: is there a sinewave on the oscilloscope now

Third-Party: no

Participant: take a screenshot

Researcher: there we go

Third-Party: screenshot taken

Participant: *...unintelligible...* yep sinewave, good. right can you measure the peak-to-peak voltage of the

Researcher: *...unintelligible...*

Participant: can you turn on averaging to times 16

Researcher: *...unintelligible...*

Participant: make sure that each of the probes at times 1 and if they're not change them to times 1

Third-Party: done

Participant: could I have the digital multimeter set to voltage

Participant: what is the reading

Third-Party: 12 volts

Participant: what is the reading

Third-Party: minus 12 volts

Participant: that's good

Participant: could you set the signal generator to 10 hertz

Third-Party: done

Participant: can you set the oscilloscope to measure the phase difference from channel two to channel one

Participant: take a screenshot

Third-Party: screenshot taken

Participant: right. can the timebase be decreased so that

Researcher: careful here, numerical values

Participant: decreased to increment of 0.01 seconds

Participant: take a screenshot

Third-Party: screenshot taken

Participant: right. can I have the timebase decreased by one anticlockwise

Participant: take a screenshot

Third-Party: screenshot taken

Participant: could I have the amplitude of channel two reduced by 2 ... could I have the amplitude of channel two

Participant: what's the amplitude of channel two

Third-Party: *...unintelligible...*

Participant: what's the amplitude of channel one

Third-Party: 100 milli volts

Participant: take a screenshot

Third-Party: screenshot taken. so can I see what the scales are?

Researcher: yeah the screenshot is that entire display

Participant: *...unintelligible...* can I have channel two amplitude reduced by two notches anticlockwise

Third-Party: done

Participant: can I have the oscilloscope to measure the peak-to-peak voltage on channel one and channel two

Third-Party: done

Participant: what is the peak-to-peak voltage of channel one

Third-Party: 100 milli volts

Participant: what is the peak-to-peak voltage of channel two

Third-Party: 3.04 volts

Participant: what is the phase difference from channel two to channel one

Third-Party: minus 1.08 degrees

Participant: could I have the signal generator set to 20 hertz

Third-Party: done

Participant: take a screenshot

Third-Party: screenshot taken

Participant: what is the peak-to-peak voltage of channel one

Third-Party: 100 milli volts

Participant: what is the peak-to-peak voltage of channel two

Third-Party: 3.04 volts

Participant: what is the phase difference between channel two and channel one

Third-Party: minus 1.44 degrees

Participant: could I have the frequency generator set to 100 hertz, signal generator should be

Third-Party: done

Participant: could I have the timebase on the oscilloscope set to ... set to 0.01 seconds per division

Third-Party: done

Participant: take a screenshot

Third-Party: screenshot taken

Participant: what is the peak-to-peak voltage of channel one

Third-Party: 100 milli volts

Participant: what is the peak-to-peak voltage of channel two

Third-Party: 3.04 volts

Participant: what is the phase difference from channel two to channel one

Third-Party: question mark

Participant: okay

Participant: could I have the signal generator set to 200 hertz

Third-Party: done

Participant: what is the peak-to-peak voltage of channel one

Third-Party: 100 milli volts

Participant: and the peak-to-peak for channel two

Third-Party: 3.04 volts

Participant: and what's the phase difference

Third-Party: 864 milli degrees

Part 2

Researcher: go on then

Participant: right ...*unintelligible*... right can I have a one volt pulse wave from the signal generator. One milli volt

Researcher: what channel

Participant: from channel one

Participant: could you increase the timebase by one increment clockwise

Participant: and again

Participant: and again

Participant: and again

Participant: could you rotate the position of the timebase clockwise until I say stop

Researcher: that's going to be an interesting one to implement

Participant: stop

Participant: could you rotate the timescale clockwise by one increment.

Participant: and again

Participant: could you rotate the position

Researcher: I'm concerned about that, because that should be centred

Researcher: ...*unintelligible*...

Participant: could you rotate the scale on the timebase clockwise by one

Participant: and again

Participant: and again

Participant: and again

Participant: and again

Participant: and again

Participant: and again

Participant: rotate it anticlockwise by one

Researcher: ...*unintelligible*...

Participant: stop

Participant: could I have the cursor for timebase

Researcher: what channel

Participant: for channel two

Participant: could I have cursor one set on the third interval left of the origin

Researcher: ...*unintelligible*... it's considering that to be origin because we've moved the horizontal position

Participant: so time equals zero

Researcher: yeah

Participant: could I have cursor two set on, set at 1.5 micro seconds

Participant: rise time is up to 90%

Researcher: yep

Participant: *...unintelligible...*

Participant: could I have cursor two so that the delta voltage is 1.35

Third-Party: done

Participant: so one micro second as the rise time okay

Participant: could you press run. run

Researcher: what are you asking for?

Participant: run

Researcher: oh run

Third-Party: sorry

Participant: could the scale be decreased anticlockwise by two increments

Researcher: so that's your period

Participant: oh could I have the trigger set to falling for external

Researcher: yeah *...unintelligible...*

Participant: could the timebase be increased by one increment clockwise

Participant: could you hit stop

Participant: could cursor set on the origin

Third-Party: done

Participant: could it be set on the origin where time is zero

Third-Party: oh sorry. yeah *...unintelligible...*

Participant: could cursor two set at two micro seconds in front of cursor one

Participant: yep voltage 1.5

Participant: could I have cursor two set to the point where the delta voltage is 1.35

Third-Party: done

Participant: yeah one micro second

D.6.4 Test #4

Part 1

Participant: set the multimeter to resistance

Third-Party: okay. resistance has been set on the digital multimeter

Participant: what number does the multimeter say

Third-Party: uh. it's not being clear

Researcher: *...unintelligible...*

Note: The DMM doesn't give clear readings if the probes aren't firmly pushed against the wires

Participant: what number does it say

Third-Party: 3.8 kilo ohms

Researcher: *...unintelligible...*

Participant: okay so that's close enough to 3.9. umm now I'm going to do the same again but with a different resistor. what does it say now

Third-Party: 0.99 kilo ohms

Participant: okay that's close enough to 1. and then *...unintelligible...* and now what does it say

Third-Party: 0.99 kilo ohms

Participant: okay that's close enough and *...unintelligible...* and what does it say now

Third-Party: 1.989 kilo ohms

Participant: okay perfect thank you. and then am I allowed to look at the function generator or not

Researcher: no

Participant: okay. umm is the function generator all plugged in

Researcher: yeah

Third-Party: yep

Participant: so on the function generator at the top it will say a long number in probably kilo hertz

Researcher: just say *...unintelligible...*

Participant: *...unintelligible...*

Researcher: pretend you're talking to the machine, something really dumb yeah

Participant: set the frequency to 5 kilo hertz

Third-Party: what channel

Participant: channel one. to 5 kilo hertz. and then adjust the amplitude to provide a 10 milli volt signal. set the function generator to a sine wave

Third-Party: yep

Participant: for channel two set the amplitude to 1 volt and set frequency to 5 kilo hertz

Third-Party: *...unintelligible...*

Participant: apply the signal at channel two to the external trigger

Researcher: that's done

Participant: okay uh set the oscilloscope to use the external trigger

Third-Party: done

Participant: connect channel one of the function generator to channel one of the oscilloscope

Researcher: done

Participant: done perfect. okay so just 9 now. switch on the power supply

Researcher: that's for you. you need to do the power supply

Participant: okay. there we go *...unintelligible...* so the oscilloscope is already connected to the function generator. is the oscilloscope still connected to the function generator.

Researcher: yeah

Participant: okay umm *...unintelligible...* set the frequency of signal one and two on the function generator to both equal 10 hertz.

Third-Party: which channel, sorry

Participant: both of them, channel one and two

Third-Party: 10 hertz

Participant: yeah

Third-Party: done

Participant: is TP3 the output of the circuit

Researcher: yeah

Participant: *...unintelligible...* connect the output two of the oscilloscope to test point 3

Researcher: done

Participant: set the oscilloscope to measure phase margin. phase difference sorry

Third-Party: what channel

Participant: for the difference between the channels

Third-Party: done

Participant: what is the phase difference

Third-Party: question mark

Note: This is what is displayed on the oscilloscope when the phase difference is immeasurable.

Participant: are both channels turned on on the oscilloscope

Third-Party: no

Participant: can you turn on channel two please

Third-Party: done

Participant: does it now have a phase difference that's not a question mark

Third-Party: no

Participant: still a question mark

Third-Party: yes

Participant: zoom in. sorry reduce the amplitude of the. yeah reduce the amplitude of the time base. no not the timebase, the voltage divisions.

Third-Party: by how much

Participant: continue reducing them until a clear wave is shown

Researcher: no

Participant: no

Researcher: numerical values

Participant: reduce it. what it is currently on

Third-Party: one volt for each channel one and channel two

Participant: okay reduce channel two to be half a volt and reduce channel one to be 50 milli volts

Third-Party: what was channel one sorry

Participant: 50 milli volts

Third-Party: done

Participant: adjust the time base to be around 10 milli seconds

Third-Party: done

Participant: what is the phase margin

Third-Party: question mark

Participant: is averaging turned on

Third-Party: no

Participant: can you turn on averaging please. are both the scopes set to be measuring at times one

Researcher: *...unintelligible...*

Third-Party: *...unintelligible...* yep it's times one now *...unintelligible...*

Participant: can you adjust the timebase down by one to make it one smaller. can you make it smaller 50 milli seconds per division

Third-Party: 15 or 50

Participant: 50

Third-Party: done

Participant: does it now have a clear phase margin

Third-Party: no

Researcher: you can take a screen shot if you want

Participant: yeah *...unintelligible...*

Participant: adjust the amplitude to be 100 millivolts per time division

Third-Party: which channel

Participant: one and two

Third-Party: what value was that again

Participant: 100 milli volts

Third-Party: done

Participant: what is the phase difference

Third-Party: it's varying from minus 2.16 degrees

Researcher: just minus 2.16 degrees

Participant: and can you set the oscilloscope to measure gain

Researcher: no. easiest thing to do is get the peak-to-peak of each one

Participant: okay yeah is there not a maths function. I thought there was a maths function on there that did gain.

Researcher: it's addition and subtraction.

Participant: oh right umm can you please umm measure the peak-to-peak voltage of both signals one and two.

Third-Party: yep. done

Participant: what is the peak to peak voltage of both

Third-Party: channel one peak-to-peak is 104 milli volts.

Participant: yep

Third-Party: and peak-to-peak of channel two is 1.01 volts.

Participant: *...unintelligible...* roughly 10. can you please adjust the function generator signal for both one and two to be 20 hertz.

Third-Party: done.

Participant: what the is the peak-to-peak voltage of both signals on the oscilloscope

Third-Party: umm channel one peak-to-peak is 104 millivolts

Participant: uhuh

Third-Party: peak-to-peak of channel two is 1.01 volts.

Participant: okay and what is the phase margin

Third-Party: minus 576 milli degrees

Participant: okay thank you. *...unintelligible...* can you change the signal on the function generator to be 100 hertz and adjust the timebase on the oscilloscope to be 50. 50 times bigger.

Third-Party: 50 times bigger.

Participant: no sorry 5 times bigger. what is the peak-to-peak voltage of both signals

Third-Party: peak-to-peak of channel one is 104 milli volts

Third-Party: channel two peak-to-peak is 1.01 volts

Participant: and what is the phase difference

Third-Party: minus 2.52 degrees

Participant: can you adjust the function generator to umm 200 hertz for both channel one and two

Third-Party: done

Participant: what is the peak-to-peak voltage of both the signals on the signal generator

Third-Party: channel one peak-to-peak is 104 milli volts

Third-Party: peak-to-peak on channel two is 1.01 volts

Participant: okay and what is the umm signal. what is the phase difference

Third-Party: minus 2.16 degrees

Part 2

Participant: okay umm set the function generator to have a umm it coming out of signals one channel one and two both to one milli volt and 5 kilohertz

Third-Party: one milli volt did you just say

Participant: that's what it says

Third-Party: okay

Researcher: one milli volt, 0 to peak

Participant: oh and set it to a pulse

Third-Party: oh yeah

Participant: and then ensure it's a pulse that's being produced. *...unintelligible...*

Participant: reduce the timebase uh by a factor of 100

Third-Party: *...unintelligible...* done

Participant: reduce the timebase by another factor of 100

Participant: reduce it by another factor of a thousand

Third-Party: done

Participant: okay so umm *...unintelligible...* reduce it by another factor of 100

Third-Party: done

Participant: turn off channel one uh channel two sorry

Third-Party: done

Participant: umm increase the umm decrease the *...unintelligible...*

Researcher: *...unintelligible...*

Participant: oh it is channel two I want. turn channel two back on

Third-Party: done

Participant: umm reduce the umm voltage divisions, sorry increase the voltage divisions by a factor of 100

Third-Party: *...unintelligible...*

Participant: for 10 just for channel two *...unintelligible...* uhh decrease the time division by a factor of a thousand

Third-Party: what was it on before, it was on that before wasn't it so

Participant: yeah

Third-Party: there done

Participant: decrease the time division by a factor of a hundred

Participant: increase them by a factor of 10 sorry

Third-Party: increase

Participant: by a factor of 10 yeah

Third-Party: of the timebase

Participant: yes

Third-Party: done

Participant: half decrease it by a factor of 2. set the origin of the signal down by 3 time bases

Participant: increase the umm. sorry decrease the voltage per. increase the voltage per division by a factor of 2

Participant: increase it by another factor of a half

Participant: yeah. umm set up the cursor to measure umm from half a time division from the origin

Third-Party: what type do you want

Participant: a time cursor, sorry for channel two

Third-Party: where do you want the first cursor

Participant: the first to be half a time division to the right of the origin, the centre of the yeah

Third-Party: this being the centre

Participant: yeah half to the right *...unintelligible...*

Participant: then set cursor two to be umm to uhh one time division, one and a half time divisions in from the left. so two and a half time divisions right of the origin

Third-Party: here

Participant: yeah *...unintelligible...* move the right hand time. the right hand time base to be a quarter of a time division to the left.

Third-Party: the right hand time base

Participant: the right hand sorry umm cursor a quarter of a time division to the left.

