Wireless-based Text Entry System for Handicap

BY

CHOOI WAI LUM

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMPUTER ENGINEERING

Faculty of Information and Communication Technology
(Perak Campus)

MAY 2018

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: ____

____

____

Academic Session: _____________

I ____

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

_________________________ _________________________

(Author's signature) (Supervisor's signature)

Address:

____

____

____

____

Supervisor's name

Date: _____________________ **Date**: ____________________

Wireless-based Text Entry System for Handicap

By

Choi Wai Lum

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMPUTER ENGINEERING

Faculty of Information and Communication Technology
(Perak Campus)

MAY 2018

DECLARATION OF ORIGINALITY

I declare that this report entitled "**Wireless-based Text Entry System for Handicap**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : _________________________

Name : _________________________

Date : _________________________

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation and gratitude to my supervisor, Mr. Leong Chun Farn who has given me this great opportunity to help the handicaps by designing a new text-entry system for them. He did a great job in giving me some new ideas for this project. Besides, this project is my first project developed using the Arduino board.

In addition, it would be not fair if I do not mention my friends and family members. With the guides and ideas provided by my friends did save me a lot of time during the development of this project. Finally, I feel grateful because of the continuous support and encouragement from my beloved family.

ABSTRACT

Text entry system can be found commonly in modern computer and mobile devices such as smartphone and tablet. Despite this, these text entry systems are not easily accessible by handicaps. The major problem with the normal text entry system such as the “QWERTY” keyboard, it would be very difficult for the handicaps to input the keys. To use an ordinary keyboard, major hand movement is needed for them because the keyboard contains over a hundred keys. They might have to move across the keyboard to enter the desired keys. Hence, the usage of a keyboard for the handicaps is not recommended especially for those who suffered from motor disabilities.

A new wireless-based text entry system is described in this research to help the handicaps to ease their lives and input text faster. This new text entry system will be much smaller than the traditional keyboard. In addition, this system would be designed to be portable, lightweight, long operating hour and user-friendly. Besides, this system enables the handicaps input text efficiently with minimum hand movement. Besides, the handicaps will only require one hand to hold the equipment and one finger to interact and to input text.

This equipment will be built using the Printed Circuit Board (PCB), the Arduino Uno board (CT-UNO) and a Bluetooth module. PCB is used instead of using breadboard is to keep the device with the minimum wires. Besides, the Bluetooth is used to ensure the device can communicate to the computer wirelessly.

The algorithm used in this text entry system is Press-Drag-Release algorithm where the user first pressed on the infrared text entry board, then the user drags the finger around the board to choose the letter and then release. Next, the respective letters will be printed on the computer. As a result, the user only needs to perform 3 steps to input a letter.

Apart from inputting letter to the computer, the text entry system is also a combination of keyboard and mouse where the user can use the text entry system as a mouse and navigate through the computer like the ordinary mouse.

The average entry rate of this text entry system is about 5 WPM. The result is better than some of the system that designed for the handicapped.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADC	Analog-to-digital converter
C#	C Sharp
COM	Communication
CGA	Character Graphical Association
CPM	Characters per minute
GUI	Graphical user interface
ICT	Information and Communication Technology
IDE	Integrated development environment
IR	Infrared
LED	Light-emitting diode
PC	Personal Computer
PCB	Printed Circuit Board
T9	Text on 9 keys
USB	Universal Serial Bus
WPM	Words per minute

CHAPTER 1: INTRODUCTION

Rapid growth in ICT causes a great demand for digital devices such as smartphones, tablets as well as desktop PC and laptop (Panwar, Sarcar, and Samanta, 2012). To interact with those devices, we need text entry systems such as keyboards. Generally, there is a lot of modern text entry system invented to ease our life. The most important matter to be concerned when user input text into a computer is they must have the ability to type speedily and precisely (Clawson et al., 2005). Currently, keyboards provide the most common way to communicate with computers. The most popular keyboard is the “QWERTY” keyboard (Noyes, 1998). This text entry system can be used on both mobile devices and computer. For example, a “QWERTY’ keyboard is commonly used in computer and for mobile devices, a virtual “QWERTY” keyboard on a touchscreen is used. Unfortunately, this text entry system may become an obstacle for the handicaps to interact with devices (Fu and Ho, 2009). A handicapped person would be able to use the “QWERTY” keyboard, but it would be very difficult for them to enter the keys.

1.1 Problem Statement and Motivation

Handicaps may find difficulties such as they may take a longer time to input when using the traditional text entry system such as using the “QWERTY” keyboard. By using this type of keyboard, they have to move their fingers across the keyboard and enter the words they want. They will feel tired as their fingers will feel fatigue and give up on typing easily. Unfortunately, input devices that are specially designed for handicaps will be expensive and they will not be able to afford the devices. A new text entry system is needed to help them and solve the problems they are facing. To solve this problem, a new text entry system is proposed to help the handicaps to input text more easily and efficiently. Besides, the system will be developed using low-cost components. The system will be a wireless-based system to make the system more portable and easy to use.

1.2 Background Information

1.2.1 Typewriter

A typewriter is an ancient text entry system used between the 1870s and 1980s. The typewriter has a "QWERTY" layout just like the modern keyboard. The typewriter is shown in Figure 1-1 below.

Figure 1-1: The typewriter

The typewriter was a success text entry system because small companies were transforming into large companies during the late 19th century, which results in a demand for enhanced communications (I. Scott MacKenzie and Tanaka-Ishii, 2007). Typing errors are one of the biggest issues in the typewriter. One of the drawbacks of using a typewriter is you cannot erase what you have written easily. But, there are some workarounds to solve this problem such as using a special eraser to rub off the mistakes, correction fluid or use a typewriter equipped with an auto-correction feature which can remove mistakes made when users clicked on the auto correct button (Chris Woodford, 2016).

1.2.2 Standard 101/104 keys "QWERTY" keyboard

"QWERTY" keyboard is a very common keyboard used as a text entry system for computers. This keyboard layout was duplicated from the typewriter (I. Scott MacKenzie and Tanaka-Ishii, 2007). The "QWERTY" keyboard is shown in Figure 1-2 below.

Figure 1-2: The “QWERTY” keyboard layout

There are many types of “QWERTY” keyboard such as the multimedia keyboard, wireless keyboard, gaming keyboard and etc. Multimedia keyboards come with additional keys compared to ordinary keyboards such as volume control and mute buttons, application launch buttons. For wireless keyboards, it is the same as ordinary keyboard except it is cordless. Users will just have to plug in a receiver on the computer side and they can start to use the keyboard. Gaming keyboards are usually designed for users who like gaming.

1.2.3 The 12-Key Keypad for Mobile Phone

Nowadays, most smartphones have one thing in common – a touchscreen capable of detecting multiple inputs at once. Touchscreen with a virtual keyboard enables users to type quickly and efficiently. But for older phones with no touchscreen, users will have to depend on the 12-Key Keypad for text entry. The layout of the 12-Key keypad is shown in Figure 1-3 below.

Figure 1-3: The 12-Key keypad on a mobile device

This 12-Key keypad gives a unique challenge to users for text entry because nearly all languages have almost 26 characters (I. Scott MacKenzie and Tanaka-Ishii, 2007). To enter text using the 12-Key keypad, each key is entered one or numerous times to determine which character is needed by users. For example, the user wants to enter a string “keyboard”, the user first need to press key 5 for 2 times, key 3 for 2 times, key 9 for 3 times, key 2 for twice, key 6 for 3 times, key 2 for once, then key 7 for 3 times, and finally key 3 for once to complete the string “keyboard”. Hence, the typing speed for this tapping method is quite time-consuming.

Fortunately, there is another better method named as “predictive” method, to be used with this 12-key keypad, rather than tapping the same key multiple times. The most famous predictive method is the T9 by Tegic. By using T9, the user would only have to tap once for each and every character, then all the recommended strings will be shown based on the phone’s dictionary. For example, the user wants to type “keypad”, the user would only have to tap key 5, key 3, key 9, key 7, key 2, and key 3, all keys at once and the user chooses the word needed from all the recommended strings. This method is more time saving compared to the previous method. However, there are some drawbacks to using this predictive method. One of the drawbacks is the fluency of typing. If the word needed by the user is not in the dictionary, then the user may have to change back to the tapping method.

1.2.4 Speech-to-Text

Speech recognition has achieved significant research concern as most users can speak more quickly than they write (I. Scott MacKenzie and Tanaka-Ishii, 2007). Error corrections are crucial in speech-to-text. The speed of speech-to-text is slow due to if users keep correcting mistakes in dictation through the speech. In the Karat et al. (1999) study, the traditional keyboard-mouse combination text transcription speed is at 33 words per minute, which was faster than the text-to-speech that is about 14 words per minute. Speech-to-text can help those users who suffer from physical impairment, as an alternative way to replace the keyboard.

1.3 Project Scope

The goals of interaction design are taken into consideration when designing a product. There are 6 usability goals in total which are effective to use (effectiveness), efficient to use (efficiency), safe to use (safety), have a good utility (utility), easy to learn (learnability) and easy to remember how to use (memorability) (Sharp et al., 2007).

In this system, usability goals such as effectiveness, efficiency, safety, learnability, and memorability are taken into consideration. The proposed text entry system must solve the text entry problem faced by the handicaps effectiveness. The input device must be efficient to use so that the handicaps can use the device without facing any difficulties. Besides, the safety of the handicaps is an important factor that needed to consider when developing the system. The device must be safe to use to avoid any unwanted accident from happening.

The system must be easy to learn and adapted by the handicaps. The input device is a 9-key input pad with the arrangement of 3 x 3 grid which makes the handicaps more easily to enter texts. Due to the small grid of arrangement of the keyboard, usability goal memorability is achieved. The users will only need minor memorizing to remember how to use this new text entry system.

1.4 Project Objective

At the end of this final year project, a new text entry system will be proposed to help the handicaps to enter texts more efficiently. This project will develop a hardware which is a low-cost input device and a software which is the GUI for this text entry system. The objective of this project is to provide a suitable and low-cost text entry system for the handicaps to let them communicate simultaneously with others without letting them feel like they are being left out or abandoned.

Sub-objectives:

- Portable – Can bring the device to anywhere and start using it.
- Lightweight – The device must be not bulky.
- Long operating hour – The device must be able to operate for longer hours. No need to charge so frequent.
- Wireless connection – Reduce the number of wires. The increasing number of wires will cause the device to become more complex.
- User-friendly – Need only minor instruction to operate the device.

However, this new text entry system is only suitable for handicaps who suffered from minor motor disabilities. To use the input device, minor hand movement is needed. Hence, handicaps with severely physically handicapped people are not advised to use this text entry system. However, the handicaps with vision impairment can still utilize this system as they can memorize the keypad pattern on the input device.

1.5 What Have Been Achieved

The text entry system is capable of typing 26 alphabets in uppercase and lowercase. To enter the alphabets in uppercase, the 'Caps Lock' and 'Shift' function are provided. Besides, it can be used to input numbers too. Other than typing alphabets and number, this system can be used to 'Select All', 'Copy', 'Cut', 'Paste' the text like the ordinary keyboard. If the user mistyped a character, 'Backspace' function is provided to remove wrongly typed characters. Furthermore, 'Tab' key and the 'Space' key are provided too for the user to insert spaces between the words. If the user wants to input a carriage return, the 'Enter' function is provided to enter a newline. Punctuations such as comma, period, exclamation mark, question mark, plus sign, minus sign and slash can be inserted with this text entry system.

Apart from the text entry function, the system is able to simulate like a mouse device. After choosing the mouse function, the system can be used to move around the mouse pointer of the computer. Besides, the 'Left Click', 'Right Click' and 'Middle Click' function are provided so that the user can use this system as an alternative to the mouse device.

1.6 Impact, Significance and Contribution

This project develops a new wireless-based text entry system for handicaps to improve their text entry experience. In addition, this project will make the handicaps' life easier by introducing them a new text entry system that is specially designed for them. The system consists of 2 parts which include hardware and the software GUI. The hardware which is the input device is designed as small as possible to let the user can easily hold the input device and operate with one hand only. For the hardware implementation part, the developed hardware is for the user to hold and do the text entry process and then send the text entered by the user wirelessly to their computer or laptop. Then, the software implementation part will take place on the user's computer and interprets the text entered by the user and show it on the computer.

Besides, this project can contribute to the handicaps to increase their productivity. Some of the handicaps work in companies which they need to do word processing. With this new text entry system, they can complete their job faster and efficient. Next, the innovation of this project is the input device of the system is wireless based, which means they can bring the device with them everywhere and start to type wirelessly.

1.6 Report Organization

The details of this research are shown in the following chapters. In Chapter 2, some reviews are made, and some comparison of previous works are done in this chapter. Besides, the designs of the text entry system and the GUI program are discussed in Chapter 3. Then, Chapter 4 discussed the methodology of this research and the tools used in this research. Furthermore, Chapter 5 discuss the implementation and testing of the text entry system to test whether the system functions as expected or not. Lastly, Chapter 6 discuss the conclusion of this research such as the project review, novelties, limitation of the system and future enhancement.

CHAPTER 2: Literature Review

There are many researchers work hard in implementing a new text entry system for the handicaps. Some of the research papers will be discussed and the advantages and the disadvantages of the systems will be compared as well.

2.1 Text Entry System Based on a Minimal Scan Matrix for Severely Physically Handicapped People

This aim of this research is to let the users which are physically handicapped to connect and communicate with other people (Miró and Bernabeu, 2008). Besides, this research describes a switch-based text entry method for handicaps to use with minimal of training and loads.

2.1.1 The Design of the System

This system is based on a 2-cell only scan matrix. The first cell contains alphabets in alphabetical order from the character ‘a’ to character ‘m’ while the second cell contains alphabet from character ‘n’ to character ‘z’. During the first stage, the automatic scan method is used to scan through the cells. The user needs to press the switch provided when the alphabet needed is highlighted. The design of the system is illustrated in Figure 2-1 below.

a	b	c	d	e	f	g	h	i	j	k	l	m

First cell

n	o	p	q	r	s	t	u	v	w	x	y	z

Second cell

Figure 2-1: 2 cells consisting of the ‘a’ to ‘z’ alphabets

The second stage starts right after the first stage ends. The cell that contained characters are shown in statistical order using the 4-gram linguistic model, hence, previous characters are taken into consideration. When a long press in the character of a word during the first stage is detected, space will be added.

Normal sentence	Unigrams (1-gram)	Bigrams (2-gram)	Trigrams (3-gram)	4-gram
Honesty is the best policy	Honesty, is, the, best, policy	Honesty is, is the, the best, best policy	Honesty is the, is the best, the best policy	Honesty is the best, is the best policy

Figure 2-2: Example of N-gram Linguistic Model

2.1.2 Advantages and Disadvantages

Advantages	Disadvantages
• **Ease of use** One of the usability goals in Human-Computer Interaction. The users do not need extra knowledge to use this system.	• **Takes longer time** The users have to wait until the desired character is highlighted, then the user only can choose the character.
• **Faster scan rate** The system scan through each cell with a scan rate of 0.5 seconds.	• **No additional keys** Keys like punctuation, number keys and function keys such as delete, insert and others cannot be entered into this system.
• **Expensive equipment is not required** Only low-cost hardware such as switches	

Table 2-1: The advantages and disadvantages of the minimal scan matrix system

2.1.3 Comparison with the proposed solution

The time needed to enter a character varies with different characters. If the user wants to enter character “A”, he can enter the mentioned character quickly since the mentioned system above uses automatic scanning method that will scan through the 2 columns consists of 26 alphabets, starts from character “A”. If the user wants to enter character “Z”, he will have to wait until the automatic scanning method scan until the character “Z”, then only he can enter the character. Unlike the proposed solution, the time to enter a character is almost the same with different characters since it does not use the automatic scanning method.

2.2 Using a Game Controller for Text Entry to Address Abilities and Disabilities Specific to Persons with Neuromuscular Diseases

In this research, a game controller is used as an input device to become an alternative text entry system for people with the neuromuscular disease (Felzer and Rinderknecht, 2011). The chosen game controller is illustrated in Figure 2-3 below.

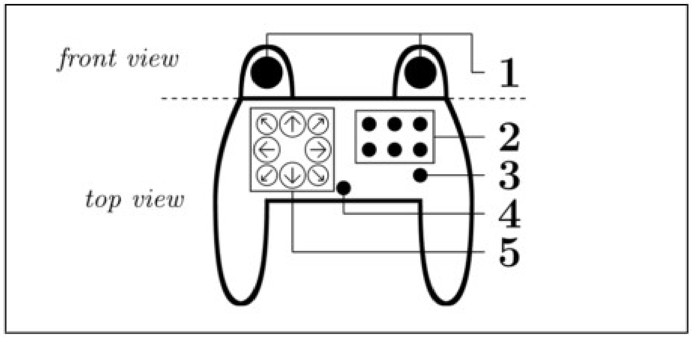


Figure 2-3: The game controller used in the text entry system

1 = trigger buttons; 2 = action buttons; 3 = mode button

4 = shift button; 5 = 8-way D-pad

2.2.1 The Design of the System

The software design of the system is shown in Figure 2-4 below.

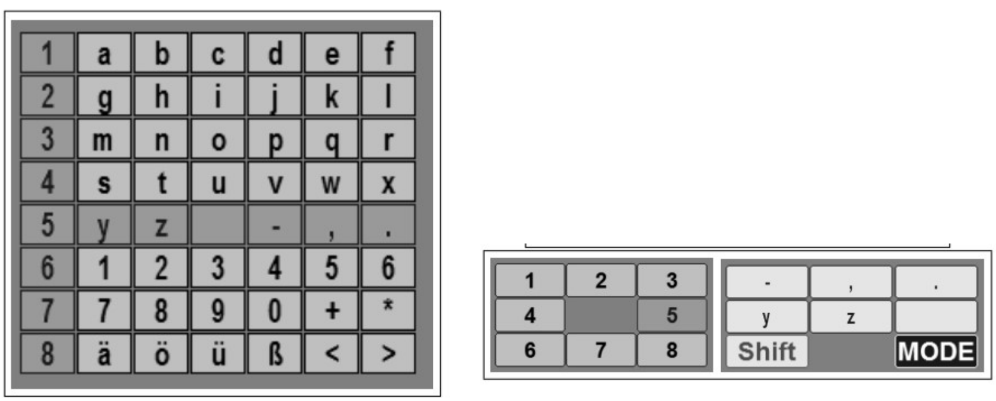


Figure 2-4: Eight-by-six grid showing characters that can be selected (left) and indicators for the buttons for the left and right thumbs (right)

The software GUI shows an on-screen keyboard, letting the user pre-select one of the 8 rows using the D-pad and press the corresponding character the user wants using the 6 action buttons on the right. After that, the character is written.

2.2.2 Advantages and Disadvantages

Advantages	Disadvantages
• **Time-saving** The user will only have to move the D-pad and choose the character they want.	• **Expensive equipment is required** A game controller is relatively expensive and may cause the users cannot afford to buy the equipment
• **Less exhausting** The user feels less tired after using this text entry system.	• **Need to get familiar with the system** The user may need to learn how to use the system.
• **No need to design and build new equipment** This system uses an end product as a device to create a new text entry system.	• **Less additional keys** Only a few punctuation keys can be entered Function keys such as delete, insert and others cannot be entered into this system.

Table 2-2: The advantages and disadvantages of using a game controller for a text entry system

2.2.3 Comparison with the proposed solution

The mentioned system above may need the user to memorize and learn how to use the system. Besides, the user may need to hold the equipment using both their hands. This may cause the handicaps user feel fatigue after holding for a long time. Unlike the proposed solution, the user needs only one hand to hold the equipment and starts typing. Besides, the proposed solution is easier to use compared to the mentioned system above. The user can enter text easily using the proposed solution.

2.3 BrailleKey: An alternative Braille text input system

Braille is a system of raised dots on paper describe the alphabet letters that used by the blind people in order to read the letters. They use their fingertips to feel the raised dots and read the paper (Groups and Cell, n.d.). The Braille alphabets and numbers are shown in Figure 2-5 below.

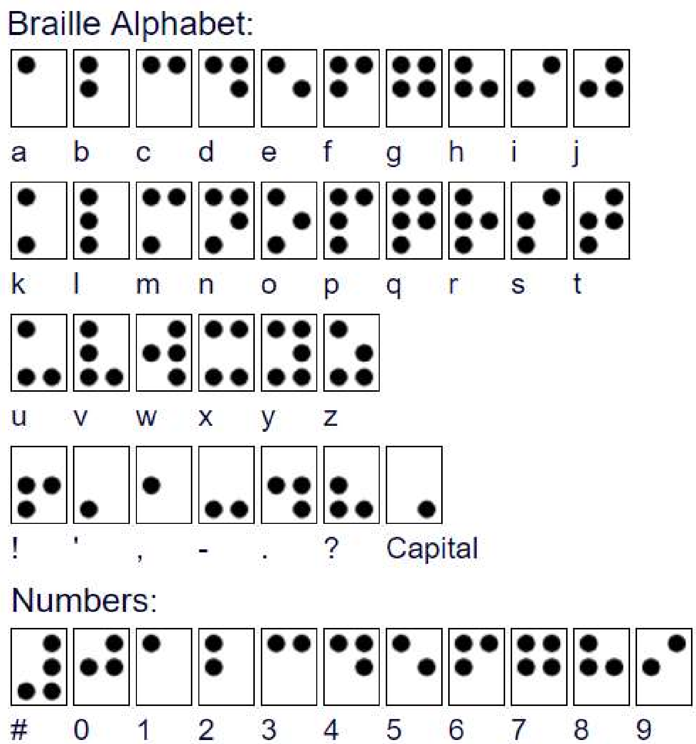


Figure 2-5: The Braille alphabets and numbers

2.3.1 The Design of the System

This research is about implementing the Braille system into touch-screen enabled smartphones such as iOS and Android devices – BrailleKey (Subash et al., 2012). The layout of the BrailleKey application for smartphones is shown in Figure 2-6 below.

Figure 2-6: The layout of BrailleKey

The left button represents the left side of the Braille character while the right button represents the right side. A single click of the left or right button triggers the first cell of the Braille character. A double-click of the left or right button triggers the second cell of the Braille character while a long hold of the left or right button triggers the third cell of the Braille character.

Besides, a single click of the enter button inputs the text while double click on the enter button inserts a space character. The user can delete a last entered character by tapping the delete button once.

A tap of the volume key of the smartphone replays everything that has been typed using the speakers of the phone to ensure that there is no error committed.

2.3.2 Advantages and Disadvantages

Advantages	**Disadvantages**
• **Voice feedback** The system provides voice feedback to the user when the user is tapping.	• **Expensive equipment is required** A smartphone is expensive and may cause the users cannot afford to buy the equipment.
• **Simplified interface** The interface only contains 4 buttons which are left, right, enter and delete buttons.	• **Not suitable for some users** Not all handicapped users suffered from visual impairments, therefore, they may not familiar with the Braille characters
• **No need to design and build new equipment** This system uses a smartphone as a device to create a new text entry system.	• **Less additional keys** No punctuation key is provided. Special function keys such as Shift and Caps Lock are also not provided.

Table 2-3: The advantages and disadvantages of the BrailleKey system

2.3.3 Comparison with the proposed solution

For the mentioned system above, not everyone knows how to read the BrailleKey. Hence, the user of the system may need to understand how BrailleKey works and it takes a lot of time. Unlike the proposed solution, the user need not have extra knowledge to use the system, just a simple understanding of how the system works will do.

2.4 An Ambiguous Keyboard Based on "Character Graphical Association"

An ambiguous keyboard is a type of keyboard that has lesser keys compared to the traditional keyboard (Miró-Borrás et al., 2010). CGA acts as a new way to select letters to keys and generate a new keyboard layout with 2, 3 to 4 keys only.

2.4.1 The Design of the System

First, the consonants are written on a ruled paper with multiple lines as shown in Figure 2-7 below.

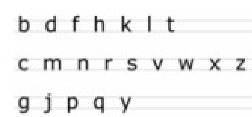


Figure 2-7: 21 consonants

Then, the consonants are grouped into 3 groups:

1. Consonants exceeding the top line (b, d, f, h, k, l, t)
2. Consonants between the lines (c, m, n, r, s, v, w, x, z)
3. Consonants exceeding the bottom line (g, j, p, q, y)

Besides, the vowels are distributed among other different keys to improve performance during the text entry process. For the user to learn the layout easily, the following requirements must be satisfied which only consecutive vowels in alphabetical order can be matched to a key and the keyboard must have the ability to display the all 5 vowels in alphabetical order. Some of the CGA layouts are shown in Figure 2-8 below.

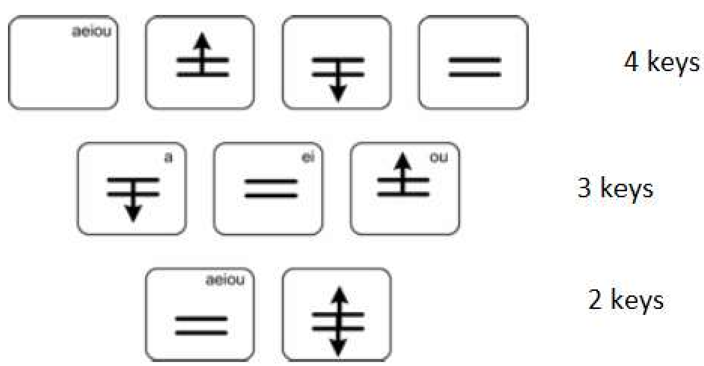


Figure 2-8: The 2-key, 3-key, and 4-key CGA layouts

There are 2 stages in this system which are cell selection stage and disambiguation phase stage. During the first stage starts, the user clicks the switch when the desired cell is highlighted using automatic scanning. To type the last letter, the user has to press and hold the switch to indicate that the user has finished typing the letters. Next, the second stage starts by showing the suggested words one by one until the user found the desired word, then only can release the switch. Finally, the word is entered followed by a space.

2.4.2 Advantages and Disadvantages

Advantages	Disadvantages
• **Lesser keys** The system has lesser keys compared to the conventional keyboard, but it preserves all the important keys.	• **Takes longer time** Need to undergo a long process before entering a word.
• **Letters are grouped** Consonants and vowels are grouped into few keys to prevent any confusion.	• **Exhausting** The user may feel tiredness or fatigue when using this system because the user must hold the switch button until the desired word appeared.

Table 2-4: The advantages and disadvantages of the CGA layout-based text entry system

2.4.3 Comparison with the proposed solution

The CGA layout-based text entry system requires the user to press and hold the switch button until the words the user wanted appears. This may take a lot of time and it will make the user feels fatigue and tired after holding the switch for a long duration. Unlike the proposed solution, the proposed solution does not have a switch, therefore, the user does not have to hold it to type, the user only needs to move through the keypad and start typing.

2.5 Infrared-based Text Entry System for Handicap

This research paper is about to create a new infrared-based text entry system for the handicaps to ease their lives when they are doing text entry task (K.W. Yong, 2015). This paper is similar to the proposed solution of this project except for the proposed text entry system, the data is transferred to the user's PC or laptop wirelessly while the text entry system of the paper transmits the data using a wired connection.

2.5.1 The Design of the System

The Figure 2-9 below shows that the text entry system which is specially designed for the handicaps.

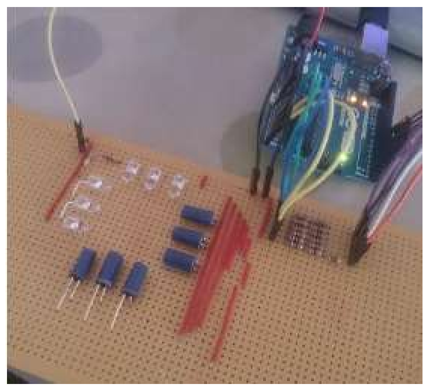


Figure 2-9: The Infrared-based Text Entry System

The circuit diagram for the arrangement of IR LED is shown in Figure 2-10 below.

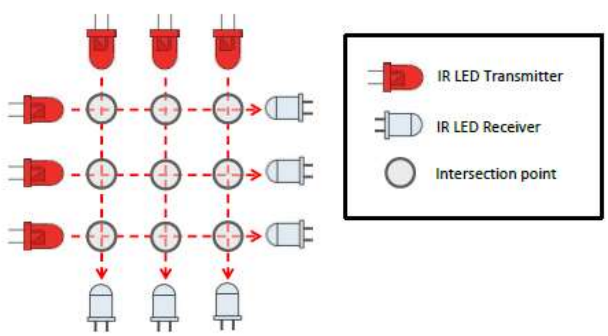


Figure 2-10: The arrangement of the IR LED for the keypad

When the user's fingertip touches the one intersection point, the corresponds IR LED will generate an interrupted and the interrupt is transmitted to the user's PC or laptop via the Arduino board and process by the software to determine which key is pressed.

