DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN AND CONSTRUCTION OF MIMIC AND SPEECH OPERATED TV CONTROL SYSTEM FOR DISABLED PERSON

by

Hakan ALTINKİLİT

September, 2016

İZMİR

DESIGN AND CONSTRUCTION OF MIMIC AND SPEECH OPERATED TV CONTROL SYSTEM FOR DISABLED PERSON

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for Master of
Science in Electrical and Electronics Engineering

by
Hakan ALTINKİLİT

September, 2016
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN AND CONSTRUCTION OF MIMIC AND SPEECH OPERATED TV CONTROL SYSTEM FOR DISABLED PERSON"** completed by **HAKAN ALTINKİLİT** under supervision of **PROF. DR. GÜLAY TOHUMOĞLU** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Gülay TOHUMOĞLU

Supervisor

Prof.Dr. Uğur Gom

(Jury Member)

Prof.Dr. Mustafa Gündüzalp

(Jury Member)

Prof.Dr. Ayşe OKUR

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Prof. Dr. Gülay TOHUMOĞLU who never lost her faith in me and always supported me through my best and worst days without forgetting his mission to lead me through the way of science and research.

I would also express my gratitude towards my parents whose love and support have been the greatest asset in my life as well in this thesis.

I have to specially thank my colleague, Gökmen AŞCIOĞLU, who has shared with me his great software development experience on mbed NXP LPC1768 microcontroller. Also, I am grateful to all volunteers who gave their time to this project**.**

I also would like to thank my dear Tuğçe AKGÖZ for her never ended support and her outstanding efforts in organizing volunteers

Hakan ALTINKİLİT

DESIGN AND CONSTRUCTION OF MIMIC AND SPEECH OPERATED TV CONTROL SYSTEM FOR DISABLED PERSON

ABSTRACT

In recent years, the problems about physically disabled people have become important day by day. These people live their lives at home most of the time due to their limited movement abilities. There are some research which can make physically disabled people's lives easier in home environment. Foremost among these, is the controlling television which is the window to the world for people who are disabled and confined to bed. The people who are highly disabled to use their hands or confined to bed spend most of their times in front of television that they cannot control it by themselves. It causes that they need a care asistant even while watching TV. To illustrate, if it was considered that 4 percent of adults who are aged 16 and over is hand disabled people in Great Britain, it has become crucial that hand disabled people can use remote control device independently.

In this thesis, it has focused on developing a remote control that can perceive short voice commands or up-down and rigth-left head movements for hand disabled people. While the developed remote control perceives voice commands by a voice recognition module, movement commands are recognized by an accelerometer placed on glasses. Issued voice and movement commands are transferred to television by an infrared transmitter. In this study, it is aimed the easier usage of older or newer televisions along with using infrared communication. Moreover, this system is budget-friendly as well as user-friendly.

Keywords: Voice recognition, movement detection, remote control

ENGELLİLER İÇİN SES VE MİMİK KUMANDALI TV KONTROL SİSTEMİ TASARIMI VE GERÇEKLEMESİ

ÖZ

Son yıllarda fiziksel engelli insanlarla ilgili sorunlar giderek daha da önem kazanmaktadır. Bu insanlar kısıtlı hareket yetenekleri nedeniyle eve bağlı bir yaşam geçirmektedirler. Ev ortamında fiziksel engelli insanların hayatlarını kolaylaştıracak projeler üzerinde çalışmalar yapılmaktadır. Bunların başında eve ve yatağa bağlı olan insanların dünyaya açılan penceresi olan televizyonun kontrolü gelmektedir. Ellerini kullanamayacak derecede engelli olan veya yatağa bağlı olan insanlar gün içerisindeki vaktinin çoğunu karşısında geçirdiği televizyonu kendi başına kontrol edemezler. Bu engelli insanların televizyon izlerken bile bir bakıcıya ihtiyaç duymalarına neden olmaktadır. Örneğin, İngiltere'de 16 yaş üzerindeki yetişkinlerin %4'ünün ellerini kullanamayan bireyler olduğunu düşünülürse ellerini kullanamayan insanların TV kumandasını kendi başlarına kullanabilmeleri önem kazanmaktadır.

Bu tez çalışmasında, ellerini kullanamayan bireyler için kısa ses komutlarını veya aşağı-yukarı ve sağa-sola baş hareketlerini algılayan bir TV kumandası geliştirilmeye odaklanılmıştır. Bu TV kumandası sesli komutları bir ses tanıma ünitesi ile algılarken, hareket komutlarını da bir gözlük üzerine yerleştirilen ivmeölçer ile algılamaktadır. Verilen ses ve hareket komutları bir kızılötesi verici ile televizyona aktarılmaktadır. Bu çalışmada kızılötesi iletişimi kullanımıyla evlerde bulunan eski veya yeni bütün televizyonlarda kolaylıkla kullanılması amaçlanmıştır. Ayrıca bu sistem kullanışlı olduğu kadar düşük maliyetlidir.

Anahtar kelimeler: Ses algılama, hareket algılama, TV kumandası

CONTENTS

Pages

THESIS EXAMINATION RESULT FORM.....ii
ACKNOWLEDGEMENTS.....iii
ABSTRACT.....iv
ÖZ.....v
LIST OF FIGURES.....ix
LIST OF TABLES.....xi

CHAPTER ONE - INTRODUCTION.....1

1.1 Introduction.....1
1.2 Thesis Outline.....4

CHAPTER TWO – MATERIALS AND ITS FEATURES.....5

2.1 Microcontroller Unit.....5
2.1.1 Features.....5
2.1.2 Benefits.....6
2.2 EasyVR Voice Recognition Module 7
2.2.1 Features.....8
2.2.2 Communication Interface.....9
2.3 Microphone.....11
2.4 The Motion Processing Unit.....12
2.4.1 General Description.....12
2.4.2 Features.....13
2.4.2.1 Accelerometer Features.....14
2.4.2.2 General Features.....14
2.5 Communication Interface.....15
2.6 SD Card Module.....16

2.7 Infrared LED.....17

CHAPTER THREE – SYSTEM DESIGN..... 18

3.1 Hardware Implementation 18
3.1.1 PCB Design..... 18
3.1.1.1 Schematic.....18
3.1.1.2 Steps for Exporting Design to Layout.....19
3.1.1.3 Layout.....20
3.2 Software Implementation.....22
3.2.1 Asynchronous Serial Communication Interface.....23
3.2.2 I^2C Interface.....24
3.2.2.1 General I^2C Operation.....24
3.2.2.2 START and STOP Conditions.....25
3.2.2.3 Data Format and Acknowledge.....26
3.2.2.4 Communication.....27

CHAPTER FOUR – INFRARED COMMUNICATION..... 30

4.1 IR Communication.....30
4.1.1 Modulation.....30
4.2 Remote Control Protocols.....31
4.2.1 Sony SIRC IR Protocol.....31
4.2.1.1 Features.....31
4.2.1.2 Modulation.....32
4.2.1.3 Protocol.....32
4.2.2 NEC Protocol.....33
4.2.2.1 Features.....33
4.2.2.2 Modulation.....33
4.2.2.3 Protocol.....34
4.2.3 Philips RC5 protocol.....35
4.2.3.1 Features.....36

4.2.3.2 Modulation.....36
4.2.3.3 Protocol.....36
4.2.4 Samsung Protocol.....37
4.2.4.1 Features.....37
4.2.4.2 Modulation.....37
4.2.4.3 Protocol . …….....38

CHAPTER FIVE - DATA COLLECTION AND ALGORITHMS.....39

5.1 Voice Data Collection.....39
5.1.1 EasyVR Commander.....39
5.1.2 Adding the Speaker Dependent Commands.....41
5.2 Gesture Data Collection.....43
5.3 Head Movement Recognition Algorithm.....44

CHAPTER SIX – CONCLUSIONS AND FUTURE WORK.....52

REFERENCES.....53

LIST OF FIGURES

Pages

Figure 2.1 mbed NXP LPC1768 prototyping board.....5
Figure 2.2 EasyVR voice recognition module.....7
Figure 2.3 A serial data frame demonstrated character “A”.....10
Figure 2.4 A typical connection between MCU-based host and EasyVR.....10
Figure 2.5 Microphone connection.....12
Figure 2.6 System block diagram.....13
Figure 2.7 Orientation of axes.....15
Figure 2.8 Recommended I^2C connections.....16
Figure 2.9 SD card module.....16
Figure 2.10 Infrared LED.....17
Figure 3.1 System block diagram.....18
Figure 3.2 Schematic design of system.....19
Figure 3.3 An example PCB decal 20
Figure 3.4 Top layer of the PCB.....21
Figure 3.5 Bottom layer of the PCB.....22
Figure 3.6 mbed online compiler.....22
Figure 3.7 A serial frame, start, parity, data and stop bits.....23
Figure 3.8 Start and stop conditions.....26
Figure 3.9 Acknowledge on the I^2C bus.....27
Figure 3.10 Complete I^2C data transfer.....28
Figure 3.11 I^2C write to slave device's register.....28
Figure 3.12 I^2C Read from slave device's register.....29
Figure 4.1 A modulated signal is operating the Infrared LED on the transmitter.....30
Figure 4.2 SIRC protocol versions.....31
Figure 4.3 Modulation of SIRC Protocol.....32
Figure 4.4 A Typical pulse scheme of the 12 bit SIRC protocol.....32
Figure 4.5 Oscilloscope view of 12 bit SIRC protocol.....33
Figure 4.6 Modulation of NEC protocol.....34
Figure 4.7 A typical pulse scheme of the NEC protocol 34
Figure 4.8 Oscilloscope view of NEC protocol.....35

Figure 4.9 Repeat code.....35
Figure 4.10 Repeat sequence.....35
Figure 4.11 Modulation of RC5 protocol.....36
Figure 4.12 A typical pulse scheme of the RC5 protocol.....37
Figure 4.13 Modulation of Samsung protocol.....38
Figure 4.14 A typical pulse scheme of the Samsung protocol.....38
Figure 4.15 Oscilloscope view of Samsung protocol.....38
Figure 5.1 Main application window.....41
Figure 5.2 Training channel up command.....42
Figure 5.3 Training.....42
Figure 5.4 Command set.....42
Figure 5.5 Glasses with motion processor unit.....44
Figure 5.6 Head front movement.....45
Figure 5.7 Head back movement.....46
Figure 5.8 Head right movement.....46
Figure 5.9 Head left movement.....47
Figure 5.10 Head front movement algorithm.....50

LIST OF TABLES

Pages

Table 2.1 Specifications of NXP LPC1768 microcontroller.....6
Table 2.2 Pin assignment of EasyVR.....9
Table 5.1 Command list.....40
Table 5.2 Gesture command list.....43
Table 5.3 Head front movement thresholds.....48
Table 5.4 Head back movement thresholds.....48
Table 5.5 Head right movement thresholds.....49
Table 5.6 Head left movement thresholds.....49

CHAPTER ONE
INTRODUCTION

1.1 Introduction

Television has become a part of our lives within its usage were incremented steadily from the time it was invented by John Logie Baird in 1924 (Fisher, D.E., & Fisher, M.J., 1997). In 2015, it was reflected to the statistics that an ordinary television user spent 5 hours and 31 minutes by watching TV (Bureau of Labor Statistics [BLS], 2016).

Considering the TV habits of individuals who suffer from physical disabilities like paraplegia, quadriplegia or losing limbs, due to their limited mobility, many disabled individuals enjoy television, not only as a device of entertainment, also as a window to the world, as well. What's more, 4 percent of adults have finger or hand disabilities in Great Britain in which 29 percent of adults had at least one impairment, aged 16 and over (Vine, Willitts, Farmer & Gunning, 2011).

By increasing the number of TV users in years, remote control technologies have begun to develop correspondingly in order to make people control their TVs from distant. The Zenith Radio Corporation created the first television remote control called ''Lazy Bone'' in 1950. The Lazy Bone could turn a television on-off and change channels. However, it was not a wireless remote control. The Lazy Bone remote control was attached to the television by a bulky cable. Later then Zenith engineer, Eugene Polley created the "Flash-matic" the first wireless TV remote in 1955. The Flash-matic operated by means of four photocells, one in each corner of the TV screen. By the early 1980s, the TV industry moved to infrared remote technology that we are still using (Luplov & Taylor , 2012).

Even though remote controls have been designed for its convenience, they do not give an opportunity for hand or finger disabled individuals due to their many-key structure. Besides, there are alternative remote controls that are usable for those who have finger or hand disabilities. These alternative remote controls make the TV usage possible for even those individuals with limited dexterity, to access at least the basic functions of a television.

Various approaches have been taken to give better user experiences to disabled people such as voice, head motion, muscle, brain activity, eye blink, chin, breath and so on. These systems enhance the quality of life of the disabled by increasing their independence of using their TVs and decreasing the load of the care assistant and increasing the self-confidence of the disabled individual (Craig, Tran, McIsaac & Boord, 2004).

The technical foundation of speech recognition and voice control were laid in 1960's the microprocessor revolution has brought it in our lives (Lindgren & Nilo, 1965). One of the first studies about voice recognition systems on Televisions is Sugaya's Voice Remote Controlled TV Set. He proposed a voice operated TV with 16 command words. The recognition rate is above 95% using the wireless microphone. However, when the microphone in the television set is used, recognition rate decreases due to the ambient noise. Although the system was successful, it was not cost effective (Sugaya, Nishimura, Shimizu & Sugiura, 1979). In the same year voice recognition system was used for severely paralyzed people (Clark, 1979).

In 2000, Hua Jiang described the design of a voice activated environmental control system to aid persons with disabilities, particularly people with quadriplegia, paraplegia, or cerebral palsy. The heart of this system was HM2007 Voice Recognition Module. With this small module, the system has been cost effective and easy to use (Jiang, Han, Scucces, Robidoux & Sun, 2000).