Third-Party: which cursor

Participant: the right

Third-Party: to which direction

Participant: left. no quarter of the little time bases. so there

Third-Party: there

Participant: yeah. umm *...unintelligible...* the difference between them is 0.96 milli seconds okay. *...unintelligible...* and then set the oscilloscope to trigger on the falling signal. *...unintelligible...*

Participant: measure the difference between the two cursors

Researcher: *...unintelligible...*

Participant: *...unintelligible...*

D.7 Commands

D.7.1 Digital Multimeter

Command	**Example Phrases**
Get Measurement	what is the measurement from the digital multimeter
	What's the reading on the multimeter
	what was the reading from the digital multimeter
	what measurement is the digital multimeter giving now
	what value is the digital multimeter giving now
	can you tell me what the dmm is reading please
Set Mode	Set the multimeter to measure resistance please
	can I have the dmm set to dc voltage please
	could I have it set to resistance
	could I have the digital multimeter set to voltage
Get Mode	what is the digital multimeter set to
	is it set to the resistance

D.7.2 Function Generator

Command	Example Phrases
Set Up Channel	Set channel one on the function generator to output a 100 milli volt 5 k sine wave set up a 1 volt square wave coming out of channel two and lock the frequency to channel one set up the function generator to provide a 100 milli volt, 5 kilo hertz sinewave from channel one please can the function generator provide 100 milli volts with a 5 kilo hertz sinewave from channel one set up the function generator for a 100 milli volt 5 kilo hertz sine wave from channel one change the frequency on channel one of the signal generator to 10 hertz change channel one frequency on the signal generator to 20 hertz change the frequency of the output on channel one on the signal generator to 20 hertz set the output frequency of channel one on the signal generator to 100 hertz set the output of channel one on the fg to an amplitude of 1 volt So on channel one of the function generator can you, can it be set to 10 hertz please. can I have the function generator set to 20 hertz please can I have the frequency lowered on channel one to 10 hertz please can you set up the function generator to provide a 1 volt square wave on channel two set up the function generator to produce a 1 volt square wave from channel two could you set the signal generator to 10 hertz could I have the signal generator set to 20 hertz can I have a one volt pulse wave from the signal generator from channel one.
Set Frequency Lock	lock the frequency to channel one

D.7.3 Oscilloscope

Command	Example Phrases
Get Measurement	what is the amplitude of the reading on channel two of the oscilloscope what is the reading on channel two amplitude what is the phase difference can I get the channel one reading for peak-to-peak on the oscilloscope can I have the oscilloscope set to measure the phase difference from channel two minus channel one please can you set the oscilloscope to measure the phase difference from channel two to channel one what is the peak-to-peak voltage of channel one what is the peak-to-peak voltage of channel two what is the phase difference from channel two to channel one what is the phase difference from channel two to channel one and whats the phase difference
Autoset	autorange autoset can you run autoset on the oscilloscope again please could you press autoset
Set Timebase	set the timebase to 200 microseconds per unit set the timebase to one tenth of a second I need the timebase to be set to 0.1 seconds please oscilloscope can we zoom the timebase in to its maximum scale please could I have the timebase on the oscilloscope set to 0.01 seconds per division could you increase the timebase by one increment clockwise could the scale be decreased anticlockwise by two increments
Set Trigger Channel	set the oscilloscope to use the external trigger
Get Trigger Channel	is the oscilloscope set to use the external trigger?
Set Trigger Mode	set the trigger to falling mode oh could I have the trigger set to falling for external

Table continued from previous page

Command	Example Phrases
Set Cursors	set a time cursor on the oscilloscope can I have a time cursor please on the Oscilloscope can I have the cursor function selected please can I have the second cursor set to the second division from the origin can the oscilloscope move the cursor half division anticlockwise set cursor one of the channel two to be at the origin. set the cursor two of the channel two on the oscilloscope to two micro seconds from the origin could I have the cursor for timebase for channel two could I have cursor one set on the third interval left of the origin could I have cursor two set at 1.5 micro seconds could cursor set on the origin could it be set on the origin where time is zero could cursor two set at two micro seconds in front of cursor one could I have cursor two set to the point where the delta voltage is 1.35
Default Setup	can I have the oscilloscope set to default setup please
Set Averaging	Can I have the oscilloscope set to 16 times averaging please. can you turn on averaging to times 16
Get Probe Attenuation	is the oscilloscope set for the probes at times one.
Set Probe Attenuation	can you set channel one to times one please make sure that each of the probes at times 1
Take Hardcopy	take a screen-shot of the oscilloscope please can I get a screen-shot of the oscilloscope please can I get a screen-shot from the oscilloscope take a screenshot
Start/Stop Oscilloscope	could you press run. could you hit stop

E Metric Study

E.1 Ethics Form

Physical Sciences Ethics Committee

FAST-TRACK ETHICAL APPROVAL FORM (STUDENTS)

This fast-track system for is for taught students only. Research students and staff must complete the full Ethical Approval Form.

If you answer **YES** to any of the following you must complete either this Fast-track ethical approval form, to be signed off by your supervisor, or a full Ethical approval application, to be approved by the Physical Science Ethics Committee (allow at least two weeks for this process).

Note that the outcome of the Fast-track system may result in you needing to complete a full ethical approval application.
Does your project involve any of the following?

	YES	NO
Human participants (adults or children)	✓	
Human material (e.g. tissue or fluid samples)		✓
Human data (e.g. surveys and questionnaires on issues such as lifestyle, housing and working environments, attitudes and preferences)		✓
Vertebrates, especially mammals and birds		✓
Any other organisms not previously mentioned		✓
Military or defence context		✓
Funding sources with potential to adversely affect existing relationships or bring the University or Department into disrepute.		✓
Restrictions on dissemination		✓
Overseas countries under regimes with poor human rights record or identified as dangerous by the Foreign & Commonwealth Office		✓
Applications that could potentially involve unethical practice, including potential dual-use applications which could be unethical		✓

Students: you should discuss the ethical considerations of your project with your project supervisor and, if necessary, fill in a full ethics form to be submitted to the Physical Sciences Ethics Committee.

Supervisors: Please ensure you are familiar with the University's 'Code of practice and principles for good ethical governance' in order to guide your student effectively. Please seek guidance from the Departmental Ethics Officer if you are uncertain about any ethical issue arising from this application.

THE UNIVERSITY of York

Physical Sciences Ethics Committee

FAST-TRACK ETHICAL APPROVAL FORM (STUDENTS)

Project Information:

Student Name: Chris Taylor

Course Title: Electronic Engineering

Tick one box:

Undergraduate project	☑	Postgraduate project	☐
Undergraduate module assignment	☐	Postgraduate module assignment	☐
Other (Please state..........................)	☐		

Title of project: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users

Project supervisor / module leader name: Tony Ward

Protocol:

a): *If you answer **NO** to any of the following you must submit a full ethical approval form*

	If you answer yes to any of the following, this must be explicit in any supporting literature (e.g. consent forms, information sheets and questionnaires)	YES	NO	N/A
1	Will you describe the procedures to participants in advance, so that they are informed about what to expect?	✓		
2	Will you tell participants that their participation is voluntary?	✓		
3	Will you inform the participants of the purpose / background of the study?	✓		
4	Will you obtain written consent for participation?	✓		
5	If the research is observational, will you ask participants for their consent to being observed?	✓		
6	Will you tell participants that they may withdraw from the research at any time and for any reason?	✓		
7	With questionnaires and interviews will you give participants the option of omitting questions they do not want to answer?	✓		
8	Will you tell participants that their data will be treated with full confidentiality and that, if published, it will not be identifiable as theirs?	✓		

THE UNIVERSITY *of York*

Physical Sciences Ethics Committee

Protocol:

b): *If you answer **YES** to any of the following you must submit a full ethical approval form.*

		YES	NO	N/A
9	Is your study designed to be challenging/disturbing (physically or psychologically)?		✓	
10	Will you deliberately mislead your participants?		✓	
11	Does your study involve taking bodily samples?		✓	
12	Is your study physically invasive?		✓	
13	Is there any obvious or inevitable adaptation of your research findings to ethically questionable aims?		✓	
14	Could the methodologies or findings of your study damage the reputation of the University of York?		✓	

Health and Safety:
Please identify any risks to the participants and state any precautions you will take to ensure their health and safety:

There are negligable risks as the participants will only be using a microphone and a desktop computer.

The participant will not be seated at the computer for an extended amount of time.

Participants: *If you answer **YES** to any of the following you must submit a full ethical approval form. If you have ticked **YES** to 15 and your participants are **patients**, in addition to the full ethical application you must follow the Guidelines for Ethical Approval of NHS Projects.*

			YES	NO	N/A
15	Does your project involve work with animals			✓	
16	Will any of the participants be from one of the following vulnerable groups? **Note that you may also need to obtain satisfactory DBS clearance (or equivalent for overseas students)**	Children under 18		✓	
		People with learning difficulties		✓	
		People who are unconscious or severely ill		✓	
		NHS patients		✓	
		Other vulnerable groups (specify)		✓	

THE UNIVERSITY *of York*

Data Protection: *If you answer **NO** to any of the following you must submit a full ethical approval form*

		YES	NO	N/A
17	Any personal / sensitive data will be stored in password protected folders on computers.	✓		
18	Any hard copies of personal data (including consent forms) will be stored in a secure place.	✓		
19	Only the student and supervisors will have access to the data generated from the study. (The supervisor may share the anonymised data with other researchers at the University of York)	✓		
20	The data will be preserved beyond the study in line with University policy and will be placed in the custody of the supervisor at the end of the project.	✓		
21	All data will be anonymised prior to analysis. Please state your method of anonymisation: No indentifiable data will be taken. All inadvertant data provided will be removed from the report.	✓		

FOR THE STUDENT TO COMPLETE:

Please complete and sign the following section and submit to your supervisor alongside any supporting documentation (this includes consent forms, information sheets and questionnaires where necessary).

Provide a brief summary of the participants and procedures of your project (max 100 words)

The participant will be asked to read a set of phrases off a screen into a microphone. The recording will then be sent to a range of Natural Language Processing platforms. The participant will then be asked to make sure that the platforms correctly parsed the commands and the result will be recorded.

THE UNIVERSITY *of York*

Physical Sciences Ethics Committee

I have considered the ethical implications of this project and have identified no significant ethical implications requiring a full ethics submission to the Physical Sciences Ethics Committee ☑

I have included all relevant paperwork (e.g. consent form, information sheet, questionnaire/interview schedules) with this application ☑

Signed.. Print name CHRIS TAYLOR Date 05/02/18
(Student)

FOR THE SUPERVISOR TO COMPLETE:
By signing this form you are taking responsibility for the ethical conduct of this project

The student has taken all reasonable steps to ensure ethical practice in this study and I can identify no significant ethical implications requiring a full ethics submission to the Physical Sciences Ethics Committee ☑

I have checked and approved all relevant paperwork required for this proposal ☑

STATEMENT OF ETHICAL APPROVAL

This project has been considered using the Physical Sciences Ethics Committee Fast-track ethical approval procedure, agreed by the Physical Sciences Ethics Committee of the University of York, and is now approved.

Signed.. Print name A. WARD Date 9/2/18
(Supervisor/Module leader)

OR

The details on this form indicate a need for a full application to PSEC. The practical aspects of this project will not proceed until this has application has been approved.

Signed.. Print name.. Date...............
(Supervisor/Module leader)

THE UNIVERSITY *of York*

E.2 Participant Consent Form

Consent Form for Participation in User Testing

Institution: University of York
Researcher: Chris Taylor (cjt534@york.ac.uk)
Supervisor: Tony Ward (tony.ward@york.ac.uk)
Project: Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users

Brief Description of Study

The purpose of this study is to investigate the accuracy and response time of two online NLP[1] platforms. The results of which will be used to decide which platform to use in the final product.

Procedure

The participant will be asked to say a series of commands that will be processed by each platform. The participant will be asked whether or not the platform correctly interpreted what they said. The recordings will not be stored after the study has finished and all identifiable information will be omitted from the transcription. The transcription will be submitted alongside the report to the Department of Electronic Engineering at the University of York.

Participant data will remain anonymous at all times and will only be used for research purposes. Participants are able to withdraw from the experiment at any time without stating a reason. If a participant decides to withdraw, their recorded data will be destroyed and not used in the study. The study is expected to take around 20 minutes and will be conducted in the 4th floor laboratory of the Department of Electronic Engineering at the University of York.

Risks

All risks for this study have been considered and are negligible as the only equipment the participant will be using is a microphone and a PC keyboard.

Consent

I have been informed about the aims and procedures involved in the experiment I am about to participate in. The data collected as part of the experiment can be used for research and all data will be kept anonymous. I reserve the right to withdraw at any stage in the proceedings. I understand that any information that I have provided as part of the study will be destroyed or my identity removed unless I agree otherwise.

Name:

Signed:

Date:

[1] Natural Language Processing

1

E.3 Participant Instructions

Participant Instructions for User Study for "Adding Voice Control and Audible Feedback to Electronic Lab Test Equipment for Visually or Physically Impaired Users"

Make sure you have read, understood, and signed the consent form before starting.

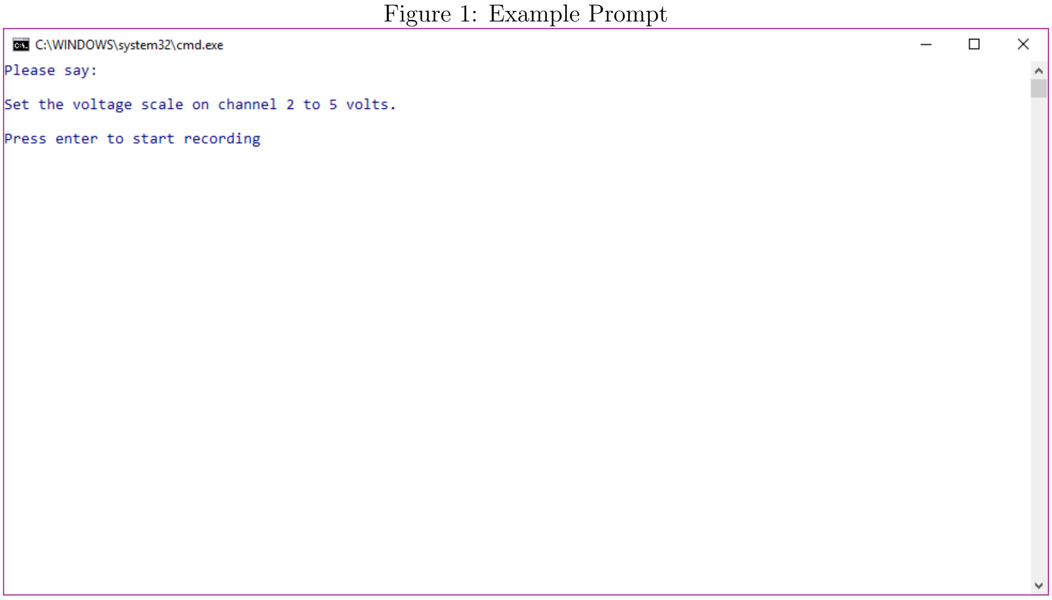
Please read through this document and let the researcher know when you are ready to begin.

For this study you will need to read a prompt off the screen and say it aloud into a microphone. Figure 1 shows an example prompt. Please make sure to speak clearly and at a natural rate. After you have finished speaking each prompt, the software will automatically stop recording and process the response. After each prompt, the software will show you two responses, relating to two different NLP[1] platforms. Please indicate if the responses are correct by entering the marked letter and pressing the return key. You may also skip the prompt if you spoke incorrectly. Figures 2 and 3 show example correct and incorrect responses.

The study is complete when there are no more prompts.

Please let the researcher know when you are completely done. You may now take this time to ask the researcher any questions or provide feedback about the study.