A software GUI acted as an on-screen keyboard is developed to interpret the data from the Arduino board. The software GUI is shown in Figure 2-11 below.

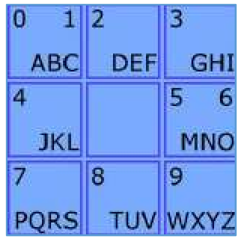


Figure 2-11: The software GUI of the infrared-based text entry system

The user first thinks of a letter to type then click on the corresponding key column which consists of the letter the user wants. Then the second stage starts as shown in Figure 2-12.

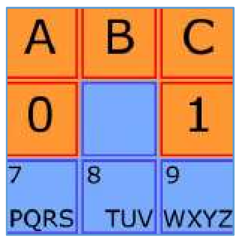


Figure 2-12: The second stage of the software GUI of the infrared-based text entry system

Then, the user may select the letter by touching the corresponding column of the keypad based on the layout of the second stage of the software GUI. Finally, the letter is entered. The Figure 2-13 below shows the interface of the GUI program.

Figure 2-13: The interface of the GUI program

2.5.2 Advantages and Disadvantages

Advantages	Disadvantages
• **Lesser keys** The system has lesser keys compared to the conventional keyboard but preserve almost all the keys of the conventional keyboard.	• **Wired connection** The connection between the input device to the user's PC or laptop may cause inconvenience or not portable
• **Relatively small** The user can hold the input device with one hand only.	• **Cannot directly input to the text editor** If the user wants to type letters in the text editor, the user first enters the letters to the GUI program then only copy the text from the GUI to the text editor.

Table 2-5: The advantages and disadvantages of the infrared-based text entry system

2.5.3 Comparison with the proposed solution

The mentioned system above uses wired connection unlike the proposed solution uses a wireless connection. With the proposed solution, the user can start typing after the wireless connection between the system and the computer is established. The wireless connection keeps the proposed text entry system less complex and wires.

CHAPTER 3: System Design

3.1 System Overview

To develop the wireless-based text entry system for the handicaps, an input device and a software are needed to perform the text entry task. Hence, the device will be built using some low-cost components and a development board.

To build the input devices, a few of components are chosen and are taken into consideration. The key arrangement of the input device is illustrated in Figure 3-1 below.

1	2	3
4	5	6
7	8	9

Figure 3-1: 3 X 3 grid key arrangement

3.2 Full System Diagram

The full system diagram of the project is shown in Figure 3-2 below.

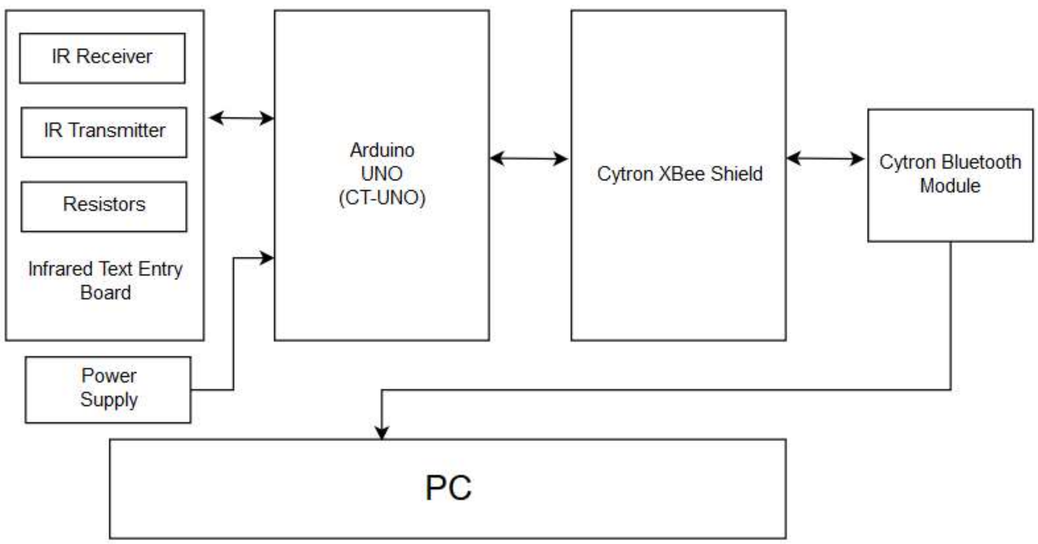


Figure 3-2: Full system diagram

In this project, the CT-UNO is used as the microcontroller to read analog signal from the IR transmitter and receiver and transmit those values to the PC via the Bluetooth module to determine which letter is pressed or function is performed. The Bluetooth module is mounted on the Cytron XBee Shield then the shield will be stacked on top of the CT-UNO. Besides, the infrared text entry board is a PCB where the IR transmitters, the IR receiver, and the resistors are soldered into the PCB. This is to keep the keypad as simple as possible without all the jumper wires. Next, the infrared text entry board is stacked on top of the Cytron XBee Shield. When the user operates on the infrared text entry board, the input data will be transferred via the Bluetooth module to the computer. The Figure 3-3 below shows the side-view illustration of the text entry system.

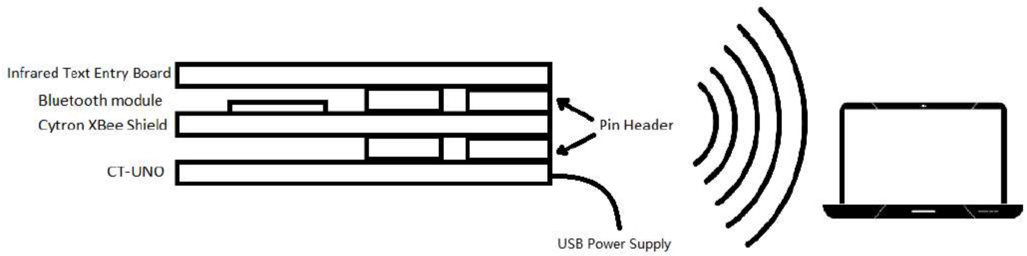


Figure 3-3: Side-view illustration of the wireless-based text entry system

3.3 Hardware Implementation

The design of the input keypad will be a 9-key arranged in a 3 by 3 grid. Multiple IR LEDs are placed on the keypad to detect the key press by the user. The IR LED arrangement is shown in Figure 3-4 below.

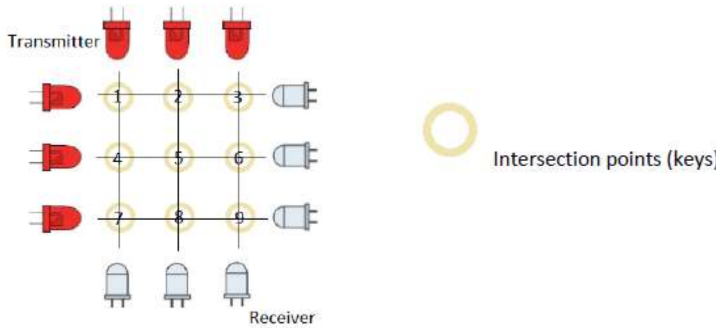


Figure 3-4: The arrangement of the IR LEDs

The IR light travel in a direct path from the transmitter to the receiver. When the user's fingertip touches one of the intersection points, the IR light emitted by the transmitter will be blocked.

The receivers are connected to the ADC inputs of the CT-UNO to get the ADC values since the CT-UNO includes a 10-bit ADC. The 10-bit ADC gives the digital value between 0 and 1023 from the input voltage ranging from 0 to 5V.

For example, the transmitter A emits IR light to the receiver B. If neither the transmitter nor the receiver is blocked, the ADC value is usually near to 1023. But, if either the transmitter or the receiver is blocked, the ADC value will be dropped to less than 1023. By using this concept, the intersection point that pressed by the user will be identified and send to the software GUI program.

The system consists of an Infrared text entry board, a Cytron XBee Shield mounted with the Bluetooth module and CT-UNO board. Figure 3-5 below shows the hardware implementation part.

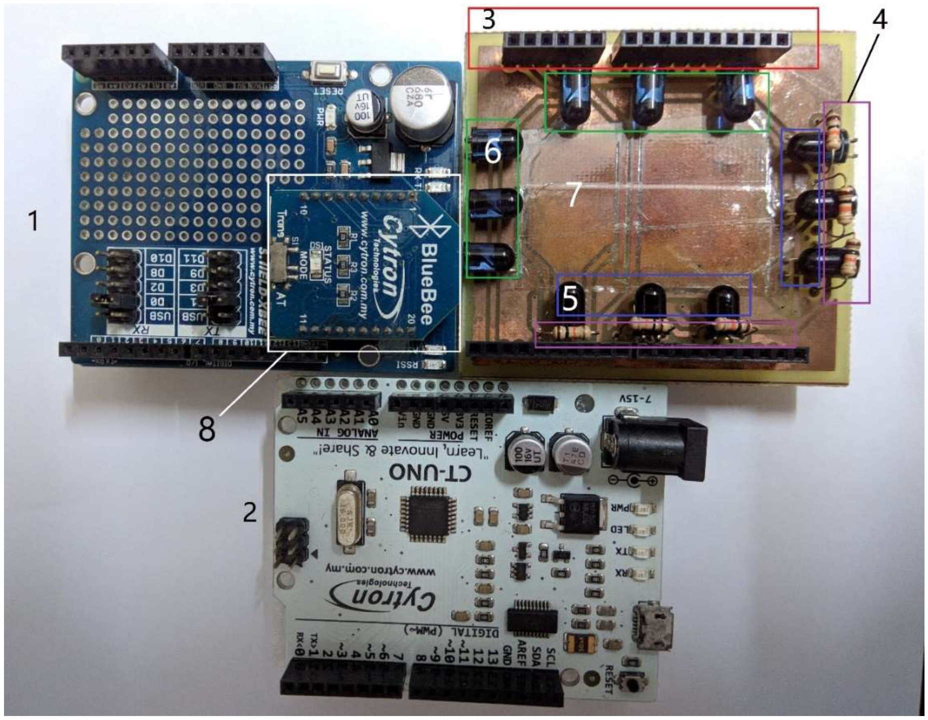


Figure 3-5: The hardware part of the text entry system

No.	Hardware Components	Amount	Total Price (RM)
1.	Cytron XBee Shield	1	17.50
2.	CT-UNO	1	29.40
3.	Header Pin	4	4.90
4.	10K ohm resistors	6	0.30
5.	IR Receiver	6	3.00
6.	IR Transmitter	6	3.00
7.	Infrared Text Entry Board (PCB)	1	0.00
8.	Cytron Bluetooth Module	1	34.00
		Total	92.10

Table 3-1: The details and price of the hardware components

The development cost of the whole text entry system is less than RM100 which is about RM92.10. The details of the hardware components and the software components used in this project will be discussed in detail in Chapter 4: Methodology and Tools.

To establish the connection between the device and the computer in Windows 10 wirelessly, the user needs to pair the Bluetooth module of the device with the Bluetooth adapter of the computer first. First, the user has to open the Bluetooth settings on the computer and click "Add Bluetooth or other device" to search for the Bluetooth module of the device as shown in Figure 3-6 below.

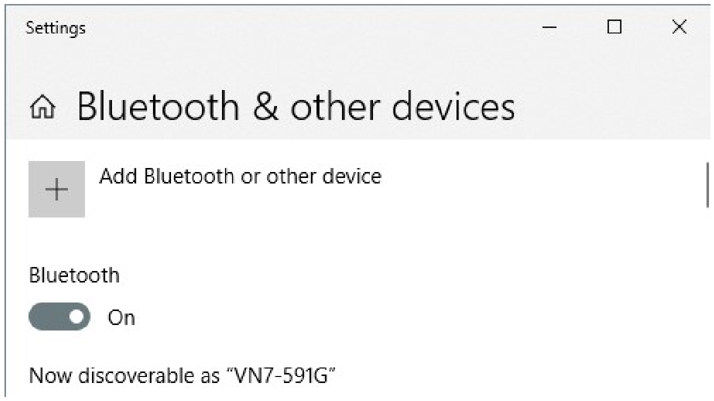


Figure 3-6: Bluetooth settings of the computer

Then, the user has to select the "Bluetooth" option and the searching process will begin. When the computer found the name of the Bluetooth module name which is "H-C-2010-06-01", the user need to click it and insert the PIN "1234" and click "Connect" to complete the pairing process as shown in Figure 3-7 below.

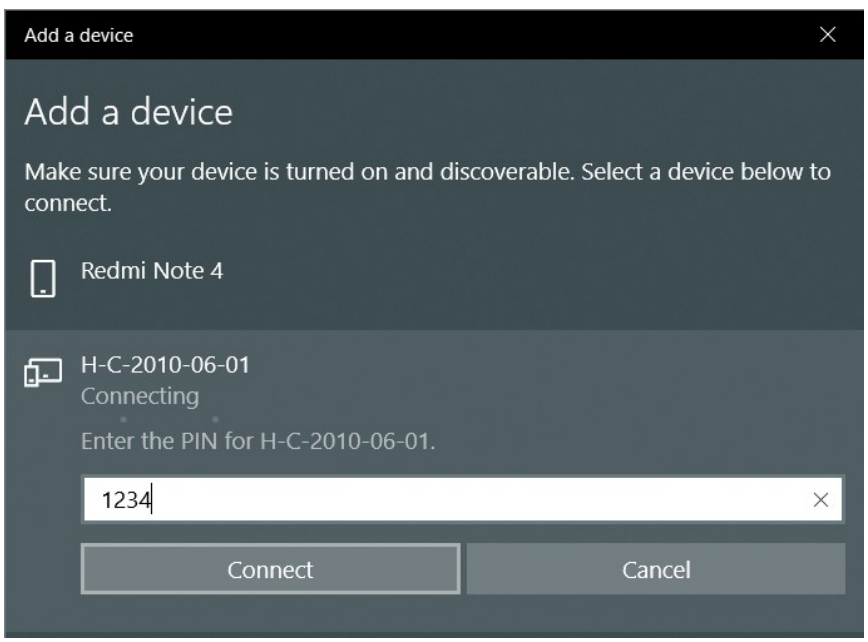


Figure 3-7: Bluetooth pairing is in progress

Next, if the PIN entered is correct, the pairing process will complete as shown in the Figure 3-8 below.

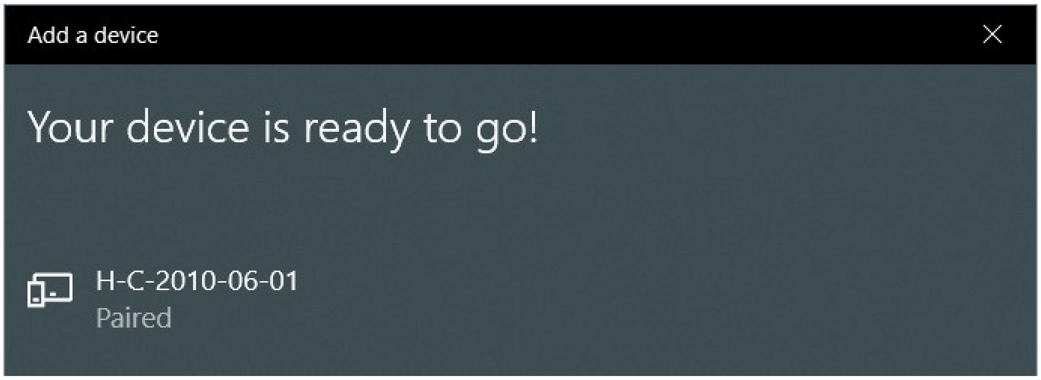


Figure 3-8: Bluetooth pairing is succeeded

Next, the user needs to know which COM port is used by the device. Hence, the user needs to go to the "Device Manager" of the computer and select the "Ports (COM & LPT)". If the pairing process is completed without any error, there will be at least one entry which is "Standard Serial Over Bluetooth link" followed by the COM port name as shown in Figure 3-9 below.

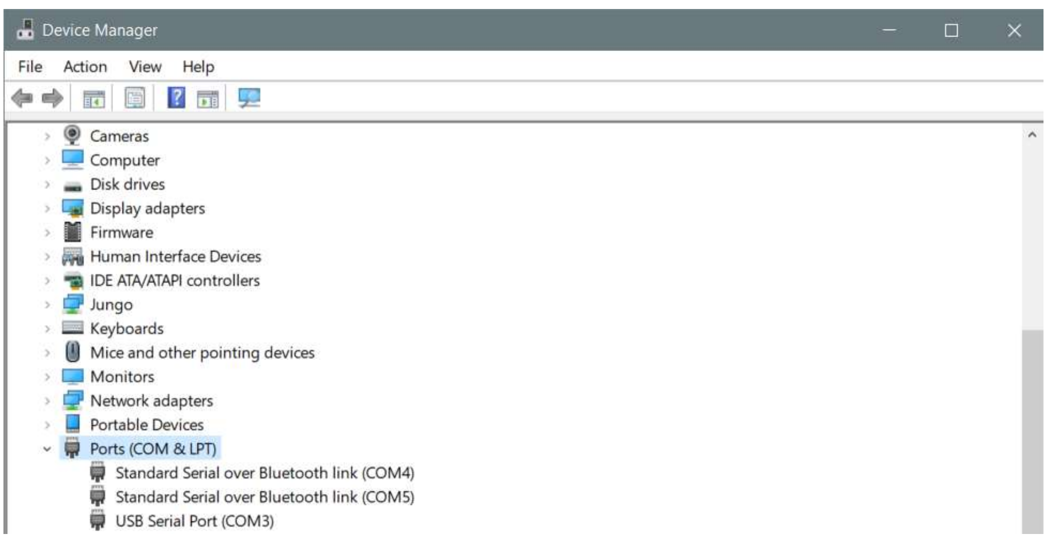


Figure 3-9: Task manager of the computer

In this case, the COM port used by the device is "COM4". This name of the COM port is important as the user need to choose the respective COM port on the software GUI later to connect the GUI program and the device.

3.3.1 Flowchart of Hardware Implementation

The Figure 3-10 below shows the flowchart of the hardware implementation part of the text entry system.

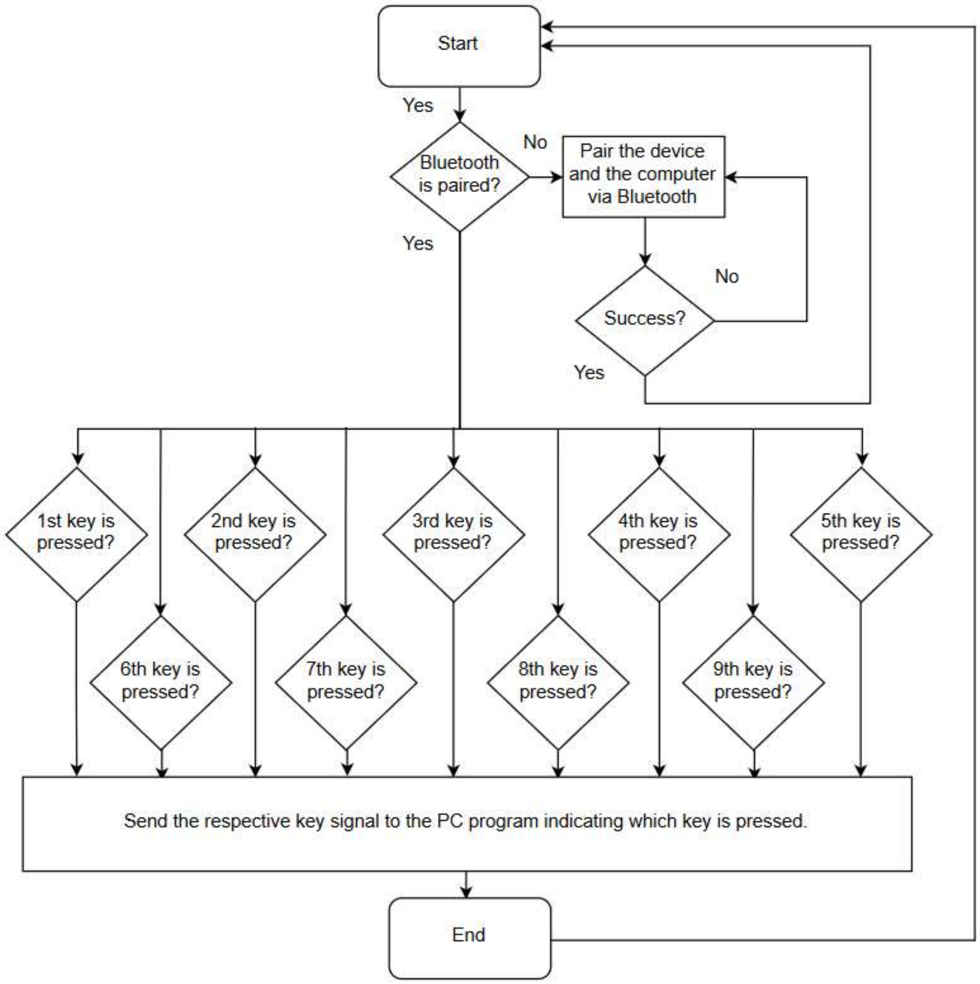


Figure 3-10: Flowchart for the hardware implementation part

Once the device is first booted up, the user has to pair the device with the computer first or else the computer will not receive any signal from the device. This is a one-time process. After the pairing process completed, it will wait for the user input and check whether the user has pressed which intersection point or key. If the first key is pressed, the first key signal will be sent to the GUI program indicating that the first key has been pressed and the GUI program will proceed.

3.4 Software Implementation

A software GUI is required to interpret the input from the input device which is the CT-UNO board. The GUI will be developed using the C# programming language. The C# programming language is chosen as the language has good compatibility with the Windows operating system. The GUI will be designed in such a way that is similar to the key arrangement of the input device where 9 buttons are arranged in a 3 by 3 grid. This GUI will act as a visual aid for the user. The principle of working for this system is straightforward. There are 9 buttons which consist of all the alphabets, numbers from 0 to 9, some special function such as Copy, Cut, Paste, Caps Lock, Shift, Select All, Space, Enter and some basic punctuations.

When the user launches the program for the first time, the user needs to choose the correct COM port first from the drop-down list and click "Connect". Then, the program will remember the COM port chosen by the user and next time when the user launches this program again, the program will just auto connect with the device using the last saved COM port. The interface of the GUI program is illustrated in Figure 3-11 below.

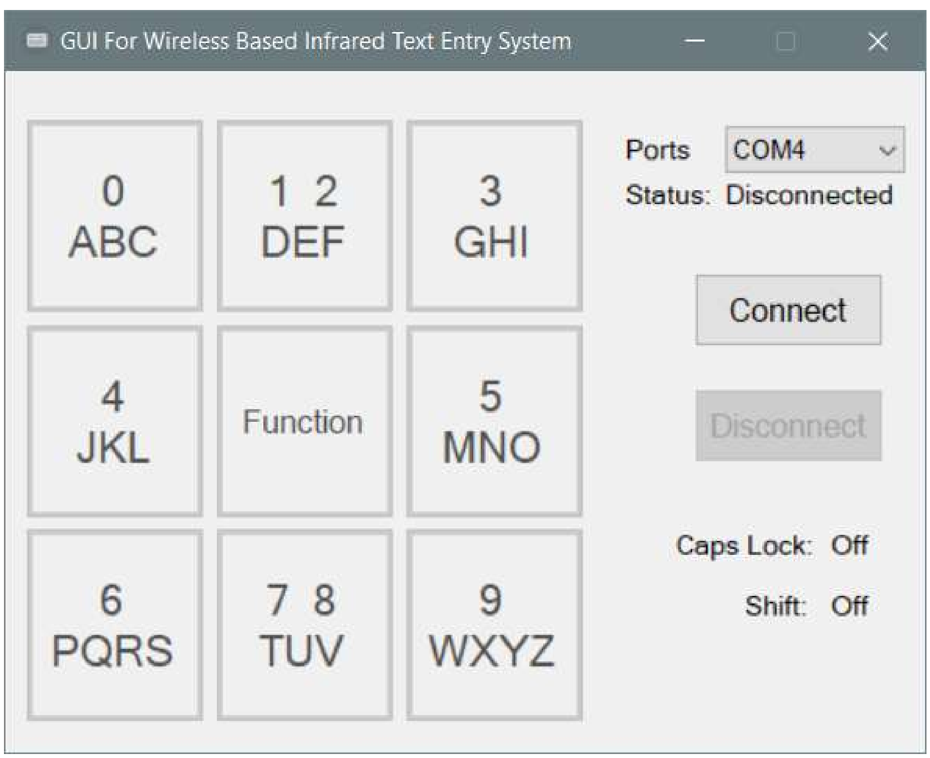


Figure 3-11: The interface of the GUI program

3.4.1 Flowchart for Software Implementation

The Figure 3-12 below shows the flowchart of the software implementation part which is the GUI program of the text entry system.

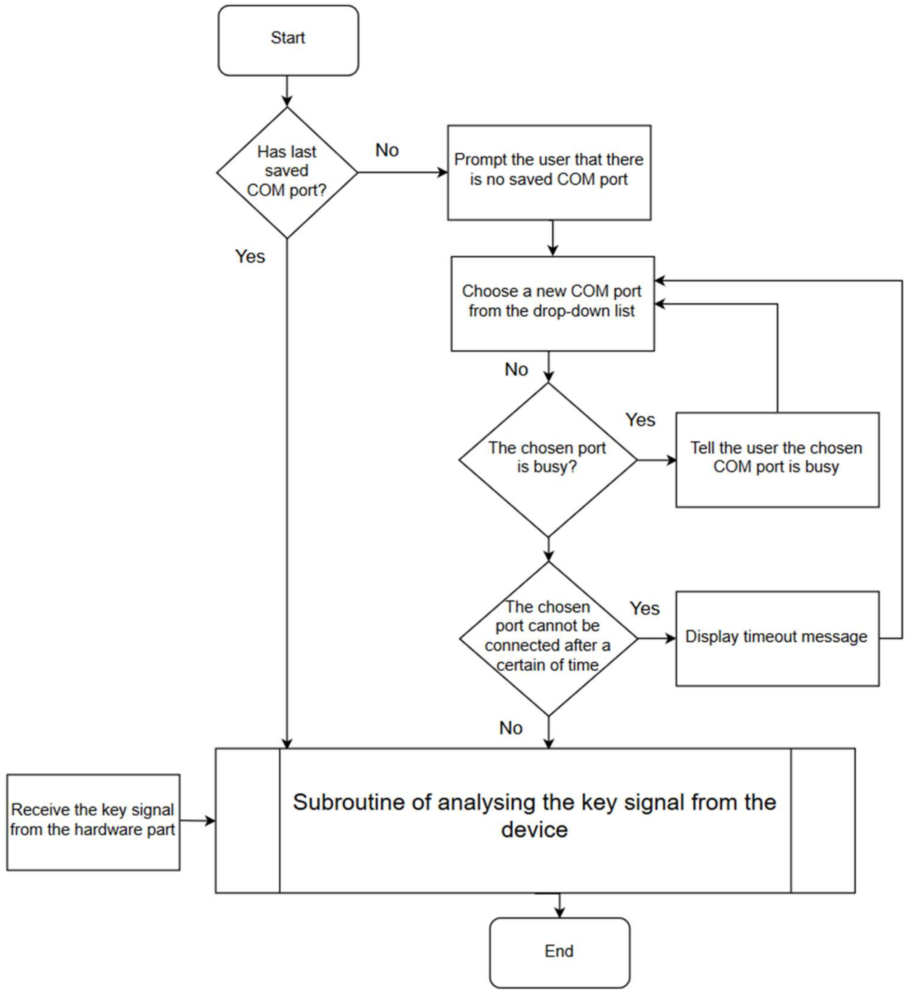


Figure 3-12: Flowchart for the software implementation part

When the user launches the program, it will check whether there is any saved port, if the program cannot find any saved port, it will ask the user to choose the COM port manually. Otherwise, it will connect with the device using the saved COM port entry automatically. This will save the user from the hassle of choosing the COM port every time when the user launches the program. After the connection is successful, the program will run the subroutine of analysing the key signal sent by the device.

The flowchart of the subroutine can be found in Appendix A because the flowchart is too complex to be put in this chapter. The subroutine will check the key signal from the device and analyse which key is pressed and eventually it will input the characters typed by the user or perform the functions as requested by the user.