In 2013, Mangrulkar proposed a voice controlled embedded for TV remote. The system was speaker dependent and easy to use. It was working only with Philips televisions due to RC5 code system (Mangrulkar, & Shah, 2013). In the same year Reddy proposed a voice operated TV remote. System was designed with HM2007 voice recognition module (Reddy et al., 2013). These two systems were very similar to each other. Both of them are very cost effective, speaker dependent and easy to use.

Another approach about remote controlled TV systems is gesture recognition. Gestures are generally recognized in two ways; using a camera (Freeman & Weissman, 2004) or a motion processing unit (acceleration sensor and gyro sensor)

(Rekimoto, 1996). Freeman et al proposed a hand gestures controlled television. They use image processing techniques in their study. Image processing based gesture recognition is not cost effective and practical. In these systems a camera and a computer are needed to run software application (Solanki & Desai, 2011). Accelerometers have been used by Sawada to control music performance (Sawada and Hashimoto, 2000) After this study Juha Kela use accelerometers to capture full hand gestures and help users to control TV or other appliances(Kela et al, 2006). Yoshitoshi Murata presents a multipurpose remote control for finger disabled people combining an Android smartphone with a Bluetooth-IR conversion unit that recognizes arm gestures for use with televisions (Murata, Sato & Takayama, 2012). However, these remote studies were not proper for hand disabled individuals.

Voice recognition remote control units are quite easier to use for hand disabled people. Individuals can control their TV units just with only voice commands without the need of any movement (Kawarazaki & Yoshidome, 2012). Unfortunately, the yield of voice recognition remote control units diminishes in crowded and noisy environments. Environmental noise is the main problem about robustness of voice recognition systems (Nakatoh, Kuwano, Kanamori & Hoshimi, 2007).This reason restricts its usage alone.

In literature, there are studies which are combined gesture and voice recognition techniques. In 1988, Eiichi Ito described multi-modal interface that allows use of voice and gesture commands for controlling distributed home appliances used by people with disabilities. It uses head gestures as a pointer to choose desired appliance while voice recognition was being used for control the appliances. Basically, a computer is used as a voice recognition unit. This study is not practical and cost effective.

When the studies above are taken into consideration, in this thesis, it is aimed that to develop a multipurpose remote control for hand disabled people combining voice recognition and gesture recognition. Voice commands which are recognized by microphone, command TV over infrared LED by being processed in EasyVR Voice Recognition Module and LPC1768 micro-controllers. In addition to this, it is provided that a user who wear glasses placed upon it a motion processing unit

(accelerometer and gyroscope), can make the basic process on TV with head movements. Gesture recognition system was designed for helping voice recognition system when environment is noisy and crowded. Thus, voice recognition and gesture recognition techniques will close their gaps by using both of them just in one remote control.

1.2 Thesis Outline

Chapter 1 presents an introduction to project.

Chapter 2 is gives information about materials and its features of the proposed voice and gesture operated remote control system. The chapter 2 also explains how to configure these devices

Chapter 3 interested in hardware and software implementation of the proposed voice and gesture operated remote control system. In hardware implementation section, Mentor Graphics PADS schematic and layout procedures of the system are given shortly. In software implementation section, mbed Compiler, I²C serial communication protocol and asynchronous serial communication protocol (UART) are explained step by step. These serial communication protocols are used to transfer data between microcontroller unit and other devices such as motion processing unit and voice recognition module.

Chapter 4 gives information about IR communication. In this chapter most common IR remote protocols are described with features and modulation techniques.

In chapter 5, data collection process is given step by step.

In the last chapter, the performance of system and conclusion will be given.

CHAPTER TWO
MATERIALS AND ITS FEATURES

The Voice and gesture operated remote control system consist of a microprocessor unit, motion processing unit voice recognition module and IR transmitter.

2.1 Microcontroller Unit

The mbed NXP LPC1768 prototyping board has been chosen as microcontroller because of its useful features for this study. The most important feature is that it allows designers to develop their design faster. The mbed NXP LPC1768 prototyping board lets the designer create prototypes without having to work with low-level microcontroller details, so the designer can work faster.

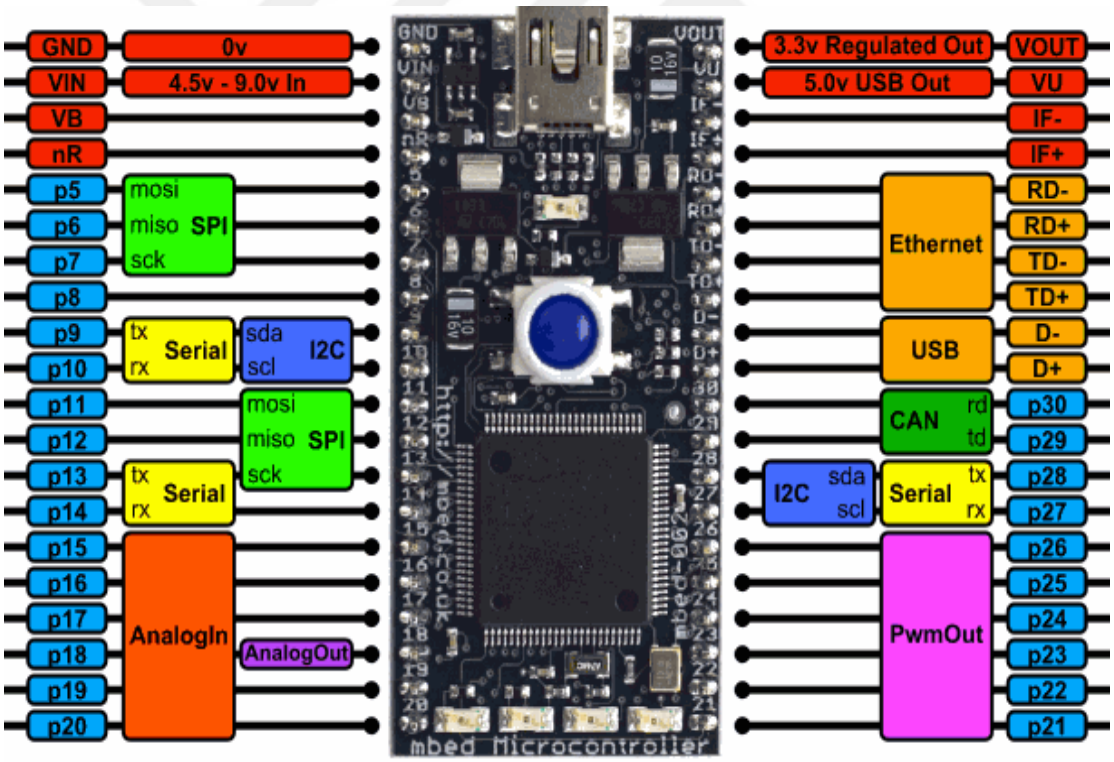


Figure 2.1 mbed NXP LPC1768 prototyping board (NXP, 2009)

2.1.1 Features

- Convenient form-factor : 40-pin DIP, 0.1-inch pitch
- Drag-and-drop programming, with the board represented as a USB drive

- Easy-to-use online tools
    - Web-based C/C++ programming environment
    - Uses the ARM RealView compile engine
    - API-driven development using libraries with intuitive interfaces
    - API-driven development using libraries with intuitive interfaces
- Best-in-class Cortex-M3 hardware
    - 100 MHz ARM with 64 KB of SRAM, 512 KB of Flash
    - Ethernet, USB OTG
    - SPI, I2C, UART, CAN
    - GPIO, PWM, ADC, DAC (See Table 2.1)

2.1.2 Benefits

- Get started right away, with nothing to install
- Get working fast, using high-level APIs
- Explore, test, and demonstrate ideas more effectively
- Write clean, compact code that's easy to modify
- Log in from anywhere, on Windows, Mac or Linux

Table 2.1 Specifications of NXP LPC1768 microcontroller (NXP, 2009)

LPC1768	
ARM Cortex M3 Core	• 100 MHz operation • Memory protection unit • Four power mode: sleep, deep sleep, power down, and deep power down
Memories	• 512 KB of flash memory • 64 KB of SRAM
Serial Peripherals	• 10/100 Ethernet MAC • USB 2.0 full-speed device/Host/ OTG controller with on-chip PHY • Four UARTs with fractional baud rate generation • Two CAN 2.0B controllers • Three SSP/SPI controllers

Table 2.1 Specifications of NXP LPC1768 microcontroller (NXP, 2009) (continue)

	• Three I2C-bus interfaces with one supporting Fast Mode Plus (1Mbit/s data rates) • I2S interface for digital audio
Analog Peripherals	• 12 bit ADC with 8 channels • 10 bit DAC
Other Peripherals	• Ultra-low power (< 1μA) RTC • Four 32-bit general purpose timers

2.2 EasyVR Voice Recognition Module

The EasyVR 2.0 module shown in Figure 2.2 was used for this thesis. EasyVR module has been chosen for its specs like cost effective and easy to find. EasyVR is designed to provide flexible, strong and cost effective speech recognition capabilities to a variety of applications. EasyVR 2.0 voice recognition module works with any device which has an asynchronous serial interface. It can be powered between 3.3V-5V. Commonly it is used with PIC and Arduino boards.

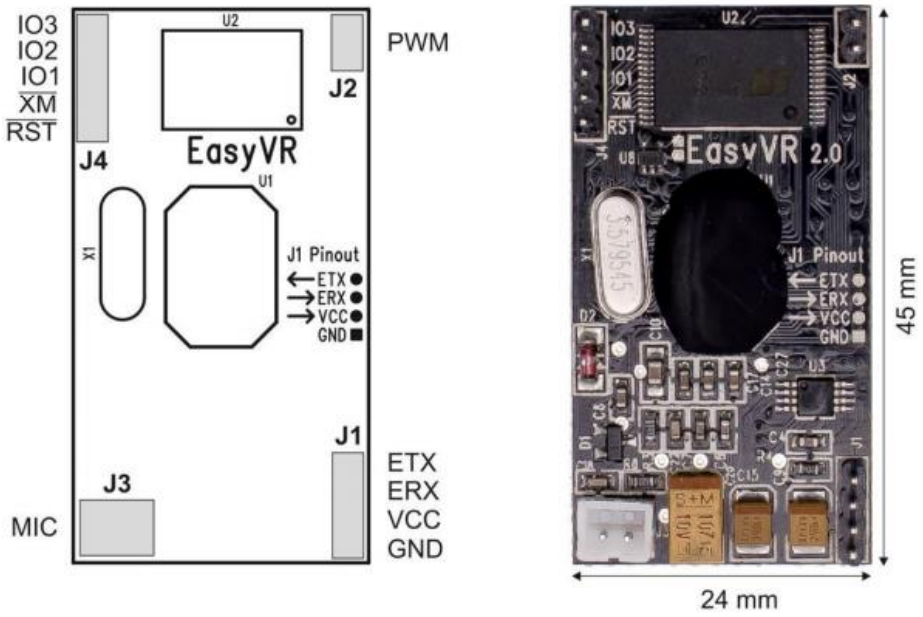


Figure 2.2 EasyVR voice recognition module

2.2.1 Features

- Up to 28 custom Speaker Independent commands are supported. Supported languages can be given as English, German, Italian, Spanish, French, Korean and Japanese
- Up to 32 user determined Speaker Dependent commands, triggers and voice passwords are supported. Custom Speaker Dependent (SD) commands can be saved in any language.
- There are ready to run built-in Speaker Independent (SI) commands to run basic operations in the languages below:
    - Japanese
    - Spanish
    - French
    - German
    - English
    - Italian
- The module has straightforward and easy to use Graphical User Interface to save and collect Voice commands
- The module communicate with host MCU with UART interface
- It is powered at 3.3V or 5V
- Straightforward and strong documented serial protocol simplifies to program through the host MCU board.
- GPIO lines (IO1, IO2, IO3) can be used for controlled the module by new protocol (see Table.2.2)
- The speakers which has 8Ω impedance can be supported by PWM audio output

Table 2.2 Pin assignment of EasyVR

Connector	Number	Name	Type	Description
J1	1	GND	-	Ground
	2	VCC	I	Voltage DC input
	3	ERX	I	Serial Data Receive (TTL level)
	4	ETX	O	Serial Data Transmit (TTL level)
J2	1-2	PWM	O	Differential audio output (can drive 8Ω speaker)
J3	1	MIC_RET	-	Microphone reference ground
	2	MIC_IN	I	Microphone input signal
J4	1	RST	I	Active low asynchronous reset (internal 100K pull-up)
	2	XM	I	Boot select (internal 1K pull-down)
	3	IO1	I/O	General purpose I/O (3.0 VDC TTL level)
	4	IO2	I/O	General purpose I/O (3.0 VDC TTL level)
	5	IO3	I/O	General purpose I/O (3.0 VDC TTL level)

2.2.2 Communication Interface

The EasyVR module is communicating with an asynchronous serial interface. This interface is generally called as UART. The serial interface of EasyVR has features below:

- The default Baud Rate is 9600 Also 19200, 38700, 57600, 115200 baud rates can be adjusted.
- The Serial Frame is consist of 8 Data bits, No Parity bit and 1 Stop bit

The receiver data line (input) is called ERX, the transmitter data line (output) is called ETX.



Figure 2.3 Serial data frame demonstrated character “A” (VeeaR 2015)

EasyVR voice recognition module communicates via an asynchronous serial interface. It works with 3.3V and 5V logical levels, in compliance with the VCC voltage of MCU (See Figure 2.4).