Figure 1: Example Prompt



[1] Natural Language Processing

Figure 2: Example Correct Response [2]

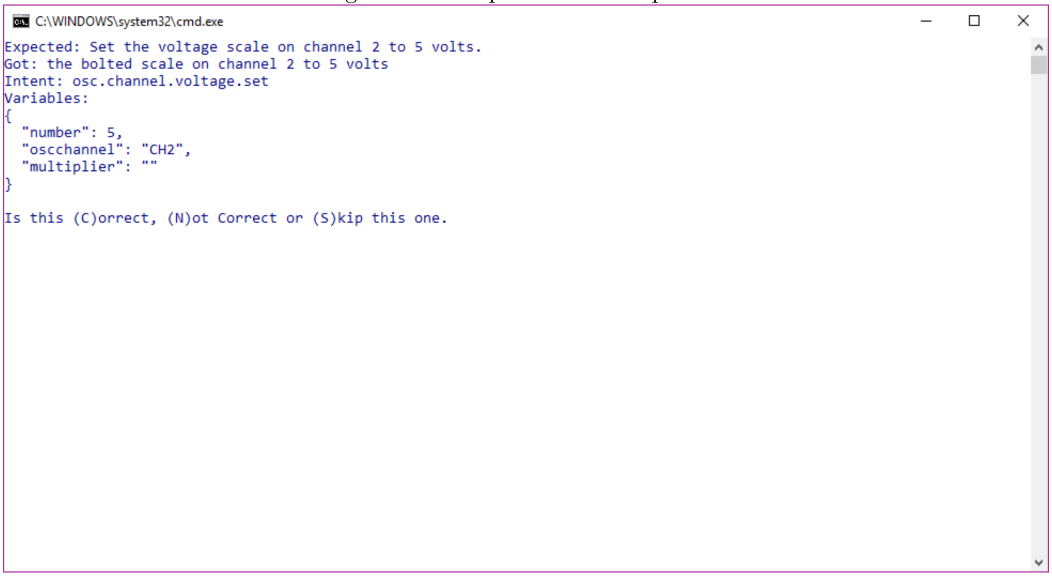
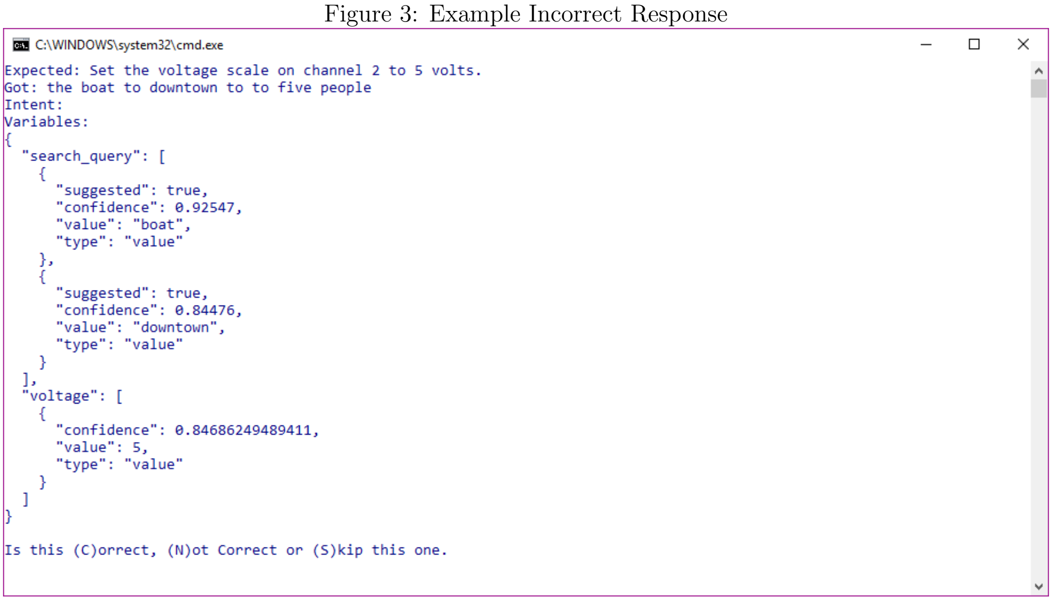


Figure 3: Example Incorrect Response



[2]Note that although the text hasn't been 100% correctly transcribed, the platform still parsed the required data and intent correctly.

Page 2 of 2

E.4 Metric Analysis Raw Results

E.4.1 Test #1

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	3723	What's the duty cycle for channel one?	duty cycle to channel one	fg.channel.duty.get fg.channel.duty.set
WIT	FALSE	3948	Set the oscilloscope averaging to times 4.	what is a averaging two point four	osc.averaging.set osc.averaging.set
WIT	FALSE	2806	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel two to one hundred million	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3268	Channel 2 frequency to 100 pico Hertz.	channel two frequency to one hundred paco	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3464	Set the osc channel one probe to times 1.	set the telescope channel one pro two times one	osc.probe.att.set osc.probe.att.set
WIT	FALSE	3300	Set the voltage scale on channel two to five volts.	the voltage scale one shown to to five people	osc.channel.voltage.set osc.channel.voltage.set
WIT	FALSE	4804	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	stop the function generator to provide a one hundred mill of all five khz going wife from channel one	fg.channel.set
WIT	FALSE	3048	Channel one frequency to 100 pico.	no one frequency to one hundred because	fg.channel.frequency.set
WIT	FALSE	4034	Voltage offset for channel two to 50 milli volts.	could you also let the channel to fifty ball	osc.channel.voltage.offset.set
WIT	FALSE	2494	Switch off frequency lock.	tell frequency lol	fg.freq.lock.set fg.freq.lock.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	4007	What's the voltage offset for channel one on the oscilloscope?	the voltage offset the trunk one on the alaska	osc.channel.voltage.offset.get
WIT	FALSE	4300	Set the frequency of channel 2 to 0.5 Hertz.	but the frequency of town to to know point five how	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	2504	Turn off the oscilloscope.	turn off the lights go	device.off device.off
WIT	FALSE	2904	Channel 1 frequency to 100 Mega Hertz.	channel one frequency to one hundred like a	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3222	Timebase to 1 second.	play one second	osc.timebase.set
WIT	FALSE	2696	Switch on the oscilloscope.	switch on the telescope	device.on device.on
WIT	FALSE	3045	Set timebase to 50 micro seconds.	set the time they to fifty my purse i can	osc.timebase.set
WIT	FALSE	2981	Set the voltage offset for channel one to 5 volts.	the voltage offset to channel one to five people	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
WIT	FALSE	3650	Channel one osc probe to times 1.	can one go pro times one	osc.probe.att.set osc.probe.att.set
WIT	TRUE	2635	Set averaging to times 16.	averaging two times sixteen	osc.averaging.set osc.averaging.set
WIT	TRUE	2900	What's the dmm reading?	what's the pin i'm reading	dmm.value.get dmm.value.get

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	TRUE	3045	Set the voltage scale of channel two to five.	the voltage scale channel two to five	osc.channel.voltage.set osc.channel.voltage.set
WIT	TRUE	2656	What's the timebase?	what's the time by	osc.timebase.get osc.timebase.get
WIT	TRUE	2727	Set the trigger to channel two.	but the trigger to channel two	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3498	What's the voltage on the dmm?	the voltage on the den and	dmm.value.get dmm.value.get
WIT	TRUE	2327	The trigger to channel one.	trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	2451	Switch the trigger to channel one.	what's the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3060	What's the measurement on the dmm?	what's the measurement on monday and then	dmm.value.get dmm.value.get
WIT	TRUE	2395	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	FALSE	2862	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel to two 100ml	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2715	Channel 2 frequency to 100 pico Hertz.	channel to frequency to 100 Peco Cox	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2181	Set averaging to times 16.	averaging 2 x 16	osc.averaging.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	2421	Set the voltage offset for channel one to 5 volts.	the voltage offset for Channel 125 volts	osc.channel.voltage.offset.set fg.channel.frequency.set
DIA	FALSE	2307	Channel one osc probe to times 1.	channel 1 oscilloscope probe 2 x 1	osc.probe.att.set fg.channel.frequency.set
DIA	FALSE	1824	What's the timebase?	what's the time base	osc.timebase.get
DIA	FALSE	3029	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	top the function generator to provide a 100mm volt 5 kilohertz sine wave from channel one	fg.channel.set fg.channel.set
DIA	FALSE	2570	Voltage offset for channel two to 50 milli volts.	page offset for Channel to 250 volts	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
DIA	FALSE	1983	Switch off frequency lock.	child frequency lock	fg.freq.lock.set osc.channel.voltage.offset.set
DIA	FALSE	2058	Switch the trigger to channel one.	it's the trigger to channel one	osc.trigger.channel.set osc.channel.voltage.offset.set
DIA	FALSE	3309	Set the frequency of channel 2 to 0.5 Hertz.	at the frequency of channel to 20.5 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2245	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	1438	Timebase to 1 second.	PL21 2nd	osc.timebase.set
DIA	FALSE	1916	Set timebase to 50 micro seconds.	set the time base 250 microseconds	osc.timebase.set osc.timebase.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2292	What's the duty cycle for channel one?	Duty cycle for channel one	fg.channel.duty.get fg.channel.duty.get
DIA	TRUE	2467	Set the oscilloscope averaging to times 4.	if the oscilloscope averaging 2 x 4	osc.averaging.set osc.averaging.set
DIA	TRUE	2012	What's the dmm reading?	what's the dmm Reading	dmm.value.get dmm.value.get
DIA	TRUE	2452	Set the osc channel one probe to times 1.	set the oscilloscope channel one probe 2 x 1	osc.probe.att.set osc.probe.att.set
DIA	TRUE	2379	Set the voltage scale of channel two to five.	the voltage scale of channel 2 to 5	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	1559	Set the trigger to channel two.	but the Trigger 2 channel 2	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2340	Set the voltage scale on channel two to five volts.	the voltage scale on channel 2 to 5 volts	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	2216	Channel one frequency to 100 pico.	channel 1 frequency to 100 Pico	fg.channel.frequency.set fg.channel.frequency.set
DIA	TRUE	2014	What's the voltage on the dmm?	what's the voltage on the dnm	dmm.value.get dmm.value.get
DIA	TRUE	1489	The trigger to channel one.	Trigger 2 Channel 1	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	1921	What's the voltage offset for channel one on the oscilloscope?	what's the voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	1920	Turn off the oscilloscope.	turn off the oscilloscope	device.off device.off
DIA	TRUE	2145	What's the measurement on the dmm?	what's the measurement on the Deer man	dmm.value.get dmm.value.get
DIA	TRUE	1929	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	2265	Switch on the oscilloscope.	pitch on the oscilloscope	device.on device.on

E.4.2 Test #2

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	1964	What's the voltage on the dmm?	what's the voltage on the dmm	dmm.value.get dmm.value.get
DIA	TRUE	1827	Switch off frequency lock.	switch off frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	1856	What's the dmm reading?	what's the dmm Reading	dmm.value.get dmm.value.get
DIA	TRUE	1939	What's the measurement on the dmm?	what's the measurement on the dnm	dmm.value.get dmm.value.get
DIA	FALSE	2227	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	3058	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel 2 to 2 to 100 minutes	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2236	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel 22 0.5 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2503	Turn on frequency lock.	turn on 6 o'clock	fg.freq.lock.set device.on
DIA	FALSE	2200	Channel 2 frequency to 100 pico Hertz.	channel to frequency to 100pk Hz	fg.channel.frequency.set device.on
DIA	FALSE	2903	Set the oscilloscope averaging to times 4.	set for a telescope Atherton 2 x 4	osc.averaging.set device.on
DIA	FALSE	1990	What's the duty cycle for channel one?	what's the duty cycle for Terminal 1	fg.channel.duty.get device.on
DIA	FALSE	2208	Set the frequency of channel 1 to 50 Hertz.	set the frequency of Channel 12 50 Hz	fg.channel.frequency.set device.on
DIA	FALSE	3394	Set the osc channel one probe to times 1.	set velocity channel 142 x 1	osc.probe.att.set device.on
DIA	FALSE	1780	What's the timebase?	what's the time base	osc.timebase.get device.on
DIA	FALSE	3540	What's the voltage offset for channel one on the oscilloscope?	what's the voltage of set for 1001 on the oscilloscope	osc.channel.voltage.offset.get device.on
DIA	FALSE	2511	Channel one frequency to 100 pico.	channel 1 frequency to 100 people	fg.channel.frequency.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	2625	Set the voltage scale on channel two to five volts.	set default scale on terminal 2 to 5 volts	osc.channel.voltage.set device.off
DIA	FALSE	2434	Voltage offset for channel two to 50 milli volts.	voltage offset for tunnel 2 to 250 mm bolts	osc.channel.voltage.offset.set device.off
DIA	FALSE	2304	The trigger to channel one.	the ticket to channel one	osc.trigger.channel.set device.off
DIA	FALSE	1979	Set timebase to 50 micro seconds.	set time based 250 microseconds	osc.timebase.set device.off
DIA	FALSE	2312	Switch the trigger to channel one.	switch for ticket to Terminal 1	osc.trigger.channel.set device.off
DIA	FALSE	3463	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	setup.cfg to Friday 100ml volt 5 kilohertz sine wave channel one	fg.channel.set device.off
DIA	FALSE	1813	Switch on the oscilloscope.	Switzerland vs scope	device.on device.off
DIA	FALSE	2164	Set the trigger to channel two.	set for to get to Terminal 2	osc.trigger.channel.set device.off
DIA	FALSE	3396	Set the voltage offset for channel one to 5 volts.	search for photos of settle tunnel 125 faults	osc.channel.voltage.offset.set Default Fallback Intent
DIA	FALSE	2005	Set averaging to times 16.	set African 2 x 16	osc.averaging.set Default Fallback Intent
DIA	FALSE	1955	Timebase to 1 second.	time based to one second	osc.timebase.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	1975	Set the voltage scale of channel two to five.	set default scale of 10:02 to 5	osc.channel.voltage.set osc.channel.voltage.set
DIA	FALSE	3700	Channel one osc probe to times 1.	Tunnel Wallasey pub 2 X phone	osc.probe.att.set osc.channel.voltage.set
WIT	TRUE	3512	What's the timebase?	what's the time they	osc.timebase.get osc.timebase.get
WIT	TRUE	5718	Set the voltage scale on channel two to five volts.	set to full scale on channel two to five so	osc.channel.voltage.set osc.channel.voltage.set
WIT	TRUE	4546	What's the measurement on the dmm?	what the measurement from the end and	dmm.value.get dmm.value.get
WIT	FALSE	5942	Turn on frequency lock.	no frequency lock	fg.freq.lock.set fg.freq.lock.set
WIT	FALSE	4274	Channel 2 frequency to 100 pico Hertz.	turn to frequency to one hundred people who	fg.channel.frequency.set
WIT	FALSE	4106	Channel 1 frequency to 100 Mega Hertz.	no one frequency to one hundred mhz	fg.channel.frequency.set
WIT	FALSE	6082	Set the oscilloscope averaging to times 4.	set the telescope oftentimes full	osc.averaging.set
WIT	FALSE	3781	What's the duty cycle for channel one?	what's up cycle ten of one	fg.channel.duty.get
WIT	FALSE	4629	Set the frequency of channel 1 to 50 Hertz.	set the frequency of the one to fifty	fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	4506	Set the osc channel one probe to times 1.	set of the channel one for old times one	osc.probe.att.set
WIT	FALSE	6107	What's the voltage offset for channel one on the oscilloscope?	what's the voltage of that with no one from a traditional	osc.channel.voltage.offset.get dmm.value.get
WIT	FALSE	5839	What's the voltage on the dmm?	what voltage from pm and	dmm.value.get
WIT	FALSE	4052	Switch off frequency lock.	switch off because the lock	fg.freq.lock.set
WIT	FALSE	4492	Channel one frequency to 100 pico.	no one frequency to one hundred people	fg.channel.frequency.set
WIT	FALSE	6238	Voltage offset for channel two to 50 milli volts.	hope it's all set for ten to the two fifty model	osc.channel.voltage.offset.set
WIT	FALSE	3092	The trigger to channel one.	the ticket to channel one to three	osc.trigger.channel.set
WIT	FALSE	3240	Set timebase to 50 micro seconds.	set time date to fifty michael second third	osc.timebase.set
WIT	FALSE	3918	Switch the trigger to channel one.	switch the ticket to one	osc.trigger.channel.set
WIT	FALSE	6014	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	to where is he to find the one hundred many volts five kids ahead of time a from china one	fg.channel.set
WIT	FALSE	3850	Switch on the oscilloscope.	switch on the digital	device.on