3.5 System Flow Diagram (Hardware and Software Implementation)

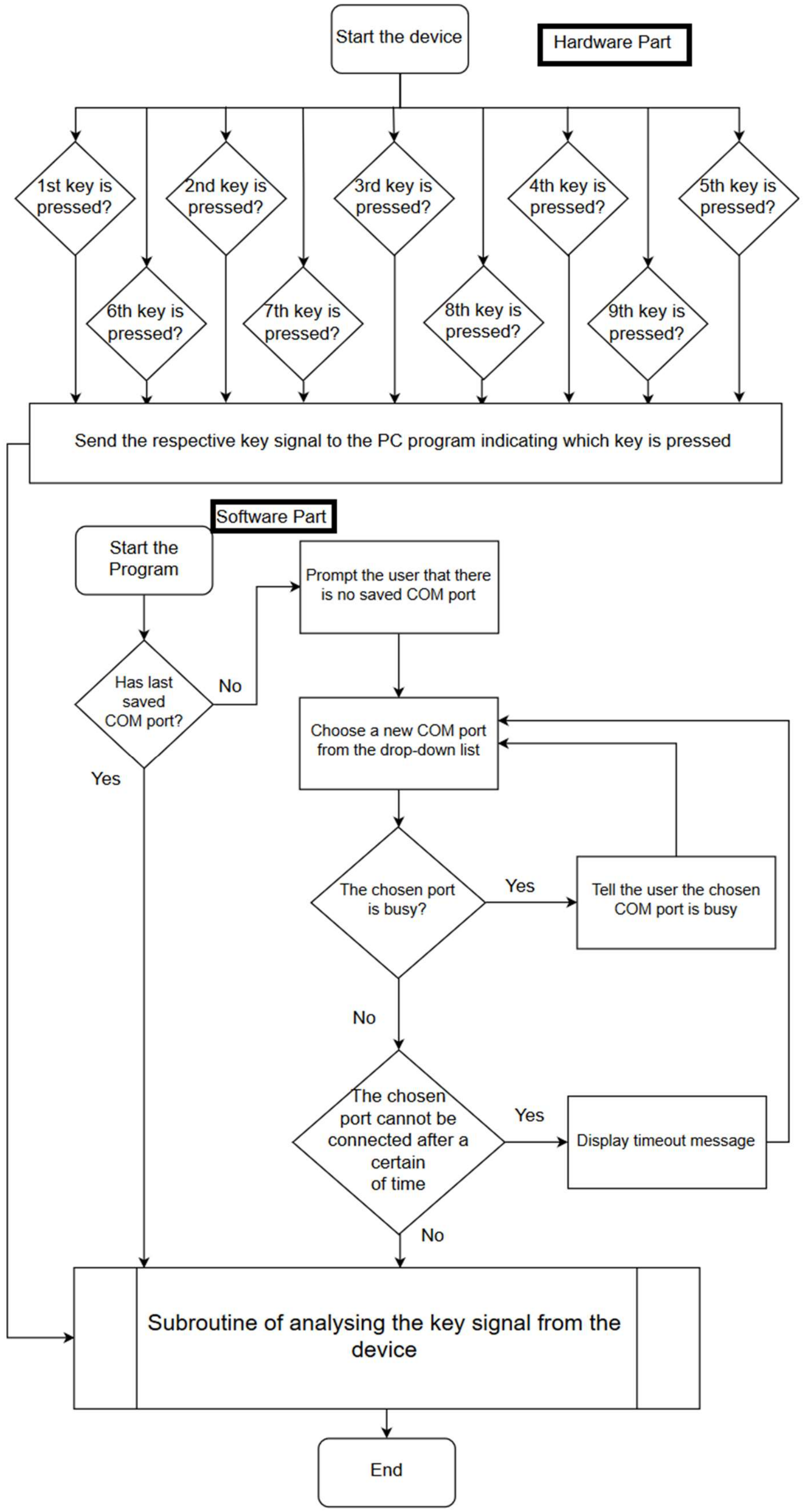


Figure 3-13: System flow diagram of both software and hardware implementation

The Figure 3-13 above shows that the system flow diagram of both software and hardware implementation. The user needs to start the device first, or else the program cannot communicate with the device. After the device is booted up, the user can now start the program. If the user launches the program for the first time, the user has to manually choose the COM port of the device and then click the "Connect" button. This step is important as choosing the wrong COM port will result in the GUI program could not communicate with the device. The program will remember the port chosen by the user so that when the user launches the program next time, the user would not need to choose the COM port again as the program will connect with the device automatically. Then, if the connection between the device and the program is established, the program will show the status "Connected". After the connection is established, the user can now start to use the text entry system. If the user wishes to stop the program, the user can click the "Disconnect" button or just close the program. The Figure 3-14 below shows the program has already established the connection with the device.

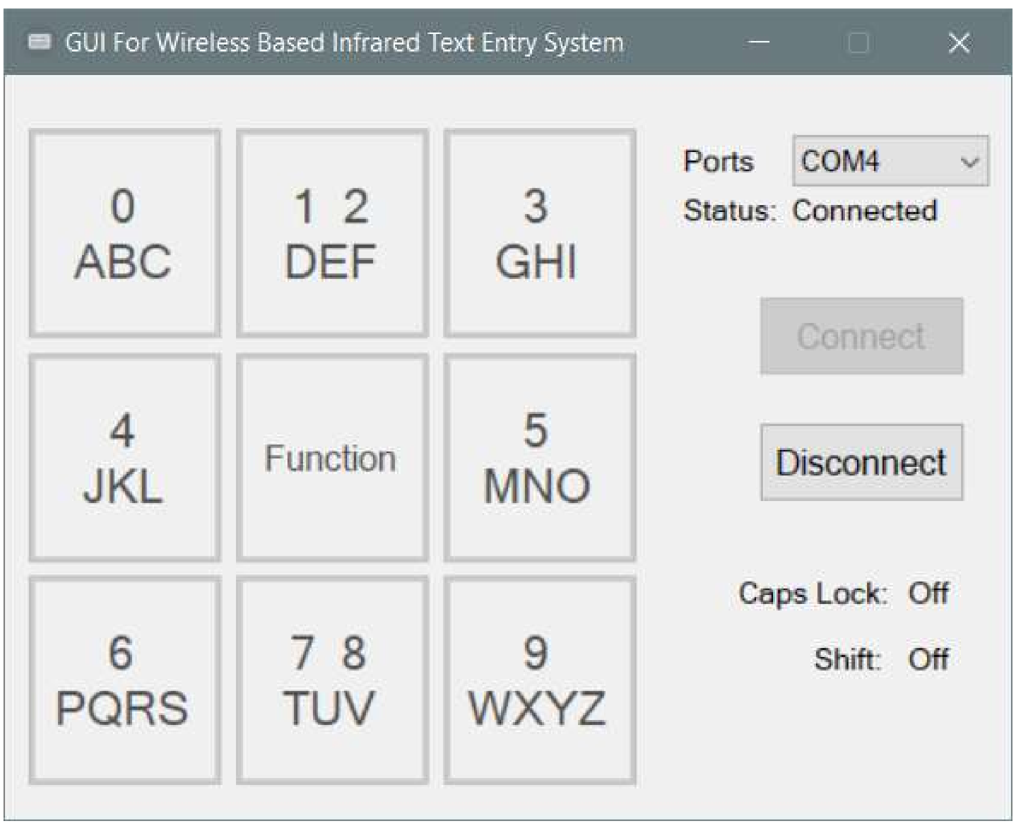


Figure 3-14: The connection between the device and the computer is established

3.6 The Usage of the GUI Program

3.6.1 Alphabets

To input the characters or perform functions, the user can hold on one of the intersection points from the 9 intersection points and drag across the intersection points. Once the user's desired letters or function appeared, the user can now release, then the desired letter will be typed, or the specific function will be performed.

For example, the user wanted to type letter 'f' onto the computer. Firstly, the user needs to choose the correct COM port and then connect first and then the user needs to hold at the second intersection point. After that, the interface of the program will switch to another page displaying letter 'd', 'e', 'f' and number '1' and '2'. The Figure 3-15 below shows the interface of the GUI program after the user holds the second intersection point.

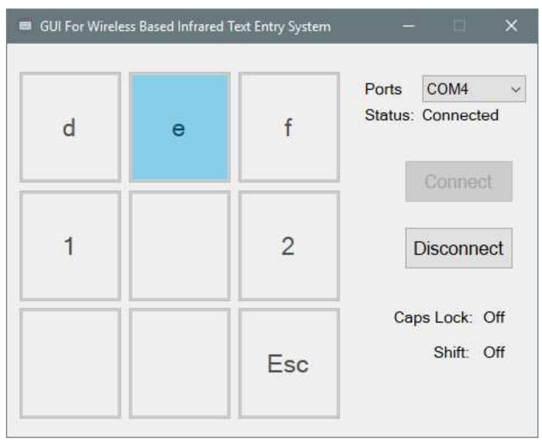


Figure 3-15: The program interface when the second intersection point is pressed

Then, the user's finger needs to be dragged to the third intersection point from the second intersection point on the device as shown in Figure 3-16 below.

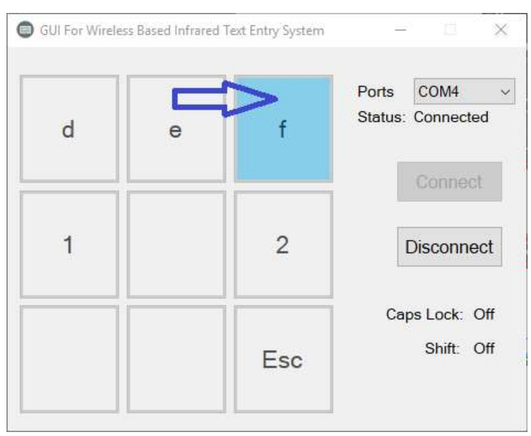


Figure 3-16: The program interface when the third intersection point is pressed

Next, the user's finger is released and letter 'f' will be printed and GUI program will return to the main menu as shown in Figure 3-17 below.

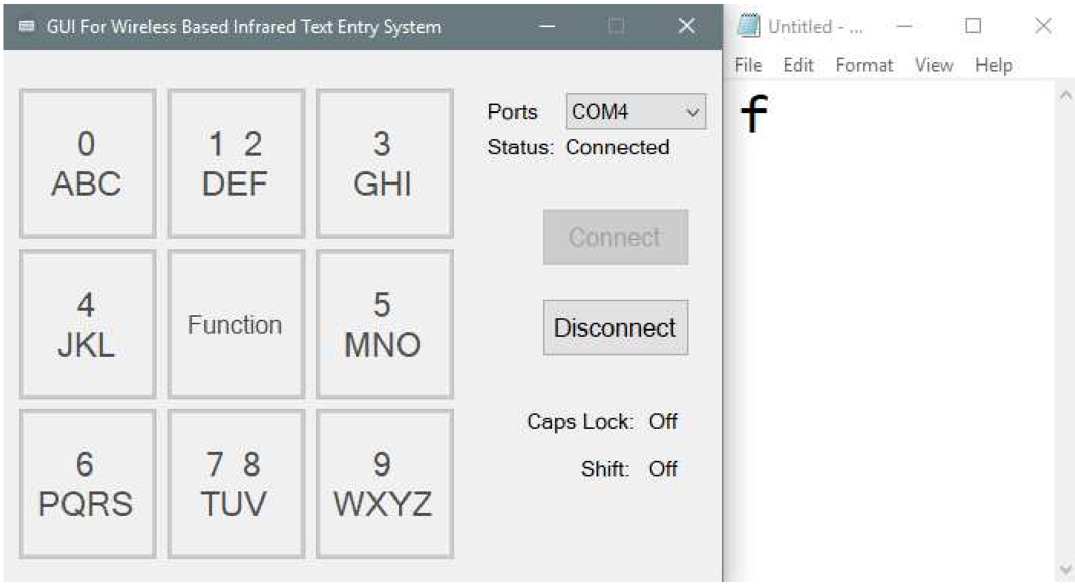


Figure 3-17: The letter 'f' has been entered

If the user wanted to hold on the third intersection point but mistakenly hold on the sixth intersection point, the user can drag to the ninth intersection point and release to return to the main menu as shown in the Figure 3-18 below.

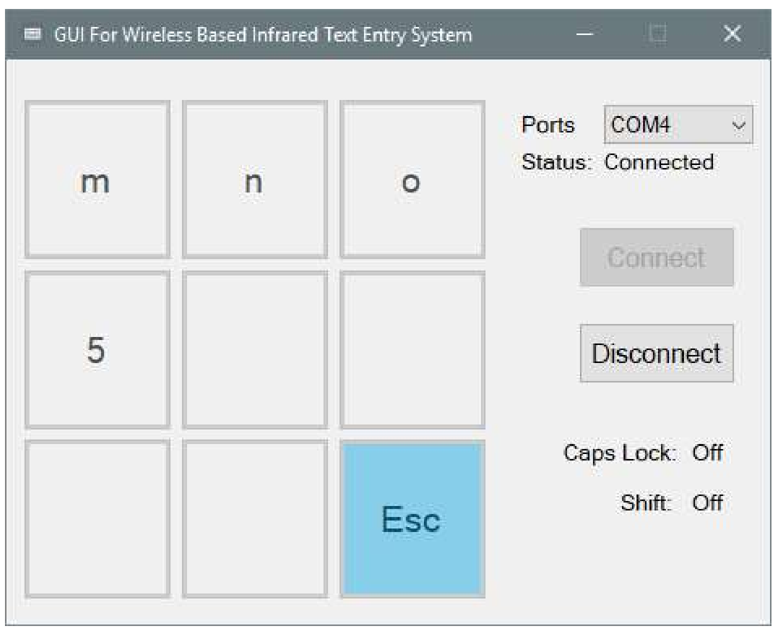


Figure 3-18: The program interface when the ninth intersection point is pressed

The user can enter other letters or numbers using the same process as above: "hold-drag-release".

The fifth intersection point is the function mode. The function mode is to make the typing experience of the user become more convenient and easy. The mode can let the user perform the following functions:

- Copy (Ctrl+C), Cut (Ctrl+X) and Paste (Ctrl+V)
- Mouse Simulation
- Tab, Enter (Carriage Return) and Select All (Ctrl+A)
- Caps Lock and Shift
- Punctuation
- Backspace
- Space
- Back to Main Menu

To enter the function mode, the user needs to hold on the fifth intersection point and release. To choose one of the function, the user's finger needs to be dragged to the specific intersection point that is related to the function. The interface of the function mode is shown in Figure 3-19 below.

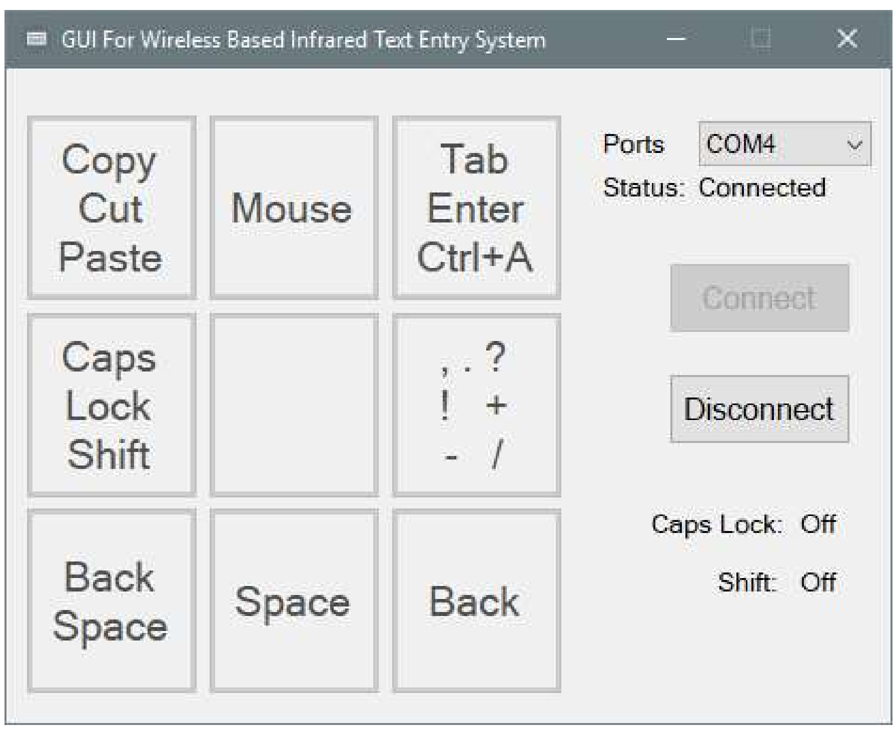


Figure 3-19: The program interface of the function mode

3.6.2 Copy, Cut and Paste Function

If the user wants to perform 'Copy', 'Cut' or 'Paste' function, the user needs to hold the fifth intersection point first, then drag to the first intersection point and release. Now, the program will enter the next page which let the user select 'Copy', 'Cut' or 'Paste' as shown in the Figure 3-20 below.

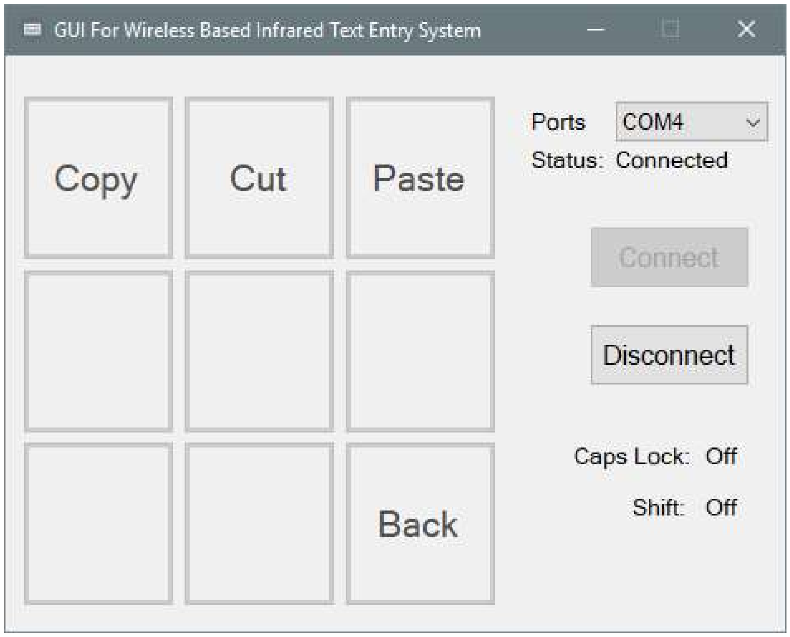


Figure 3-20: The program interface showing the 'Copy', 'Cut' and 'Paste' function

If the user wants to perform 'Copy' function, the user needs to press the first intersection point. The user needs to press the second intersection point if the user wants to perform 'Cut' function. Besides, the user needs to press the third intersection point if the user wants to perform 'Paste' function. If the user chooses to go back to the main menu, the user can press the ninth intersection point. In addition, the program will go back to the main menu once the user performs one of the functions.

These functions are useful for the user because the user can copy, cut or paste faster. Once the text is highlighted, the user only needs to enter the function mode, press the first intersection point and then choose either copy, cut and paste, rather than right-clicking on the highlighted text and choose copy, cut or paste.

3.6.3 Mouse Simulation Function

If the user wants to use the mouse function, the user needs to hold the fifth intersection point first, then drag to second intersection point and release. Now, the program will enter the next page which let the user move the mouse pointer on the computer and perform 'Left Click', 'Right Click' and 'Middle Click' as shown in the Figure 3-21 below.

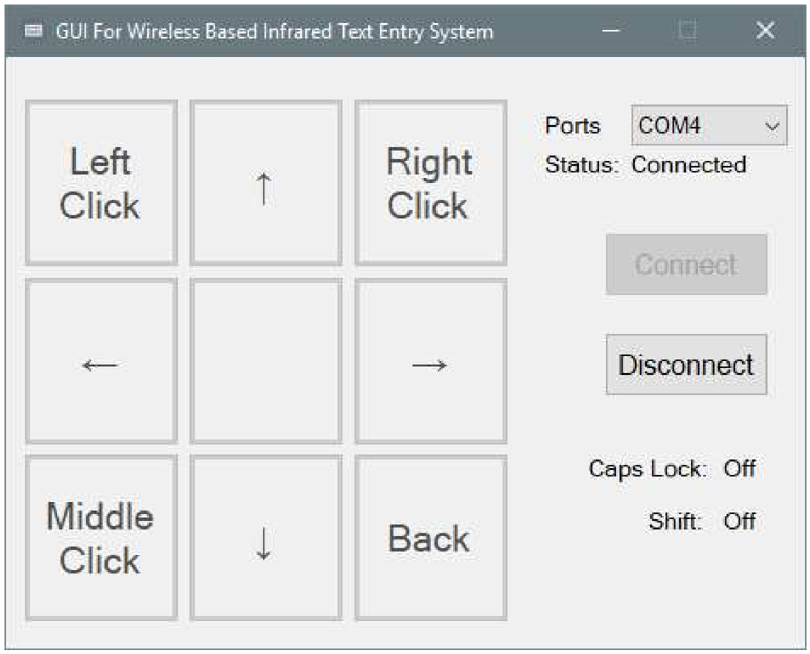


Figure 3-21: The program interface showing the mouse function

To simulate the 'Left Click', 'Right Click' and 'Middle Click', the user needs to press the first, third or seventh intersection point respectively as shown in Figure above. Besides, if the user wants to move the mouse pointer upward or downward, the user needs to hold on the second intersection point to move the mouse pointer upward or eighth intersection point to move the mouse pointer downward. In addition, if the user wants to move the mouse pointer left or right, the user needs to press and hold on the forth intersection point or sixth intersection point to move the mouse pointer left or right respectively.

To simulate the action of scrolling the middle wheel of the ordinary mouse, the user has to press on the seventh intersection point first which is the middle click button and then hold on second or eighth intersection point to simulate the action of scrolling upwards and downwards respectively. This function is useful when the user is browsing the web pages.

3.6.4 Tab, Enter and Select All Function

If the user wants to use the ‘Tab’, ‘Enter’, or ‘Select All’ function, the user needs to hold the fifth intersection point first, then drag to third intersection point and release. Now, the program will enter the next page which let the user choose whether to click ‘Tab’, ‘Enter’ or perform ‘Select All’ function as shown in the Figure 3-22 below.

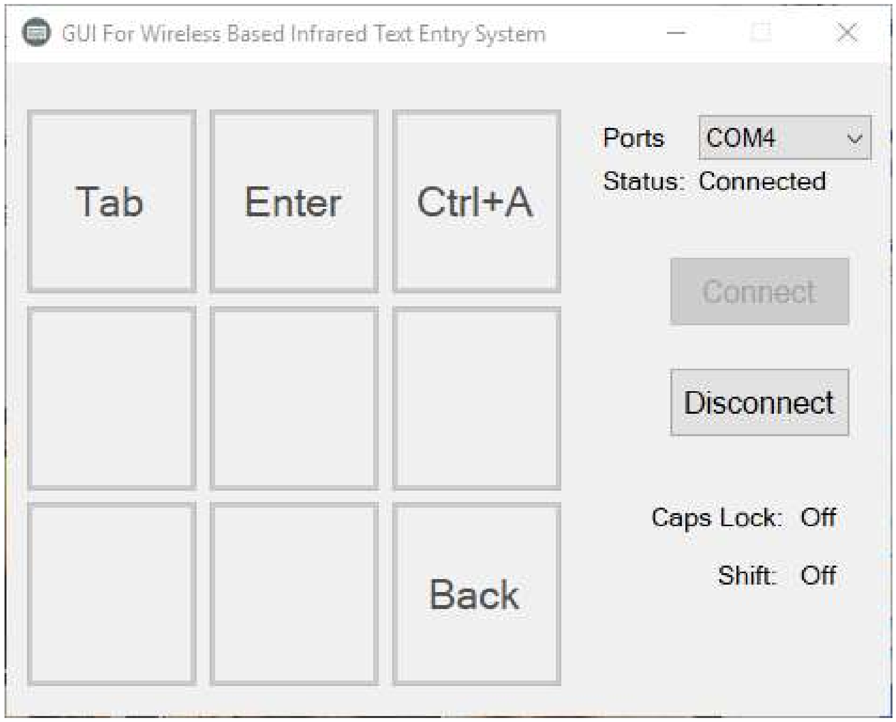


Figure 3-22: The program interface showing the tab, enter and select all function

When the user wants to input ‘Tab’, ‘Enter’, or ‘Select All’ (Ctrl+A), the user needs to press the first, second or third intersection point respectively. If the user wants to input ‘Tab’ followed by ‘Enter’, the user just need to press the first intersection point then followed by pressing the second intersection point. The program would not jump back to the main menu after inputting either ‘Tab’, ‘Enter’, or ‘Select All’ function. To return the program to the main menu, the user would need to press the ninth intersection point to go back to the main menu.

3.6.5 Caps Lock and Shift

When the user wants to use the 'Caps Lock' or 'Shift' function, the user needs to hold the fifth intersection point first, then drag to the forth intersection point and release. Now, the program will enter the next page which let the user choose whether to toggle 'Caps Lock' or 'Shift' function as shown in the Figure 3-23 below.

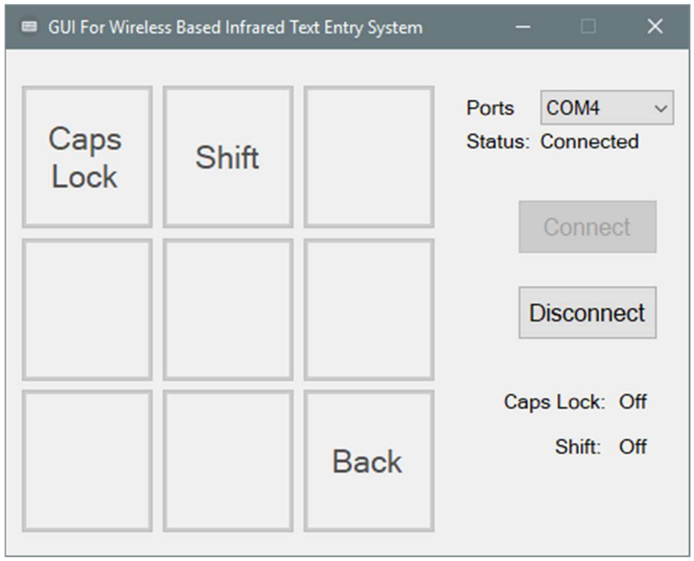


Figure 3-23: The program interface showing the caps lock and shift function

The 'Caps Lock' toggle is the same as the 'Caps Lock' button on the ordinary keyboard. The function switches all input characters to the upper case until the toggle has been switched off. The user can enable the 'Caps Lock' toggle by pressing the first intersection point. Then, there is an indicator showing the status of the 'Caps Lock' toggle as shown in Figure 3-24 below.

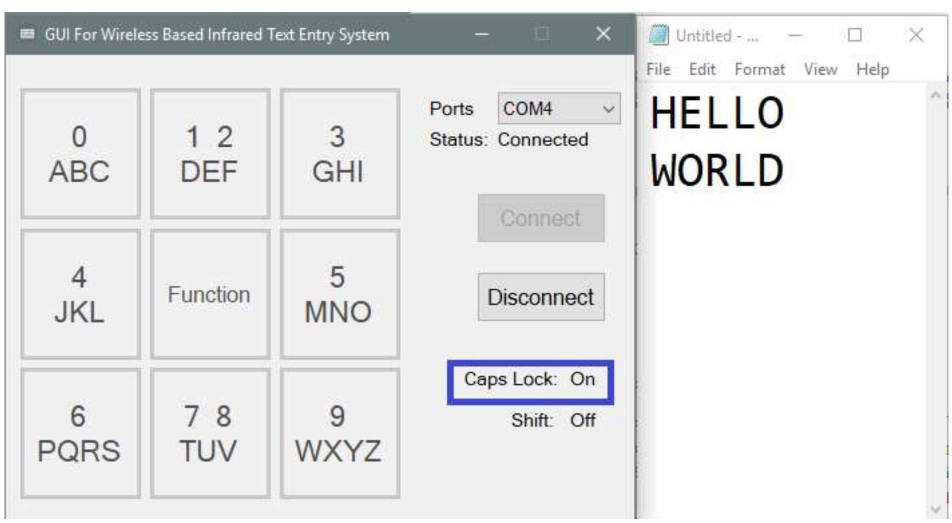


Figure 3-24: The program interface showing the caps lock indicator and the text

The ‘Shift’ toggle is also the same as the ‘Shift’ button on the ordinary keyboard. The function switches the first input characters to upper case and then the rest of the input characters will be in lower case if the ‘Caps Lock’ toggle is not enabled. The user can enable the ‘Shift’ toggle by pressing the second intersection point. Then, there is an indicator showing the status of the ‘Shift’ toggle too as shown in Figure 3-25 below.