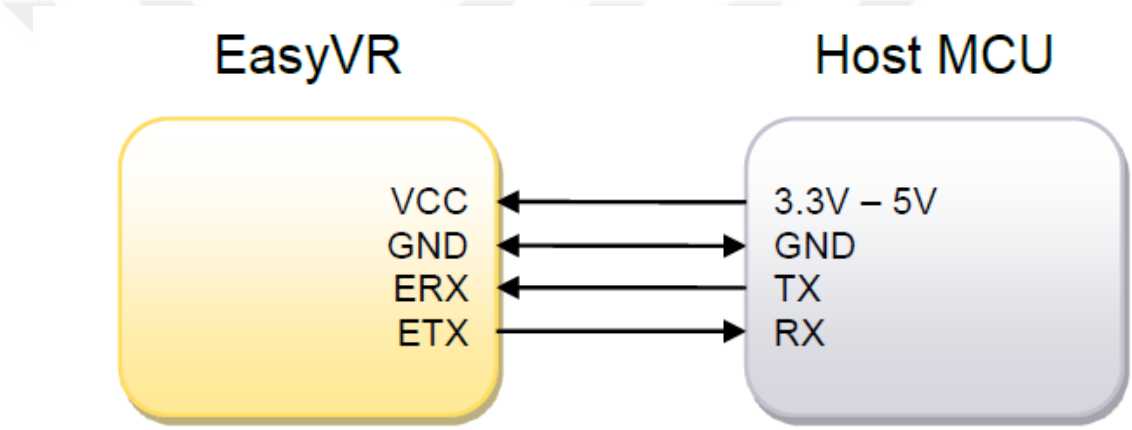


Figure 2.4 A typical connection between an MCU based host and EasyVR

The initial configuration is 9600 baud, 8 data bit, no parity, 1 stop bit when power is on. Afterwards the baud rate can be selected between 9600 and 115200 baud.

The ASCII characters which are printable must be used in the communication protocol. It can be separated in two fundamental groups:

- Command and status characters, in order of the TX and RX lines. They are chosen from lower case letters.
- Command arguments or status details, on the TX and RX lines, containing the range of capital letters.

The commands are sent via the TX line, with zero or more additional argument bytes, receives an answer on the RX line in the form of a status byte followed by zero or more arguments.

Before each byte sent out to the RX line from Easy VR module there is a minimum delay. This delay is set to 20 ms in the beginning and afterwards it can be adjusted in ranges 0-9ms, 10-90ms and 100ms – 1s. These options are suitable for slower or faster master systems and software based serial communication.

The serial interface of EasyVR voice recognition module is software-based. Therefore a short wait time should be required prior to transmitting a character to the module. Particularly, if the master device is too fast to allow the EasyVR module to turn back listening to a new character.

The communication is driving by master. To receive additional status data the master has to acknowledge to every byte of the answer to a command. The answer is cancelled if any other character is received. Thus it is not necessary to read all the bytes of an answer if it is not required.

If all the required arguments of a command aren't sent by host, the module ignores the command, without another notification, thus the host can start to send another command.

After power on, EasyVR voice recognition module automatically enters to sleep mode for lowest power consumption. To start communication, it is necessary to send a character to wake-up the module (VeeaR, 2015).

2.3 Microphone

In this thesis Horn EM9745P-382 omnidirectional electret condenser microphone is used with EasyVR module (See Figure 2.5). The microphone has following features:

- Sensitivity -38dB (0dB=1V/Pa @1KHz)
- Load Impedance 2.2K
- Operating Voltage 3V

- Almost flat frequency response in the range 100Hz – 20kHz

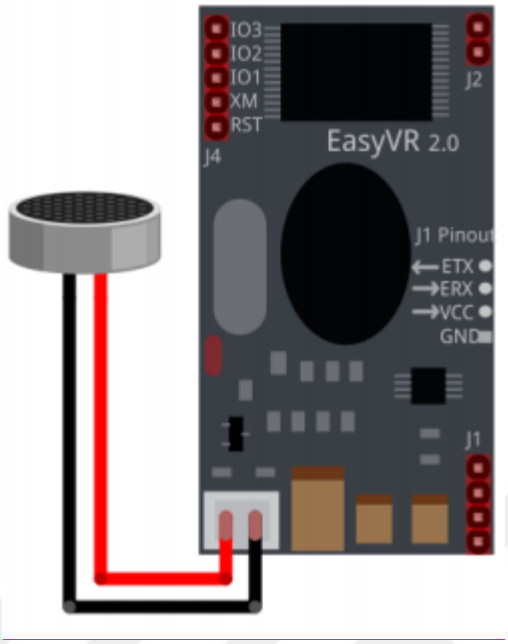


Figure 2.5 Microphone connection

The recognition precision can be reduced if a different kind of microphone is used. The EasyVR voice recognition module is not supported different microphones (Veear, 2015).

2.4 The Motion Processing Unit

2.4.1 General Description

In voice and gesture operated remote control system, MPU6050 Motion Processing Unit which has an embedded 3-axis MEMS gyroscope, a 3-axis MEMS accelerometer, has been chosen because of its features such as measurement range (for user-programmable accelerometer full-scale range of ±2g, ±4g, ±8g, and ±16g and for user-programmable gyroscope full-scale range of ±250, ±500, ±1000, and ±2000°/sec), high resolution (16 bit ADCs), I2C and SPI serial interfacing, ultra-low power (10 µA to 500 µA for accelerometer, 3,6mA for gyroscope), small size. This combination of features makes the MPU6050 an ideal Motion Processor Unit for voice and gesture operated remote control system. Figure 2.6 shows the system block diagram and pin definitions of the MPU6050 (Invensense, 2013).

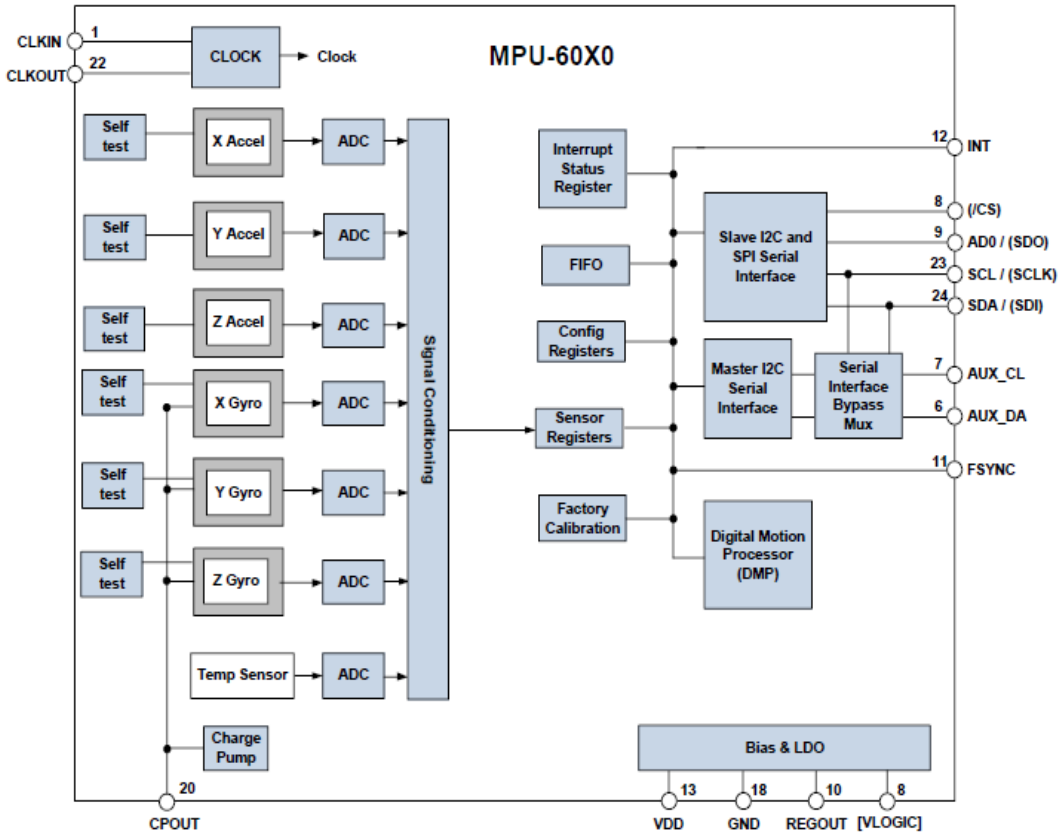


Figure 2.6 System block diagram (Invensense, 2013)

2.4.2 Features

2.4.2.1 Accelerometer Features

MPU-6050 has a triple axis MEMS based accelerometer with following comprehensive features:

- Triple axis accelerometer has digital output
- The full scale range is ± 2g, ± 4g, ± 8g and ± 16g
- 16 bit analog digital converter is integrated
- The standard operating current is 500 µA
- The power saving mode current is 10 µA at 1.25 Hz, 20 µA at 5 Hz, 60 µA at 20 Hz, 110 µA at 40 Hz
- Orientation detection and signaling
- Tap detection
- User programmable interrupts

- High G interrupt
- User self test

2.4.2.2 General Features

The MPU-6050 includes the following additional features:

- Digital Motion Processor chip has 9 axis Motion-Fusion
- It has an auxiliary I^2C bus to communicate with external sensors (for example, magnetometer)
- The operating current is 3.9 mA while six motion sensing axes and the DMP are working
- VDD supply voltage is between 2.375 V and 3.46 V
- Flexible VLOGIC reference voltage works with multiple I^2C interface voltages
- Small and thin QFN package can be used with portable devices: 4 x 4 x 0.9 mm
- The sensitivity cross axis is minimum between the accelerometer and gyroscope axes
- 1 megabyte FIFO buffer is helped to reduce the consume of power by waking up the host processor to read the data in bursts and then go into a low power mode as the Motion processor unit collects more data
- It has a digital output temperature sensor
- Gyroscope, accelerometer and temperature sensor have user programmable digital filter
- 10000g shock tolerance
- 400 kHz faster mode I^2C for connecting with all registers

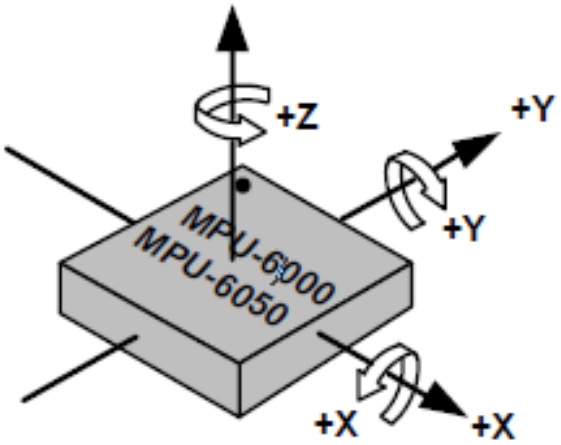


Figure 2.7 Orientation of axes (Invensense, 2013)

2.5 Communication Interface

MPU6050 communication is done with I^2C protocol. I^2C is a two-wire interface consisted of the signals serial data (SDA) and serial clock (SCL) (Figure 2.8). Generally, the lines are open-drain and bi-directional. In a universalized I^2C interface implementation, connected devices can be a master or a slave. The master device puts the slave address on the bus, and the slave device with the matching address acknowledges the master.

The MPU6050 always operates as a slave device when communicating to the CPU, which acts as the master. SDA and SCL lines need pull-up resistors to VDD. The maximum bus speed is 400 kHz.

The slave address of the MPU6050 is b110100X which is 7 bits long. The LSB bit of the 7 bit address is determined by the logic level on AD0 (Pin 9). This allows two MPU6050s to be connected to the same I^2C bus. When used in this configuration, the address of the one of the devices should be b1101000 when AD0 (Pin 9) is logic low and the address of the other should be b1101001 when AD0 (Pin 9) is logic high.

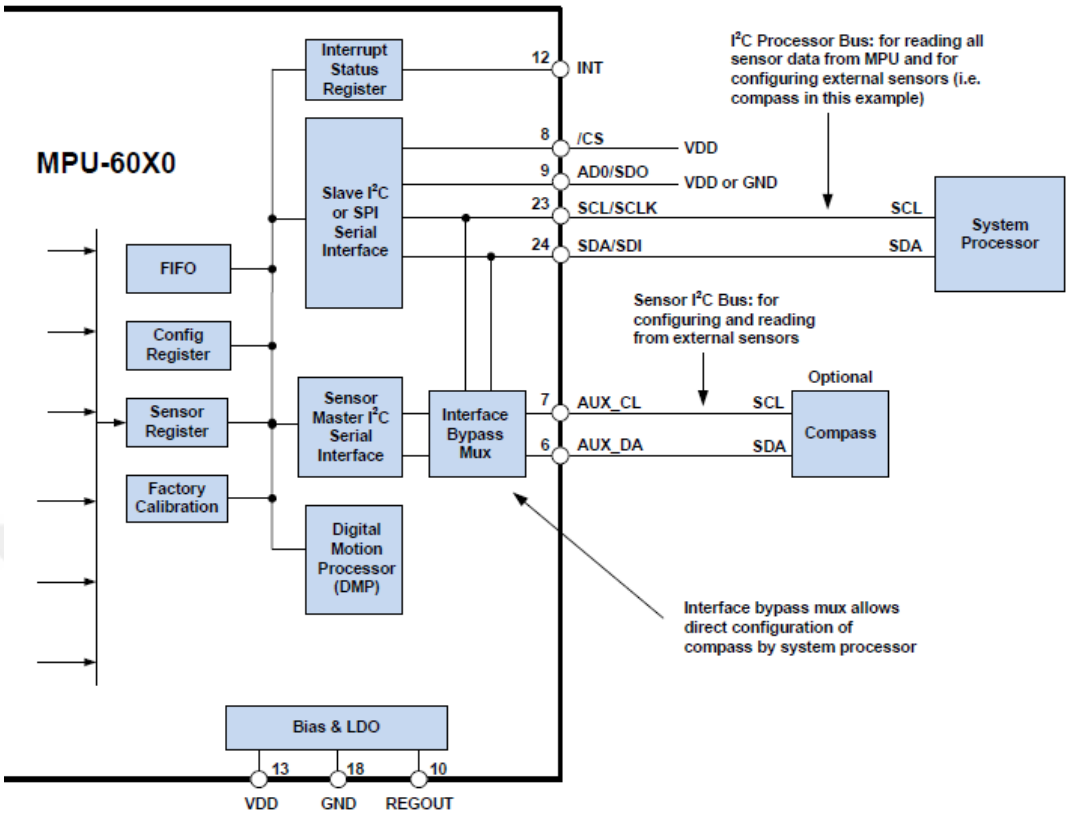


Figure 2.8 Recommended I²C connections (Invensense, 2013)

2.6 SD Card Module

SD card module (Figure 2.9) is not a part of our final prototype. However, this part was used in data acquisition system. SD card acted like a bridge between our data acquisition system and computer, while data was collecting.