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	3804	What's the dmm reading?	what city am and leading	dmm.value.get
WIT	FALSE	5218	Set the trigger to channel two.	to get to channel two	osc.trigger.channel.set
WIT	FALSE	6227	Set the voltage offset for channel one to 5 volts.	that's a full of that no one to five will not	osc.channel.voltage.offset.set
WIT	FALSE	4441	Set averaging to times 16.	set up a meeting time sixteen the	osc.averaging.set
WIT	FALSE	4939	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel two to two to one hundred twenty had	fg.channel.frequency.set
WIT	FALSE	4892	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of tendencies to know point five	fg.channel.frequency.set
WIT	FALSE	3630	Timebase to 1 second.	time they c section three	osc.timebase.set
WIT	FALSE	3839	Set the voltage scale of channel two to five.	set the whole scale of ten to two five	osc.channel.voltage.set osc.channel.voltage.set
WIT	FALSE	3918	Channel one osc probe to times 1.	one of the pope two times one	osc.probe.att.set

E.4.3 Test #3

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	1959	Turn off the oscilloscope.	turn off the oscilloscope	device.off device.off
DIA	TRUE	2234	Set the voltage scale of channel two to five.	set the voltage scale of channel 2 to 5	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	2153	Set the oscilloscope averaging to times 4.	set the oscilloscope averaging 2 x 4	osc.averaging.set osc.averaging.set
DIA	TRUE	1904	Switch the trigger to channel one.	switch the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2144	Set the osc channel one probe to times 1.	set the off channel 1 probe 2 x 1	osc.probe.att.set osc.probe.att.set
DIA	TRUE	1811	What's the dmm reading?	what's the dmm Reading	dmm.value.get dmm.value.get
DIA	TRUE	1859	Set the trigger to channel two.	set the trigger to channel 2	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	1867	Switch off frequency lock.	switch off frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	1921	What's the voltage on the dmm?	what's the voltage on the dmm	dmm.value.get dmm.value.get
DIA	TRUE	1809	Switch on the oscilloscope.	switch on the oscilloscope	device.on device.on
DIA	TRUE	2137	Set the voltage scale on channel two to five volts.	set the voltage scale on channel 2 to 5 volts	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	1859	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	1902	What's the duty cycle for channel one?	what's the duty cycle for channel one	fg.channel.duty.get fg.channel.duty.get
DIA	TRUE	1878	Timebase to 1 second.	timebase to 1 second	osc.timebase.set osc.timebase.set
DIA	TRUE	2144	What's the voltage offset for channel one on the oscilloscope?	what's the voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
DIA	TRUE	1925	Set averaging to times 16.	set averaging 2 x 16	osc.averaging.set osc.averaging.set
DIA	TRUE	1840	What's the measurement on the dmm?	what's the measurement on the dmm	dmm.value.get dmm.value.get
DIA	TRUE	1775	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	FALSE	2529	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	1917	What's the timebase?	what's the time base	osc.timebase.get Default Fallback Intent
DIA	FALSE	2263	Voltage offset for channel two to 50 milli volts.	voltage offset for channel 2 to 15 minutes	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
DIA	FALSE	2898	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the function generator to provide a 100 mg 5 kilohertz sine wave from channel one	fg.channel.set fg.channel.set
DIA	FALSE	2163	Set the voltage offset for channel one to 5 volts.	set the voltage offset for Channel 125 volts	osc.channel.voltage.offset.set osc.channel.voltage.offset.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	2803	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel to 20.5 Hz	fg.channel.frequency.set osc.channel.voltage.offset.set
DIA	FALSE	2047	Set timebase to 50 micro seconds.	set time based 250 microseconds	osc.timebase.set osc.channel.voltage.offset.set
DIA	FALSE	2233	Set the frequency of channel 1 to 50 Hertz.	set the frequency of Channel 12 50 Hz	fg.channel.frequency.set osc.channel.voltage.offset.set
DIA	FALSE	2379	Channel 2 frequency to 100 pico Hertz.	penalty frequency T100 pico Hz	fg.channel.frequency.set osc.channel.voltage.offset.set
DIA	FALSE	2350	Channel one osc probe to times 1.	channel 1 oscilloscope probe 2 x 1	osc.probe.att.set osc.channel.voltage.offset.set
DIA	FALSE	2282	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel 2 to 100 minutes	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	1947	Channel one frequency to 100 pico.	channel 1 frequency to 100pk	fg.channel.frequency.set fg.channel.frequency.set
WIT	TRUE	3824	Switch the trigger to channel one.	switch the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3917	Set the osc channel one probe to times 1.	set the channel one pray to times one	osc.probe.att.set osc.probe.att.set
WIT	TRUE	3900	Set the trigger to channel two.	set the trigger the channel to	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3733	Switch off frequency lock.	switch off the frequency look	fg.freq.lock.set fg.freq.lock.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	TRUE	2968	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3128	Turn on frequency lock.	turn on frequency look	fg.freq.lock.set fg.freq.lock.set
WIT	TRUE	3400	What's the timebase?	what's the time they think	osc.timebase.get osc.timebase.get
WIT	TRUE	3197	What's the dmm reading?	what's the d and i'm reading	dmm.value.get dmm.value.get
WIT	TRUE	3388	What's the voltage on the dmm?	what's the voltage on the m and	dmm.value.get dmm.value.get
WIT	TRUE	3223	Set the frequency of channel 1 to 50 Hertz.	set the frequency of channel one to fifty her	fg.channel.frequency.set fg.channel.frequency.set
WIT	TRUE	3025	What's the measurement on the dmm?	what's the measurement on the damn m three	dmm.value.get dmm.value.get
WIT	FALSE	5052	Turn off the oscilloscope.	turn off the elevator	device.off device.off
WIT	FALSE	7788	Channel 1 frequency to 100 Mega Hertz.	can one frequency one hundred make	fg.channel.frequency.set
WIT	FALSE	3134	Set the voltage scale of channel two to five.	set the voltage scale of channel tv to five	osc.channel.voltage.set osc.channel.voltage.set
WIT	FALSE	4368	Voltage offset for channel two to 50 milli volts.	voltage of that which not to fifty million vote	osc.channel.voltage.offset.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	4144	Set the oscilloscope averaging to times 4.	set the escape averaging two times will	osc.averaging.set osc.averaging.set
WIT	FALSE	4370	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the function generated to provide a one hundred milli vote five khz finally from channel one	fg.channel.set
WIT	FALSE	3935	Switch on the oscilloscope.	switch on the light	device.on device.on
WIT	FALSE	5990	Set the voltage scale on channel two to five volts.	set the voltage scale internal taste five fl	osc.channel.voltage.set osc.channel.voltage.set
WIT	FALSE	3966	Set the voltage offset for channel one to 5 volts.	set the voltage of that for general one to five	osc.channel.voltage.offset.set
WIT	FALSE	6995	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel tv to their point five so hot	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	4488	Set timebase to 50 micro seconds.	set time based fifty microseconds	osc.timebase.set
WIT	FALSE	3941	Channel 2 frequency to 100 pico Hertz.	tell the frequency to one hundred because	fg.channel.frequency.set
WIT	FALSE	5103	Channel one osc probe to times 1.	can one of a private times one	osc.probe.att.set
WIT	FALSE	3348	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel tv to one hundred million s three	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3144	What's the duty cycle for channel one?	what's the cycle to channel one	fg.channel.duty.get

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	3154	Timebase to 1 second.	time based on second thought	osc.timebase.set
WIT	FALSE	3385	What's the voltage offset for channel one on the oscilloscope?	what's the voltage offset the channel one on the interstate	osc.channel.voltage.offset.get dmm.value.get
WIT	FALSE	3024	Channel one frequency to 100 pico.	channel one frequency to one hundred pk	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	4993	Set averaging to times 16.	set for jim to times sixteen	osc.averaging.set

E.4.4 Test #4

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	3402	Set the voltage scale on channel two to five volts.	set the voltage scale on channel 2 to 5 volts	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	1966	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2256	Switch the trigger to channel one.	switch the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2228	Timebase to 1 second.	timebase to 1 second	osc.timebase.set osc.timebase.set
DIA	TRUE	2353	What's the duty cycle for channel one?	what's the duty cycle for channel one	fg.channel.duty.get fg.channel.duty.get

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2463	Channel one frequency to 100 pico.	channel 1 frequency to 100 Pico	fg.channel.frequency.set fg.channel.frequency.set
DIA	TRUE	2750	Switch on the oscilloscope.	switch on the oscilloscope	device.on device.on
DIA	TRUE	2160	Set the trigger to channel two.	set the trigger to channel 2	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2171	Set the voltage scale of channel two to five.	set the voltage scale of channel 2 to 5	osc.channel.voltage.set osc.channel.voltage.set
DIA	TRUE	2139	Switch off frequency lock.	switch off frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	2154	What's the voltage on the dmm?	what's the voltage on the dmm	dmm.value.get dmm.value.get
DIA	TRUE	1911	What's the dmm reading?	what's the dmm Reading	dmm.value.get dmm.value.get
DIA	TRUE	1955	Turn off the oscilloscope.	turn off the oscilloscope	device.off device.off
DIA	TRUE	3198	What's the voltage offset for channel one on the oscilloscope?	what's that voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
DIA	TRUE	1957	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	2693	Set averaging to times 16.	set averaging 2 x 16	osc.averaging.set osc.averaging.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2820	Set the osc channel one probe to times 1.	set the oscilloscope channel one probe 2 x 1	osc.probe.att.set osc.probe.att.set
DIA	TRUE	2564	Set the oscilloscope averaging to times 4.	set the oscilloscope averaging 2 x 4	osc.averaging.set osc.averaging.set
DIA	FALSE	2461	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2828	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel to 20.5 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2548	Channel one osc probe to times 1.	channel 1 oscilloscope probe 2 x 1	osc.probe.att.set fg.channel.frequency.set
DIA	FALSE	2282	Set timebase to 50 micro seconds.	set the time base 250 micro seconds	osc.timebase.set osc.timebase.set
DIA	FALSE	2500	Set the voltage offset for channel one to 5 volts.	set voltage offset for Channel 125 volts	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
DIA	FALSE	3420	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the function generator to provide a 100mm volt 5 kilohertz sine wave from channel one	fg.channel.set osc.channel.voltage.offset.set
DIA	FALSE	1937	What's the timebase?	what's the time base	osc.timebase.get Default Fallback Intent
DIA	FALSE	2436	Set the frequency of channel 1 to 50 Hertz.	set the frequency of Channel 12 50 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2517	Channel 2 frequency to 100 pico Hertz.	channel to frequency to 100 Peco hertz	fg.channel.frequency.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	2310	What's the measurement on the dmm?	what's the measurement on the deed mmm	dmm.value.get fg.channel.frequency.set
DIA	FALSE	2728	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel to two 100ml	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2700	Voltage offset for channel two to 50 milli volts.	voltage offset for Channel to 250 mg	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
WIT	TRUE	4868	Switch the trigger to channel one.	switch the trigger channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	4062	What's the voltage on the dmm?	what's the voltage on the d and and	dmm.value.get dmm.value.get
WIT	TRUE	3422	Turn on frequency lock.	turn on the frequency lol	fg.freq.lock.set fg.freq.lock.set
WIT	FALSE	7381	Set the voltage scale on channel two to five volts.	set the voltage dial on time okay to size xl	osc.channel.voltage.set
WIT	FALSE	38891	The trigger to channel one.		osc.trigger.channel.set
WIT	FALSE	4503	Set averaging to times 16.	set average intertwined sixteen	osc.averaging.set
WIT	FALSE	4661	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel days to zero point five	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	4179	Channel one osc probe to times 1.	can no one will escape route to tines want	osc.probe.att.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	5523	Set the osc channel one probe to times 1.	that being said gaetano worn pray to times one	osc.probe.att.set
WIT	FALSE	4208	Set timebase to 50 micro seconds.	set the time they fifty micro second	osc.timebase.set
WIT	FALSE	3887	Channel 1 frequency to 100 Mega Hertz.	channel one frequency to one hundred matter how	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3446	Timebase to 1 second.	time they to one second	osc.timebase.set
WIT	FALSE	3932	What's the duty cycle for channel one?	what are the cycle for channel one the	fg.channel.duty.get fg.channel.duty.get
WIT	FALSE	4554	Channel one frequency to 100 pico.	can no one frequency of one hundred p cuz will	fg.channel.frequency.set
WIT	FALSE	5152	Set the voltage offset for channel one to 5 volts.	set the voltage of that the channel one to five so	osc.channel.voltage.offset.set osc.channel.voltage.set
WIT	FALSE	6295	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the function generator to provide a one hundred ml of old five kilo fine life from channel one	fg.channel.set
WIT	FALSE	2918	What's the timebase?	what's the time they	osc.timebase.get osc.timebase.get
WIT	FALSE	4047	Set the frequency of channel 1 to 50 Hertz.	set the frequency of channel one to fifty her	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	4619	Channel 2 frequency to 100 pico Hertz.	cannot say frequency to one hundred p kyle	fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	5279	Switch on the oscilloscope.	which film this guy	device.on
WIT	FALSE	3280	What's the measurement on the dmm?	what's a measure man on the deed and and	dmm.value.get dmm.value.get
WIT	FALSE	5715	Set the trigger to channel two.		osc.trigger.channel.set
WIT	FALSE	3518	Set the frequency of channel 2 to 100 milli hertz.		fg.channel.frequency.set
WIT	FALSE	5520	Set the voltage scale of channel two to five.	thats not scale of channel two days to thaw a	osc.channel.voltage.set
WIT	FALSE	5550	Switch off frequency lock.	switch off frequency lol	fg.freq.lock.set fg.freq.lock.set
WIT	FALSE	6824	Set the oscilloscope averaging to times 4.	set the date on fridge and keep in full	osc.averaging.set
WIT	FALSE	3172	What's the dmm reading?	what's the d and and reading	dmm.value.get dmm.value.get
WIT	FALSE	6254	Voltage offset for channel two to 50 milli volts.	go to that the channel two days to fifteen illegal	osc.channel.voltage.offset.set
WIT	FALSE	4924	Turn off the oscilloscope.	turn all the fullest i	device.off
WIT	FALSE	6265	What's the voltage offset for channel one on the oscilloscope?	what voltage of that the channel warm on the interstate	osc.channel.voltage.offset.get dmm.value.get