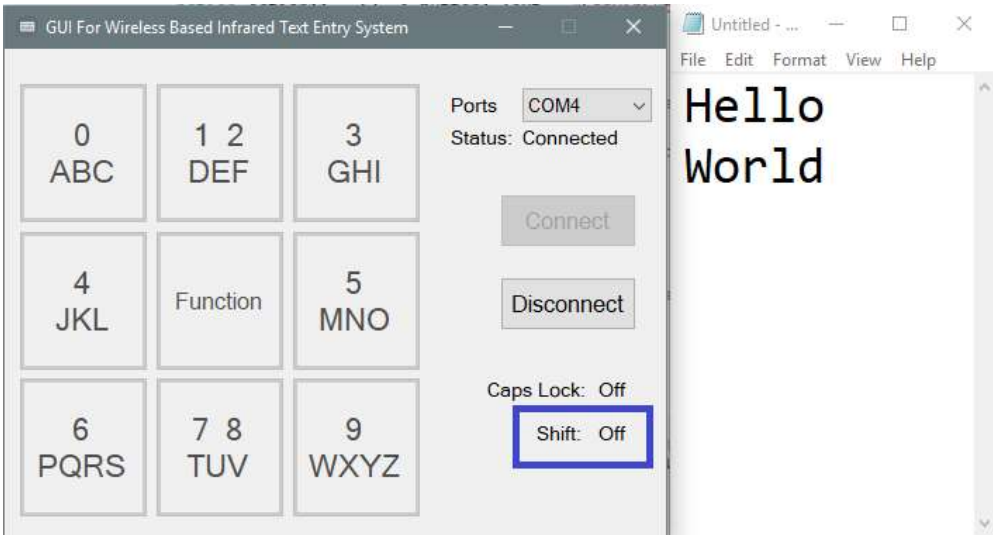


Figure 3-25: The program interface showing the shift indicator and the text

The Figure below shows the “Hello World” text entered using the device with the ‘Caps Lock’ and ‘Shift’ toggles are enabled. Note that the indicator ‘Shift’ is off as shown in Figure 3-26 below as the ‘Shift’ function only enabled for entering one character only and will be toggled off when the user has entered one character.

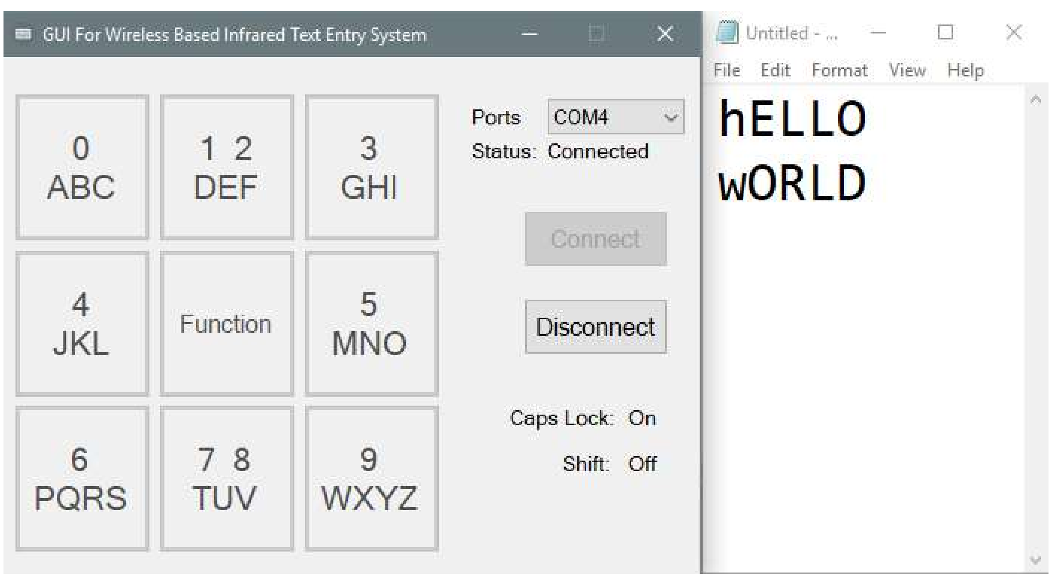


Figure 3-26: The program interface showing the text entered by the device

3.6.6 Punctuation

When the user wants to enter punctuations, the user needs to hold the fifth intersection point first, then drag to sixth intersection point and release. Now, the program will enter the next page which let the user choose whether which punctuation to insert as shown in the Figure 3-27 below.

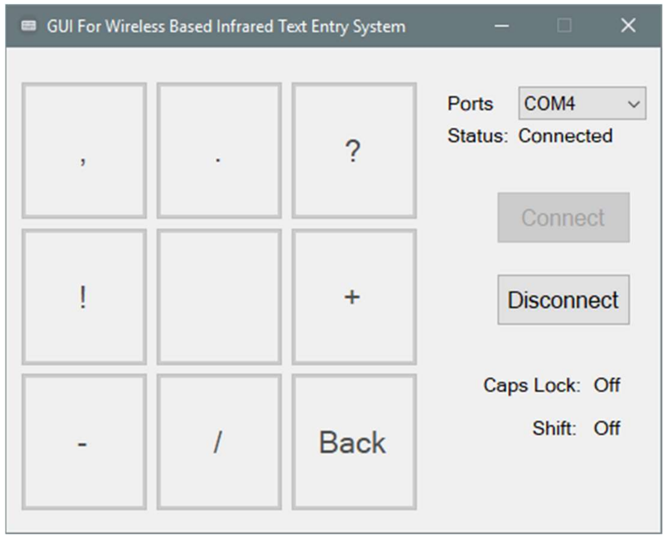


Figure 3-27: The punctuations available on this device

However, there are only some of the common use punctuations are available for use such as comma (,), period (.), question mark (?), exclamation mark (!), plus sign (+), minus sign (-) and slash (/). The program would not jump back to the main menu after entering one of the punctuations. To return the program to the main menu, the user would need to press the ninth intersection point to go back to the main menu. The Figure 3-28 below shows the punctuations are inserted into the text editor "Notepad".

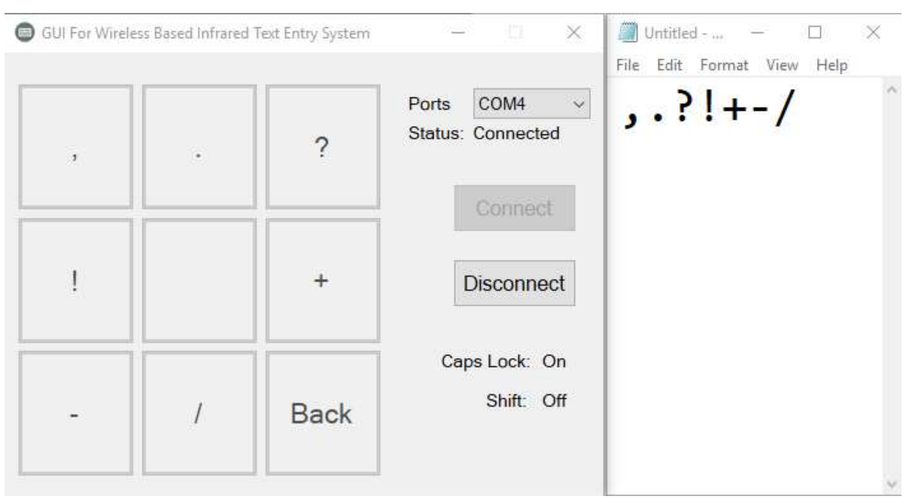


Figure 3-28: The punctuations are inserted into the text editor "Notepad"

3.6.7 Backspace

To remove a character entered, the user will first have to hold the fifth intersection point first, then drag to seventh intersection point and release. Then, one character will be removed. The user can press the seventh intersection point repeatedly as long as the program is still in the function mode to remove one or more characters. To exit the function mode, the user just needs to press the ninth intersection point.

3.6.8 Space

To enter a space character, the user will first have to hold the fifth intersection point first, then drag to eighth intersection point and release. Then, one space character will be added. The user can press the eighth intersection point repeatedly as long as the program is still in the function mode to add one or more space characters. To exit the function mode, the user just needs to press the ninth intersection point.

CHAPTER 4: Methodology and Tools

4.1 Design Methodology

The design methodology indicates the method for the development of a system. The objective of the design methodology is to find a perfect solution for every design situations. A good design methodology is needed to develop a good and high utility system. Therefore, the Software Development Life Cycle (SDLC) is studied and researched. The Software Development Life Cycle (SDLC) is basically a framework for the developer that define tasks need to be performed at each step during the development process (tutorialspoint, 2017). There are several SDLC models defined such as the Waterfall Model, Iterative Model, Spiral Model and the Prototype Model. The Figure 4-1 below shows the stages of the SDLC.

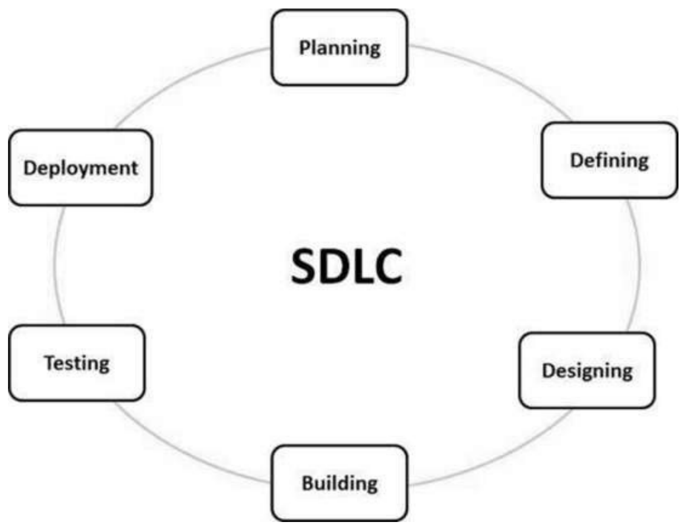


Figure 4-1: The stages of the SDLC

In this research, the Prototype Model is selected to build a good and high utility system. The Prototype Model indicates that building a prototype which roughly shows the functionalities of the product and it also can let the developer figure out the customer requirements at the very early stage of the development. Besides, it helps to acquire the comments or feedback from the customers and then helps the developer to know the expectations of the customers of the final product. In addition, Prototype Model is an interesting approach that helps in developing complex and huge systems where there is no existing system or manual process to assist the developers in determining the requirements (Verma, 2015). The Figure 4-2 below shows the general steps of the Prototype Model.

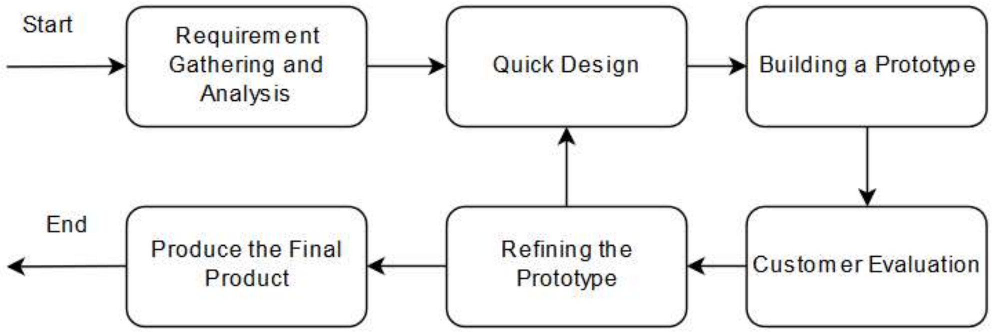


Figure 4-2: The general steps of the Prototype Model

There are 6 general steps of the Prototype Model as shown in Figure 4-2 above which include:

1. **Requirement Gathering and Analysis –** The developer will collect the requirements from the customers first. Then, the requirements will be analysed, and the product requirement documents will be prepared.
2. **Quick Design** – The developer will start to design the prototype based on the requirements given by the customers. However, the preliminary design is not a detailed design, but it includes all the main aspects of the system which it will give an idea to the customers.
3. **Building a Prototype** – A prototype is developed from the preliminary design. The prototype will give an idea to the customer what the final product will probably look like.
4. **Customer Evaluation** – Then, the prototype will be sent to the customer and let them evaluate the prototype. After evaluating, the customer comments and feedbacks will be recorded for the developer to improve the product.
5. **Refining the Prototype** – During this step, the comments and feedback from the customer will be studied and discussed. The developer will try to refine the product based on the comments and feedback and then go back to the Quick Design stage. If the customer satisfies with the prototype, then the developer only can proceed to the next step.
6. **Produce the Final Product** – The actual product will be developed based on the prototype.

4.2 Hardware Components

4.2.1 IR LED Transmitter and Receiver

An infrared light emitting diode is a solid-state lighting device that emits infra-red light. The transmitter will transmit the infrared light when the voltage is supplied to the transmitter. The receivers are connected to the ADC inputs of the CT-UNO and the changes of the ADC values for each receiver will be used to determine which intersection point is pressed by the user. The IR LED receiver and transmitter are shown in Figure 4-3.

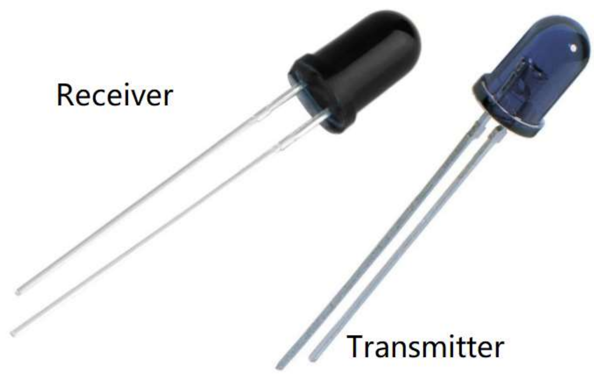


Figure 4-3: The IR LED receiver and transmitter

The IR LED is very suitable for this project since it is low cost and energy consumption is low as well. Finally, the IR LED is chosen as a component to build the input device.

4.2.2 10K ohm Resistors

The resistors used in this project are the 10K ohm resistors as shown in Figure 4-4 below.

Figure 4-4: The 10K ohm resistors

4.2.3 Arduino Header Pin

The header pins are used in this project and the pins are soldered into the infrared text entry board to let the board can stack on top of the Arduino or other Arduino Shield. The Figure 4-5 below shows the header pins used in this project.

Figure 4-5: The header pins

4.2.4 CT-UNO Development Board

Arduino is an open-source platform that used to build electronics projects. Arduino has its own IDE to write, edit or upload codes to the microcontroller board. Arduino platform is famous because its IDE uses the C++ programming language rather than Assembly Language compared to the older development board. The Cytron version of Arduino Uno (CT-UNO) is chosen for this project. Cytron company is based in Penang, Malaysia. They sell a series of development boards, modules. In addition, the company developed their version of Arduino UNO called CT-UNO. The CT-UNO is a more economical selection compared to the usual Arduino Uno. Besides, the CT-UNO included mostly all the features that are available in the Arduino UNO. The CT-UNO board is shown in Figure 4-6 below.

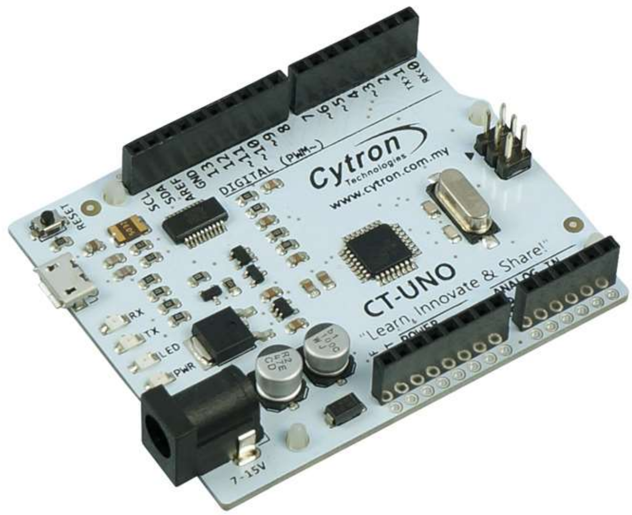


Figure 4-6: CT-UNO

CT-UNO is based on the SMD ATmega328P microcontroller and it offers some features like 14 Digital I/O pins with 6 PWM pins, 6 Analog inputs, UART, SPI, I2C, and external interrupts and of course, a reset button. The technical specifications of the board are shown in Table 4-1 below.

Microcontroller	SMD ATmega328P
Input Voltage	DC7-15V
5V Voltage Regulator (Maximum)	1A
3.3V Voltage Regulator (Maximum)	500mA
Digital I/O Pins	14 (6PWM outputs)
Analog Input Pins	6
Flash Memory	32 KB
Clock Speed	16 MHz

Table 4-1: Technical Specifications of CT-UNO

4.2.5 Cytron Bluetooth Module and the Cytron XBee Shield

There are 2 options to make the system works wirelessly by transferring the data to the computer wirelessly using Bluetooth or Wi-Fi. After some consideration, by using Bluetooth technology is more than enough in this system. Hence, a Cytron Bluetooth Module is used to allow the CT-UNO to communicate wirelessly with the PC and laptop. The Cytron Bluetooth Module is shown in Figure 4-7 below.

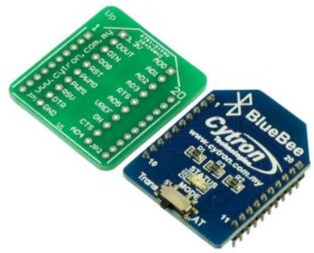


Figure 4-7: Cytron Bluetooth Module

This module is based on the CSR BC04 chipset with Bluetooth version 2.0 + EDR. It operates in an operating frequency within 2.4 to 2.8GHz. It can transmit within a range of 20m to 30m in free space with no obstacle. The maximum transfer rate for asynchronous transmission is 2.1Mbps download, 160kbps upload while the module can achieve maximum 1Mbps download and upload speed. The baud rate can be adjusted from 1200bps to 1382400bps.

To fit in the Bluetooth module into the CT-UNO, an Arduino shield is required. The Cytron XBee Shield is chosen for this system. The purpose of this shield is to mount the Bluetooth module on top of it and mount to the CT UNO. The shield is shown in Figure 4-8 below.

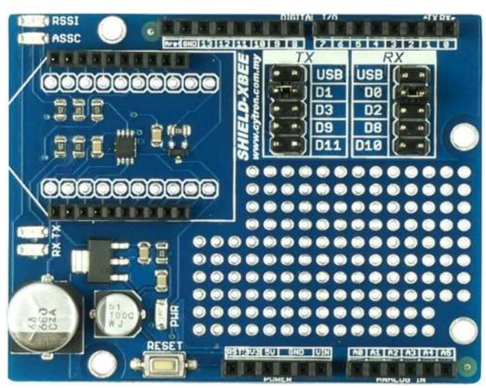


Figure 4-8: Cytron XBee Shield

4.2.6 The Infrared Text Entry Board (PCB)

The Infrared Text Entry Board is developed using the PCB design software, EAGLE, which will be discussed later. The schematic and the board diagram of the Infrared Text Entry Board are shown in Figure 4-9 and Figure 4-10 respectively.

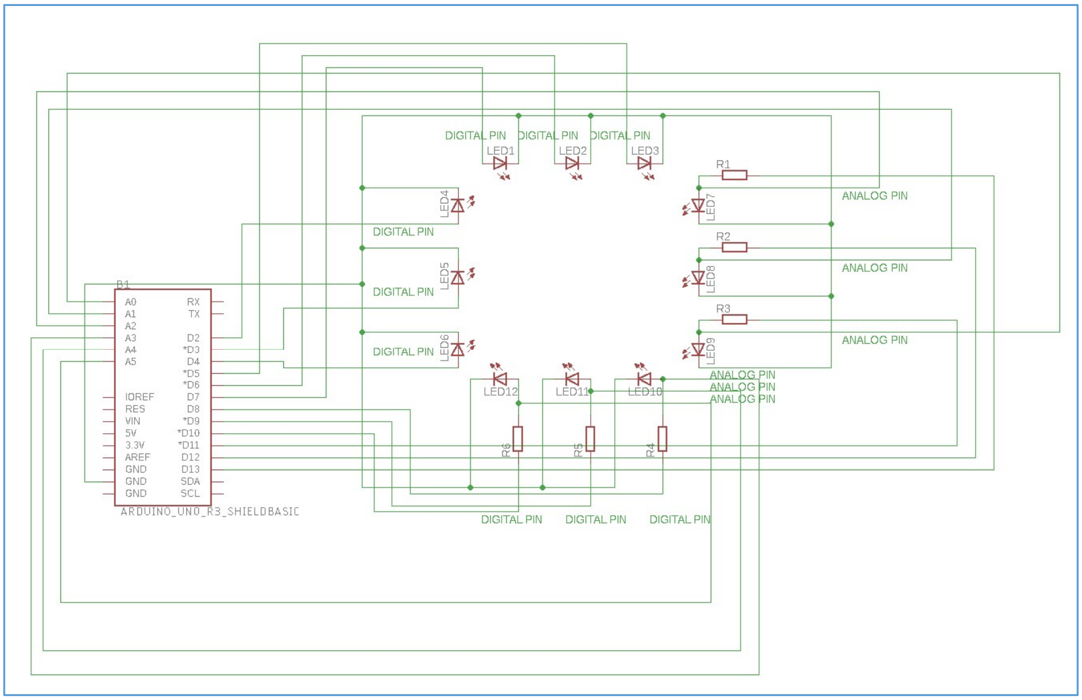


Figure 4-9: The schematic diagram of the infrared text entry board

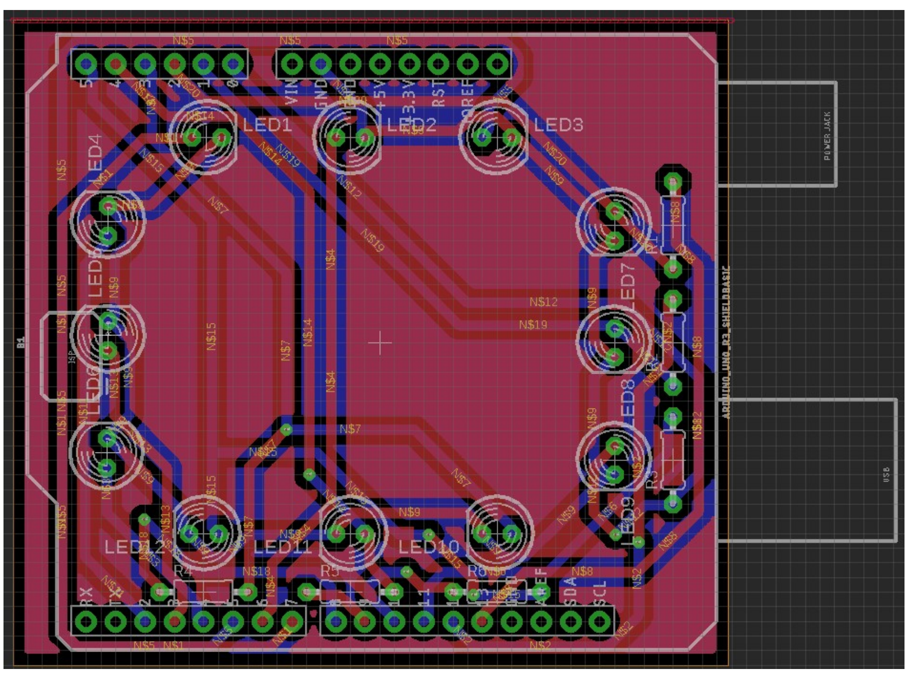


Figure 4-10: The board diagram of the infrared text entry board

4.3 Hardware Development Tools

4.3.1 Arduino IDE

The open-source Arduino IDE makes developers life easier to write code and upload it to the board. It runs on Windows OS, Mac OS X, and Linux OS and it can be used with any Arduino board. The environment is written in Java and other open-source software. To run this Arduino IDE, the computer must install the Java Runtime Environment first. Arduino IDE will be used to upload codes to the CT-UNO. The IDE can verify the correctness of the codes first then only it can compile and upload to the CT-UNO board. The IDE also includes the serial monitor without having the need of downloading another software such as PuTTY. Figure 4-11 below shows the interface of the Arduino IDE software.

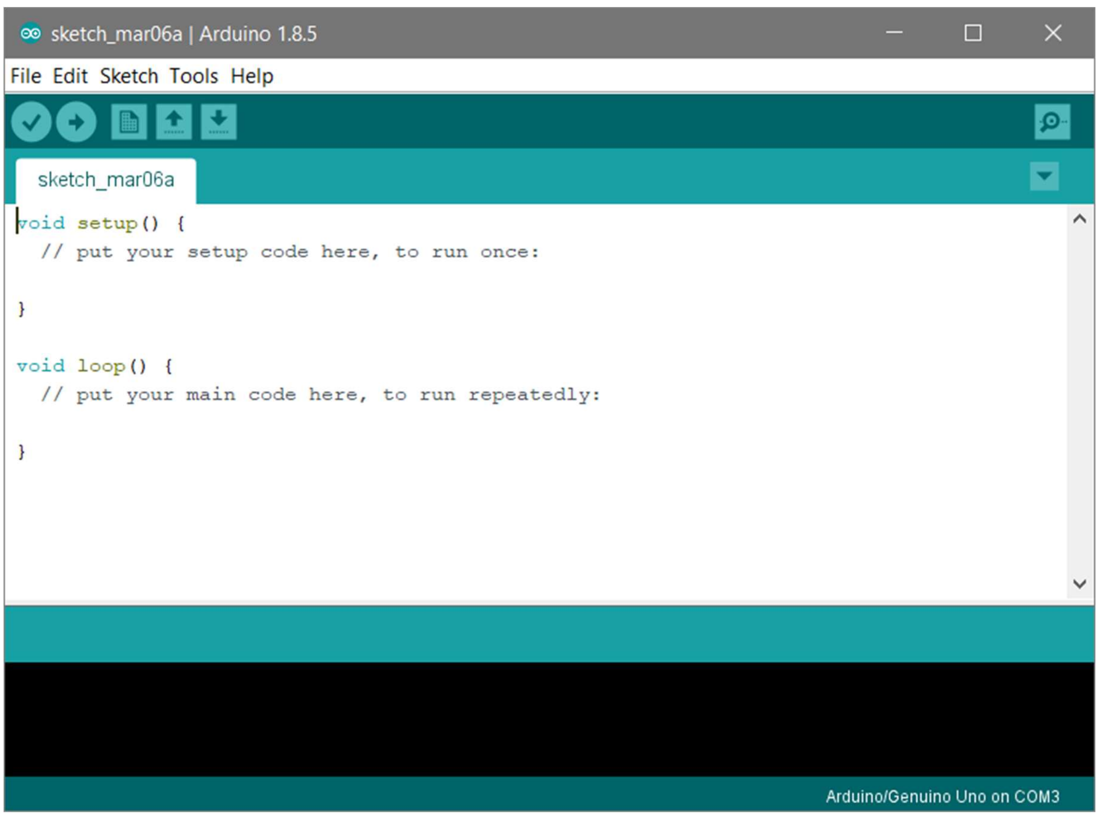


Figure 4-11: The interface of the Arduino IDE

4.3.2 EAGLE Software

The schematic and the board diagram of the Infrared Text Entry Board is designed the software EAGLE. The software EAGLE is a PCB design software, it can let the user design the schematic diagram of the PCB first then convert to board diagram. Then, the software can export the board diagram to Gerber file format (.gbr) to be sent for fabrication. The Figure 4-12 shows that the interface of the EAGLE software.