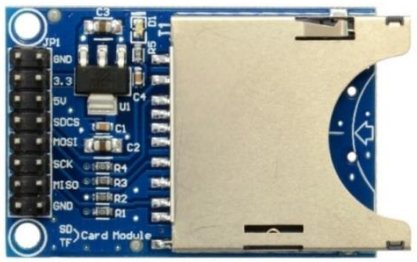


Figure 2.9 SD card module

This module uses Serial Peripheral Interface (SPI) to connect master device. Since it has internal 3.3V regulator, it can be used with 3.3V or 5V systems.

2.7 Infrared LED

In this project a very simple, clear infrared LED has been used. It operates between 840-850nm and work well for generic infrared systems including remote control and touchless object sensing. The forward voltage is 1.5V and the forward current is 50mA.



Figure 2.10 Infrared LED

CHAPTER THREE
SYSTEM DESIGN

3.1 Hardware Implementation

Voice and gesture operated remote control system consists of six parts including voice recognition module, microcontroller unit, motion processing unit, led & battery & button. A block diagram of the system is shown in Figure 3.1.

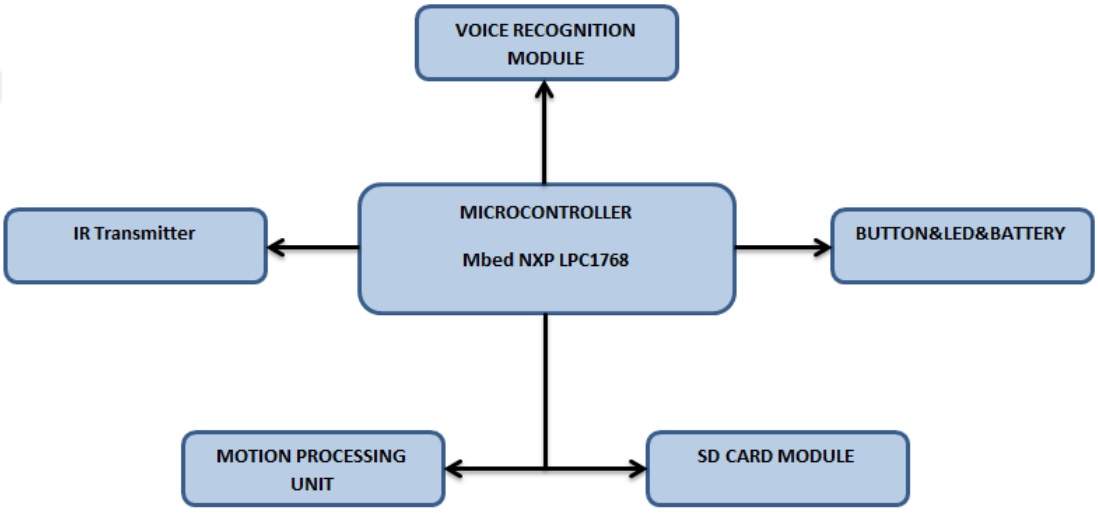


Figure 3.1 System block diagram

3.1.1 PCB Design

3.1.1.1 Schematic

In this study, printed circuit board is designed with Mentor Graphics PADS. It is a suitable tool from Mentor Graphics for the schematic and layout of printed circuit boards (PCBs). In this study, PADS version X has been used to design the circuit board from beginning to end. PADS comprised of two tools. PADS Logic is used for schematic design. PADS Logic is used to generate the schematic diagram of voice and gesture operated remote control system. The other tool is called PADS Layout for designing the component placement and layout of circuits on a PCB.

The subsequent must be performed for drawing schematic diagram of voice and gesture operated remote control system (Figure 3.2);

- Starting a New Schematic Project
- Creating a Schematic Parts Library
- Creating CAE Decals: CAE Decals are the schematic parts symbols that are used in PADS Logic.
- Setting up the Design Rules
- Placing the parts and making the connections

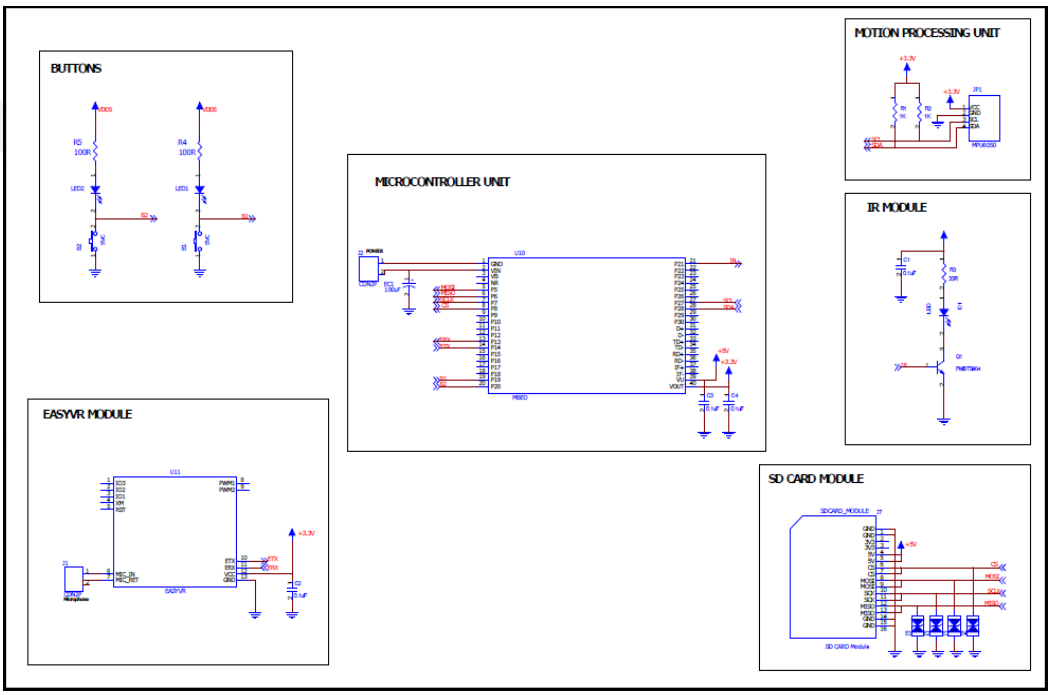


Figure 3.2 Schematic design of system

3.1.1.2 Steps for Exporting Design to Layout

After completing schematic diagram of voice and gesture operated remote control system, several steps must be performed to export the design to Layout:

- Creating PCB Decal Libraries: The physical area that a part covers on PCB is called PCB Decal or footprint. Figure 3.2 shows an example PCB decal. A PCB decal is a set of copper pads that directly comes up to the component pins. This is the one of the most important steps of the PCB design. Any

mismatch between PCB and CAE decal can ruin entire PCB design. For this reason, extra care should be taken while matching decals.

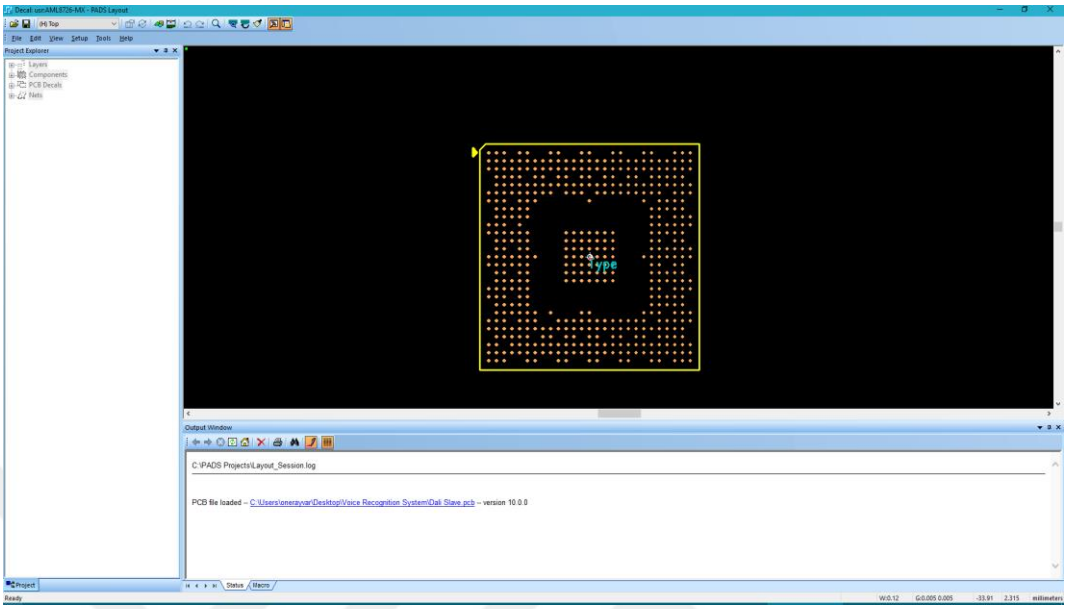


Figure 3.3 An example PCB decal

- Assigning Decals to Parts
- Creating the Netlist: Netlist is a file that has all the decals, parts, part attributes, nets and design rules for the layout design in a format that can be read by Pads Layout. The netlist must be created before exporting design to layout.

3.1.1.3 Layout

To route voice and gesture operated remote control system on Pads Layout, subsequent steps must be performed.

- Importing netlist
- Creating a board outline
- Dispersing and placing components
- Routing nets
- Routing ground, power planes and copper pours
- Checking for errors

- Creating Gerber files: Gerber is a standard file format that is used to communicate design information to manufacture for many types of PCBs. It can be said that Gerber is PDF of electronics world.

PCB of the voice and gesture operated remote control system consists of 2 layers. Placed parts and routed nets are shown on the top (figure 3.4) and bottom (figure 3.5) layers. The following Gerber layers are needed for sending design to fabricate:

TOP: Layer 1 – Routed Nets on Top Side

BOT: Layer 2 – Routed Nets on Bottom Side

SMT: Solder Mask on Top Layer

SMB: Solder Mask on Bottom Layer

SST: Silk Screen on Top Layer

DRD: Coordinates and Size of the Drills

FAB: Fabrication Drawing (Board Dimensions and screw holes)

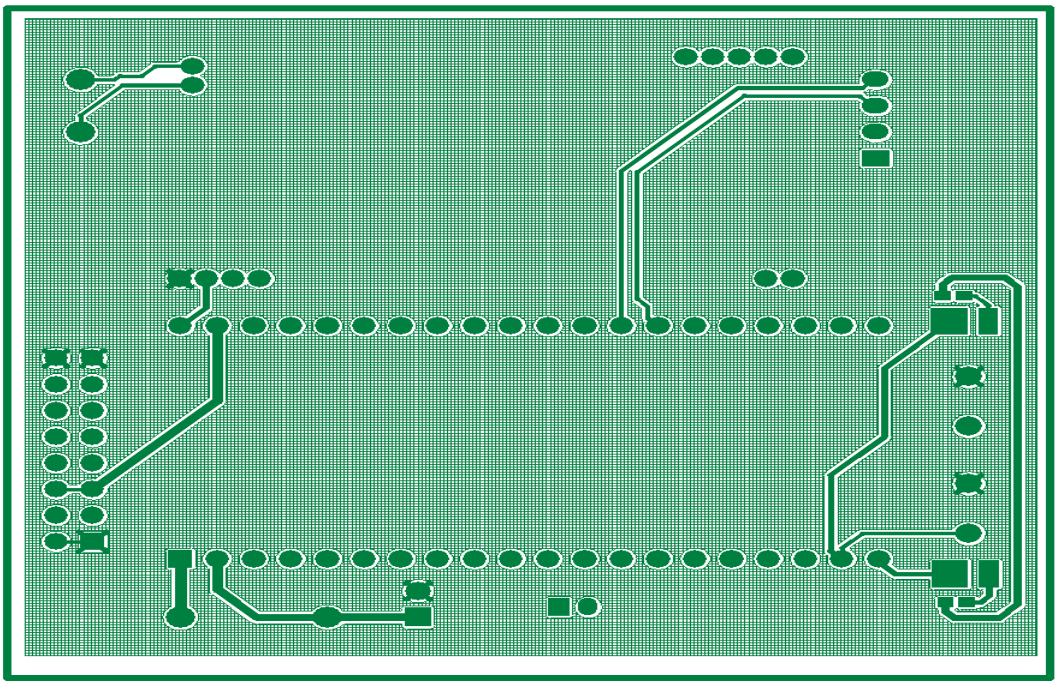


Figure 3.4 Top layer of the PCB

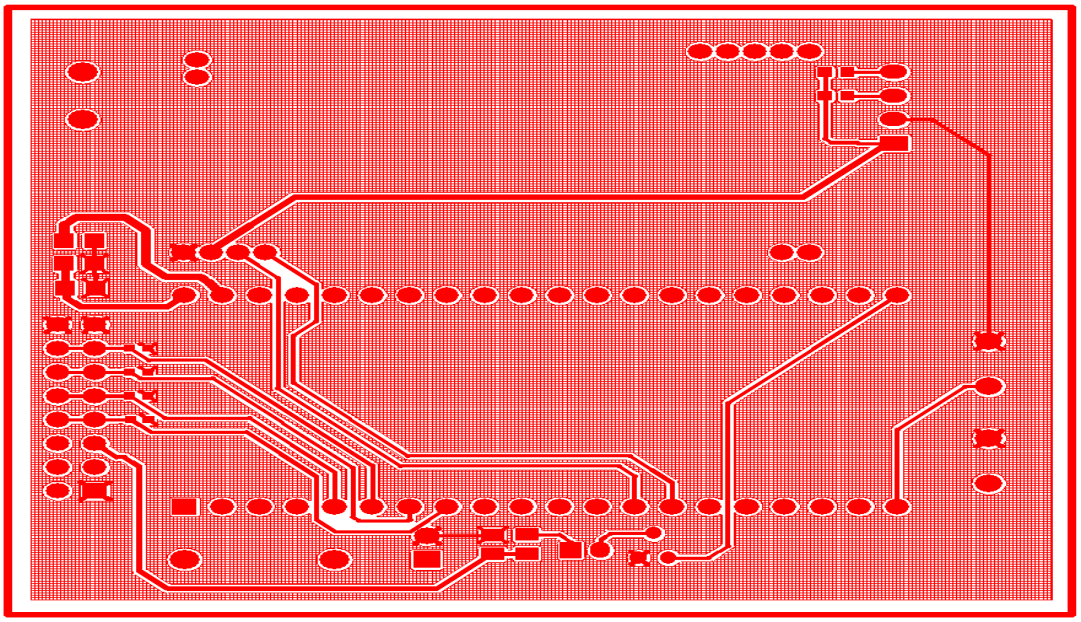


Figure 3.5 Bottom layer of the PCB

3.2 Software Implementation

In this study, all the codes have been written in mbed Compiler. The mbed Compiler runs online, it is not necessary to setup or install the compiler. The codes can be written in C++ language. Afterwards, the codes compiled and downloaded for running on the LPC1768 microcontroller. The compiler is shown in figure 3.6