E.4.5 Test #5

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	2535	Set the voltage scale on channel two to five volts.	check the voltage scale on channel 2 to 5 volts	osc.channel.voltage.set osc.channel.voltage.set
DIA	FALSE	1975	Turn off the oscilloscope.	off the oscilloscope	device.off device.off
DIA	FALSE	2299	Set the voltage scale of channel two to five.	the voltage scale of channel 2 to 5	osc.channel.voltage.set osc.channel.voltage.set
DIA	FALSE	2298	Set averaging to times 16.	set averaging 2 x 16	osc.averaging.set
DIA	FALSE	2874	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel to 20.5 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2428	Set timebase to 50 micro seconds.	set time base 250 microseconds	osc.timebase.set fg.channel.frequency.set
DIA	FALSE	2591	Set the oscilloscope averaging to times 4.	set the oscilloscope averaging 2 x 4	osc.averaging.set fg.channel.frequency.set
DIA	FALSE	3033	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel 2 to 100 minutes	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2208	What's the voltage on the dmm?	what's the voltage on the dnm	dmm.value.get dmm.value.get
DIA	FALSE	2255	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2657	Voltage offset for channel two to 50 milli volts.	voltage offset for channel 2 to 15 minutes	osc.channel.voltage.offset.set osc.channel.voltage.offset.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	1970	What's the timebase?	what's the time base	osc.timebase.get fg.channel.frequency.set
DIA	FALSE	2005	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set fg.channel.frequency.set
DIA	FALSE	2422	Set the frequency of channel 1 to 50 Hertz.	set the frequency of Channel 12 50 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2475	Channel one osc probe to times 1.	hello won Oscar Pro 2 x 1	osc.probe.att.set fg.channel.frequency.set
DIA	FALSE	2141	Timebase to 1 second.	timebase to 1 second	osc.timebase.set fg.channel.frequency.set
DIA	FALSE	2772	Set the voltage offset for channel one to 5 volts.	set the voltage offset for Channel 125 volt	osc.channel.voltage.offset.set fg.channel.frequency.set
DIA	FALSE	3089	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the voltage generator to provide 100mg 5 kilohertz sine wave from channel one	fg.channel.set fg.channel.set
DIA	FALSE	2393	What's the voltage offset for channel one on the oscilloscope?	what's the voltage offset for Channel wand on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
DIA	FALSE	2370	Set the osc channel one probe to times 1.	set the hospital one probe 2 x 1	osc.probe.att.set osc.channel.voltage.offset.get
DIA	TRUE	1954	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	2177	Set the trigger to channel two.	set the trigger to channel 2	osc.trigger.channel.set osc.trigger.channel.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2334	Channel one frequency to 100 pico.	channel 1 frequency to 100 Pico	fg.channel.frequency.set fg.channel.frequency.set
DIA	TRUE	2002	Switch on the oscilloscope.	switch on the oscilloscope	device.on device.on
DIA	TRUE	2416	Channel 2 frequency to 100 pico Hertz.	channel to frequency to 100pk Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	TRUE	2239	What's the dmm reading?	what's the digital multimeter Reading	dmm.value.get dmm.value.get
DIA	TRUE	2290	What's the duty cycle for channel one?	what's the duty cycle for channel one	fg.channel.duty.get fg.channel.duty.get
DIA	TRUE	1951	Switch off frequency lock.	switch off frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	1947	Switch the trigger to channel one.	switch the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2281	What's the measurement on the dmm?	what's the measurements on the digital multimeter	dmm.value.get dmm.value.get
WIT	FALSE	5749	Set the voltage scale on channel two to five volts.	set your volume okay on channel three to five will	osc.channel.voltage.set
WIT	FALSE	3903	Turn off the oscilloscope.	all the the of	device.off
WIT	FALSE	3000	Set the voltage scale of channel two to five.	the voltage go on channel two to five	osc.channel.voltage.set osc.channel.voltage.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	3269	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of a child to two zero point five	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	4009	Set timebase to 50 micro seconds.	set time they to fifty micro second	osc.timebase.set
WIT	FALSE	3555	Set the oscilloscope averaging to times 4.	set the cisco averaging two times for	osc.averaging.set osc.averaging.set
WIT	FALSE	3298	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel two to one hundred million	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3130	What's the voltage on the dmm?	what's the voltage on the d and and	dmm.value.get dmm.value.get
WIT	FALSE	3544	Channel 1 frequency to 100 Mega Hertz.	channel one frequency to one hundred nigga	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	2751	Channel one frequency to 100 pico.	channel one frequency to one hundred speaker	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	2849	Switch on the oscilloscope.	switch on the a telescope	device.on device.on
WIT	FALSE	4856	Voltage offset for channel two to 50 milli volts.	voltage offset the channel to to the mini vault	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
WIT	FALSE	3962	Channel 2 frequency to 100 pico Hertz.	channel two frequency to one hundred pickup	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	2827	What's the timebase?	what's the time they	osc.timebase.get osc.timebase.get

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	2870	Set the frequency of channel 1 to 50 Hertz.	set the frequency of channel one to fifty	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3021	What's the duty cycle for channel one?	what's good ut cycle to channel one	fg.channel.duty.get
WIT	FALSE	3079	Channel one osc probe to times 1.	one of pro two times one	osc.probe.att.set
WIT	FALSE	2539	Timebase to 1 second.	time base to one second	osc.timebase.set
WIT	FALSE	2657	Switch off frequency lock.	which of the frequency lock	fg.freq.lock.set fg.freq.lock.set
WIT	FALSE	5616	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set the voltage generator to provide one hundred million gold wife kilohertz find ways from chow one	fg.channel.set
WIT	FALSE	2836	Switch the trigger to channel one.	question the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	FALSE	4626	What's the voltage offset for channel one on the oscilloscope?	what voltage of the channel one on the	osc.channel.voltage.offset.get
WIT	FALSE	3233	Set the osc channel one probe to times 1.	set the office to channel one pro to tines one	osc.probe.att.set
WIT	TRUE	2641	Set averaging to times 16.	set averaging two times sixteen	osc.averaging.set osc.averaging.set
WIT	TRUE	2348	Turn on frequency lock.	turn on the frequency lock	fg.freq.lock.set fg.freq.lock.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	TRUE	3300	Set the trigger to channel two.	set the trigger to channel two	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	2424	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3793	What's the dmm reading?	what the digital multi meter reading	dmm.value.get dmm.value.get
WIT	TRUE	4060	Set the voltage offset for channel one to 5 volts.	set the voltage offset the channel one to five volts	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
WIT	TRUE	2569	What's the measurement on the dmm?	what are the measurements on the digital multi meter	dmm.value.get dmm.value.get

E.4.6 Test #6

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2277	Set averaging to times 16.	set averaging 2 x 16	osc.averaging.set osc.averaging.set
DIA	TRUE	2256	Set the oscilloscope averaging to times 4.	set the oscilloscope averaging 2 x 4	osc.averaging.set osc.averaging.set
DIA	TRUE	2385	Channel one osc probe to times 1.	channel 1 oscilloscope probe 2 x 1	osc.probe.att.set osc.probe.att.set
DIA	TRUE	2402	Switch on the oscilloscope.	switch on the oscilloscope	device.on device.on

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2089	Switch the trigger to channel one.	switch the trigger to channel one	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	1945	Switch off frequency lock.	switch off frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	2224	What's the measurement on the dmm?	what is the measurement on the dmm	dmm.value.get dmm.value.get
DIA	TRUE	1868	What's the duty cycle for channel one?	what's the duty cycle for channel one	fg.channel.duty.get fg.channel.duty.get
DIA	TRUE	1442	Turn off the oscilloscope.	turn off the oscilloscope	device.off device.off
DIA	TRUE	2239	Channel one frequency to 100 pico.	channel 1 frequency to 100 Pico	fg.channel.frequency.set fg.channel.frequency.set
DIA	TRUE	1964	Set the trigger to channel two.	set the trigger to channel 2	osc.trigger.channel.set osc.trigger.channel.set
DIA	TRUE	2281	What's the voltage offset for channel one on the oscilloscope?	what's the voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
DIA	TRUE	1501	Turn on frequency lock.	turn on frequency lock	fg.freq.lock.set fg.freq.lock.set
DIA	TRUE	1807	What's the dmm reading?	what's the dmm Reading	dmm.value.get dmm.value.get
DIA	TRUE	2964	Timebase to 1 second.	timebase to 1 second	osc.timebase.set osc.timebase.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	TRUE	2107	What's the voltage on the dmm?	what's the voltage on a dmm	dmm.value.get dmm.value.get
DIA	TRUE	1853	Set the voltage scale on channel two to five volts.	set the voltage scale on channel 2 to 5 volts	osc.channel.voltage.set osc.channel.voltage.set
DIA	FALSE	2666	Voltage offset for channel two to 50 milli volts.	voltage offset for Channel to 250 mm	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
DIA	FALSE	2345	Set timebase to 50 micro seconds.	set time verse 250 microseconds	osc.timebase.set osc.channel.voltage.offset.set
DIA	FALSE	2602	Set the osc channel one probe to times 1.	set the oscilloscope channel one probe 2 x 1	osc.probe.att.set osc.channel.voltage.offset.set
DIA	FALSE	3022	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	set up the function generator to provide 100mm 5 kilohertz sine wave from channel one	fg.channel.set fg.channel.set
DIA	FALSE	2717	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel to 20.5 Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	3024	Set the voltage scale of channel two to five.	set the voltage scale of channel 2 to 5	osc.channel.voltage.set fg.channel.frequency.set
DIA	FALSE	2150	Channel 1 frequency to 100 Mega Hertz.	channel 1 frequency to 100 mHz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	1751	Channel 2 frequency to 100 pico Hertz.	channel to frequency to 100pk Hz	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	2120	Set the frequency of channel 1 to 50 Hertz.	set the frequency of Channel 12 50 Hz	fg.channel.frequency.set fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
DIA	FALSE	1864	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of Tamil 2 to 100 minutes	fg.channel.frequency.set fg.channel.frequency.set
DIA	FALSE	1341	What's the timebase?	what's the time base	osc.timebase.get fg.channel.frequency.set
DIA	FALSE	2352	Set the voltage offset for channel one to 5 volts.	set the voltage offset for Channel 125 volts	osc.channel.voltage.offset.set fg.channel.frequency.set
DIA	FALSE	2031	The trigger to channel one.	the trigger to channel one	osc.trigger.channel.set fg.channel.frequency.set
WIT	TRUE	4939	Set the voltage scale of channel two to five.	set the voltage to channel two to five	osc.channel.voltage.set osc.channel.voltage.set
WIT	TRUE	5467	What's the measurement on the dmm?	what is the measurement on the them on	dmm.value.get dmm.value.get
WIT	TRUE	4967	Set the trigger to channel two.	set the trigger the channel to	osc.trigger.channel.set osc.trigger.channel.set
WIT	TRUE	3376	Turn on frequency lock.	turn on frequency lol	fg.freq.lock.set fg.freq.lock.set
WIT	TRUE	3358	What's the timebase?	what's the time the	osc.timebase.get osc.timebase.get
WIT	FALSE	5556	Voltage offset for channel two to 50 milli volts.	who would you for coming to two fifteen million bottles	osc.channel.voltage.offset.set
WIT	FALSE	3956	Set timebase to 50 micro seconds.	set time to fifty micro seconds	osc.timebase.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	5079	Set the osc channel one probe to times 1.	set the shows go someone pro two times warm	osc.probe.att.set
WIT	FALSE	7340	Set up the fg to provide a 100 milli volt 5 kilo hertz sine wave from channel 1.	so the function general provide a one hundred million ball five to a song live in china warm	fg.channel.set
WIT	FALSE	5577	Switch on the oscilloscope.	switch on the sale of the	device.on
WIT	FALSE	4625	Set the frequency of channel 2 to 0.5 Hertz.	set the frequency of channel two to not point five	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3558	Switch the trigger to channel one.	switch to go to come on warm	osc.trigger.channel.set
WIT	FALSE	2885	Switch off frequency lock.	switch all frequency lol	fg.freq.lock.set fg.freq.lock.set
WIT	FALSE	4257	Set averaging to times 16.	set everything time sixteen	osc.averaging.set
WIT	FALSE	3892	What's the duty cycle for channel one?	what do the cycle for come along	fg.channel.duty.get fg.channel.duty.get
WIT	FALSE	3641	Set the oscilloscope averaging to times 4.	set the philips go publishing to console	osc.averaging.set
WIT	FALSE	4399	Channel 1 frequency to 100 Mega Hertz.	turn alarm frequency twelve hundred mhz	fg.channel.frequency.set fg.freq.lock.set
WIT	FALSE	4140	Channel 2 frequency to 100 pico Hertz.	can also frequency to one hundred because	fg.channel.frequency.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	3955	Set the frequency of channel 1 to 50 Hertz.	set the frequency of telling want to fifty hz	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	3583	Turn off the oscilloscope.	turn off their skills	device.off device.off
WIT	FALSE	4312	Channel one frequency to 100 pico.	come on one frequency to one hundred people	fg.channel.frequency.set
WIT	FALSE	5357	What's the voltage offset for channel one on the oscilloscope?	what's the volume is set for a one on the two	osc.channel.voltage.offset.get dmm.value.get
WIT	FALSE	2787	What's the dmm reading?	also damn i'm reading	dmm.value.get
WIT	FALSE	3314	Timebase to 1 second.	time to one second	osc.timebase.set
WIT	FALSE	3735	What's the voltage on the dmm?	what's the volume of the m m	dmm.value.get dmm.value.get
WIT	FALSE	23691	Channel one osc probe to times 1.		osc.probe.att.set
WIT	FALSE	4780	Set the voltage scale on channel two to five volts.	set the voltage go on channel two to five souls	osc.channel.voltage.set osc.channel.voltage.set
WIT	FALSE	3519	Set the frequency of channel 2 to 100 milli hertz.	set the frequency of channel two to one hundred million	fg.channel.frequency.set fg.channel.frequency.set
WIT	FALSE	5224	Set the voltage offset for channel one to 5 volts.	set the voltage offset some warm thoughtful	osc.channel.voltage.offset.set

Table continued from previous page

Provider	Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
WIT	FALSE	6794	The trigger to channel one.	okay to come on	osc.trigger.channel.set

F Miscellaneous

F.1 Export Mockup

Pegasus 18th December

Export Details

Version: 0.0.1
Bench: 32
Instruments:

- Tektronix Oscilloscope
- Tektronix Function Generator
- Digital Multimeter

Session

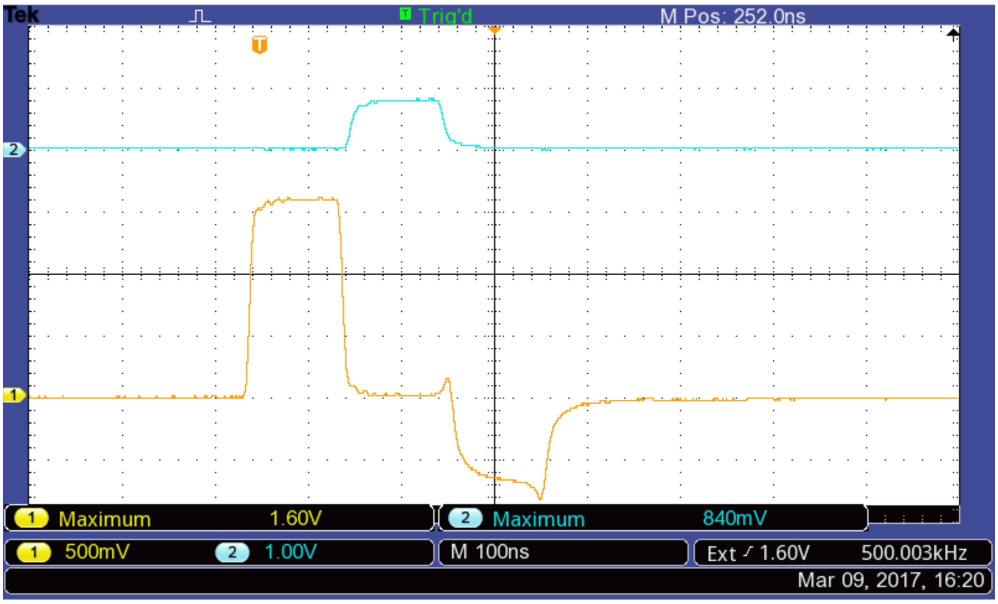
1. **Command:** Set channel 2 on function generator to 50 Hz.