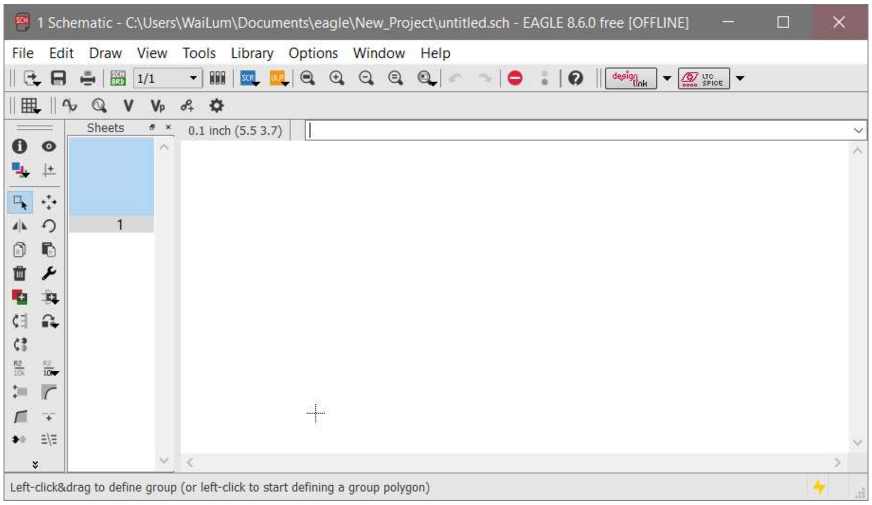


Figure 4-12: The interface of the EAGLE software

To add the components such as resistors and IR LED into the schematic diagram, the user can search for the required components on the Library Manager. Then, the components can be added to the schematic diagram. Figure 4-13 shows that the interface of the Library Manager of the EAGLE software.

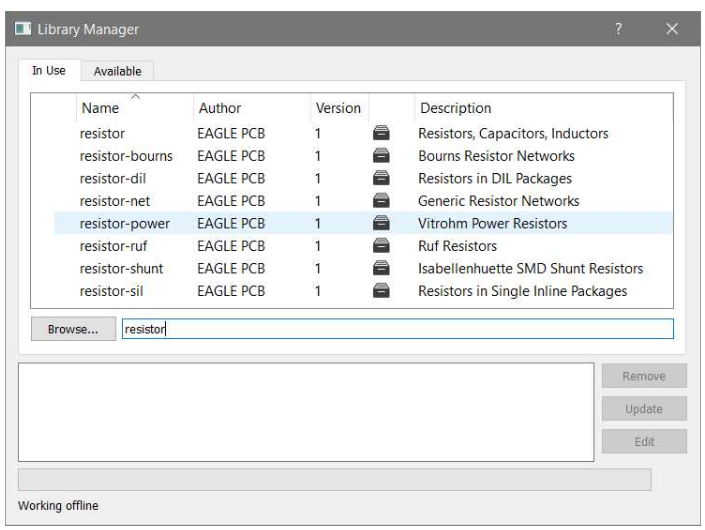


Figure 4-13: The interface of the Library Manager

4.4 Software Development Tools

4.4.1 Visual Studio 2013

In this project, the software GUI program is built using the C Sharp (C#) programming language to get the data from the device, and then output the text. C# programming language is a general-purpose programming language that designed to develop apps on the Microsoft platform and it requires the .NET framework on Windows to work. Besides, C# is commonly used to develop Windows desktop applications and games too. Hence, C# programming language is used to develop the GUI program in this project.

Apart from choosing the programming language, a powerful IDE needs the ability to build C# program, editing the C# source code and a debugger. A debugger can help to test and debug the program. In this project, the Microsoft Visual Studio 2013 is chosen as the IDE in building the GUI program. Microsoft Visual Studio is an IDE developed by Microsoft and the IDE is written in C++ and C# and it supports many different programming languages other than C#. The logo of the Microsoft Visual Studio IDE is shown as in Figure 4-14 below and the interface of the Visual Studio 2013 with the project opening is shown in Figure 4-15 below.

Visual Studio

Figure 4-14: The logo of the Microsoft Visual Studio

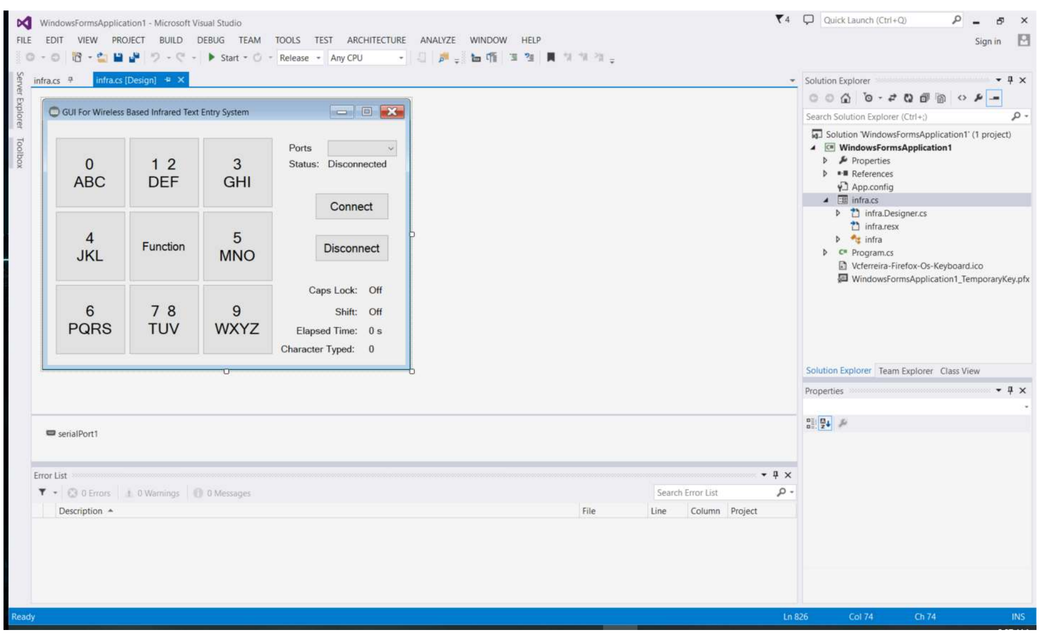


Figure 4-15: The interface of the Visual Studio 2013 with the project opening

CHAPTER 5: Implementation and Testing

After implementing the hardware device and the software GUI, some tests are done to verify and test the functionalities of the text entry system. In addition, the typing speed of the text entry system will be determined through the usage of the text entry system. Besides verifying the functionalities of the system, the power consumption of the text entry system will be calculated as well to see whether the system can operate for a long hour or not.

5.1 Final Product

The final hardware implementation part of the text entry system is shown in Figure 5-1 and Figure 5-2 below.

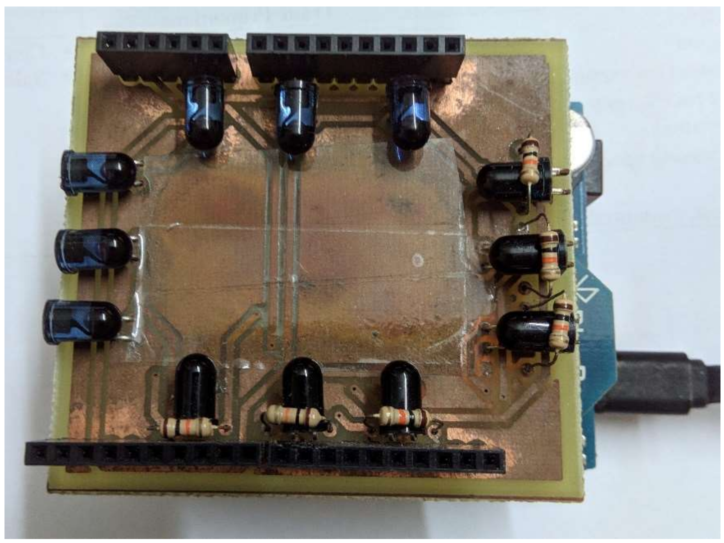


Figure 5-1: The top view of the hardware part of the text entry system

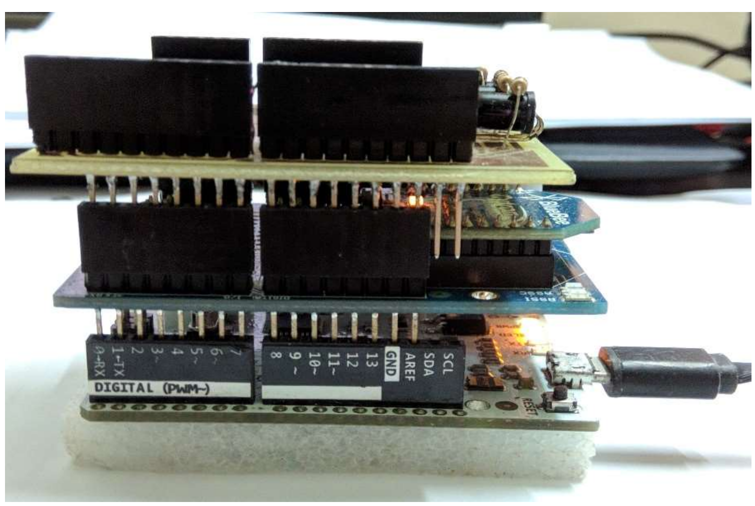


Figure 5-2: The side view of the hardware part of the text entry system

The final software GUI program interface implementation part of the text entry system is shown in Figure 5-3 below.

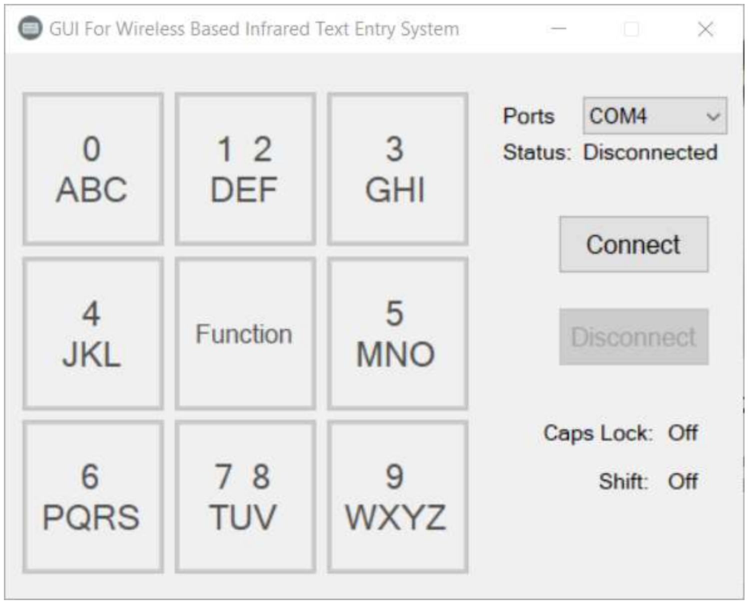


Figure 5-3: The interface of the Software GUI program of the text entry system

5.2 Functions

Other than inputting alphabets in lowercase and uppercase using this text entry system, other functions such as ‘Space’, ‘Tab’, ‘Backspace’ and ‘Enter’ can be done using this system as well. Besides, the functions such as ‘Copy’, ‘Cut’, ‘Paste’ and ‘Select All’ are included too. In addition, this text entry system can be used as a mouse too. In order to ensure the system is working, the test plan is needed to evaluate the system. The Table 5-1 below shows all test cases done using the text entry system.

No.	Test Case	Details	Example
1	Lowercase alphabets and ‘Space’	To input text with lowercase alphabets and ‘Space’	the quick brown fox jump over the lazy dog
2	‘Copy’, ‘Cut’, ‘Paste’, and ‘Select All’	- To test the ‘Copy’, ‘Cut’. ‘Paste’ and ‘Select All’ functions	the quick brown fox jump over the lazy dog
3	‘Backspace’	To remove a word from the text	the quick brown fox jump over the lazy dog
4	Uppercase alphabets and ‘Tab’	To input text with uppercase alphabets and ‘Tab’	THE QUICK BROWN FOX JUMP OVER THE LAZY DOG
5	Numbers, Punctuations, ‘Shift’ Key and ‘Enter’ Key	- To input the numbers and punctuations - To test the ‘Shift’ key and the ‘Enter’ key	Hi! I am Adam, my number is 0123456789. 10+5-3 Coffee/ Tea?
6	Mouse Function	- To move the mouse pointer using this function - To issue left click, right click and middle click like the ordinary mouse	- To control the mouse pointer and open the program - To test the ‘Left Click’, ‘Right Click’ and ‘Middle Click’ using the program

Table 5-1: The test plan for the text entry system

5.2.1 Test Case 1: Lowercase Alphabets and 'Space' Key

To complete the Test Case 1, the following text, "the quick brown fox jump over the lazy dog" is used as an example to input to the computer using the text entry system as the text contains all the 26 alphabets and some spaces. The details and the results of the Test Case 1 are stored in Table 5-2 and Figure 5-4 below respectively.

Test Case	Example	Expected Result	Actual Result
Lowercase alphabets and 'Space'	the quick brown fox jump over the lazy dog	All the letters of the text and the 'Space' key are inserted successfully.	Same as the expected result

Table 5-2: The details of the Test Case 1

Figure 5-4: The results of the Test Case 1

5.2.2 Test Case 2: 'Copy', 'Cut', 'Paste', 'Select All'

To complete the Test Case 2, the following text, "the quick brown fox jump over the lazy dog" from the previous test case is used. The text on the "Notepad" program is highlighted using the 'Select All' function. Then, the text is copied from the "Notepad" program and paste into the "Microsoft Word" program. Next, the sentence is being cut from the "Microsoft Word" program and paste it to another program called "Notepad++" using the 'Cut' function and the 'Paste' function. The details of the Test Case 2 are stored in Table 5-3 below.

Test Case	Example	Expected Result	Actual Result
'Copy', 'Cut', 'Paste', 'Select All'	the quick brown fox jump over the lazy dog	The 'Copy', 'Cut'. 'Paste' and 'Select All' are tested successfully.	Same as the expected result

Table 5-3: The details of the Test Case 2

The Figure 5-5 below shows the text on the "Notepad" program is highlighted and copied.

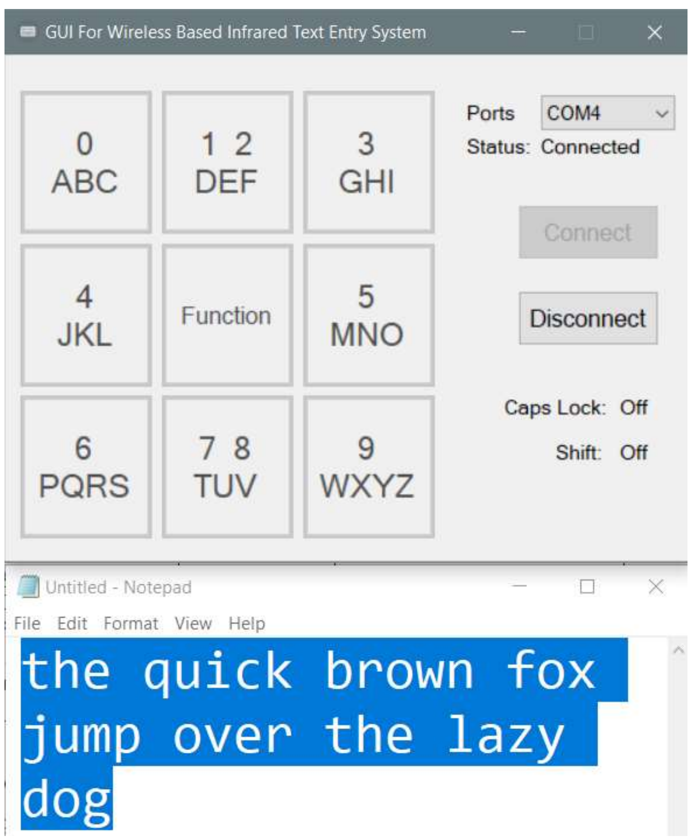


Figure 5-5: The text on the "Notepad" program is highlighted and copied

Then, the Figure 5-6 below shows the copied text is pasted into the "Microsoft Word" program.

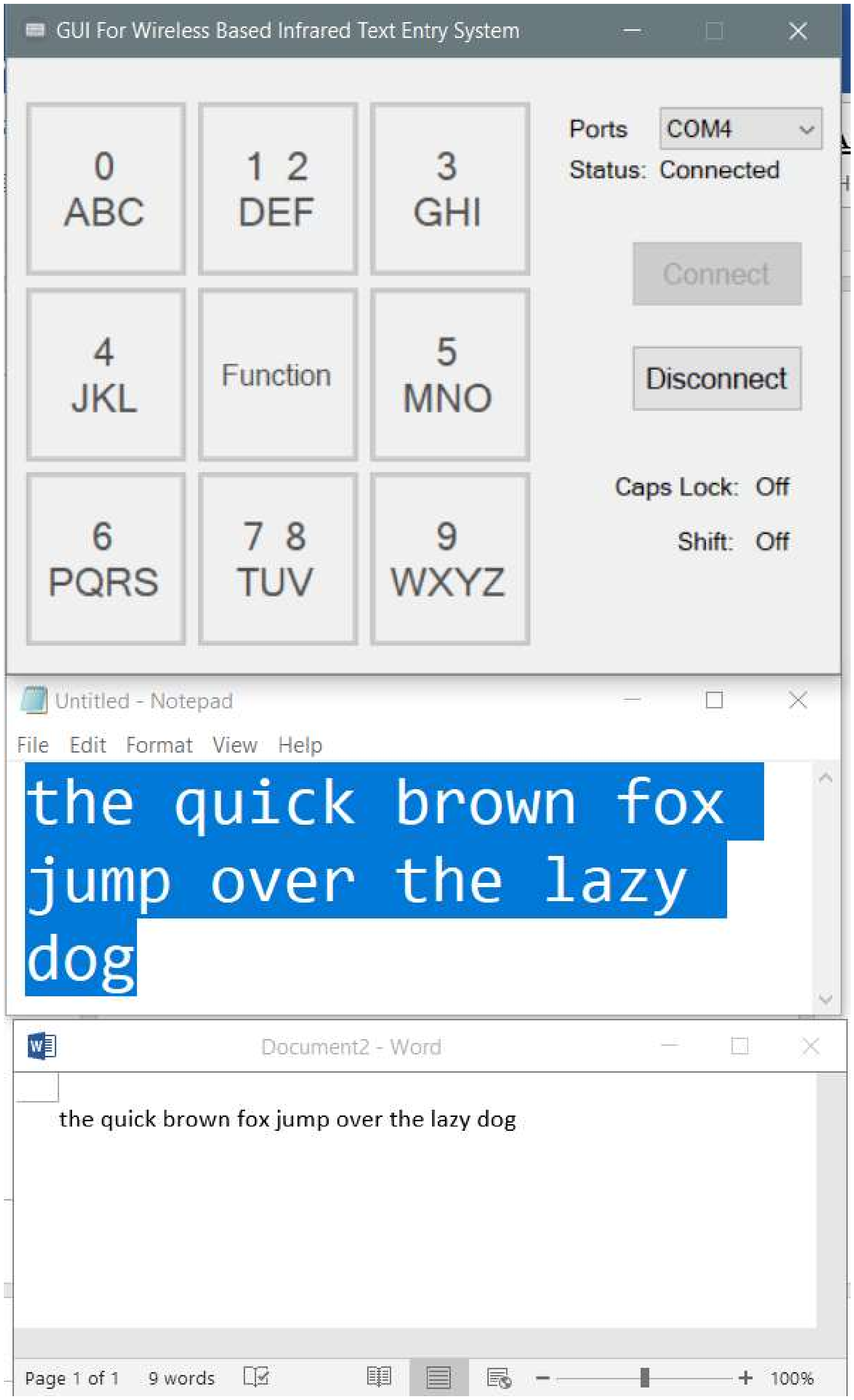


Figure 5-6: The copied text is pasted into the "Microsoft Word" program

Next, the Figure 5-7 below shows the text in the “Microsoft Word” is being cut and paste into another program called “Notepad++”.

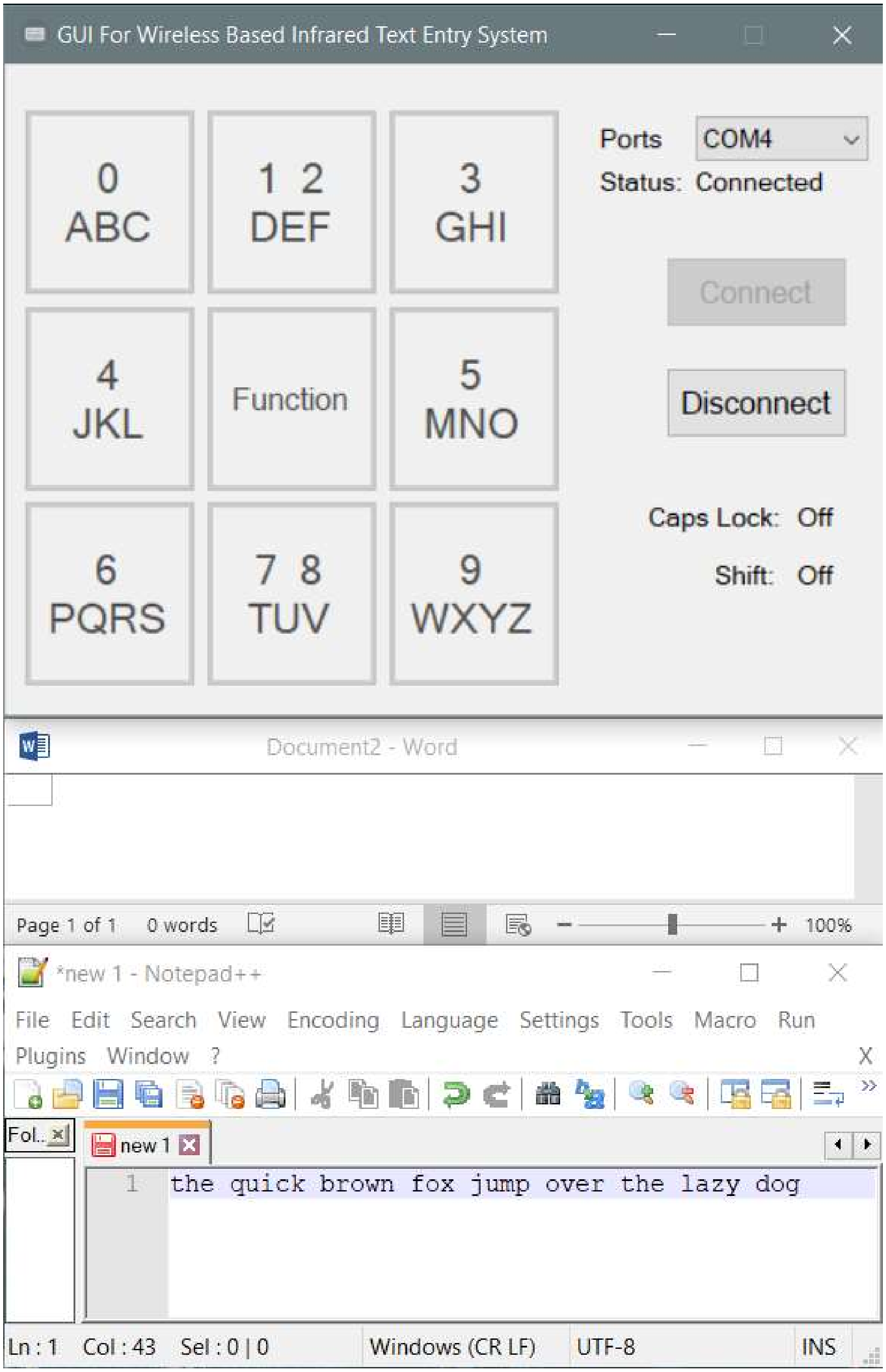


Figure 5-7: The text is being cut and paste into another program called “Notepad++”

5.2.3 Test Case 3: 'Backspace' Key

To complete the Test Case 3, the following text, "the quick brown fox jump over the lazy dog" from the previous test case is used and the word "dog" is to be removed from the text to test the 'Backspace' function. The details and the results of the Test Case 3 are stored in Table 5-4 and Figure 5-8 below respectively.

Test Case	Example	Expected Result	Actual Result
'Backspace'	the quick brown fox jump over the lazy dog	The "dog" word is removed.	Same as the expected result

Table 5-4: The Details of the Test Case 3

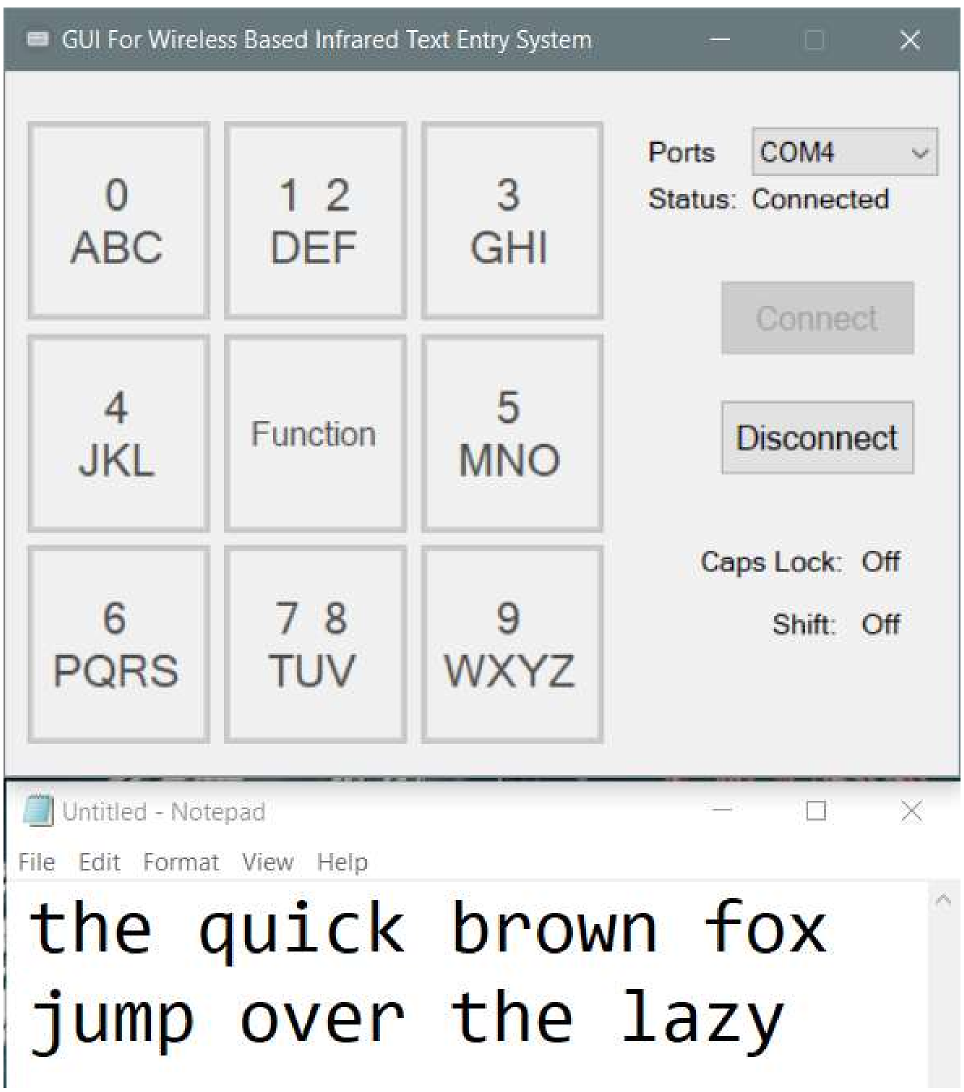


Figure 5-8: The Results of the Test Case 3

5.2.4 Test Case 4: Uppercase Alphabets and 'Tab' Key

To complete the Test Case 4, the following text, "THE QUICK BROWN FOX JUMP OVER THE LAZY DOG" is used as an example to input to the computer using the text entry system. Besides, the 'Tab' key will be added before the text. The details and the results of the Test Case 4 are stored in Table 5-5 and Figure 5-9 below respectively.

Test Case	Example	Expected Result	Actual Result
Uppercase alphabets and 'Tab'	THE QUICK BROWN FOX JUMP OVER THE LAZY DOG	All the letters of the text are added and the 'Tab' key is added before the text.	Same as the expected result

Table 5-5: The Details of the Test Case 4

Figure 5-9: The Results of the Test Case 4

5.2.5 Test Case 5: Numbers, Punctuations, ‘Shift’ Key and ‘Enter’ Key

To complete the Test Case 5, the digits from 0 to 9 and punctuations are inserted into the computer using the text entry system. Besides, the ‘Shift’ key is tested too by capitalizing the first letter of the first word. In addition, the ‘Enter’ key is tested as well to input the carriage return. The details and the results of the Test Case 5 are stored in Table 5-6 and Figure 5-10 below respectively.