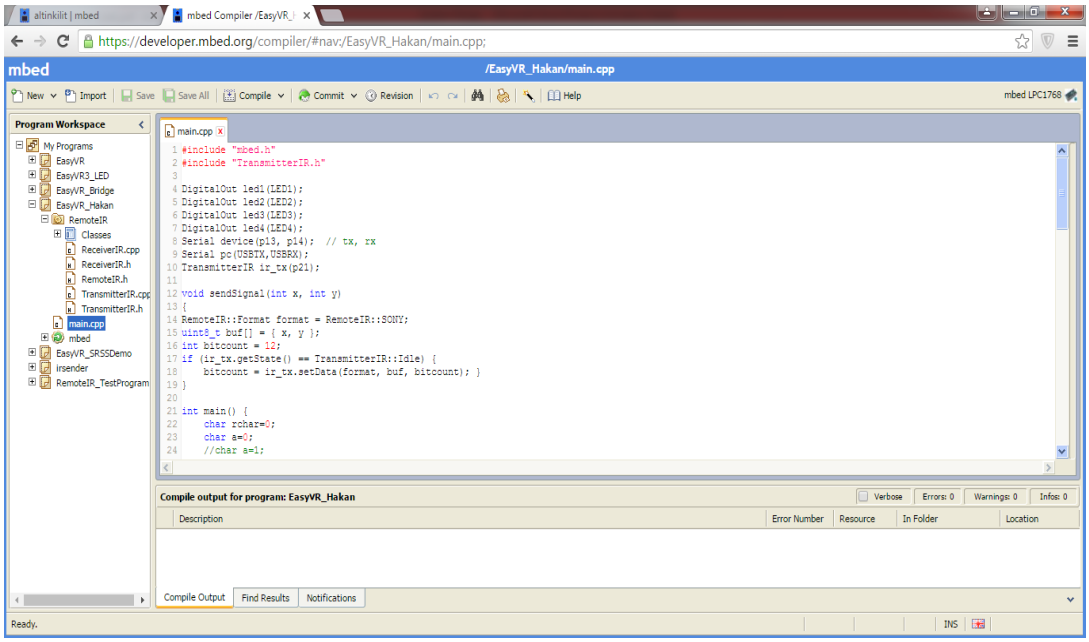


Figure 3.6 mbed online compiler

In this section, serial communication between mbed microcontroller unit and other devices (voice recognition module, motion processing unit and SD card module) is shortly described. The data between MCU and EasyVR Voice Recognition Module have been transferred with asynchronous serial communication protocol (UART), and I^2C communication protocol has been used to transfer data between MCU and motion processing unit.

3.2.1 Asynchronous Serial Communication Interface

The asynchronous serial protocol has several characteristic rules and procedures that help guarantee certain and errorless data transfers. These rules and procedures are:

- Data bits,
- Synchronization bits,
- Parity bits,
- Baud rate.

Because of the diversity of these signaling procedures, the asynchronous serial protocol is pretty much configurable. The crucial part is making certain the two devices on a serial communication bus are set to use the same protocols. The transmitted data block consists of a packet or frame of bits. When the synchronization and parity bits are added to data, the frames are occurred.

The data speed on a serial line specified by the baud rate. Baud rate is generally signified in units of bps (bit per second). If the baud rate is inverted, it can be found out the transmitting time of an only bit. The baud rate value defines the time which the transmitting device holds a serial line high and low.



Figure 3.7 A serial frame, start, parity, data, and stop bits (Sparkfun, n.d.)

The heart of every serial frame is the data bits that are carried. The amount of data in each packet can be set to between 5 and 9 bits. Definitely, the standard data size is a byte (8 bit), but other sizes may also be used (See Figure 3.7).

There are two or three specific bits in every stack of data which named synchronization bits. These are the start bit and the stop bit or bits. The start bit marks the beginning of a packet and the stop bit(s) marks the end. There is always a start bit however the number of stop bits can be arranged one or two. The start bit is represented by an empty data line going logical "1" through logical "0". The stop bits are represented by going from logical "0" to logical "1".

There is a straightforward, low level error checking procedure. It called parity. To generate the parity bit, each bit on the data byte is added. Then looked the summary is even or odd. The evenness of summary defines the parity. The serial bus comprise of two wires. The first wire is used for receiving data and the other one is transmitting. Therefore the serial devices must have two serial pins: the first one is the receiver (RX) and the second one is transmitter (TX). The serial interface can be full duplex or half duplex. When the both devices can transmit and receive data at the same time, this type is called Full Duplex. When the serial devices wait their turn to transmitting and receiving, this type is Half Duplex (Sparkfun, n.d.).

3.2.2 I^2C Interface

I^2C is a serial bus short distance protocol developed by Philips Semiconductor about forty years ago to enhance communication between the core CPU on the board and various other ICs around the core (nxp, 2003).

3.2.2.1 General I^2C Operation

The I^2C bus is a standard bidirectional interface for communicating with master and slave devices. Each device on the I^2C bus has a unique device address to discriminate between other devices that are on the same I^2C bus. And the devices can have one or multiple registers where data is written, read, or stored. The physical I^2C interface comprises of the SCL (Serial clock) and SDA (Serial data) lines. Both SDA and SCL lines must be connected to VCC through a pull-up resistor. Data

transfer initiated only when the bus is unused. A bus is supposed unused if both SDA and SCL lines are high after a STOP condition (Valdez & Becker, 2015).

Following procedure is for a master to access a slave device:

- When a master wants to send data to a slave:
    - Master-transmitter sends a START condition and addresses the slave-receiver
    - Master-transmitter sends data to slave-receiver
    - Master-transmitter terminates the transfer with a STOP condition
- When a master wants to read or receive data from a slave:
    - Master-receiver sends a START condition and addresses the slave-transmitter
    - Master-receiver sends the requested register to read to slave-transmitter
    - Master-receiver receives data from the slave-transmitter
    - Master-receiver terminates the transfer with a STOP condition

3.2.2.2 START and STOP Conditions

I^2C communication is initiated by the master device sending a START condition and completed by the master sending a STOP condition. START condition is defined as a high-to-low transition on SDA line while SCL is high. STOP condition is defined as a low-to-high transition on SDA line while SCL is high (See Figure 3.8).

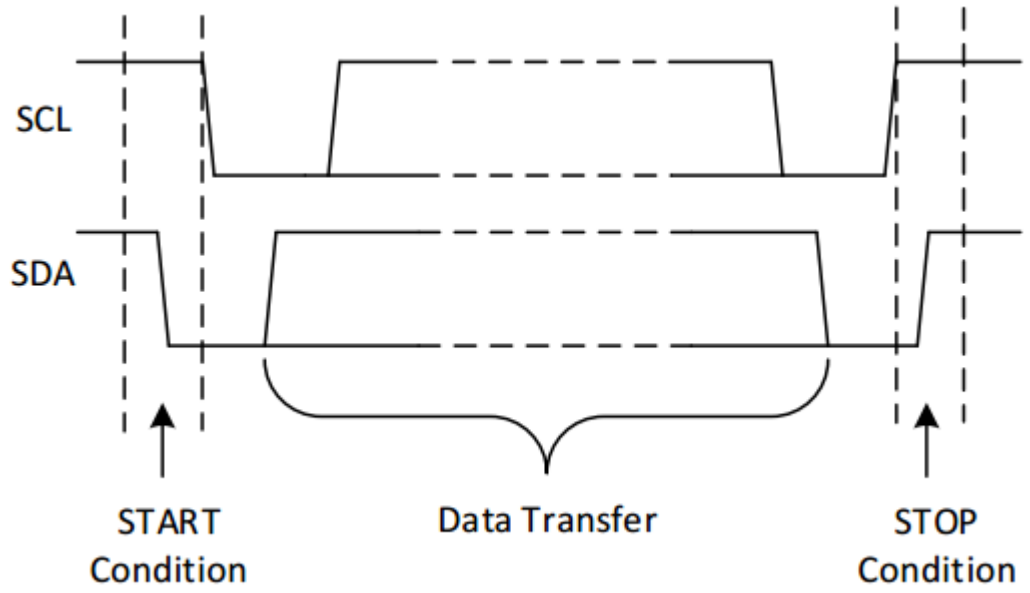


Figure 3.8 Start and stop conditions (Valdez & Becker , 2015)

3.2.2.3 Data Format and Acknowledge

During each clock pulse of Serial Clock, one data bit is transferred. One data byte is consisted of eight bits on the Serial Data line. Each transferred byte must be followed by ACK (acknowledge) signal. Master generates the clock for the ACK signal, while the actual ACK signal is generated by receiver device as pulling down SDA and holding it low during the high portion of the ACK clock pulse.

If a slave device is busy and not available transmit or receive another data until some other task has been performed, SCL is held LOW. So master is forced into a wait state. Data transfer returns to normal when the slave is ready.

It can hold SCL LOW, thus forcing the master into a wait state. Normal data transfer resumes when the slave is ready, and releases the clock line (see Figure 3.9).

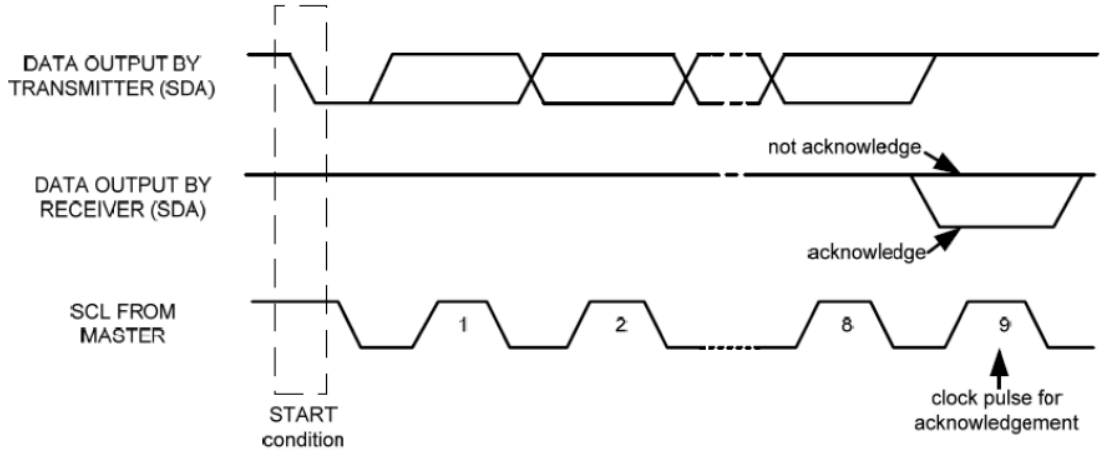


Figure 3.9 Acknowledge on the I^2C bus (InvenSense Inc., 2010)

3.2.2.4 Communication

The communication starts with the START condition (S), after START condition (S), 7 bit slave address and a read/write bit are sent by master. The read/write bit points out whether the master is writing data to slave device or receiving data from it. Afterwards the master device releases SDA line, then waits for the ACK (acknowledge signal) from the slave. Each transferred byte must be followed by an ACK bit. To acknowledge, SDA line is pulled LOW and kept LOW by slave device, for the high period of SCL line. Data transmission is always ended by the master with a STOP condition (P) (see figure 3.10). STOP condition is setting free the communication line. Stop condition is defined by A LOW to HIGH transition on the serial data (SDA) line while serial clock (SCL) is HIGH. All serial data line changes occur when serial clock is low with exception of start, stop conditions.