18-12-17 12:20:00

2. **Measurement from Oscilloscope:** $8Hz$

18-12-17 12:25:30

3. **Screenshot from Oscilloscope:**



- 15:45:25 09/03/2017

18-12-17 12:26:30

4. **Note:** This is what we expected.

18-12-17 12:39:30

Page 1 of 1

F.2 Exclusion Metrics

F.2.1 Without Accessibility Software

Category	**Assessment Level**
Vision	As visually clear as ordinary newspaper print
Hearing	Understand loud speech in a quiet room
Concentration	Concentrate enough to make toast without getting distracted
Long-term Memory	Remember names of friends & family whom you see regularly
Literacy	Read and understand individual common words, e.g. cat, house
Speech comprehension	No need to understand speech
Speaking	No need to speak
Dominant hand - Lifting strength	Strength is not required from dominant hand/arm
Dominant hand - Dexterity	Pick up a safety pin from a table-top using your fingers
Dominant hand - Reaching forward and up	Reach forward to shake hands
Dominant hand - Reaching down	No need to reach down
Non-dominant hand - Lifting strength	Strength is not required form dominant hand/arm
Non-dominant hand - Dexterity	No need to make any precision hand movements
Non-dominant hand - Reaching forward and up	No need to reach forward or up
Non-dominant hand - Reaching down	No need to reach down
Walking	No need to walk without help
Stair climbing	No need to climb steps without help
Standing and balancing	No need to stand without holding on to anything

F.2.2 With Accessibility Software

Category	**Assessment Level**
Vision	As visually clear as a newspaper headline
Hearing	No need to hear anything
Concentration	Concentrate on a short TV ad without getting distracted
Long-term Memory	No need to remember anything
Literacy	Read and understand a sentence, e.g. in a tabloid newspaper
Speech comprehension	No need to understand speech
Speaking	Ask a simple question clearly enough that others understand it
Dominant hand - Lifting strength	Strength is not required from dominant hand/arm
Dominant hand - Dexterity	No need to make any precision hand movements
Dominant hand - Reaching forward and up	No need to reach forward or up
Dominant hand - Reaching down	No need to reach down
Non-dominant hand - Lifting strength	Strength is not required from dominant hand/arm
Non-dominant hand - Dexterity	No need to make any precision hand movements
Non-dominant hand - Reaching forward and up	No need to reach forward or up
Non-dominant hand - Reaching down	No need to reach down
Walking	No need to walk without help
Stair climbing	No need to climb steps without help
Standing and balancing	No need to stand without holding on to anything

F.3 System Overview

Controllers contain the system logic that binds together Models and Views.

Models access and process the data required for the Views.

Services

`SessionManager.cs` saves and loads the current sessions to an XML file. This includes all session elements: notes, commands, measurements & macro results.

`Exporter.cs` takes a session and writes it to a PDF.

`Dialogflow.cs` allows data to be sent to the Dialogflow API.

`MacroRunner.cs` runs the macro scripts.

Utilities

`FileUtils.cs` has some helper functions to access bundled application resources.

`KeywordListener.cs` sets up and manages the callbacks for the speech activation.

`Logger.cs` contains helper functions to log messages and debug data to the system log.

`PegasusException.cs` contains a definition of a custom exception, used to safely catch error messages to show to the user.

`Recorder.cs` contains all the processing to record audio with automatic silence detection.

`SettingsUtils.cs` contains helper functions to process the application settings.

`SoundEffects.cs` contains simple static functions to play certain sound effects.

`TemplateUtils.cs` processes raw data from devices and allows the system to present the data in a readable manner.

Views display the data to the user.

Forms are complete models that appear to the user as windows. These are directly related to the windows discussed in Section 4.2.

`FormDevice.cs`

`FormMain.cs`

`FormNote.cs`

`FormSettings.cs`

`FormStart.cs`

Controls are partial models that are embedded within forms.

`ComponentCommand.cs` shows the result of any commands given.

`ComponentGraph.cs` shows a graph plot, for example a hardcopy from an Oscilloscope.

`ComponentMeasurement.cs` show the result of a measurement.

`ComponentNote.cs` shows the content of any notes the user writes.

`ComponentTable.cs` shows a table of values, for example a macro result.

`CustomProgressBar.cs` shows the peak microphone level.

F.4 Testing

F.4.1 Commands

Oscilloscope

Command	Phrase	Works as Intended	Comments
osc.autoset.set	Autoset the oscilloscope.	✓	
osc.averaging.set	Set the averaging to times 4.	✓	There was a slight typo with the command.
osc.channel.state.set	Hide channel two.	✓	The NLP platform has trouble with the word ‘hide’.
osc.channel.voltage.get	Get the voltage scale on channel 1.	✓	
osc.channel.voltage.offset.get	What’s the voltage offset for channel one on the oscilloscope?	✓	
osc.channel.voltage.offset.set	Voltage offset for channel two to 50 milli volts.	✓	
osc.channel.voltage.set	Set the voltage scale on channel 1 to 50 milli volts.	✓	
osc.cursor.delta.get	What’s the horizontal delta for the cursors?	✓	
osc.cursor.position.set	Set the first cursor to 1 milli second.	✓	
osc.cursor.source.set	Set the cursor source to channel two.	✓	
osc.cursor.type.set	Switch on the time cursor.	✓	
osc.hardcopy.get	Take a hard copy.	✓	

Table continued from previous page

Command	Phrase	Works as Intended	Comments
osc.horizontal.position.get	What's the horizontal position?	✓	
osc.horizontal.position.set	Set the horizontal position to minus 25 milli seconds.	✓	
osc.measure.get	What's the peak to peak of channel one on the oscilloscope?	✓	
osc.probe.att.get	Get the osc channel one probe attenuation.	✓	The attenuation is returned as '`1.0E0`', not '`1`', as the documentation states.
osc.probe.att.set	Set the osc channel one probe to times 1.	✓	
osc.state.set	Start the oscilloscope.	✓	
osc.timebase.get	Get the time base.	✓	
osc.timebase.set	Set the time base to 50 micro seconds.	✓	
osc.trigger.autoset.set	Set trigger to 50%	✓	
osc.trigger.channel.get	What's the trigger channel?	✓	
osc.trigger.channel.set	Set the trigger to channel one.	✓	
osc.trigger.level.set	Set trigger to 1.4 volts	✓	The oscilloscope won't allow the trigger level to be set higher than 8 times the voltage scale for the channel.
osc.trigger.slope.set	Set the trigger to rising edge.	✓	

Function Generator

Command	Phrase	Works as Intended	Comments
fg.channel.duty.get	What’s the duty cycle for channel one?	✓	
fg.channel.duty.set	Set the duty cycle for channel one to 85 percent	✓	
fg.channel.frequency.set	Channel 1 frequency to 100 milli Hz.	✓	
fg.channel.onoff.set	Turn on channel one.	✓	
fg.channel.voltage.set	Channel 1 voltage to 100 milli volts.	✓	The NLP platform has difficulty separating ‘milli’ from the number.
fg.channel.waveform.set	Set channel one on the function generator to a square wave.	✓	
fg.freq.lock.get	Are the frequencies locked?	✓	
fg.freq.lock.set	Switch off frequency lock	✓	

F.4.2 Metric Study Repeat

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
TRUE	2480	Channel 1 voltage to 100 milli volts.	1 voltage 100mm bolts	fg.channel.voltage.set fg.channel.voltage.set
TRUE	2995	What's the horizontal position?	horizontal position	osc.horizontal.position.get osc.horizontal.position.get
TRUE	3535	What's the peak to peak of channel one on the oscilloscope?	peak to peak of channel 1 on the oscilloscope	osc.measure.get osc.measure.get
FALSE	3581	Start the oscilloscope.	oscilloscope	osc.state.set osc.autoset.set
TRUE	1982	Take a hard copy.	hard copy	osc.hardcopy.get osc.hardcopy.get
TRUE	2316	Set trigger to 50%	trigger to 50%	osc.trigger.autoset.set osc.trigger.autoset.set
TRUE	4461	Set the osc channel one probe to times 1.	set the oscilloscope channel one probe x 1	osc.probe.att.set osc.probe.att.set
FALSE	19560	Set the time base to 50 micro seconds.	the coinbase 15 microseconds	osc.timebase.set osc.timebase.set
FALSE	3027	Are the frequencies locked?	all the Frequency blocked	fg.freq.lock.get fg.channel.frequency.set
TRUE	2231	What's the duty cycle for channel one?	the duty cycle for Channel 1	fg.channel.duty.get fg.channel.duty.get
FALSE	3276	What's the horizontal delta for the cursors?	the horizontal delta for the curses	osc.cursor.delta.get fg.channel.frequency.set
FALSE	1945	Get the time base.	coinbase	osc.timebase.get fg.channel.frequency.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
FALSE	2184	Set the cursor source to channel two.	Elsa sauce to Charlton	osc.cursor.source.set fg.channel.frequency.set
FALSE	2429	Set the trigger to rising edge.	Twitter rising Edge	osc.trigger.slope.set fg.channel.frequency.set
FALSE	2249	Set trigger to 1.4 volts	F 1.4 volts	osc.trigger.level.set fg.channel.frequency.set
FALSE	2446	Set the first cursor to 1 milli second.	the first cutter one millisecond	osc.cursor.position.set fg.channel.frequency.set
TRUE	2797	Get the osc channel one probe attenuation.	the oscilloscope channel one probe attenuation	osc.probe.att.get osc.probe.att.get
FALSE	2000	Hide channel two.	channel 2	osc.channel.state.set osc.measure.get
FALSE	2256	Set the trigger to channel one.	ticket to channel one	osc.trigger.channel.set osc.measure.get
FALSE	4739	Set the averaging to times 4.	garaging 2 x 4	osc.averaging.set osc.measure.get
FALSE	5211	Autoset the oscilloscope.	set the oscilloscope	osc.autoset.set osc.measure.get
FALSE	7689	Switch off frequency lock	top frequency lock	fg.freq.lock.set osc.measure.get
TRUE	1984	What's the trigger channel?	the trigger channel	osc.trigger.channel.get osc.trigger.channel.get
FALSE	1954	Turn on channel one.	the channel one	fg.channel.onoff.set osc.measure.get
TRUE	2661	Set channel one on the function generator to a square wave.	channel one on the function generator to a sine wave	fg.channel.waveform.set fg.channel.waveform.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
TRUE	2424	What's the voltage offset for channel one on the oscilloscope?	voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
FALSE	1542	Switch on the time cursor.	on the tynecastle	osc.cursor.type.set
FALSE	2210	Get the voltage scale on channel 1.	altered scale on Channel 1	osc.channel.voltage.get osc.channel.voltage.set
TRUE	2535	Set the duty cycle for channel one to 85 percent	Duty cycle for Channel 1 85%	fg.channel.duty.set fg.channel.duty.set
FALSE	3827	Channel 1 frequency to 100 milli Hz.	1 frequency 100ml	fg.channel.frequency.set osc.channel.voltage.set
FALSE	2852	Set the horizontal position to minus 25 milli seconds.	presenting position - 25 milliseconds	osc.horizontal.position.set osc.timebase.set
FALSE	3646	Set the voltage scale on channel 1 to 50 milli volts.	voltage scale on channel 150 ml bolts	osc.channel.voltage.set osc.channel.voltage.set
FALSE	2463	Voltage offset for channel two to 50 milli volts.	to offset 15mm bolts	osc.channel.voltage.offset.set osc.channel.voltage.set
TRUE	1996	What's the horizontal position?	horizontal position	osc.horizontal.position.get osc.horizontal.position.get
FALSE	2023	Set trigger to 50%	bigger 50%	osc.trigger.autoset.set osc.channel.voltage.set
FALSE	2543	Set the time base to 50 micro seconds.	the time base 50 micro seconds	osc.timebase.set osc.channel.voltage.set
FALSE	2024	Turn on channel one.	one channel one	fg.channel.onoff.set osc.channel.voltage.set
FALSE	2014	Are the frequencies locked?	frequency is locked	fg.freq.lock.get fg.freq.lock.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
FALSE	2583	Channel 1 frequency to 100 milli Hz.	the Frequency 100ml	fg.channel.frequency.set fg.freq.lock.set
FALSE	2354	Set the cursor source to channel two.	the cast of source channel 2	osc.cursor.source.set fg.freq.lock.set
FALSE	1529	Set the trigger to channel one.	trigger channel one	osc.trigger.channel.set fg.freq.lock.set
TRUE	3000	Set the horizontal position to minus 25 milli seconds.	the horizontal position - 25 milliseconds	osc.horizontal.position.set osc.horizontal.position.set
TRUE	2547	Set channel one on the function generator to a square wave.	channel one on the function generator to a square wave	fg.channel.waveform.set fg.channel.waveform.set
TRUE	1994	What's the trigger channel?	the trigger channel	osc.trigger.channel.get osc.trigger.channel.get
TRUE	2082	Autoset the oscilloscope.	to set the oscilloscope	osc.autoset.set osc.autoset.set
TRUE	2307	What's the horizontal delta for the cursors?	horizontal delta for the curses	osc.cursor.delta.get osc.cursor.delta.get
FALSE	2570	Voltage offset for channel two to 50 milli volts.	it's offset for Channel to 15 million volts	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
FALSE	2044	Start the oscilloscope.	the oscilloscope	osc.state.set osc.autoset.set
TRUE	2391	Set the first cursor to 1 milli second.	first Casa one millisecond	osc.cursor.position.set osc.cursor.position.set
FALSE	1898	Take a hard copy.	coffee	osc.hardcopy.get
FALSE	2430	Channel 1 voltage to 100 milli volts.	on voltage 100ml notes	fg.channel.voltage.set fg.channel.voltage.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
FALSE	2370	Get the voltage scale on channel 1.	the voltage scale on Channel 1	osc.channel.voltage.get fg.channel.voltage.set
FALSE	2038	Switch on the time cursor.	on the Tyne Gazza	osc.cursor.type.set
FALSE	2545	What's the peak to peak of channel one on the oscilloscope?	topeak of channel 1 on the oscilloscope	osc.measure.get osc.measure.get
FALSE	1988	Hide channel two.	channel 2	osc.channel.state.set osc.measure.get
FALSE	2099	Set the trigger to rising edge.	biggest rising Edge	osc.trigger.slope.set osc.measure.get
FALSE	3044	Set the voltage scale on channel 1 to 50 milli volts.	bookish girl on channel 150 Milly bolts	osc.channel.voltage.set osc.measure.get
FALSE	2119	Set the averaging to times 4.	bridging x 4	osc.averaging.set osc.measure.get
TRUE	2349	What's the voltage offset for channel one on the oscilloscope?	voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
FALSE	2612	Set the duty cycle for channel one to 85 percent	the duty cycle for Channel 1 85%	fg.channel.duty.set fg.channel.duty.set
TRUE	2919	Get the osc channel one probe attenuation.	oscilloscope channel one probe attenuation	osc.probe.att.get osc.probe.att.get
FALSE	2473	Set the osc channel one probe to times 1.	telescope channel one probe 2 x 1	osc.probe.att.set fg.channel.duty.set
TRUE	1983	Get the time base.	the time base	osc.timebase.get osc.timebase.get
TRUE	2033	Switch off frequency lock	off frequency lock	fg.freq.lock.set fg.freq.lock.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
TRUE	3654	What's the duty cycle for channel one?	the duty cycle for Channel 1	fg.channel.duty.get fg.channel.duty.get
TRUE	1839	Set trigger to 1.4 volts	trigger 1.4 volts	osc.trigger.level.set osc.trigger.level.set
FALSE	3007	Set the voltage scale on channel 1 to 50 milli volts.	voltage scale on channel 150 Milly bolts	osc.channel.voltage.set osc.channel.voltage.set
FALSE	1980	Take a hard copy.	cold coffee	osc.hardcopy.get osc.channel.voltage.set
TRUE	1994	Set channel one on the function generator to a square wave.	channel one on the function generator to a square wave	fg.channel.waveform.set fg.channel.waveform.set
FALSE	2060	Switch on the time cursor.	the time capsule	osc.cursor.type.set osc.channel.voltage.set
FALSE	2133	Set the trigger to channel one.	the triggered channel one	osc.trigger.channel.set osc.channel.voltage.set
FALSE	2066	Turn on channel one.	is on Channel 1	fg.channel.onoff.set osc.channel.voltage.set
FALSE	2384	Set trigger to 1.4 volts	trigger 1.4 volts	osc.trigger.level.set osc.channel.voltage.set
TRUE	2096	Set the trigger to rising edge.	Trigger 2 rising Edge	osc.trigger.slope.set osc.trigger.slope.set
FALSE	2020	What's the voltage offset for channel one on the oscilloscope?	the voltage offset for Channel 1 on the oscilloscope	osc.channel.voltage.offset.get osc.channel.voltage.offset.get
FALSE	2625	Set the osc channel one probe to times 1.	the oscilloscope channel one probe 2 x 1	osc.probe.att.set osc.probe.att.set
FALSE	1983	Are the frequencies locked?	frequency is locked	fg.freq.lock.get osc.probe.att.set