Test Case	Example	Expected Result	Actual Result
Numbers, Punctuations, ‘Shift’ Key and ‘Enter’ Key	Hi! I am Adam, my number is 0123456789. 10+5-3 Coffee/ Tea?	- All the numbers and punctuations are inserted successfully - ‘Shift’ key and ‘Enter’ key function well	Same as the expected result

Table 5-6: The Details of the Test Case 5

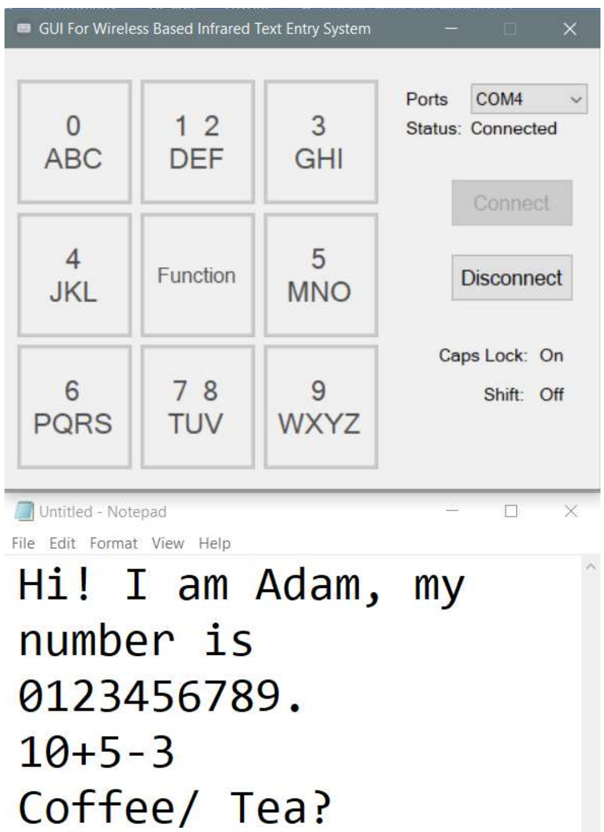


Figure 5-10: The Results of the Test Case 5

5.2.6 Test Case 6: Mouse Function

To complete the Test Case 6, the mouse function is used to move the mouse pointer of the computer using the text entry system. Besides moving the mouse pointer, the mouse function can also use to issue ‘Left Click’, ‘Right Click’ and ‘Middle Click’ like the ordinary mouse. Hence, a small program is developed to detect the ‘Left Click’, ‘Right Click’ and ‘Middle Click’ event. The details of the Test Case 6 are stored in Table 5-7 below.

Test Case	Example	Expected Result	Actual Result
Mouse Function	- To control the mouse pointer and open the program - To test the ‘Left Click’, ‘Right Click’ and ‘Middle Click’ using the program	All the functions are working	Same as the expected result

Table 5-7: The Details of the Test Case 6

The Figure 5-11 below shows the program is launched by controlling the mouse pointer using the text entry system.

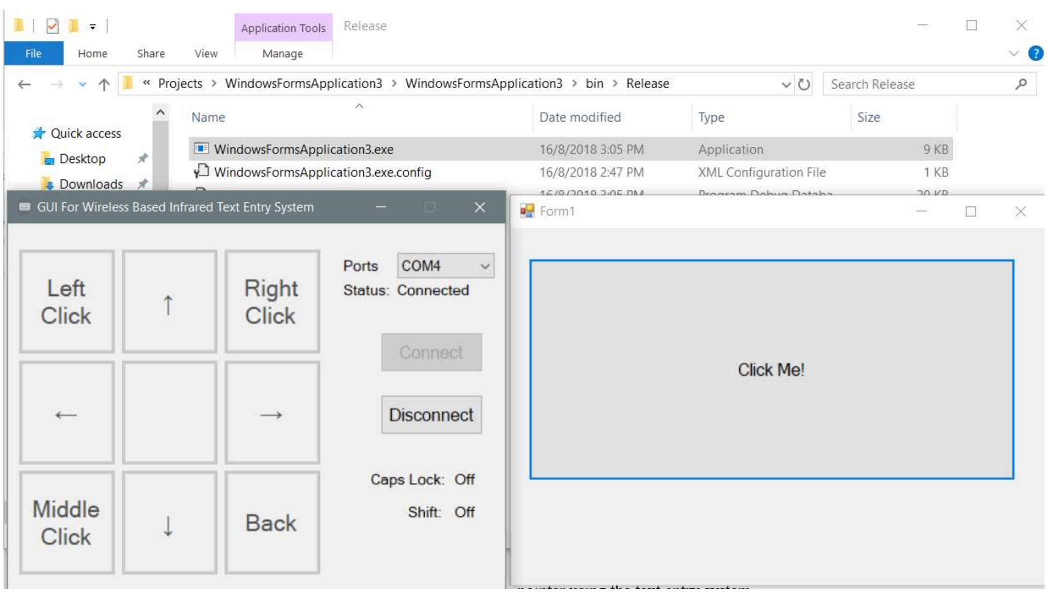


Figure 5-11: The program is launched by controlling the mouse pointer using the text entry system

The program is to test whether which mouse button is clicked. To start the test, the mouse pointer is moved to the center of the program. Then, the "Left Click" mouse function is pressed. The Figure 5-12 below shows the result of the program after the "Left Click" mouse function is pressed.

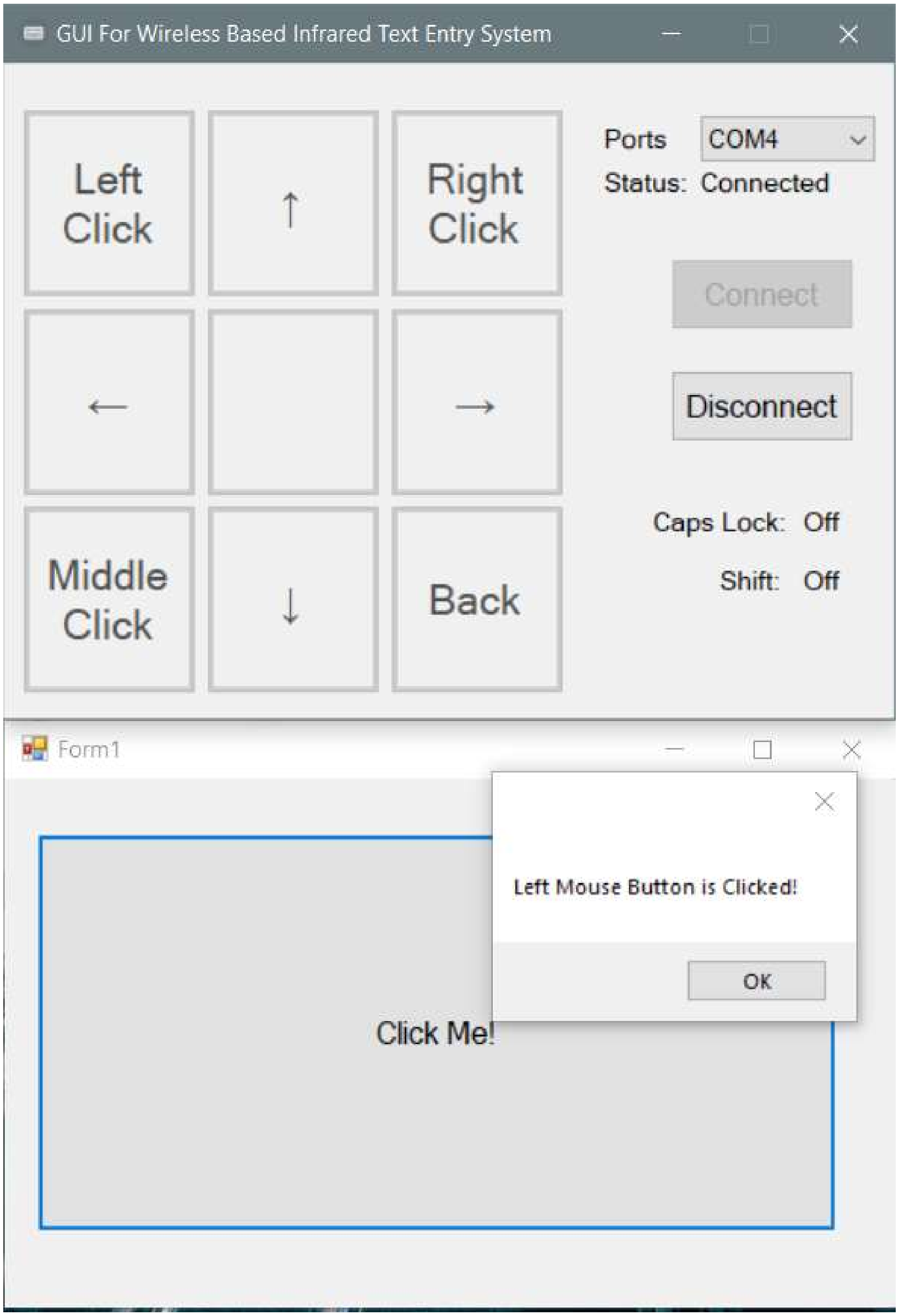


Figure 5-12: The result of the program after the "Left Click" mouse function is pressed

Next, the "Right Click" mouse function is pressed. The Figure 5-13 below shows the result of the program after the "Right Click" mouse function is pressed.

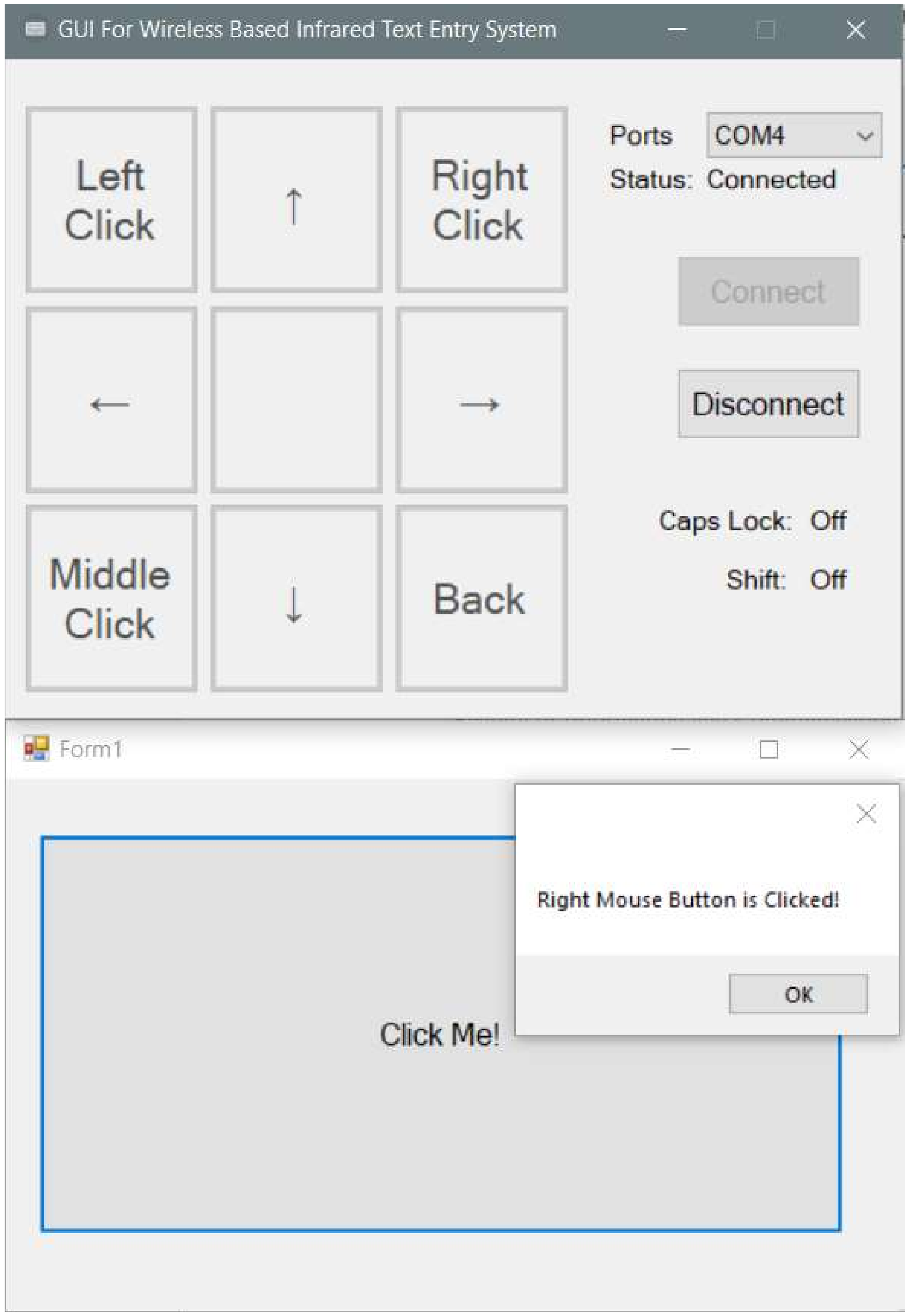


Figure 5-13: The result of the program after the "Right Click" mouse function is pressed

Then, the “Middle Click” mouse function is pressed. The Figure 5-14 below shows the result of the program after the “Middle Click” mouse function is pressed.

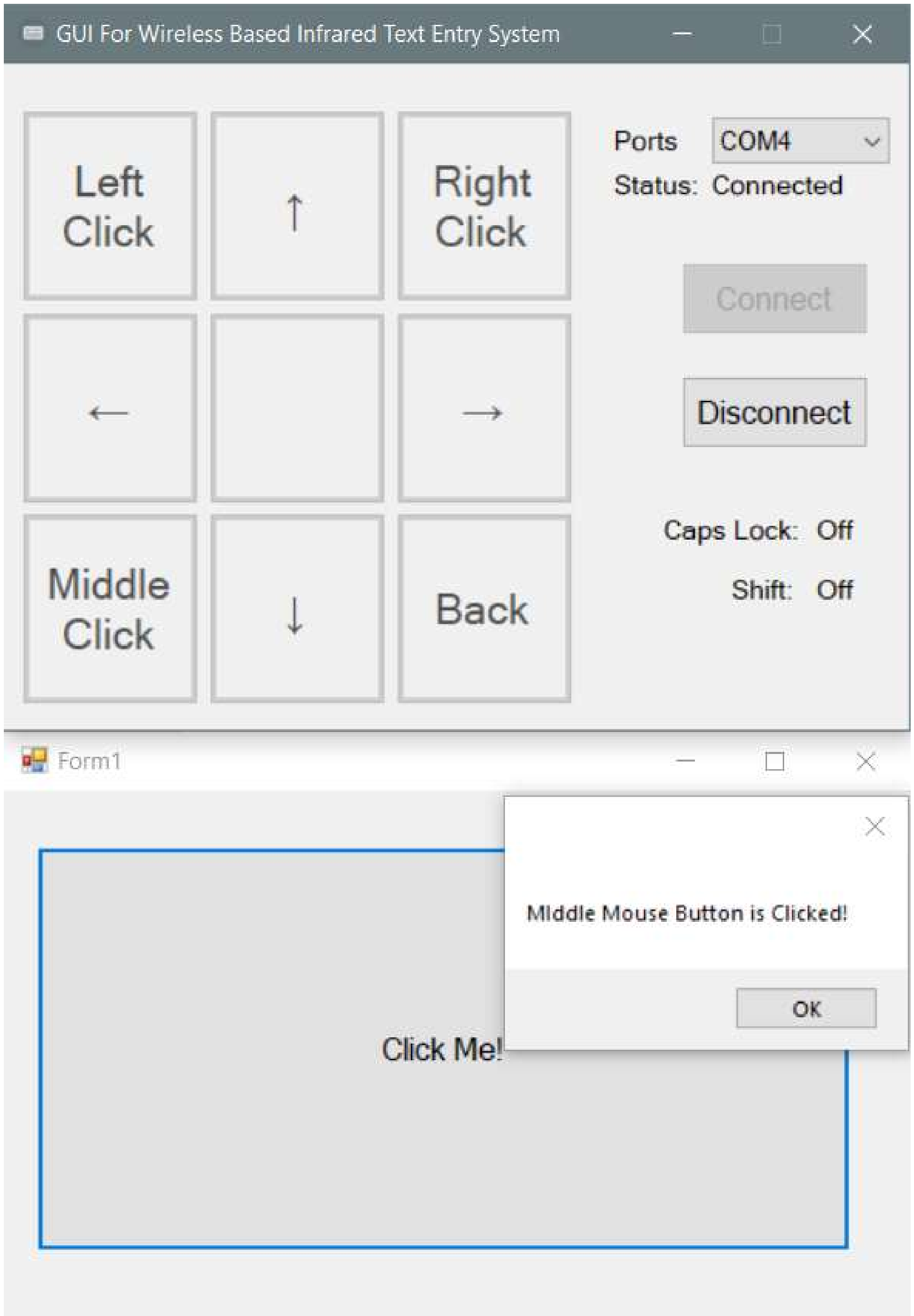


Figure 5-14: The result of the program after the “Middle Click” mouse function is pressed

5.3 Speed

In order to determine the text entry speed of the system, a special version of the software GUI program is developed, with all the functions from the original software GUI program, but it will record down the elapsed time, characters typed, character per minute (CPM), words typed and words per minute (WPM). CPM and WPM are good methods to explore the typing capabilities and efficiency of the text entry system. To calculate the CPM of the system, the number of total characters typed is multiplied by 60 seconds and then divided by the elapsed time. The calculation of the WPM is almost the same with the calculation of the CPM where the number of words typed is multiplied by 60 seconds and then divided by the elapsed time.

The number of words will be identified from the characters typed by the users automatically by the program. One of the ways to detect the words is by detecting the space entered by the user. The words are separated by spaces. Hence, when a space is detected, the characters typed before the space would be considered as a word. Besides, normally, the last word entered by the users will end with punctuations or without punctuations. Eventually, the program will count it as a word.

The Figure 5-15 below shows the interface of the special version of the software GUI program.

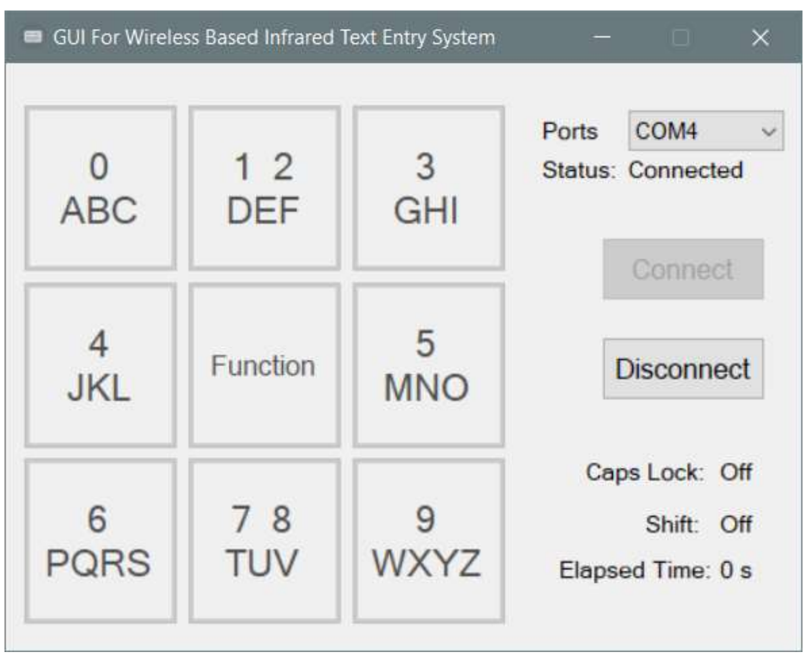


Figure 5-15: The interface of the special version of the software GUI program

The timer starts when the user presses the 'Connect' button, and when the user has finished typing the text and close the program, there will be a pop-up message showing the elapsed time, characters typed, CPM, words typed and WPM. The Figure 5-16 below shows the pop-up message after the user typed the text "the quick brown fox jump over the lazy dog" and close the program or press the 'Disconnect' button.

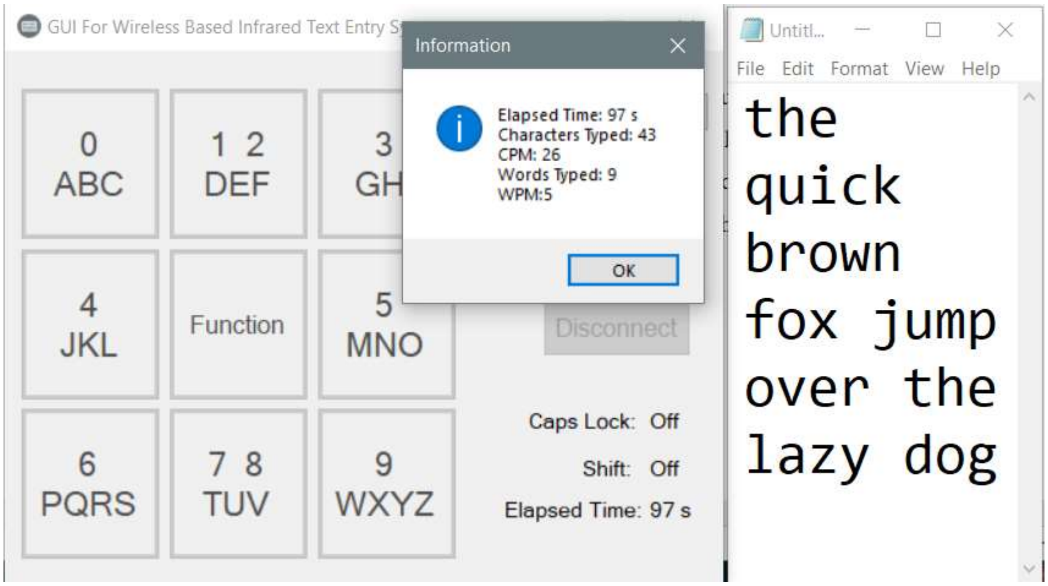


Figure 5-16: The result of the elapsed time, character typed, CPM, words typed, and WPM

The Table 5-8 below shows the CPM and WPM for 10 times by typing the same text "the quick brown fox jump over the lazy dog".

No. of attempts	CPM	WPM
1	22	4
2	23	5
3	23	5
4	24	5
5	24	5
6	26	5
7	26	5
8	26	5
9	26	5
10	27	6

Table 5-8: The CPM and the WPM for 10 attempts by typing the same text

Based on the text entry method proposed by Felzer, T. and Rinderknecht, S. on 2011, as discussed in Chapter 2.2, the entry rate of the mentioned text entry system has an entry rate ranging from 0.8 WPM to 2 WPM compared to the Infrared Text Entry System in this research which has an average of about 5 WPM.

Besides, based on the BrailleKey text entry method proposed by Subash, N.S., Nambiar, S. and Kumar, V. on 2012, as discussed in Chapter 2.3, the entry rate of this text entry system is shown in Figure 5-17 below.

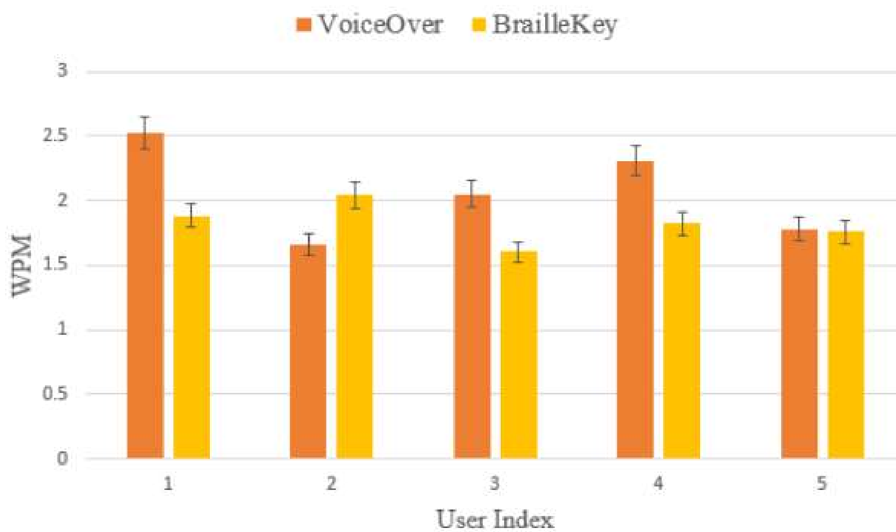


Figure 5-17: WPM for each participant

Based on the Figure 5-17 above, the entry rate of the BraileKey text entry system is about 2 WPM among the 5 participants. As a result, the entry rate of the system is lower than the Infrared Text Entry System.

5.4 Power Consumption

The power consumption of the text entry system needs to be calculated to know its efficiency. First, the current drawn by the system is calculated using a power supply, a multimeter, and a breadboard. The power supply is set to supply +5V and the multimeter is switched to measure current mode since the multimeter can measure thr voltage and the resistance also. Then, the 5V positive wire is connected to the red probe of the multimeter while the black probe of the multimeter is connected to the Vin port of the CT-UNO. Next, the ground wire from the power supply is connected to the GND port of the CT-UNO. The Figure 5-18 shows the setup on how to measure the current drawn by the system.

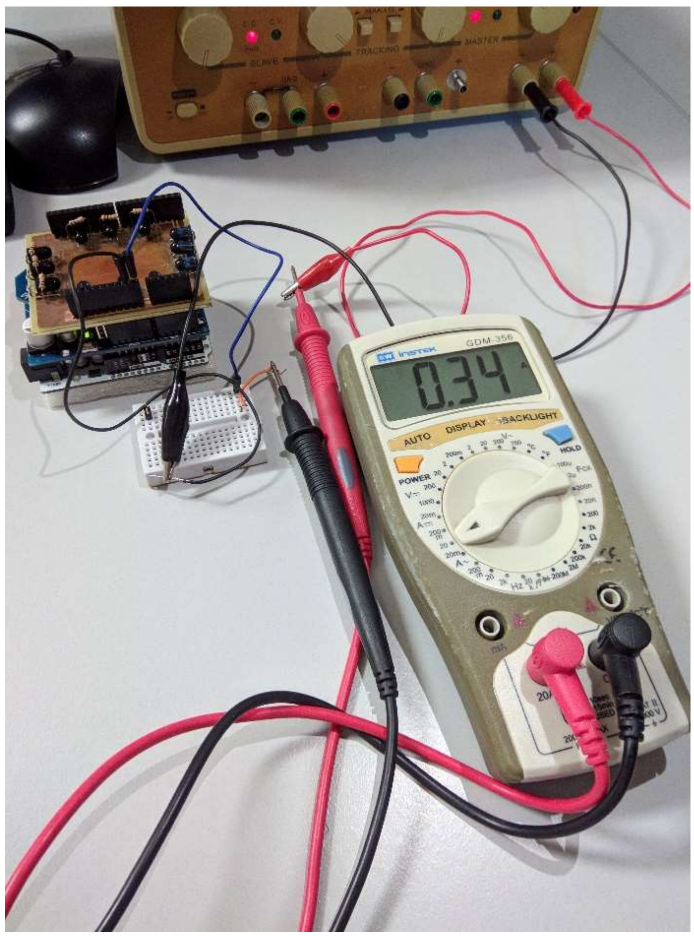


Figure 5-18: The setup on how to measure the current drawn by the system

After the current drawn by the system is determined, the power consumption of the text entry system can be calculated by multiplying the current drawn by the system with the supply voltage. The measurement of the current drawn by the system is shown in Figure 5-19 below.

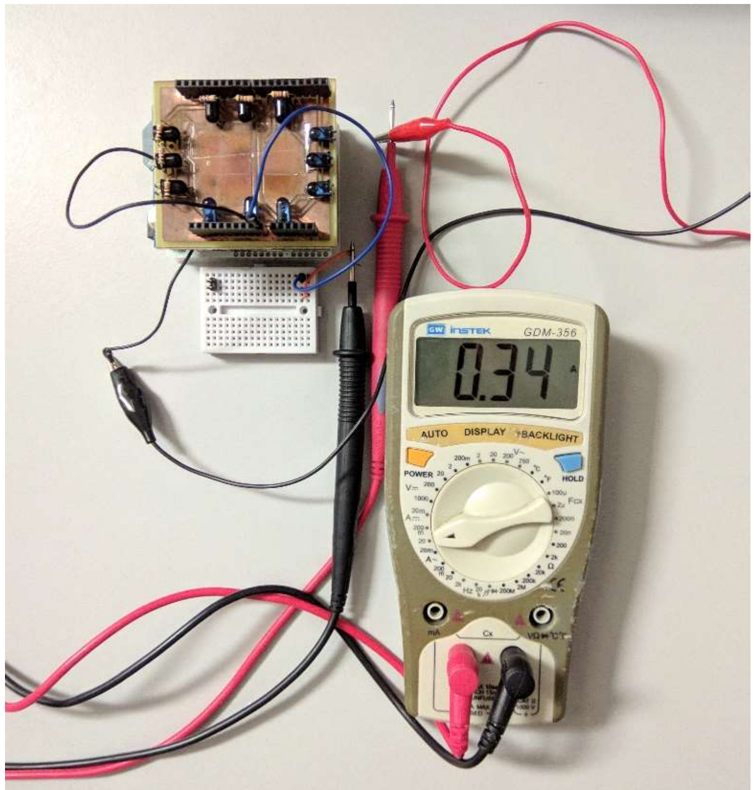


Figure 5-19: The measurement of the current drawn by the system

The calculation of the power consumption is shown below.