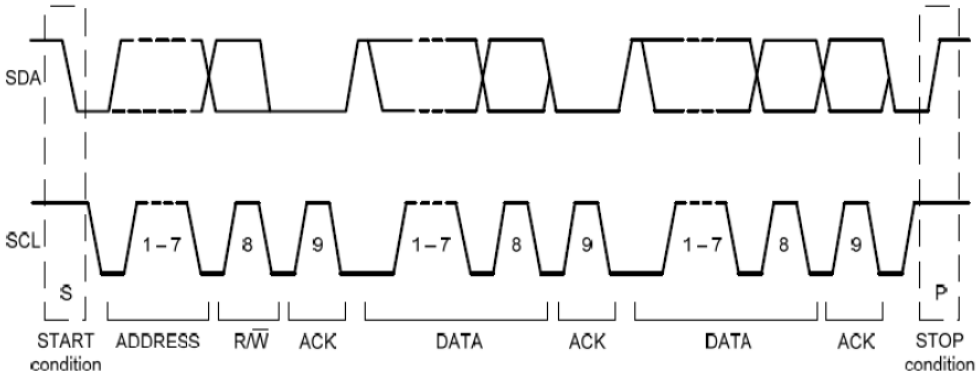


Figure 3.10 Complete I^{2}C data transfer (InvenSense Inc., 2010)

To write our motion processing unit's registers, first the master (LPC1768) transmits the start conditions (S). After start condition (S) I^{2}C address and the write bit (0) are transmitted. At the 9th clock cycle, the motion processing unit acknowledges the transfer. Thereafter the master (LPC1768) releases the register address on the bus. Then the accelerometer acknowledges the register address, the master (LPC1768) release the register data onto the bus. ACK signal follows the register data. Data transfer completed by the stop condition (P) (see figure 3.11).

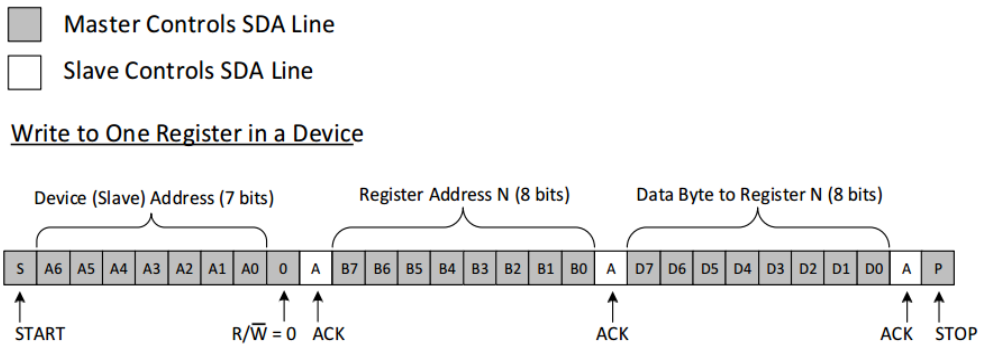


Figure 3.11 I^{2}C write to slave device's register (Valdez & Becker ,2015)

Reading from our accelerometer registers is very similar to writing, but it consists of some extra steps. First, the master (LPC1768) transmits the start condition (S). After start condition (S), I^{2}C address and the write bit (0) are transmitted. At the 9th clock cycle, the accelerometer acknowledges the transfer. Thereafter the master (LPC1768) releases the register address on the bus that is going to be read. Once the accelerometer acknowledges this register address, the master device (LPC1768) will send a START condition again, followed by the slave address that is going to be read

with the R/W bit set to 1 (signifies to read). As a result, the slave device acknowledges the read request, and sends the data. At the end of every byte of data, the master (LPC1768) sends an acknowledge signal (ACK) to the slave (accelerometer), letting the slave know that it is ready for more data. Once the master has received the number of bytes it is expecting, it sends a not acknowledge signal (NACK), to the slave device to terminate communications. Lastly, the master sends a STOP condition.

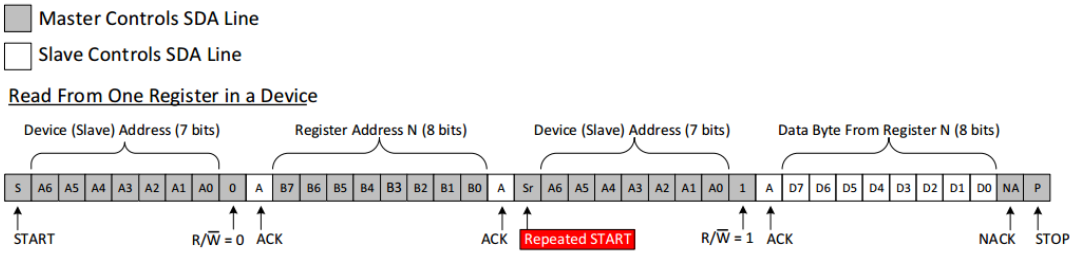


Figure 3.12 I^{2}C read from slave device's register (Valdez & Becker , 2015)

CHAPTER FOUR
INFRARED COMMUNICATION

4.1 IR Communication

Infrared communication is a common, cheap and easy to apply wireless communication technology. The main difference between IR and visible light is the wavelength. IR light has a little longer wavelength and it is undetectable with human eye. This feature makes IR light perfect for wireless communication. For example, when someone hit a button on the TV remote, the IR LED turns on and off 38,000 times in a second to transmit information to the IR receiver on the TV.

4.1.1 Modulation

IR sources are all around us, the sun, light bulbs or anything radiates heat. IR light need to modulate to avoid interfere other IR sources, while using it in communication. With modulation Infrared light source blinked at a certain frequency as the IR receiver previously tuned.

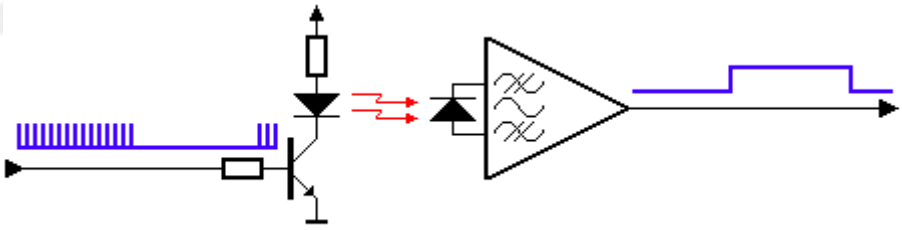


Figure 4.1 a modulated signal is operating the Infrared LED (San Bergmans, n.d.)

In the Figure 4.1 it can be seen a modulated signal is operating the Infrared LED on the transmitter. On the right side of figure detected signal can be seen. Between 30 kHz and 60 kHz carrier frequencies are commonly used in consumer electronics. The most common carrier frequency is 38 kHz.

4.2 Remote Control Protocols

There are various IR remote controller protocols currently used. Consumer product manufacturers use their own remote controller protocols. The main reason of this to avoid to interfere with other manufacturers' remote signals. Philips's RC5, Sony's SIRC and NEC's are well-known protocols.

4.2.1 Sony SIRC IR Protocol

There are three different versions of Sony IR protocol: 12, 15 and 20 bit versions. 7 bits are reserved for the command for all three versions. The 12-bit version has 5 bits reserved for the address, while the 15-bit version has 8 bits reserved for the address. The 20-bit version has a 7-bit command and a 5-bit address length. The extra 8 bits, which follow the address, are called the extended bits (See Figure 4.2).

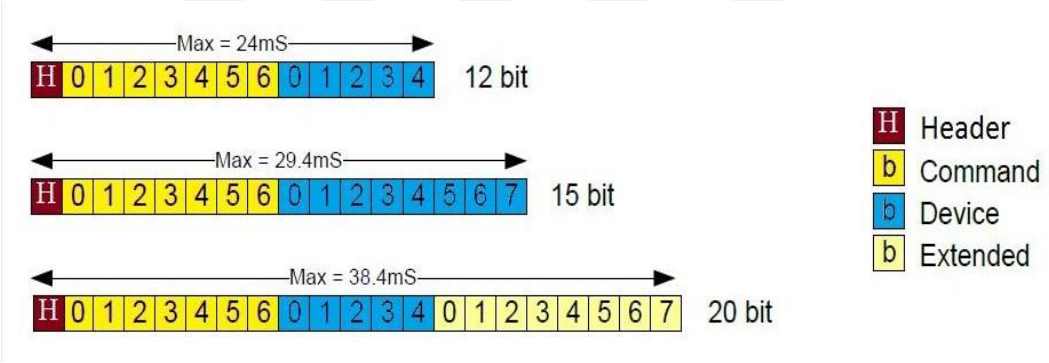


Figure 4.2 SIRC protocol versions

4.2.1.1 Features

- 12-bit version, 7 command bits, 5 address bits.
- 15-bit version, 7 command bits, 8 address bits.
- 20-bit version, 7 command bits, 5 address bits, 8 extended bits.
- Pulse width modulation is used
- Carrier frequency is selected 40 kHz
- Bit times are 1.2 ms and 0.6 ms.

4.1.1.2 Modulation

Sony's SIRC remote protocol is based pulse width encoding of the bits. Logical "1" is represented with 1.2ms burst and 0.6ms space, while logical "0" is represented with 0.6ms burst and 0.6ms space on the 40 kHz carrier frequency (See Figure 4.3).

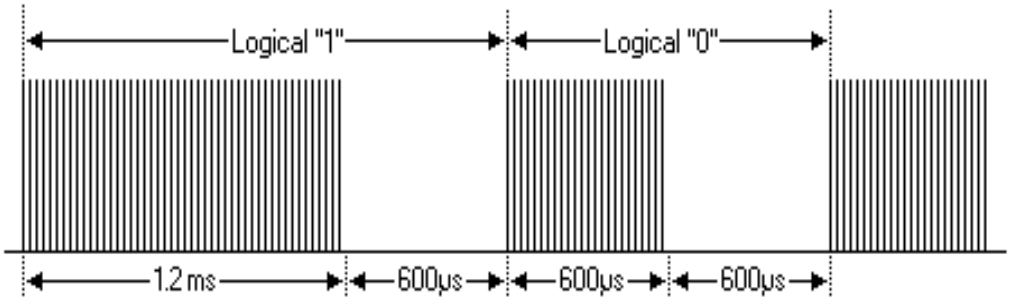


Figure 4.3 Modulation of SIRC protocol (San Bergmans, n.d)

4.1.1.3 Protocol

The Figure 4.4 demonstrates a typical pulse scheme of the 12-bit SIRC remote protocol. The least significant bit is transmitted first in this protocol. The code always starts with the header of 2.4 ms wide burst, a standard space of 0.6 ms follows the burst. Header is also helped to set the gain of the receiver. Seven command bits that shows the action to be performed follows the header. And these command codes are followed by five address bits that shows which device is going to act according to command code (See Figure 4.5)

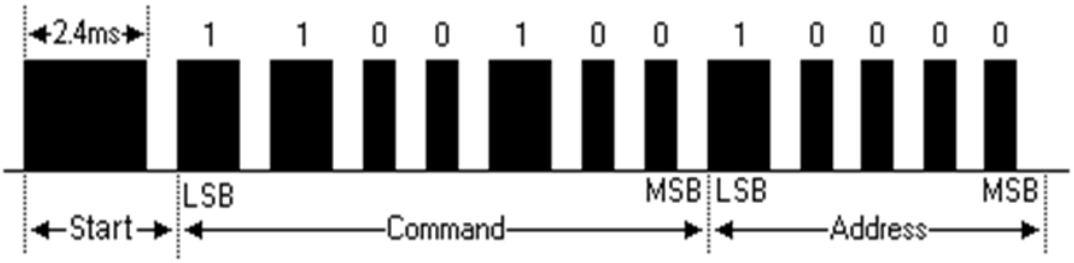


Figure 4.4 A Typical pulse scheme of the 12 bit SIRC protocol (San Bergmans, n.d)

As long as the button on the remote control is pressed, commands are repeated every 45ms.

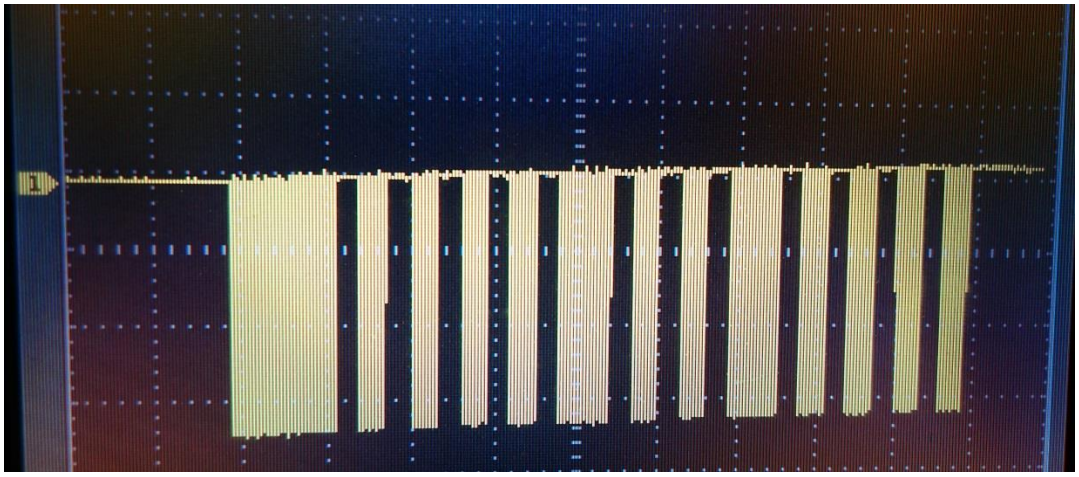


Figure 4.5 Oscilloscope view of 12 bit SIRC protocol

After 12 or 15 bits the receiver must wait at least 10ms to make sure that no more pulses follow. That way the receiver can decide which one of the 3 possible SIRC protocols it is receiving.

4.2.2 NEC Protocol

This protocol was developed by a semiconductor company. The NEC code uses bursts at a carrier frequency of 38 kHz. The NEC code starts the transmission a burst with 9 ms length, followed by 4.5 ms pause and then the data word.

4.2.2.1 Features

- 8-bit address and 8-bit command length.
- Extended mode available. This mode is doubling the address size.
- Commands and addresses are transmitted two times for trustworthiness.
- Pulse distance modulation is used.
- Carrier frequency is selected 38 kHz
- 1.125ms bit time is for logical “0” and 2.25ms for logical “1”.

4.2.2.2 Modulation

The NEC IR transmission protocol is based pulse distance encoding of the message bits. Every pulse burst is a 560 μs long at a carrier frequency of 38 kHz. The bit time for a logical "1" is 2.25ms and the bit time for a logical "0" is only 1.125ms half-length of logical “1” (See Figure 4.6).