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
FALSE	3026	Set the time base to 50 micro seconds.	the time base 50 micro seconds	osc.timebase.set osc.probe.att.set
FALSE	2602	Channel 1 voltage to 100 milli volts.	bon voltage 100ml volts	fg.channel.voltage.set osc.probe.att.set
FALSE	2570	Set the averaging to times 4.	the averaging 2 x 4	osc.averaging.set osc.probe.att.set
TRUE	3217	What's the horizontal delta for the cursors?	what's the horizontal delta for the curses	osc.cursor.delta.get osc.cursor.delta.get
TRUE	2793	What's the peak to peak of channel one on the oscilloscope?	the peak-to-peak of channel 1 on the oscilloscope	osc.measure.get osc.measure.get
FALSE	2468	Voltage offset for channel two to 50 milli volts.	voltage offset for channel 2 50 Billy Goats	osc.channel.voltage.offset.set osc.channel.voltage.offset.set
FALSE	2674	Channel 1 frequency to 100 milli Hz.	1 frequency 100ml	fg.channel.frequency.set osc.channel.voltage.offset.set
TRUE	2142	Set trigger to 50%	Peugeot 50%	osc.trigger.autoset.set osc.trigger.autoset.set
FALSE	2018	Start the oscilloscope.	the oscilloscope	osc.state.set osc.autoset.set
TRUE	2887	Set the horizontal position to minus 25 milli seconds.	horizontal position - 25 milliseconds	osc.horizontal.position.set osc.horizontal.position.set
TRUE	2034	What's the horizontal position?	horizontal position	osc.horizontal.position.get osc.horizontal.position.get
FALSE	1957	Hide channel two.	channel 2	osc.channel.state.set osc.measure.get
TRUE	1990	Get the time base.	the time base	osc.timebase.get osc.timebase.get

Table continued from previous page

Passed	Delay (ms)	Expected Text	Parsed Text	Expected/Parsed Intent
TRUE	2226	What's the duty cycle for channel one?	the duty cycle for Channel 1	fg.channel.duty.get fg.channel.duty.get
FALSE	2592	Set the first cursor to 1 milli second.	get the first Curse of one millisecond	osc.cursor.position.set osc.cursor.position.set
FALSE	3098	Get the osc channel one probe attenuation.	the oscilloscope channel one probe attend attenuation	osc.probe.att.get osc.cursor.position.set
FALSE	2039	Autoset the oscilloscope.	the oscilloscope	osc.autoset.set osc.cursor.position.set
FALSE	2709	Switch off frequency lock	cough frequency lock	fg.freq.lock.set osc.cursor.position.set
FALSE	2381	Set the cursor source to channel two.	Persil sauce to channel 2	osc.cursor.source.set osc.cursor.position.set
FALSE	2624	Set the duty cycle for channel one to 85 percent	recycle for Channel 1 85%	fg.channel.duty.set osc.cursor.position.set
TRUE	1967	What's the trigger channel?	trigger channel	osc.trigger.channel.get osc.trigger.channel.get
FALSE	2306	Get the voltage scale on channel 1.	voltage scale on channel one	osc.channel.voltage.get osc.cursor.position.set

F.5 Example Export

Export Details

Version: 0.5.0
Devices:

- TBS 1052B-EDU Oscilloscope
- AFG1062 Function Generator

Commands

1. Command to AFG1062: Switch SOURCE1 to a sinusoid wave.
17/04/2018 10:17

2. Command to AFG1062: Switch SOURCE2 to a square wave.
17/04/2018 10:17

3. Command to AFG1062: Set the voltage for SOURCE1 to 100.000 mVolts.
17/04/2018 10:28

4. Command to AFG1062: Set the frequency for SOURCE1 to 5.000 kHz.
17/04/2018 10:28

5. Command to AFG1062: Turn on the frequency lock.
17/04/2018 10:29

6. Command to AFG1062: Set the voltage for SOURCE2 to 1 Volts.
17/04/2018 10:29

7. Command to TBS 1052B-EDU: Set the trigger to EXT.
17/04/2018 10:30

8. Command to AFG1062: Switch OUTPUT1 on.
17/04/2018 10:30

9. Command to AFG1062: Switch OUTPUT2 on.
17/04/2018 10:31

10. Command to TBS 1052B-EDU: Turn on CH2.
17/04/2018 10:33

11. Command to TBS 1052B-EDU: Autoset the oscilloscope.
17/04/2018 10:34

12. Command to TBS 1052B-EDU: Set the oscilloscope CH1 probe to x1.
17/04/2018 10:36

13. Command to TBS 1052B-EDU: Set the oscilloscope CH2 probe to x1.
17/04/2018 10:36

14. Measurement from TBS 1052B-EDU: PK2PK for CH1 is 112.000 mV

14. Measurement from TBS 1052B-EDU: PK2PK for CH1 is 112.000 mV.
17/04/2018 10:39

15. Measurement from TBS 1052B-EDU: PK2PK for CH2 is 1.600 V.
17/04/2018 10:39

16. Note:

Gain: 14.2857

17/04/2018 10:40

17. Measurement from TBS 1052B-EDU: PHASE for CH1 is -6.448 degrees.
17/04/2018 10:40

18. Command to AFG1062: Set the frequency for SOURCE1 to 10 Hz.
17/04/2018 10:47

19. Note:

Section 2

17/04/2018 10:53

20. Command to AFG1062: Switch OUTPUT2 off.
17/04/2018 10:54

21. Command to AFG1062: Switch OUTPUT1 off.
17/04/2018 10:54

22. Command to AFG1062: Switch SOURCE1 to a square wave.
17/04/2018 10:57

23. Command to AFG1062: Set the voltage for SOURCE1 to 10.000 mVolts.
17/04/2018 11:03

24. Command to AFG1062: Switch OUTPUT1 on.
17/04/2018 11:03

25. Command to AFG1062: Switch OUTPUT2 on.
17/04/2018 11:03

26. Command to TBS 1052B-EDU: Autoset the oscilloscope.
17/04/2018 11:04

27. Command to TBS 1052B-EDU: Set the timebase to 50.000 μs.
17/04/2018 11:05

28. Command to TBS 1052B-EDU: Set the timebase to 10.000 μs.
17/04/2018 11:05

29. Command to TBS 1052B-EDU: Set the timebase to 1.000 μs.

17/04/2018 11:05

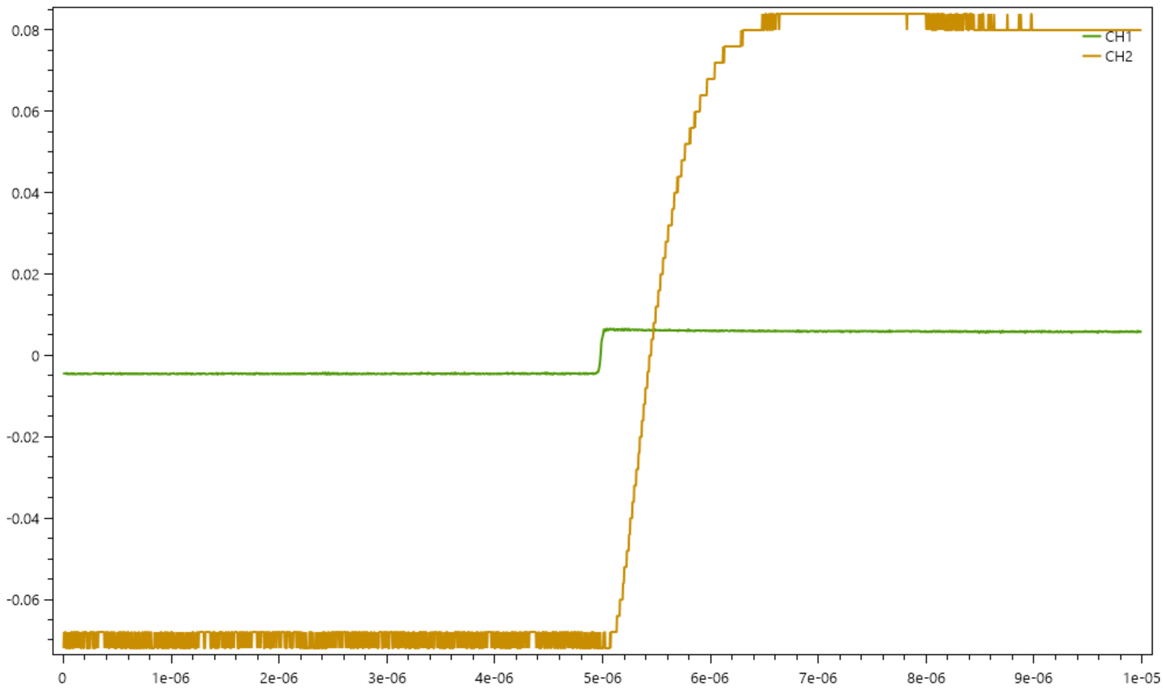
30. Command to TBS 1052B-EDU: Set the trigger to CH1.

17/04/2018 11:06

31. Command to TBS 1052B-EDU: Set the averaging to times 16.

17/04/2018 11:07

32. Graph: TBS 1052B-EDU



17/04/2018 11:07

33. Command to TBS 1052B-EDU: Set the cursors to VBARS.

17/04/2018 11:09

34. Command to TBS 1052B-EDU: Set cursor to 0 Seconds.

17/04/2018 11:09

35. Command to TBS 1052B-EDU: Set cursor to 3.000 μSeconds.

17/04/2018 11:19

36. Command to TBS 1052B-EDU: Set the cursor source to CH2.

17/04/2018 11:20

37. Measurement from TBS 1052B-EDU: The vertical delta is 152.000 mVolts.

17/04/2018 11:20

38. Note:

90% is 136.8 mV

17/04/2018 11:20

39. Command to TBS 1052B-EDU: Set cursor to 1.000 µSeconds.

17/04/2018 11:20

40. Measurement from TBS 1052B-EDU: The vertical delta is 136.000 mVolts.

17/04/2018 11:21

41. Note:

Therefore Rise Time is 1 uS

17/04/2018 11:21

42. Measurement from TBS 1052B-EDU: RISE for CH2 is 784.000 ns.

17/04/2018 11:21

43. Note:

I believe this is from 10% to 90% ?

17/04/2018 11:22

Glossary

enum is a data type consisting of a set of distinct named values. For instance, the four suits in a deck of playing cards. 25

mnemonic is "a system such as a pattern of letters, ideas, or associations which assists in remembering something" [74]. 19

parse is to "resolve (a sentence) into its component parts and describe their syntactic roles" [75]. 2, 25, 33, 35, 36, 63

reflection (computer science) allows compiled programs to inspect, modify, or invoke methods and fields during runtime usually using string variables to refer to class/field names. 61

vibrafication is a way of conveying information using small vibrations, usually applied over a surface of a users skin. 21

Acronyms

API Application Programming Interface. 5, 22, 24–26, 49, 59, 61, 69, 76, 78, 182

ASCII American Standard Code for Information Interchange. 19

CHM Compiled HTML. 54

DLL Dynamic-link library. 50, 71

DMM Digital Multimeter. 3, 5, 6, 8, 15, 18, 20, 31, 47, 59, 127, 133

GPIB General Purpose Interface Bus. 19, 50

GUI Graphical User Interface. 38, 43, 51

HTML Hypertext Markup Language. 54, 59, 65

HTTP Hypertext Transfer Protocol. 49

IDE Intergrated Development Enviroment. 9, 26, 38, 51, 70

JSON JavaScript Object Notation. 49, 58

LAN Local Area Network. 50

LF Line Feed. 19

MVC Model-view-controller. 48

NDA National Disability Authority. 66, 67

NI National Instruments. 70

NLP Natural Language Processing. 2–5, 15, 22, 24–26, 29, 33, 34, 36, 43, 44, 47, 49, 62–64, 68, 69, 75, 183, 185

OS Operating System. 24, 54, 70

OSK On-Screen Keyboard. 24, 66

PDF Portable Document Format. 4, 5, 37, 44, 47, 52, 59, 61, 64, 182

RP Received Pronunciation. 30

SALEIE Strategic ALignment of Electrical and Information Engineering in European Higher Education Institutions. 12

SCM Source Control Management. 28

SCPI Standard Commands for Programmable Instruments. 19, 20, 71

UI User Interface. 4, 5, 24, 26, 37, 38, 51, 65, 69, 70

USB Universal Serial Bus. 19, 20, 50

VISA Virtual Instrument Software Architecture. 70

VXI VME Extensions for Instrumentation. 50

WPF Windows Presentation Foundation. 70

XAML Extensible Application Markup Language. 70

XML Extensible Markup Language. 6, 11, 43, 44, 54, 56, 58, 90, 91, 182

References

[1] Cambridge Engineering Design Centre, *Exclusion Calculator*, 2015. [Online]. Available: http://calc.inclusivedesigntoolkit.com/ (Accessed: May 5, 2018).

[2] Royal National Institute of Blind People, *How many people in the UK have sight loss?* [Online]. Available: https://help.rnib.org.uk/help/newly-diagnosed-registration/registering-sight-loss/statistics (Accessed: May 5, 2018).

[3] ——, *The criteria for certification.* [Online]. Available: http://www.rnib.org.uk/eye-health/registering-your-sight-loss/criteria-certification (Accessed: May 4, 2018).

[4] The Understood Team and S. H. Horowitz, *Dyspraxia by the Numbers*, 2014. [Online]. Available: https://www.understood.org/en/learning-attention-issues/child-learning-disabilities/dyspraxia/dyspraxia-by-the-numbers (Accessed: May 5, 2018).

[5] Dyspraxia Foundation, *Dyspraxia in Adults*. [Online]. Available: https://dyspraxiafoundation.org.uk/dyspraxia-adults/ (Accessed: May 5, 2018).