Power (P) = Current (I) x Voltage (V)

Voltage Supply = 5V

Current Drawn by the System = 0.34A

Power Consumption of the system per hour = $0.34A \times 5V = 1.7W$

From the calculations above, the results show that the power consumption of the system is only about 1.7W. If the user powers the text entry system using a 10000mAh power bank, the maximum operating hour of the text entry system is shown below.

Maximum Operating Hour = $\frac{10000mAh}{340mA} \approx 30hours$

Hence, the system can operate close to 30 hours long with a 10000mAh power bank.

CHAPTER 6: Conclusion

In this chapter, the objectives of the project that has been achieved will be discussed here. Besides, the issues and challenges faced during the development of the device and the GUI program will also be explained here. Furthermore, the personal insight into the total research experience of the researcher is added in this chapter. In addition, the novelties and the limitations of the system are discussed in this chapter too. Lastly, this chapter will discuss the possible future enhancement of the system.

6.1 Project Review

The main objective of this project is to provide a suitable and low-cost text entry system for the handicaps to let them communicate simultaneously with others. This objective is achieved because the hardware components used in this project such as the IR transmitter and receiver are cheap. There are 5 sub-objectives of this project which is to make the device portable, lightweight, long operating hour, has a wireless connection and user-friendly are achieved too. The device is portable and lightweight because the dimension of the device is about 7cm x 5.5cm x 5.5cm and it is very light to hold. Besides, the device can last for a long time even by using the power bank. In addition, the device can connect wirelessly to the computer. It means the user does not need to connect the device to the computer via the USB cable to operate. Furthermore, the device is user-friendly as the GUI program will guide the user throughout the text input process and the user only needs minor instruction and memorisation to operate the device.

During the hardware implementation of the text entry system, there are many issues and challenges happened. One of the issues is about the components. Some components such as the IR LEDs are broken. Hence, before the soldering process of the components into the PCB, the components need to be checked first to ensure the components are functioning.

In addition, after the soldering of components into the PCB, some components are actually not fully soldered into the PCB. Hence, the system may not work properly and further checking on the soldered components need to be done and extra time is consumed.

Besides, the designation of the PCB become one of the issues. The idea of designing the PCB instead of using the ordinary breadboard and jumper wires is because the PCB has no jumper wire at all. This is to keep the text entry system less complex and simple. Due to zero knowledge of designing the PCB, some studies are needed on how to design a PCB. It takes extra time to design the system.

Furthermore, the development of the GUI program become an issue. The GUI program is developed with the Microsoft Visual Studio 2013 IDE using C# programming language. Due to zero knowledge of developing the program in C# language, the best and fastest way to learn the language is to search the tutorials available online. It takes extra time in developing the program.

Besides, the program needs to be set to constantly check for the signal sent from the device unless the user clicks the "Disconnect" button to stop the program. The program would hang because the program will have stuck in the infinite loop checking for the signal sent by the device. Hence, the system threading of the C# programming language is implemented here. Another thread is created to monitor if the "Disconnect" button is clicked. Then, the thread will inform the main thread to exit from the infinite loop and stop the program.

Despite a lot of difficulties and challenges happened during the development of this project, fortunately, those issues and challenges can be solved and overcame. Hence, many new pieces of knowledge such as designing a PCB and developing a C# program are gained.

6.2 Novelties

One of the novelties of this project is that the text entry system is developed using only low-cost components such as IR transmitter, IR receiver and some resistors only. Besides, a PCB is used to fit all the components together rather than using a breadboard. This is to reduce the number of jumper wire and to keep the device as simple as possible.

6.3 Limitations of the System

The CT-UNO microcontroller has only 32KB of flash memory for the users to store their program. Due to the limited memory space of the microcontroller, more functions cannot be implemented to the device because after the sketch is uploaded to the microcontroller, there is not much space left for the memory.

Besides, the infrared text entry board may not identify the intersection points pressed by the user correctly as the finger size of every user is not the same. Hence, the device may need to be optimised to suit every user.

6.4 Future Enhancement

Further improvement can be made to solve the limitation of the system mentioned above. One of the enhancement is to replace the microcontroller with a larger memory space. Arduino Mega 2560 is a better alternative compared to the CT-UNO which only has 32KB of flash memory. Arduino Mega 2560 has 54 digital I/O pins and 256KB of flash memory. Hence, more functions can be provided to the device with this large amount of flash memory, but it may increase the development cost.

Besides, further enhancement can be done by further reducing the cost of the development material. The price of the Cytron XBee Shield with the Bluetooth module is more expensive as compared to the CT-UNO microcontroller. Hence, cheaper Bluetooth module is needed in this project.

For this project, the text entry system can only work on the Windows computer with the GUI program. Hence, the device can be improved so that the device can be used on other operating system such as Linux or even on mobile devices such as Android or iOS.

REFERENCES

Clawson, J., Lyons, K., Starner, T. and Clarkson, E., 2005. The impacts of limited visual feedback on mobile text entry for the twiddler and mini-QWERTY keyboards. *Proceedings - International Symposium on Wearable Computers, ISWC*, 2005, pp.170–177.

Felzer, T. and Rinderknecht, S., 2011. Using a Game Controller for Text Entry to Address Abilities and Disabilities Specific to Persons with Neuromuscular Diseases. *The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility*, pp.299–300. Available at: http://doi.acm.org/10.1145/2049536.2049616.

Fu, Y.F. and Ho, C.S., 2009. A fast text-based communication system for handicapped aphasiacs. *IIH-MSP 2009 - 2009 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp.583–594.

Groups, B.C. and Cell, S., B1-B9 Cell Groups (Serotonergic Cell Groups).

I. Scott MacKenzie and Tanaka-Ishii, K., 2007. *Text Entry Systems: Mobility, Accessibility, Universality*,

Miró-Borrás, J., Bernabeu-Soler, P., Llinares, R. and Igual, J., 2010. An ambiguous keyboard based on "Character Graphical Association" for the severely physically handicapped. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6180 LNCS(PART 2), pp.140–143.

Miró, J. and Bernabeu, P.A., 2008. Text entry system based on a minimal scan matrix for severely physically handicapped people. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5105 LNCS(1992), pp.1216–1219.

Noyes, J., 1998. QWERTY - the immortal keyboard. *Computing & Control Engineering Journal*, 9(3), p.117.

Panwar, P., Sarcar, S. and Samanta, D., 2012. EyeBoard: A fast and accurate eye gaze-based text entry system. *4th International Conference on Intelligent Human Computer Interaction: Advancing Technology for Humanity, IHCI 2012*.

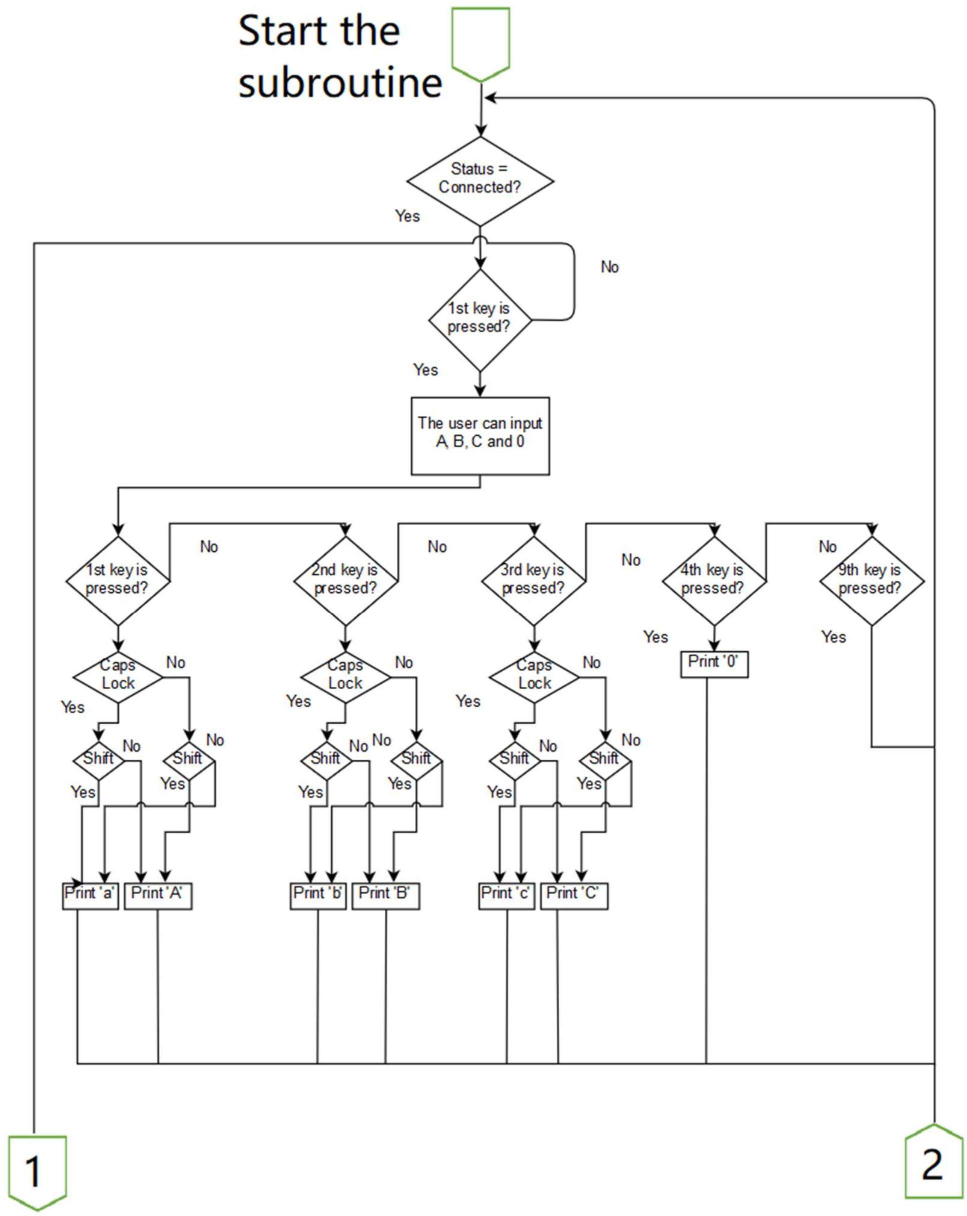
Sharp, H., Rogers, Y. and Preece, J., 2007. Interaction design: beyond human-computer interaction. *Book*, 11, p.773. Available at: http://www.id-book.com/secondedition/.

Subash, N.S., Nambiar, S. and Kumar, V., 2012. BrailleKey: An alternative Braille text input system: Comparative study of an innovative simplified text input system for the visually impaired. *4th International Conference on Intelligent Human Computer Interaction: Advancing Technology for Humanity, IHCI 2012*, pp.4–7.

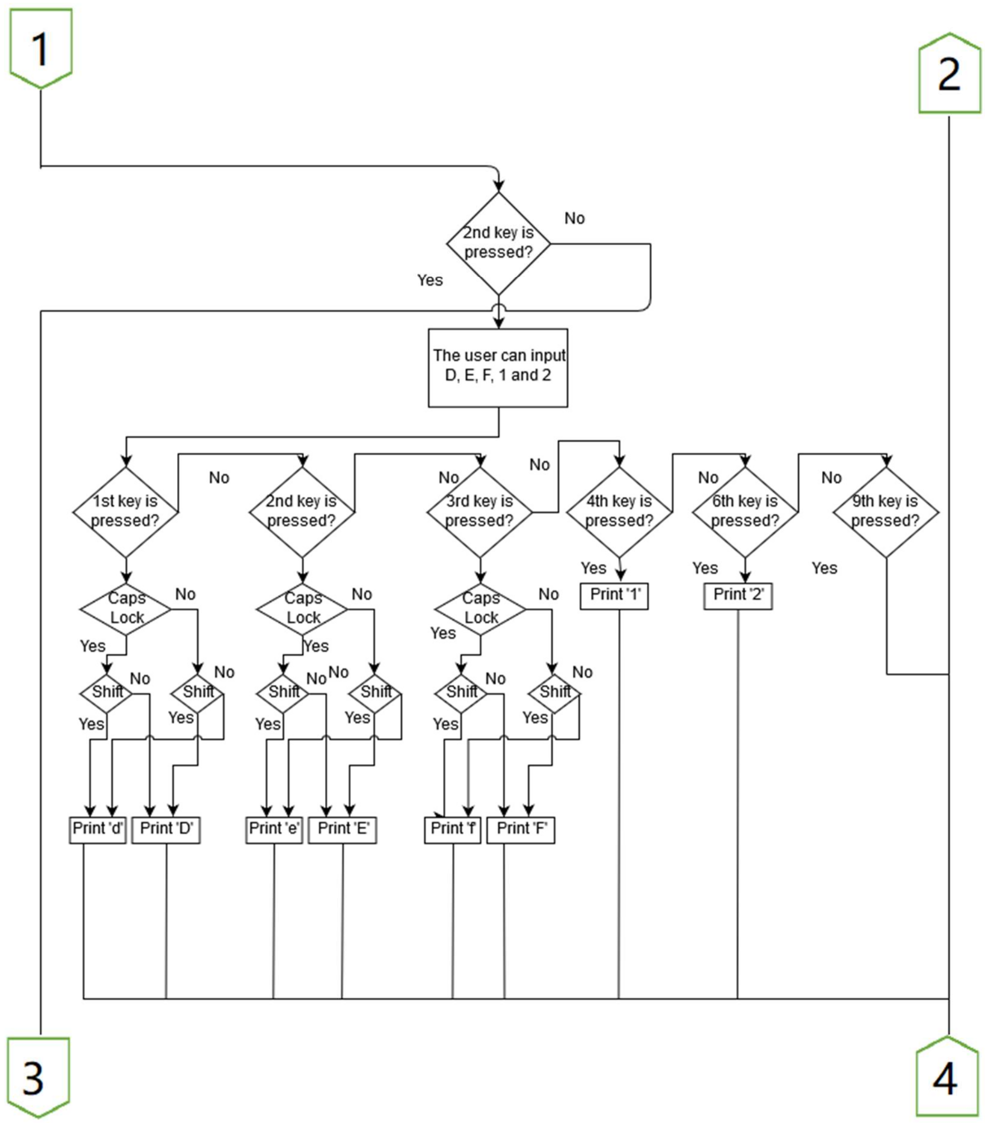
tutorialspoint, 2017, *SDLC - Overview* [Online]. Available at: https://www.tutorialspoint.com/sdlc/sdlc_overview.htm.

Verma, E.P., 2015, *SDLC Prototype model : Design , advantages , disadvantages and applications* [Online]. Available at: https://er.yuvayana.org/sdlc-prototype-model-design-advantages-disadvantages-and-applications/.

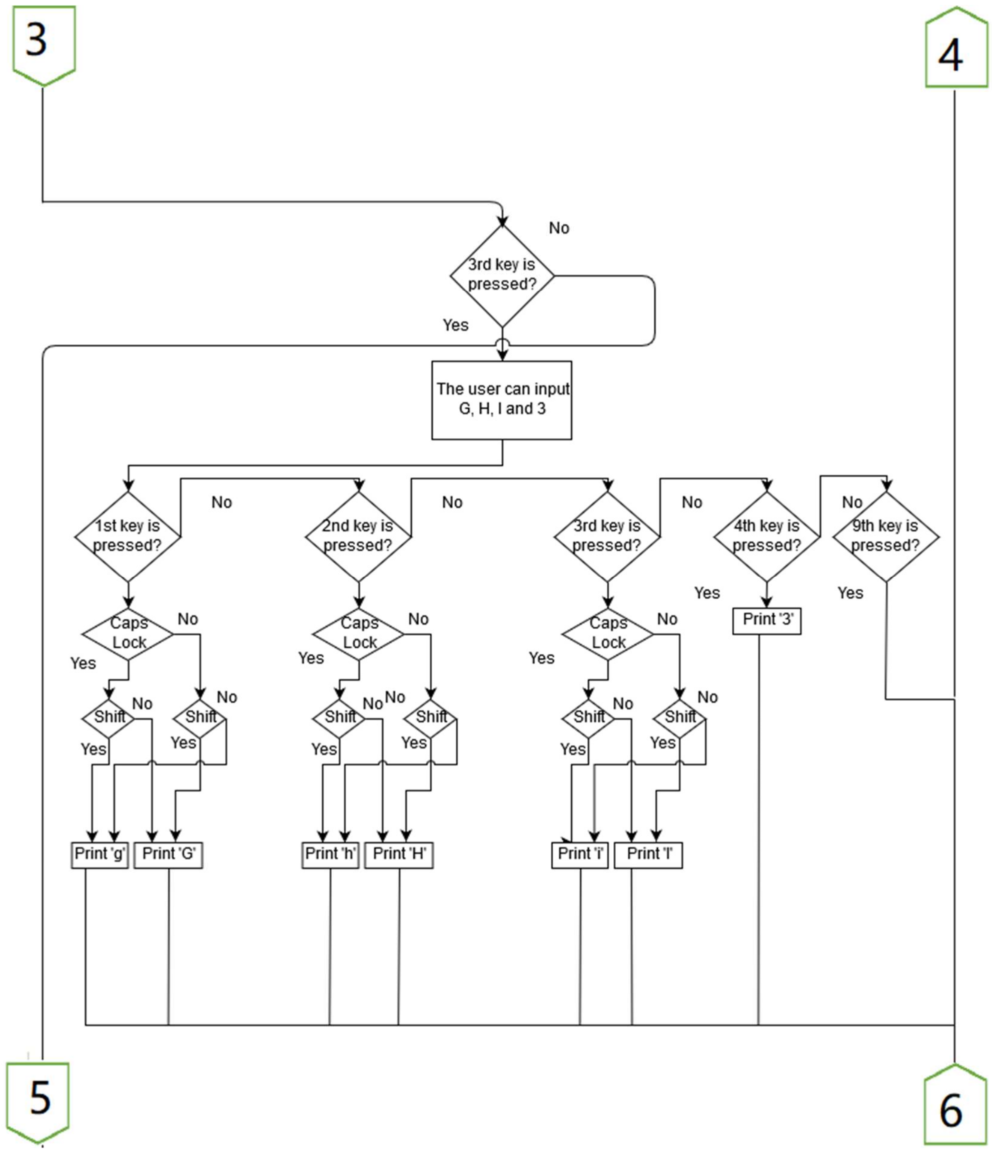
APPENDIX A: The Subroutine of Analysing the Key Signal from the Device



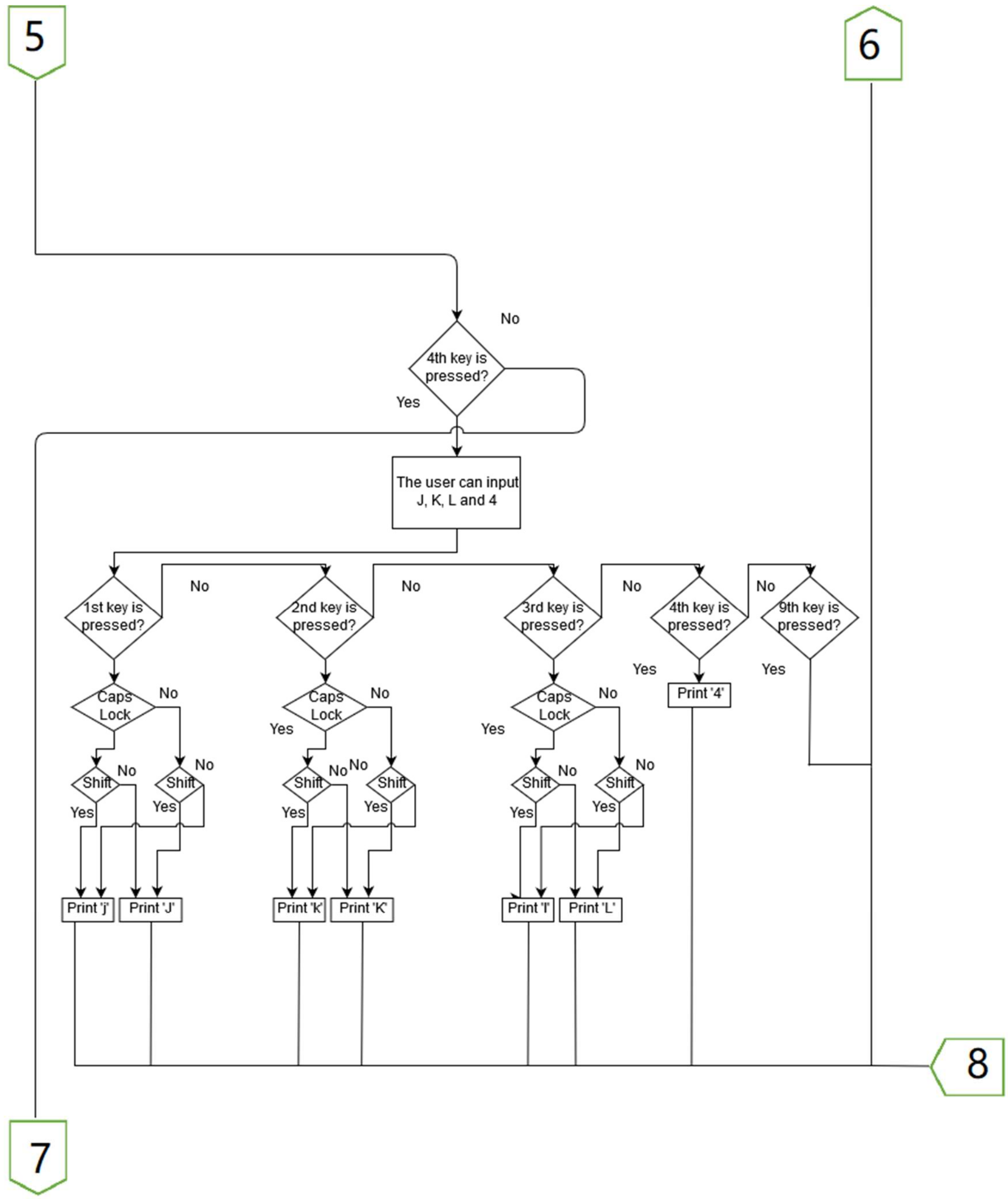
Appendix A: The Subroutine of Analysing the Key Signal from the Device



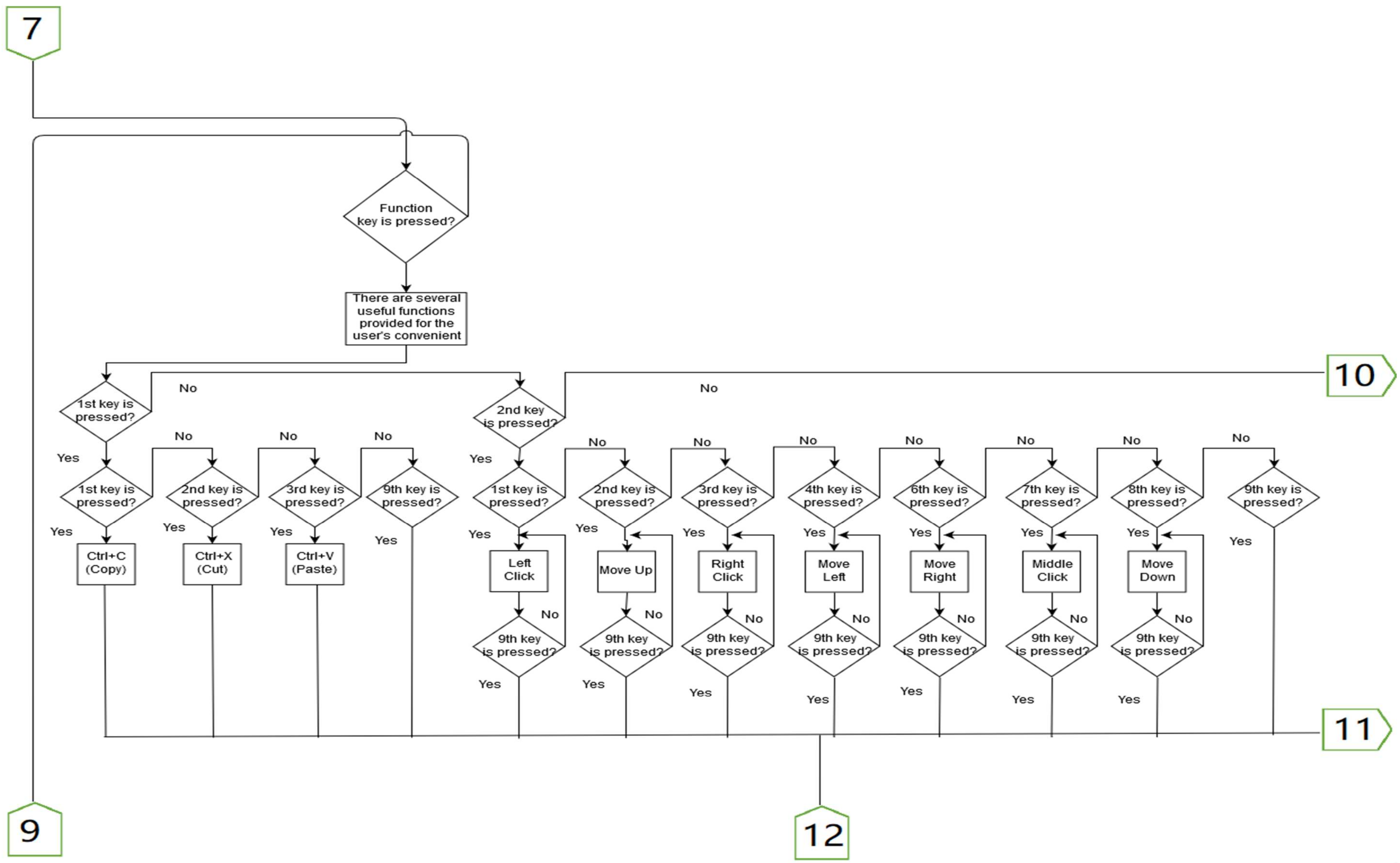
A-2



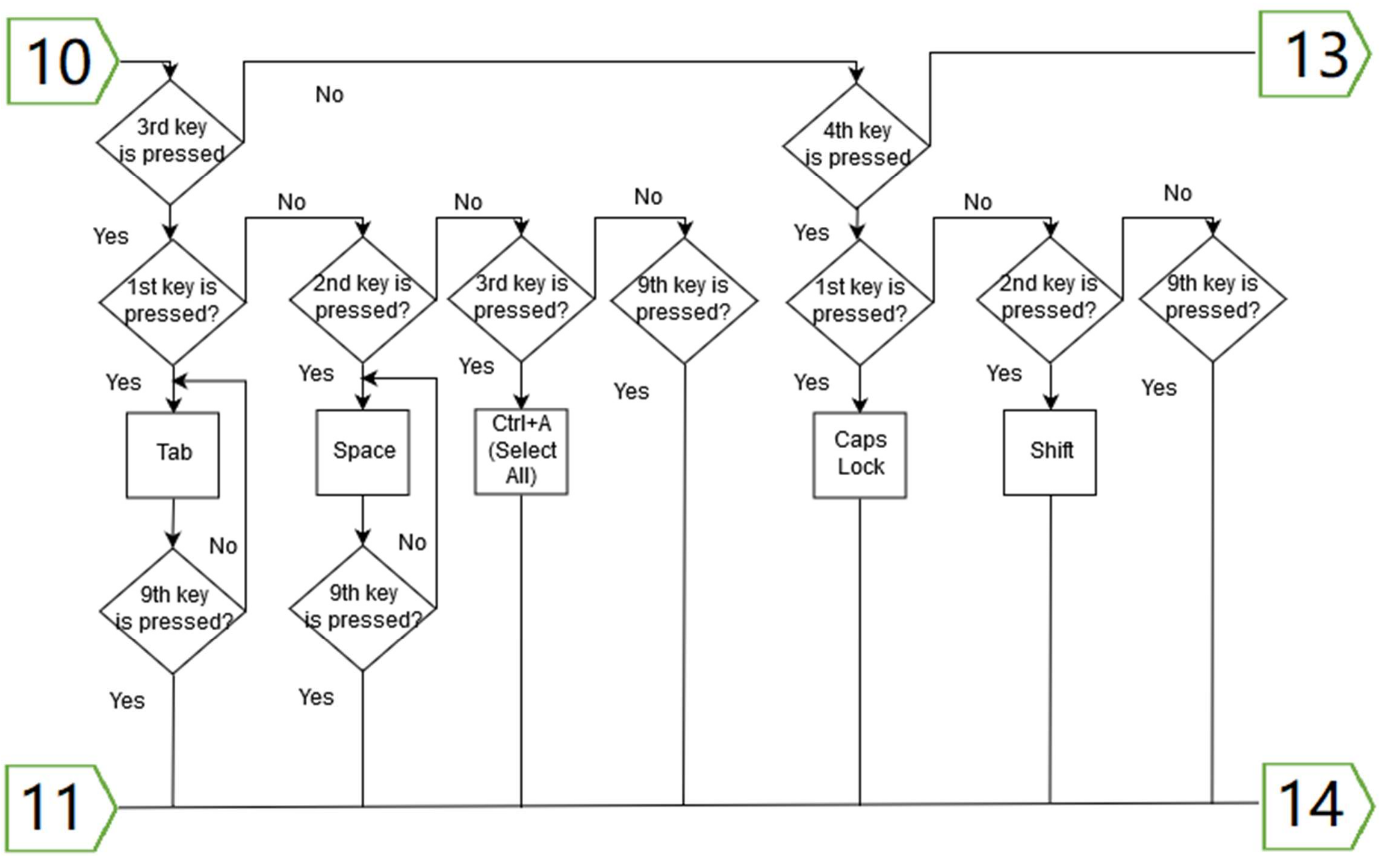
3
4
No
3rd key is pressed?
Yes
The user can input G, H, I and 3
1st key is pressed?
No
2nd key is pressed?
No
3rd key is pressed?
No
4th key is pressed?
No
9th key is pressed?
Yes
Yes
Print '3'
Caps Lock
No
Yes
Shift
No
Yes
Shift
No
Yes
Print 'g'
Print 'G'
Print 'h'
Print 'H'
Print 'i'
Print 'I'
5
6