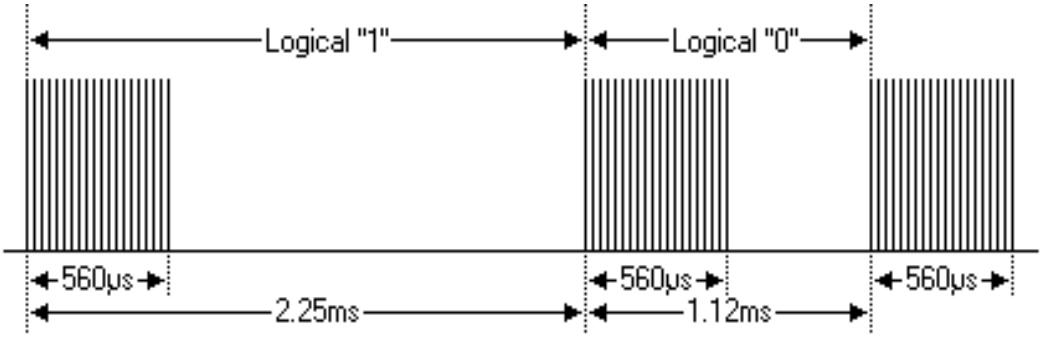


Figure 4.6 Modulation of NEC protocol (San Bergmans, n.d)

4.2.2.3 Protocol

The Figure 4.7 demonstrates a typical pulse scheme of the NEC protocol. Firstly, the least significant bit is transmitted in this protocol. The code is always starts with a 9ms AGC burst. This AGC burst was used to adjust the gain of the older Infrared receivers. Afterwards a 4.5 ms space follows the AGC burst. Then address and command bits follow the 4.5 ms space (See Figure 4.8)



Figure 4.7 A typical pulse scheme of the NEC protocol (San Bergmans, n.d.)

The Address and command bits are sent two times. The first time bits are sent as they are. The inversions of bits are sent in the second time. This technique is used for confirmation of the received message. If reliability is not important, the inverted values can be ignored (Vishay, 2013).

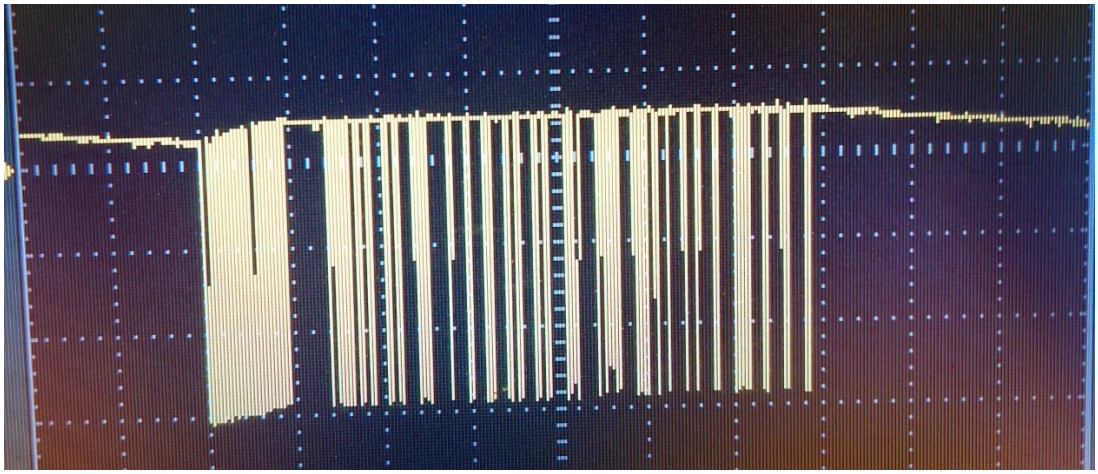


Figure 4.8 Oscilloscope view of NEC protocol

The repeat code is consists of a 9 ms AGC pulse, a 2.25 ms space and a 560 µs burst (See Figure 4.9).

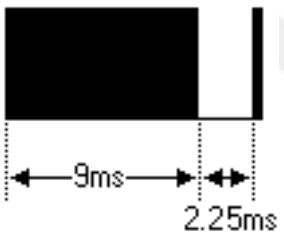


Figure 4.9 Repeat code

As long as the button on the remote control is pressed, the command is transmitted only one time but a repeat code transmitted every 110ms (Figure 4.8).

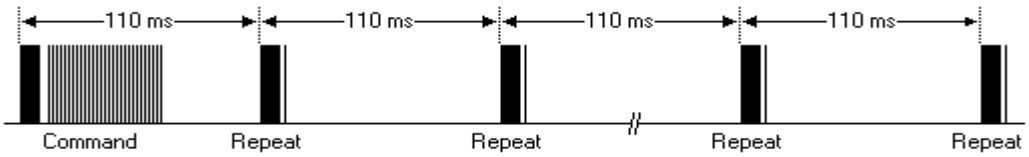


Figure 4.10 Repeat sequence (San Bergmans, n.d)

4.2.3 Philips RC5 Protocol

Philips' RC5 IR transmission protocol is one of the most popular protocols to control TV, home appliances, entertainment systems and car security systems. This

protocol invented by Philips and all remote controllers of Philips use this protocol. The protocol has the capacity to send 2048 different commands.

4.2.3.1 Features

- Consists of 5 bit of address and 6 bit of command length.
- Manchester coding (as known as Bi phase coding).
- Carrier frequency is selected 38 kHz
- Constant bit time of 1.778ms (64 cycles of 36 kHz).
- Manufacturer Philips.

4.2.3.2 Modulation

The RC5 protocol is based Manchester coding system. It uses 36 kHz Infrared carrier frequency. Each bit is 1.778ms length in R5 protocol. 0.889ms of bit time is a burst of the 36 kHz carrier frequency. The other 0.899ms of bit time is blank. The logical “1” is corresponded by a burst in second half of the bit time while the logical “0” is represented by first half (See Figure 4.11).

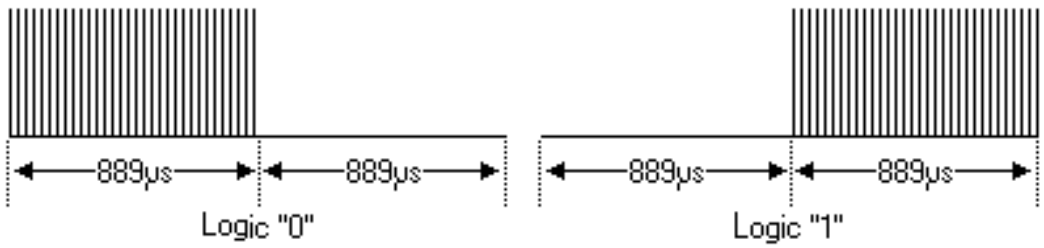


Figure 4.11 Modulation of RC5 protocol (San Bergmans, n.d.)

4.2.3.3 Protocol

The code has 14-bit data word, 2 start bits, 1 control bit, 5 address bits, 6 command bits. First two bursts are start bits at the same time they are always logical “1” to control the AGC levels (Auto gain control) in the receiver IC. The third bit is control bit. This bit is reversed all the time user released the key. It is used to recognize the key which is holding down, or is pressed repeatedly. The five address bits follow the control bit and represent which equipment is being controlled. Finally

the six command bits comprise the control information. Figure 4.12 demonstrates a typical pulse scheme of the RC5 protocol

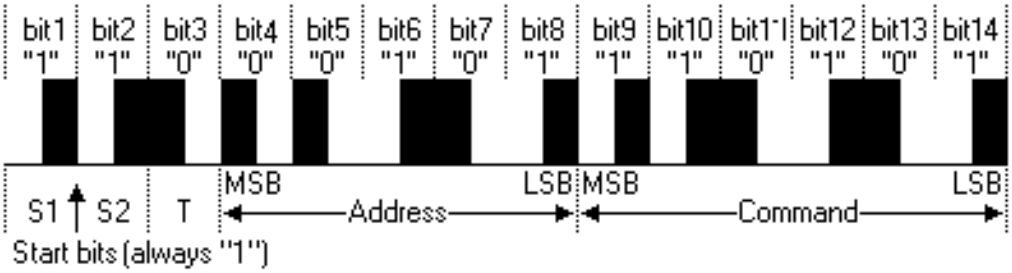


Figure 4.12 a typical pulse scheme of the RC5 protocol (San Bergmans, n.d)

Recently Philips began to use a new protocol named RC-6 that has more convenient than RC-5.

4.2.4 Samsung Protocol

Samsung IR transmission protocol is very similar with NEC IR protocol. The Samsung code uses bursts at a carrier frequency of 38 kHz. The Samsung code starts the transmission a burst with 4.5ms length, followed by 4.5 ms pause and then the data word.

4.2.4.1 Features

- 8-bit address and 8-bit command length.
- Commands and addresses are transmitted two times for trustworthiness.
- Carrier frequency is selected 38 kHz
- 1.125ms bit time is for logical “0” and 2.25ms for logical “1”.

4.2.4.2 Modulation

Each pulse burst is a 560µs long at a carrier frequency of 38 kHz. The bit time for a logical "1" is 2.25ms and the bit time for a logical "0" is only 1.125ms half-length of logical “1” (See Figure 4.13).

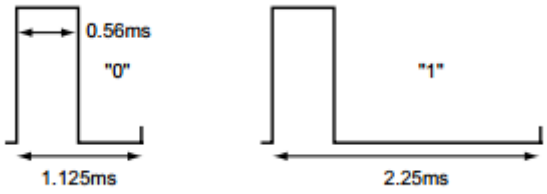


Figure 4.13 Modulation of Samsung protocol (Samsung, 2008)

4.2.4.3 Protocol

The code has 32-bit data word after leader bit containing 8 address bits, 8 address bits again, 8 command bits and inverse of 8 command bits. The code is always starts with a 4.5ms AGC burst. This AGC burst sets the gain of the IR receiver. Afterwards a 4.5 ms space follows the AGC burst. Then address and command bits follow the 4.5 ms space. Carrier frequency of transmission is 38 kHz (See Figure 4.14 & Figure 4.15)

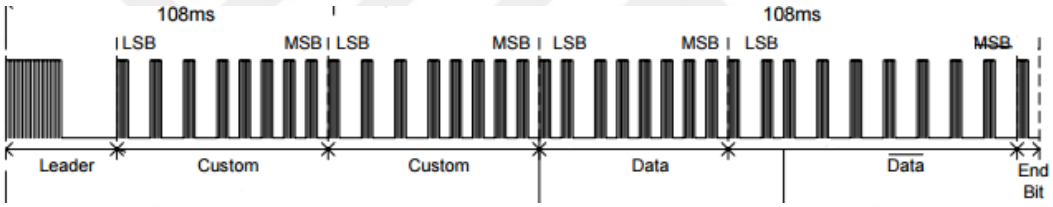


Figure 4.14 A typical pulse scheme of the Samsung protocol (Samsung, 2008)

As long as the button on the remote control is pressed, the command is repeated every 60ms.

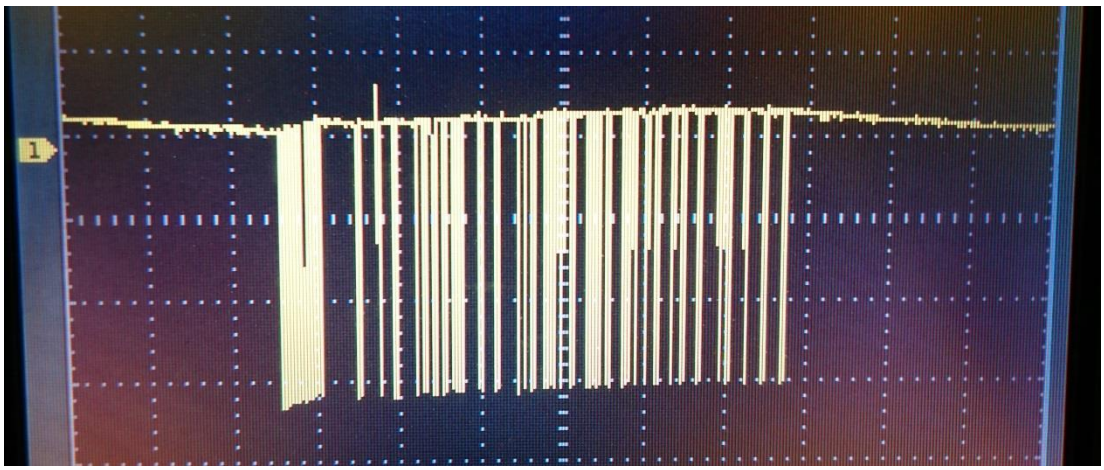


Figure 4.15 Oscilloscope view of Samsung protocol

CHAPTER FIVE
DATA COLLECTION AND ALGORITHMS

5.1 Voice Data Collection

In this study, twenty voice commands have been used. Since EasyVR Voice Recognition Module recognized English language better, all the determined commands are English. Commands and its functions can be seen in Table.5.1. EasyVR Commander Program has been used as a bridge between EasyVR and computer.

5.1.1 EasyVR Commander

EasyVR Commander is software that is used for configuring the EasyVR Voice Recognition Module connected to computer by using the microcontroller host board. The inbuilt Speaker Independent (SI) Voice Commands can be grouped with this program or new Speaker Dependent (SD) Commands can be created.

When EasyVR module connects to the computer by using the microcontroller unit, EasyVR Commander reads and synchronizes all the custom commands and groups, which are previously stored on the nonvolatile memory of the module.

Since the voice recognition function of the module operates on only one command group at same moment, it is necessary to group together all of the commands which is used at the same moment.