[6] *Impairment Simulator Software.* [Online]. Available: http://www.inclusivedesigntoolkit.com/simsoftware/simsoftware.html (Accessed: May 4, 2018).

[7] Tektronix Inc., *TBS1000B/EDU, TBS1000, TDS2000C/TDS1000C-EDU, TDS2000B/TDS1000B, TDS2000/TDS1000, TDS200 and TPS2000B/TPS2000 Series Digital Oscilloscopes Programmer*. [Online]. Available: http://download.tek.com/manual/TBS1000-B-EDU-TDS2000-B-C-TDS1000-B-C-EDU-TDS200-TPS2000-Programmer_EN-US-RevA.pdf.

[8] European SCPI Consortium, "Volume 1 : Syntax and Style," vol. 1, 1999. [Online]. Available: http://www.ivifoundation.org/docs/scpi-99.pdf.

[9] Tektronix Inc., *AFG1000 Series Arbitrary Function Generator Programmer Manual*, 2016. [Online]. Available: http://download.tek.com/manual/AFG1000-Programmer-Manual-EN-077112901(20160719)-RevA.pdf.

[10] UNI-T, *Benchtop Digital Multimeter UT804*. [Online]. Available: http://www.uni-trend.com/productsdetail_1927_1169_1169.html (Accessed: Jan. 18, 2018).

[11] Tektronix Inc., *VocalLink Voice Control Software for Tektronix Oscilloscopes*, 2002. [Online]. Available: https://www.tek.com/datasheet/vclnkb-vclnkp (Accessed: Dec. 2, 2017).

[12] Keysight Technologies, *54845A Infiniium Oscilloscope: 4 Channels, 1.5 GHz, up to 8 GSa/s*. [Online]. Available: https://www.keysight.com/en/pd-1000001468%3Aepsg%3Apro-pn-54845A/4-channel-15-ghz-8-gsa-s-infiniium-oscilloscope?pm=PL&nid=-32415.536881903&cc=GB&lc=eng (Accessed: Dec. 2, 2017).

[13] C. Nachtmann, *Tektronix Android App*. [Online]. Available: http://blog.chasenachtmann.com/p/tektronix-android-app.html (Accessed: Dec. 2, 2017).

[14] C. R. Stevens and D. E. Reamer, “Method and apparatus for activating switches in response to different acoustic signals,” US 5493618A, 1993. [Online]. Available: https://patents.google.com/patent/US5493618.

[15] M. Kauffman, *Thumbs Down For The Clapper*, 2005. [Online]. Available: http://articles.courant.com/2005-02-06/business/0502060305_1_clapper-plugs-auld-lang.

[16] Amazon, *Alexa Skills Kit - Build for Voice with Amazon*, 2017. [Online]. Available: https://developer.amazon.com/alexa-skills-kit (Accessed: Apr. 9, 2018).

[17] Apple, *iOS - Siri*. [Online]. Available: https://www.apple.com/ios/siri/ (Accessed: Apr. 23, 2018).

[18] Google, *Google Home - Smart Speaker & Home Assistant*. [Online]. Available: https://store.google.com/product/google_home (Accessed: Apr. 23, 2018).

[19] Radiocentre, “How voice-activated devices are increasing radio listening and elevating audio branding,” Tech. Rep., 2017. [Online]. Available: http://www.radiocentre.org/wp-content/uploads/2017/05/Getting-Vocal-Final-Report-Web-14-figure.pdf.

[20] J. Fitzpatrick, *How to Make Alexa Understand You Better*, 2017. [Online]. Available: https://www.howtogeek.com/286076/how-to-make-the-amazon-echo-understand-you-better/ (Accessed: May 1, 2018).

[21] E. Sloss, “Voice Activated Web Browser Interface,” University of York, Tech. Rep., 2014. [Online]. Available: http://nugget.its.york.ac.uk/ProjectLibrary/search/record.php?ID=777.

[22] S. Clarke, “A Wearable Interface for a Blind Radio Presenter,” University of York, York, Tech. Rep., 2017. [Online]. Available: http://nugget.its.york.ac.uk/ProjectLibrary/search/record.php?ID=1452.

[23] A. Figueiredo, “Voice-control of robot swarms using IBM Watson,” University of York, Tech. Rep., 2017. [Online]. Available: http://nugget.its.york.ac.uk/ProjectLibrary/search/record.php?ID=1478.

[24] IBM, *Watson Speech To Text Documenation*, 2017. [Online]. Available: https://console.bluemix.net/docs/services/text-to-speech/index.html#about (Accessed: Apr. 10, 2018).

[25] S. Cleaver, “Voice Controlled Recipe App,” University of York, Tech. Rep., 2013. [Online]. Available: http://nugget.its.york.ac.uk/ProjectLibrary/search/record.php?ID=177.

[26] O. Corpe, “Voice Activated Web Browser Interface,” University of York, Tech. Rep., 2012. [Online]. Available: http://nugget.its.york.ac.uk/ProjectLibrary/search/record.php?ID=695.

[27] Nuance Communications Inc., *Dragon Software Developer Kit — Nuance UK*. [Online]. Available: https://www.nuance.com/en-gb/dragon/for-developers/dragon-software-developer-kit.html (Accessed: Apr. 10, 2018).

[28] National Disability Authority, *Guidelines for Application Software Accessibility - Printable Version*. [Online]. Available: http://universaldesign.ie/Technology-ICT/Irish-National-IT-Accessibility-Guidelines/Application-software/Guidelines-for-Application-Software-Accessibility-Printable-Version/Guidelines-for-Application-Software-Accessibility-Printable-Version.Html (Accessed: Dec. 2, 2017).

[29] J. Nielsen, "Response Times: The 3 Important Limits," Jan. 1993. [Online]. Available: https://www.nngroup.com/articles/response-times-3-important-limits/.

[30] Apdex Alliance Inc., "Application Performance Index - Apdex Technical Specification," Tech. Rep., 2007. [Online]. Available: http://apdex.org/documents/ApdexTechnicalSpecificationV11_000.pdf.

[31] S. A. Brewster, P. C. Wright, and A. D. N. Edwards, "AN EVALUATION OF EARCONS FOR USE IN AUDITORY HUMAN-COMPUTER INTERFACES," in *HUMAN FACTORS IN COMPUTING SYSTEMS*, S. Ashlund, A. Henderson, E. Hollnagel, K. Mullet, and T. White, Eds. I O S PRESS, 1993, ISBN: 90-5199-133-9.

[32] Freesound, *Freesound - Freesound*. [Online]. Available: https://freesound.org/ (Accessed: Dec. 2, 2017).

[33] StatCounter, *Desktop Operating System Market Share Worldwide - September 2017*, 2017. [Online]. Available: http://gs.statcounter.com/os-market-share/desktop/united-kingdom/2016 (Accessed: Jan. 22, 2018).

[34] Microsoft, *Accessibility Tools for Windows — Microsoft*, 2017. [Online]. Available: https://www.microsoft.com/en-us/accessibility/windows (Accessed: Dec. 18, 2017).

[35] ——, *Hear text read aloud with Narrator - Windows Help*. [Online]. Available: https://support.microsoft.com/en-us/help/17173/windows-10-hear-text-read-aloud (Accessed: Jan. 15, 2018).

[36] NV Access Limited, *NV Access*. [Online]. Available: https://www.nvaccess.org (Accessed: Dec. 2, 2017).

[37] Serotek, *System Access to Go*. [Online]. Available: https://www.satogo.com/en/ (Accessed: Jan. 15, 2018).

[38] Microsoft, *How Active Accessibility Works (Windows)*. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/dd318447(v=vs.85).aspx (Accessed: Jan. 15, 2018).

[39] ——, *Speech Synthesis (Microsoft.Speech)*. [Online]. Available: https://msdn.microsoft.com/en-us/library/hh362831(v=office.14).aspx (Accessed: Jan. 15, 2018).

[40] A. Copestack, "Natural Language Processing," *Natural Language Processing*, 2004. [Online]. Available: https://www.cl.cam.ac.uk/teaching/2002/NatLangProc/revised.pdf.

[41] Wit.ai Inc., *Wit.ai*, Palo Alto, California. [Online]. Available: https://wit.ai/ (Accessed: Jan. 15, 2018).

[42] Wit.ai, *Wit - HTTP API*. [Online]. Available: https://wit.ai/docs/http/20170307 (Accessed: Jan. 15, 2018).

[43] Dialogflow, *Dialogflow*. [Online]. Available: https://dialogflow.com/ (Accessed: May 2, 2018).

[44] ——, *Google Project FAQ — Dialogflow*. [Online]. Available: https://dialogflow.com/docs/concepts/google-projects-faq (Accessed: Jan. 15, 2018).

[45] R. Bell, *A beginner's guide to Big O notation - Rob Bell*, 2015. [Online]. Available: https://rob-bell.net/2009/06/a-beginners-guide-to-big-o-notation/ (Accessed: May 2, 2018).

[46] Microsoft Visual Studio Team, *Visual Studio IDE, Code Editor, VSTS & App Center*. [Online]. Available: https://www.visualstudio.com/ (Accessed: Dec. 28, 2017).

[47] ECMA International, "C# Language Specification," 2017. [Online]. Available: https://www.ecma-international.org/publications/files/ECMA-ST/Ecma-334.pdf.

[48] Oracle, *How to Support Assistive Technologies*. [Online]. Available: https://docs.oracle.com/javase/tutorial/uiswing/misc/access.html (Accessed: Apr. 25, 2018).

[49] K. Beck, M. Beedle, A. V. Bennekum, *et al.*, *Manifesto for Agile Software Development*, 2001. [Online]. Available: http://agilemanifesto.org/ (Accessed: Jan. 23, 2018).

[50] ——, "Principles behind the Agile Manifesto," *Agile Manifesto*, 2001. [Online]. Available: http://agilemanifesto.org/principles.html (Accessed: Jan. 23, 2018).

[51] Github Inc, *Github*. [Online]. Available: https://github.com/ (Accessed: Jan. 23, 2018).

[52] Git, *Git*. [Online]. Available: https://git-scm.com/ (Accessed: Jan. 23, 2018).

[53] Axosoft LLC., *Axosoft is the Scrum software for agile teams*. [Online]. Available: https://www.axosoft.com/ (Accessed: Jan. 23, 2018).

[54] ——, *Burndown Charts*. [Online]. Available: https://support.axosoft.com/reporting/burndown (Accessed: Jan. 23, 2018).

[55] Center for Hearing and Communication, *Common environmental noise levels*, 2017. [Online]. Available: http://chchearing.org/noise/common-environmental-noise-levels/ (Accessed: Apr. 25, 2018).

[56] National Disability Authority, *Checklist for Application Software Accessibility — Centre for Excellence in Universal Design*. [Online]. Available: http://universaldesign.ie/Technology-ICT/Irish-National-IT-Accessibility-Guidelines/Application-software/Checklist-for-Application-Software-Accessibility/ (Accessed: Jan. 25, 2018).

[57] R. Millar, "Breadboarding Workshop," University of York, Tech. Rep., 2017.

[58] Phantom Open Emoji maintainers and contributors, *File:PEO-pegasus black.svg*, 2014. [Online]. Available: https://commons.wikimedia.org/wiki/File:PEO-pegasus_black.svg (Accessed: Apr. 22, 2018).

[59] JetBrains, *Install and set up IntelliJ IDEA*, 2018. [Online]. Available: https://www.jetbrains.com/help/idea/install-and-set-up-product.html#first-run (Accessed: Apr. 13, 2018).

[60] Microsoft, "The Windows Interface Guidelines - A Guide for Designing Software," 1995. [Online]. Available: https://www.ics.uci.edu/~kobsa/courses/ICS104/course-notes/Microsoft_WindowsGuidelines.pdf.

[61] Audacity, *File:VerticalMeters3.png - Audacity Manual*, 2009. [Online]. Available: http://ttmanual.audacityteam.org/man/File:VerticalMeters3.png (Accessed: Apr. 22, 2018).

[62] Rosetta Code, *Named parameters - Rosetta Code*. [Online]. Available: http://rosettacode.org/wiki/Named_parameters#Python (Accessed: Mar. 26, 2018).

[63] Emoacht, *Color Customizable ProgressBar for Visual Style — Sometimes Coding*, 2011. [Online]. Available: https://emoacht.wordpress.com/2011/10/16/color-customizable-progressbar-for-visual-style/ (Accessed: Apr. 8, 2018).

[64] Microsoft, *Microsoft HTML Help 1.4 (Windows)*. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ms670169(v=vs.85).aspx (Accessed: Apr. 8, 2018).

[65] ——, *How to write to an event log by using Visual C#*, 2017. [Online]. Available: https://support.microsoft.com/en-us/help/307024/how-to-write-to-an-event-log-by-using-visual-c (Accessed: May 7, 2018).

[66] C. Taylor, *Incomplete Hardcopy Retrieved from TBS 1052B-EDU - Tektronix and Keithley*, 2018. [Online]. Available: https://forum.tek.com/viewtopic.php?f=580&t=140298&p=284585 (Accessed: Apr. 11, 2018).

[67] Microsoft, *NuGet Gallery — HtmlRenderer.PdfSharp 1.5.0.6*. [Online]. Available: https://www.nuget.org/packages/HtmlRenderer.PdfSharp (Accessed: Apr. 5, 2018).

[68] Oxyplot, *OxyPlot*, 2016. [Online]. Available: http://www.oxyplot.org/ (Accessed: Apr. 11, 2018).

[69] Microsoft, *Create a UI with XAML - Windows app development*. [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/apps/mt228349.aspx (Accessed: Apr. 9, 2018).

[70] Mono Project, *Home — Mono*, 2018. [Online]. Available: https://www.mono-project.com/ (Accessed: Apr. 10, 2018).

[71] ——, *WPF — Mono*. [Online]. Available: https://www.mono-project.com/docs/gui/wpf/ (Accessed: Apr. 10, 2018).

[72] National Instruments, *Instrument Control in Linux Tutorial - Developer Zone - National Instruments*, 2015. [Online]. Available: http://www.ni.com/white-paper/2816/en (Accessed: Apr. 10, 2018).

[73] Regex101, *Online regex tester and debugger: PHP, PCRE, Python, Golang and JavaScript*. [Online]. Available: https://regex101.com/ (Accessed: Jan. 15, 2018).

[74] Oxford Dictionaries, *mnemonic — Definition of mnemonic in English by Oxford Dictionaries*. [Online]. Available: https://en.oxforddictionaries.com/definition/mnemonic (Accessed: Jan. 18, 2018).

[75] ——, *parse — Definition of parse in English by Oxford Dictionaries*. [Online]. Available: https://en.oxforddictionaries.com/definition/parse (Accessed: May 9, 2018).

Bibliography

M. Stafford and J. W. Thompson, "Speak Easy: The Future Answers To You," 2017. [Online]. Available: https://www.mindshareworld.com/sites/default/files/Speakeasy.pdf.

G. Moulton, H. LaDeana, J. Hertz, and M. Levenson, *Accessible Technology in Today's Business*, 1st ed. Microsoft Press, 2002, ISBN: 0735615012.

C. Nicolle and J. Abascal, *Inclusive Design Guidelines for HCI*, 1st ed., C. Nicolle and J. Abascal, Eds. Taylor & Francis, 2001, ISBN: 0748409483.

A. Sears and J. A. Jacko, *The Human-Computer Interaction Handbook.* 2008, ISBN: 978-0-8058-5870-9.