5
6
No
4th key is pressed?
Yes
The user can input J, K, L and 4
1st key is pressed?
No
2nd key is pressed?
No
3rd key is pressed?
No
4th key is pressed?
No
9th key is pressed?
Yes
Yes
Print '4'
Caps Lock
No
Caps Lock
No
Yes
Caps Lock
No
Yes
Shift
Shift
Yes
No
Print 'j'
Print 'J'
Print 'k'
Print 'K'
Print 'l'
Print 'L'
8
7

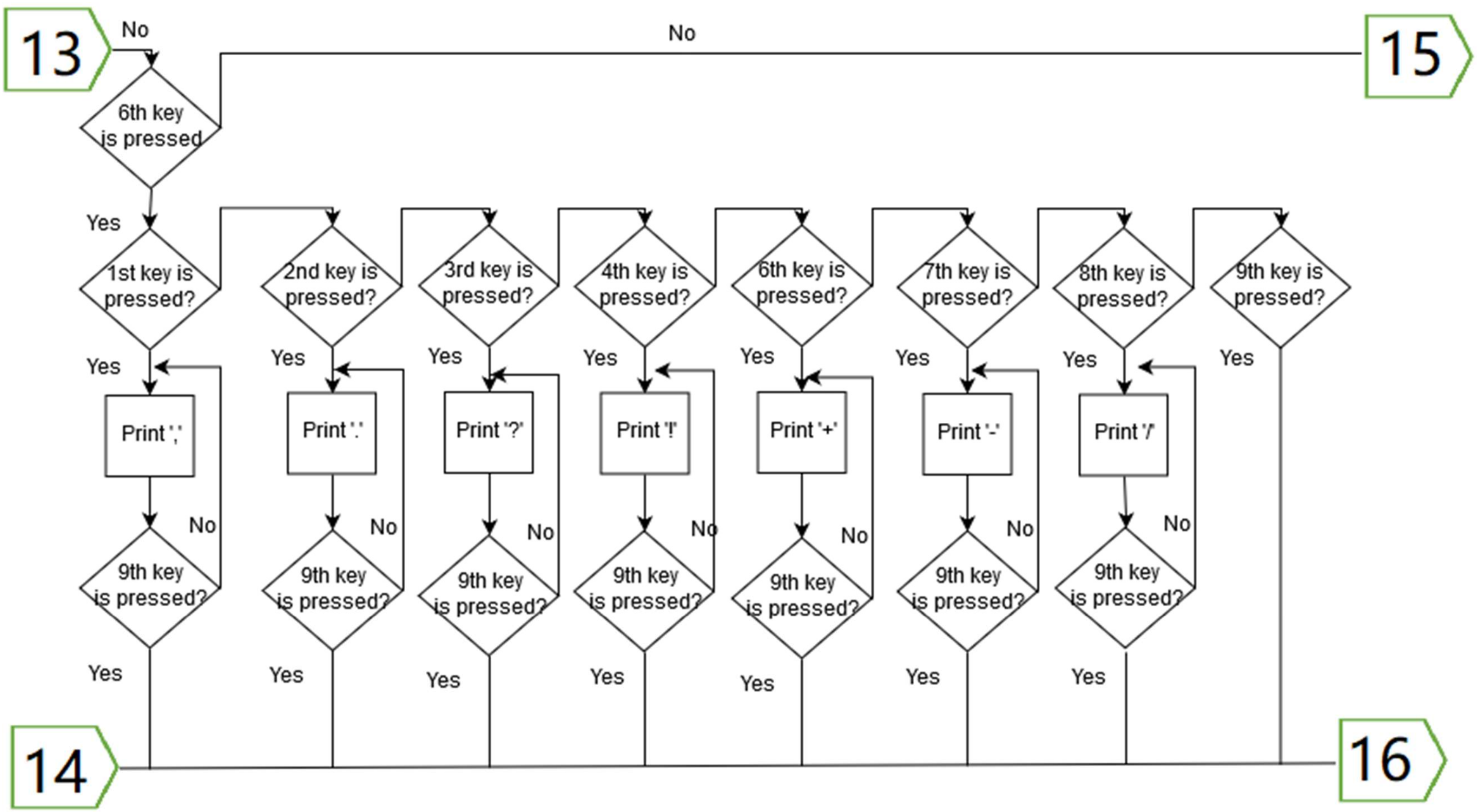
Appendix A: The Subroutine of Analysing the Key Signal from the Device



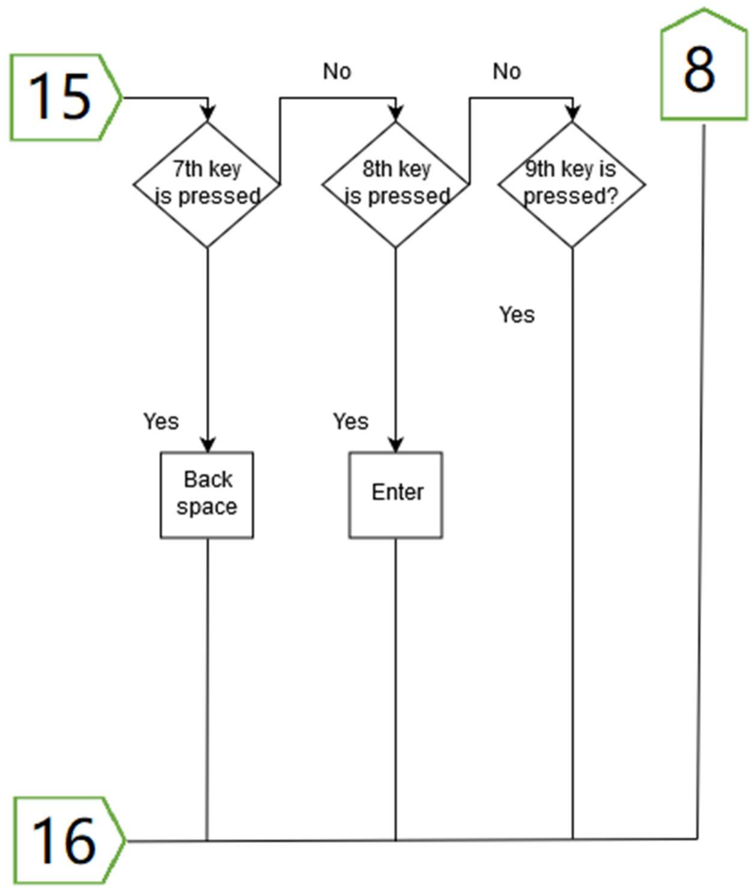
Appendix A: The Subroutine of Analysing the Key Signal from the Device

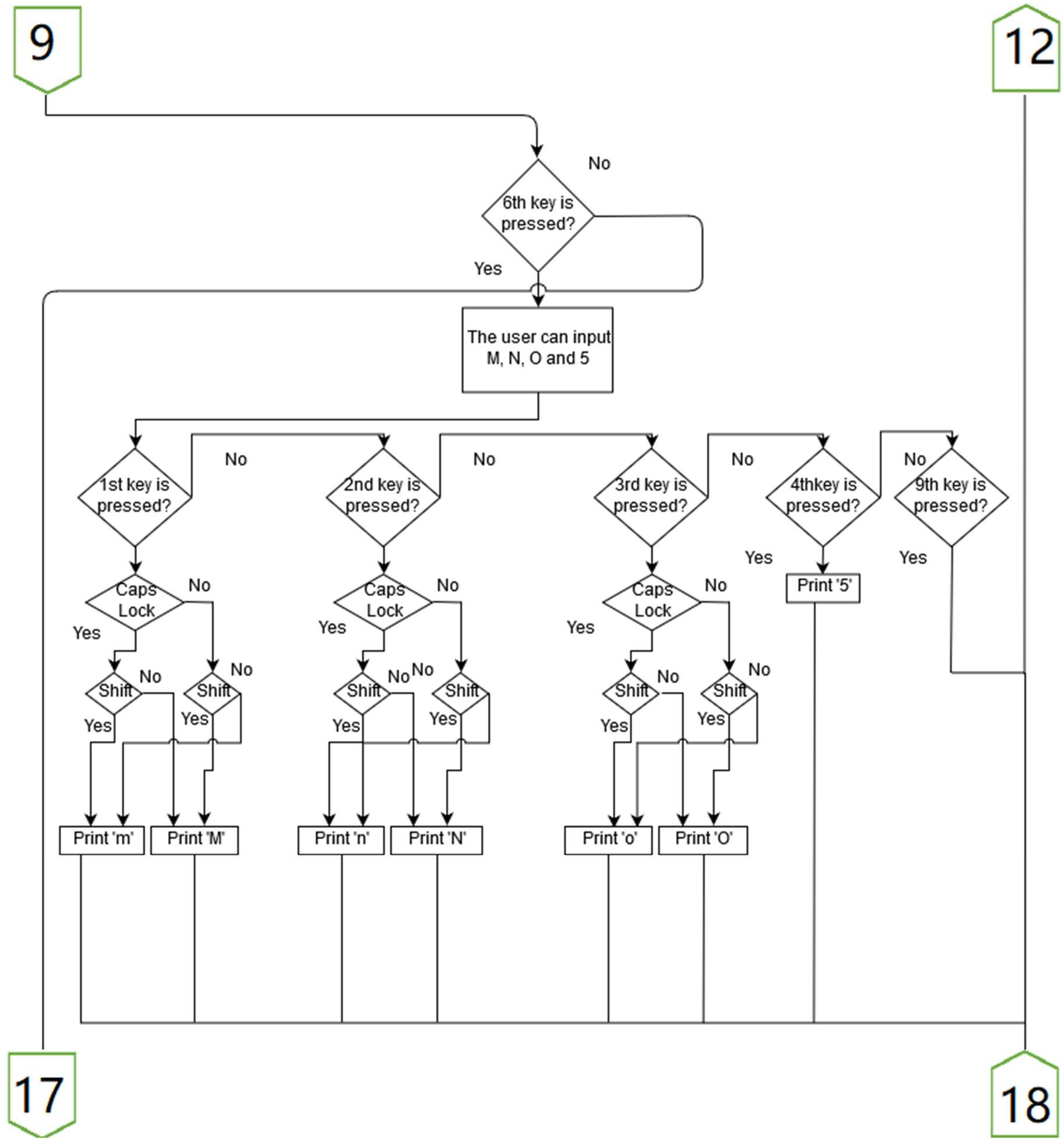


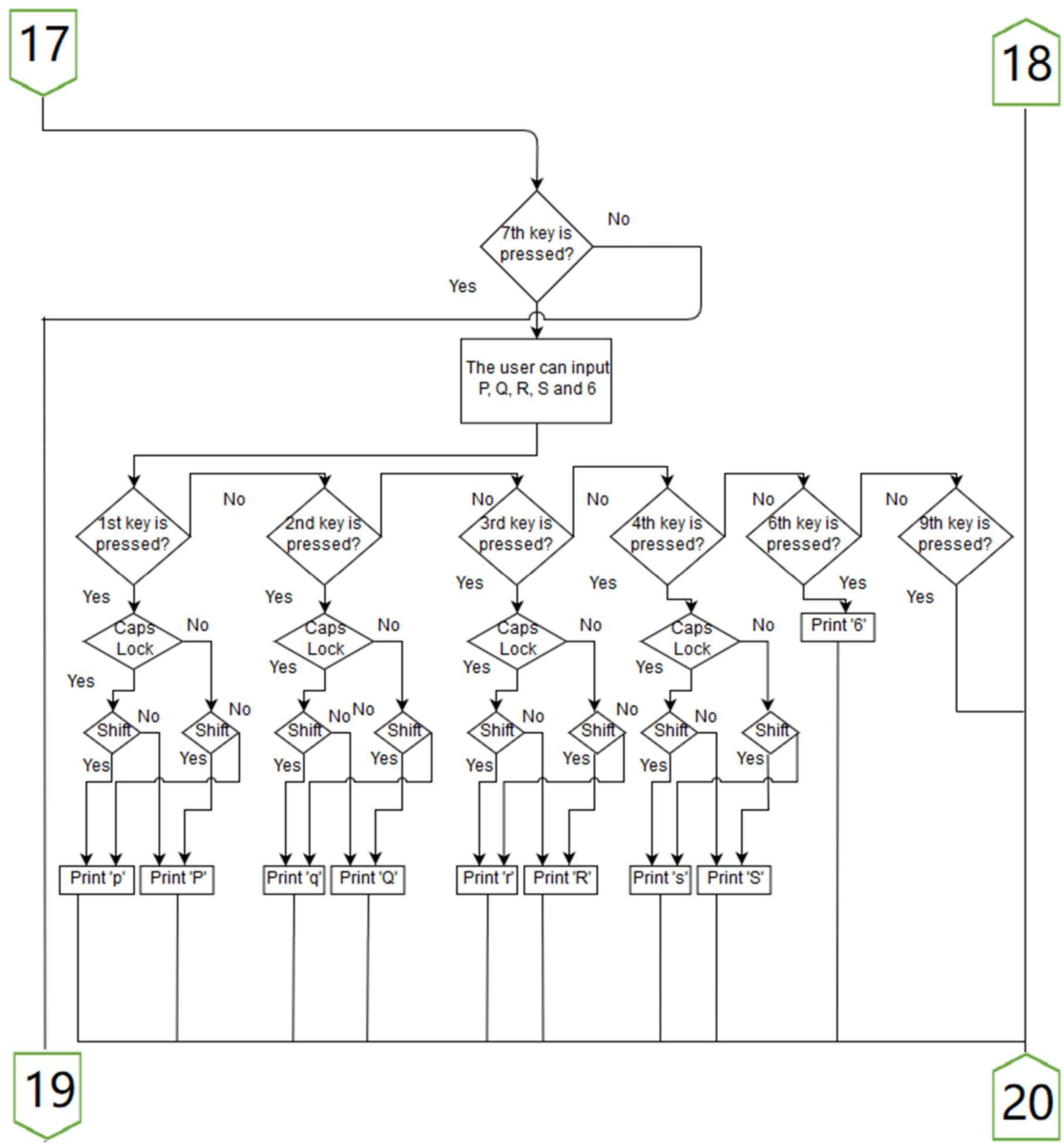
Appendix A: The Subroutine of Analysing the Key Signal from the Device

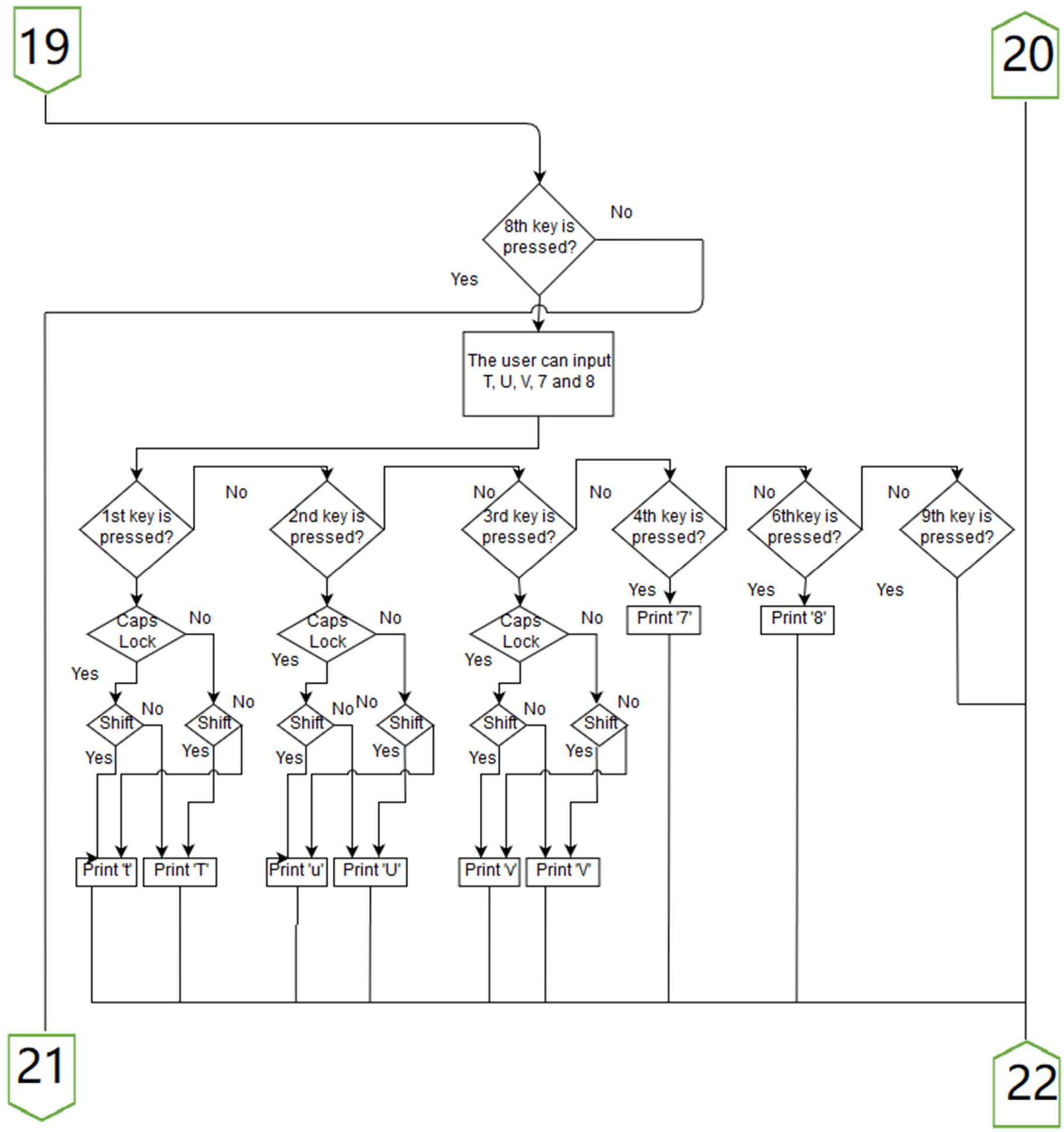


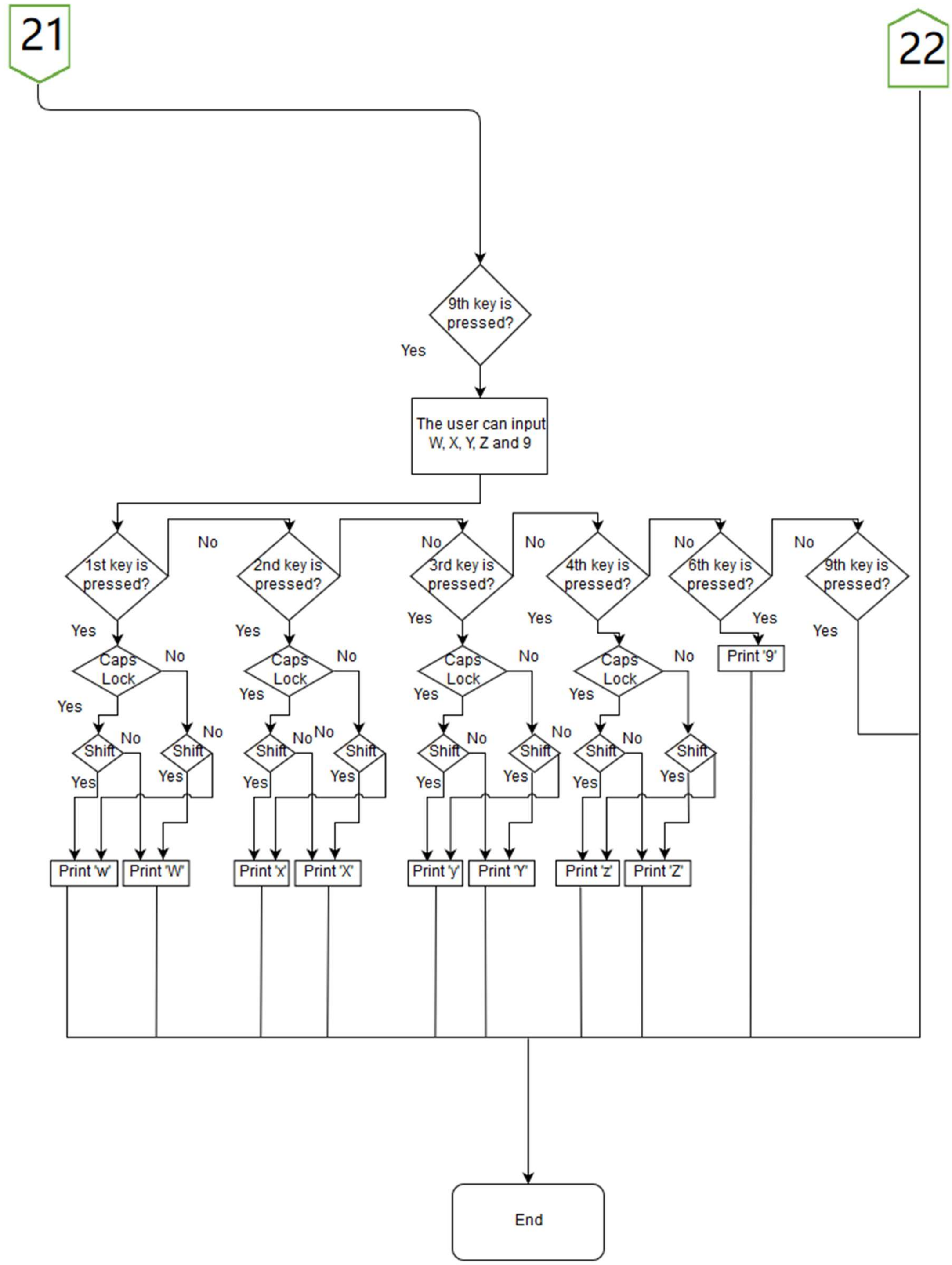
Appendix A: The Subroutine of Analysing the Key Signal from the Device




9
12
No
6th key is pressed?
Yes
The user can input M, N, O and 5
1st key is pressed?
No
2nd key is pressed?
No
3rd key is pressed?
No
4thkey is pressed?
No
9th key is pressed?
Yes
Print '5'
Yes
Caps Lock
No
Yes
Shift
No
Shift
No
Yes
Yes
Print 'm'
Print 'M'
Caps Lock
No
Yes
Shift
No
Shift
No
Yes
Yes
Print 'n'
Print 'N'
Caps Lock
No
Yes
Shift
No
Shift
No
Yes
Yes
Print 'o'
Print 'O'
17
18


17
18
7th key is pressed?
No
Yes
The user can input P, Q, R, S and 6
1st key is pressed?
2nd key is pressed?
3rd key is pressed?
4th key is pressed?
6th key is pressed?
9th key is pressed?
Caps Lock
Shift
Print 'p'
Print 'P'
Print 'q'
Print 'Q'
Print 'r'
Print 'R'
Print 's'
Print 'S'
Print '6'
19
20


19
20
8th key is pressed?
No
Yes
The user can input T, U, V, 7 and 8
1st key is pressed?
No
2nd key is pressed?
No
3rd key is pressed?
No
4th key is pressed?
No
6thkey is pressed?
No
9th key is pressed?
Caps Lock
Yes
Shift
Print 't'
Print 'T'
Print 'u'
Print 'U'
Print 'v'
Print 'V'
Print '7'
Print '8'
21
22


21
22
9th key is pressed?
Yes
The user can input W, X, Y, Z and 9
1st key is pressed?
2nd key is pressed?
3rd key is pressed?
4th key is pressed?
6th key is pressed?
9th key is pressed?
Yes
No
Caps Lock
Shift
Print 'w'
Print 'W'
Print 'x'
Print 'X'
Print 'y'
Print 'Y'
Print 'z'
Print 'Z'
Print '9'
End

APPENDIX B: Poster

Faculty of Information and Communication Technology

Project Title: Wireless-based Text Entry System for Handicap

Project Developer: Chooi Wai Lum   Project Supervisor: Mr. Leong Chun Farn

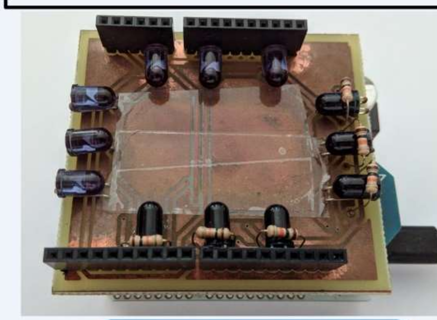
Introduction

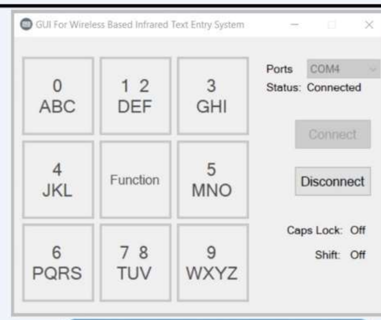
Rapid growth in ICT causes a great demand for digital devices such as smartphones, tablets as well as desktop PC and laptop. To interact with those devices, text entry system is needed such as keyboards. Currently, keyboards provide the most common way to interact with those devices. Unfortunately, keyboards may not be useful for the handicaps to communicate with those devices. Hence, a new text entry system is needed to help the handicaps to input text more easily.

Objectives

- To provide a suitable and low-cost text entry system for the handicaps to type more efficiently
- To develop a portable, lightweight, long operating hour, user friendly wireless text entry system

Discussion and Results





HARDWARE IMPLEMENTATION

1. The user swipe through the 3 x 3 grid key arrangement using Press-Drag-Release algorithm
2. This device will send the input signal to the software to interpret.

SOFTWARE IMPLEMENTATION

1. Act as a software GUI to interpret the input from the hardware
2. Let the user know which positions of the 3 x 3 key arrangement will print which characters and numbers or perform which functions
3. Enter the user's desired character or perform other functions to the computer
4. Can be used as a mouse

Conclusion: This project is developed using Arduino C language for the hardware part and C# language for the software part. Besides, this project will definitely help the handicaps to enter text more efficiently.

APPENDIX C: Turnitin Report