There are four kinds of commands in the EasyVR Commander (see Figure 5.1 and Figure 5.4):

- Trigger - is a special group where users have the built-in Speaker Independent trigger word "Robot" and users may add one user-defined SD trigger word. Trigger words are used to start the recognition process
- Group - User can define their own Speaker Dependent (SD) command words.
- Password – it is a specific group for "vocal passwords", this group uses Speaker Verification (SV) technology

- Wordset – The prerecorded set of Speaker Independent commands (In Figure 5.1 the Wordset 1 can be seen)

Table 5.1 Command list

Commands	**Function Performed**
Power	Power ON / Power OFF
Channel Up	Change the Channel +
Channel Down	Change the Channel -
Previous Channel	Tune the Previous Channel
Volume Up	Increase the Volume level
Volume Down	Increase the Volume level
Mute	Mute
Menu	Open Menu
HDMI	Open HDMI
Return	Return to TV
One	
Two	
Three	
Four	
Five	
Six	
Seven	
Eight	
Nine	
Zero	

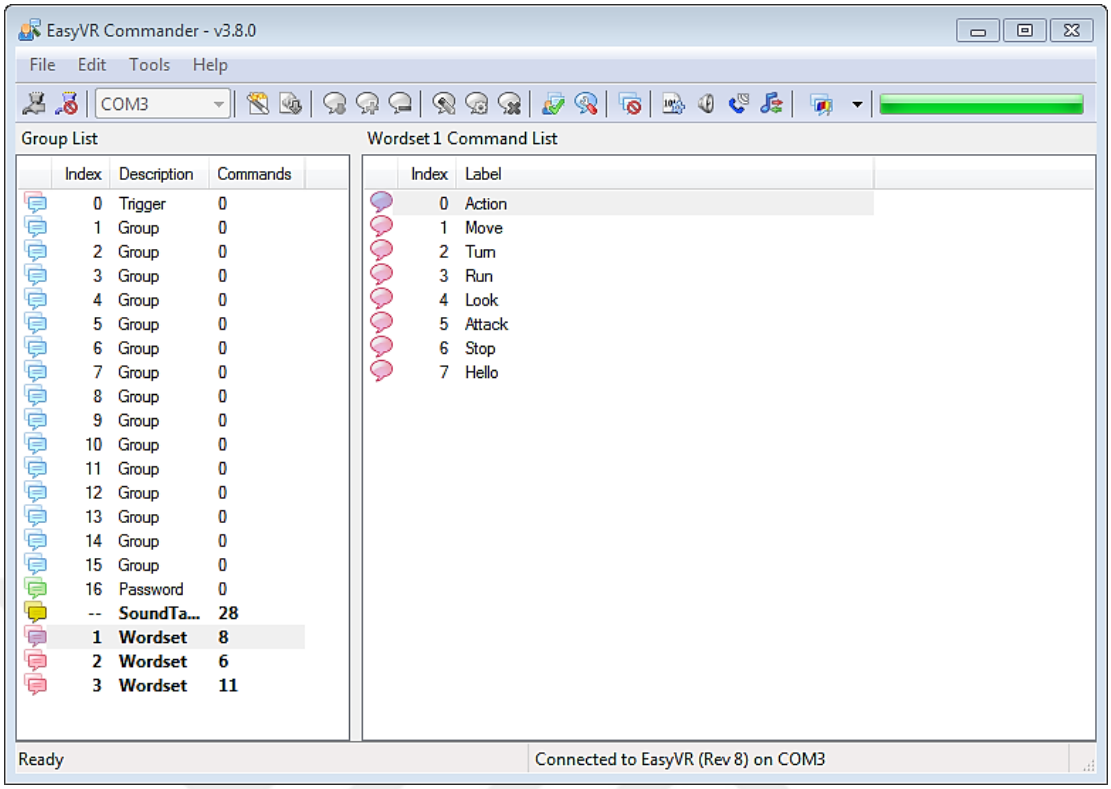


Figure 5.1 Main application window

5.1.2 Adding the Speaker Dependent Commands

All the commands that will be used in voice and gesture operated remote control system should be added in one group. A new speaker dependent command can be added by users. With this purpose, a new group must be selected from group list that consists of other commands. Then, process is completed by pressing on “Add Command” tool bar icon.

Later, a label is given to the commands and they are adjusted twice according to user’s voice. Along this course, the user will be guided when the "Train Command" action is started. Training process of channel up command was displayed in Figure 5.2 and Figure 5.3.

.

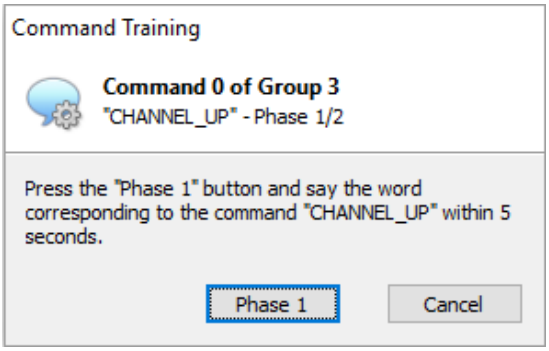


Figure 5.2 Training channel up command

After “Phase 1” or “Phase 2” buttons is pressed on, user starts to speak only when the window is seen like Figure 5.3.

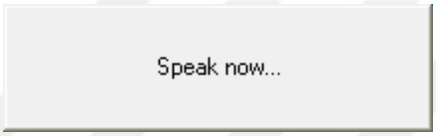


Figure 5.3 Training

In this phase, training the command will be stopped if any error occurs. When there is too much background noise, such errors can occur. The voice of user is perceived improperly if the environmental noise is too much. Then the second training will be more perceptible from the first one.

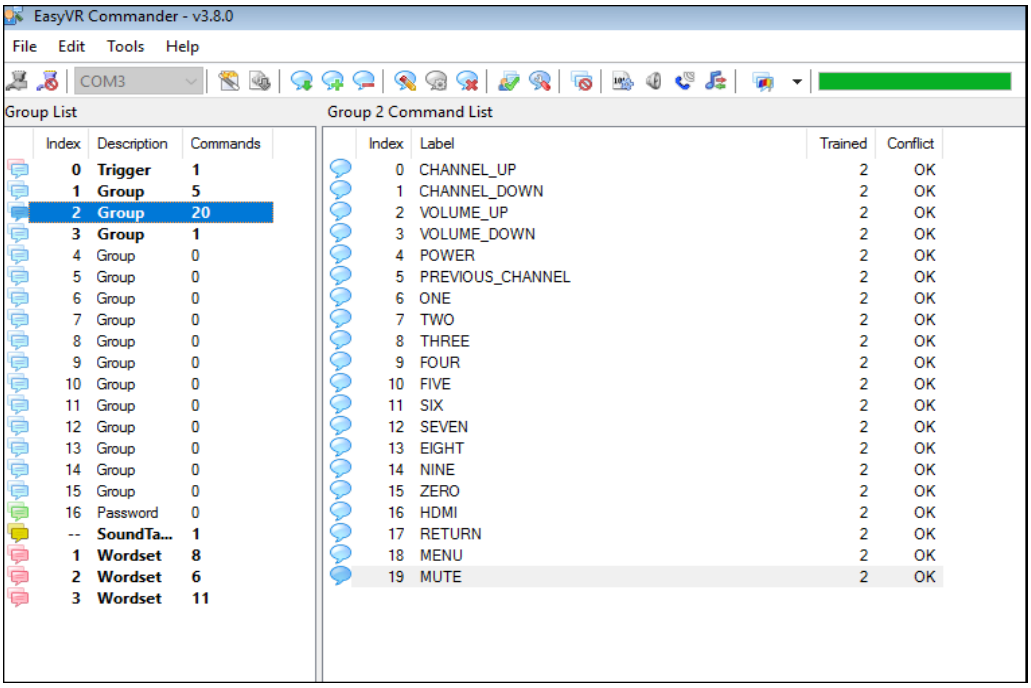


Figure 5.4 Command set

A group of commands are chosen after a class of commands is trained. Then these may be checked up by operating the icon on the toolbar menu in order to ensure the trained voice commands that can be detected properly.

5.2 Gesture Data Collection

A data acquisition system is designed with Motion Processor Unit (MPU6050) which is placed on glasses, Microcontroller Unit (mbed LPC1768) and SD Card Module to collect acceleration data (See Figure 5.5).

There are four functions which are assigned a different movement of head for each of them. The commands consist of two repeated movement to avoid interfering ordinary human movements. This functions and movements are given in the Table 5.2.

Table 5.2 Gesture command list

Function	**Head Movement**
Channel Up	Back Head Movement
Channel Down	Front Head Movement
Volume Up	Right Head Movement
Volume Down	Left Head Movement



Figure 5.5 Glasses with motion processor unit

Acceleration data are measured with MPU6050 and saved to SD card. The saved data graphs are plotted with using MATLAB. These graphs are interpreted to make algorithms.

In this study, data was collected from five different healthy people: 28 years old man, 23 years old man, 25 years old woman, 50 years old woman and 55 years old man. The data which are collected from five different people has similar characteristics. In the figures below (Fig. 5.6, 5.7, 5.8, 5.9), the data sets which is collected from 28 years old man can be seen.

Apart from gesture commands, data sets were collected during daily activities like talking, looking someone, walking stand up and sit down. All the Collected data sets were compared each other. Threshold values determined to set up Head Movement Recognition Algorithm.

5.3 Head Movement Recognition Algorithm

Accelerometer data sets were collected for different speed of each head movement. These data have not any significant difference in amplitude value for each axis. However, movement completion period has changed as expected. In

Figure 5.6, 5.7, 5.8, 5.9, obtained accelerometer data sets from subject are given for x, y, z axis along slowest and fastest head movement. For each movement, subjects provided approximately same speed. Figure 2.7 shows orientation of axis while the data sets were being collected.

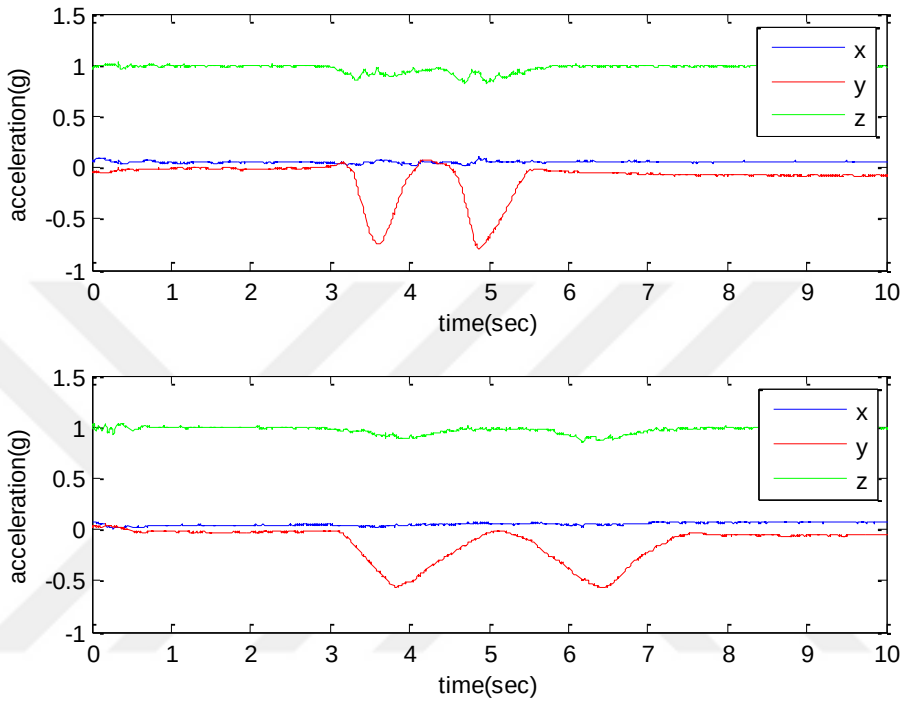


Figure 5.6 Head front movement

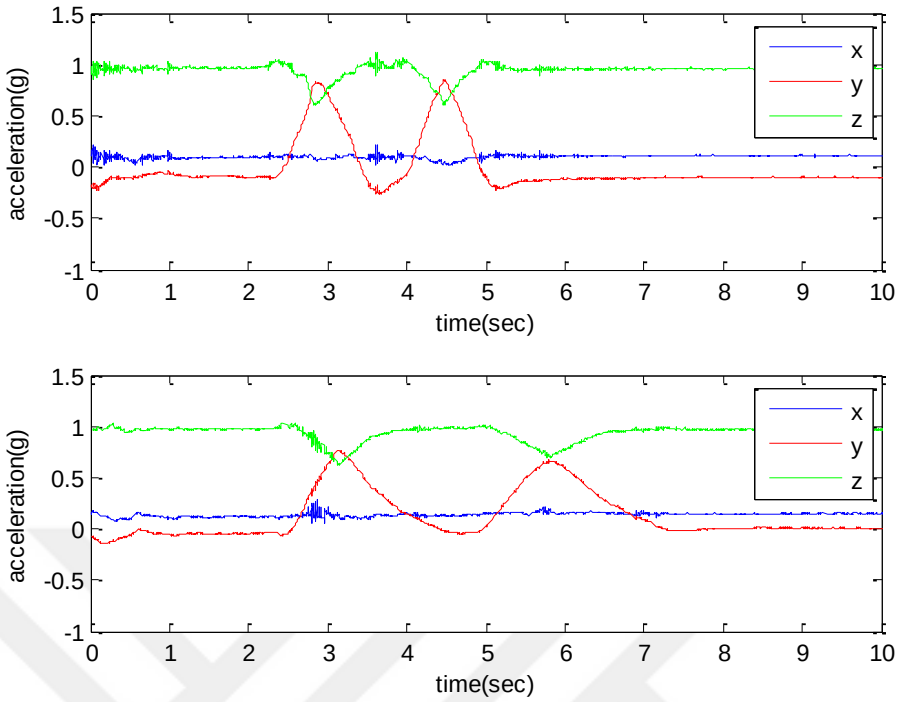


Figure 5.7 Head back movement

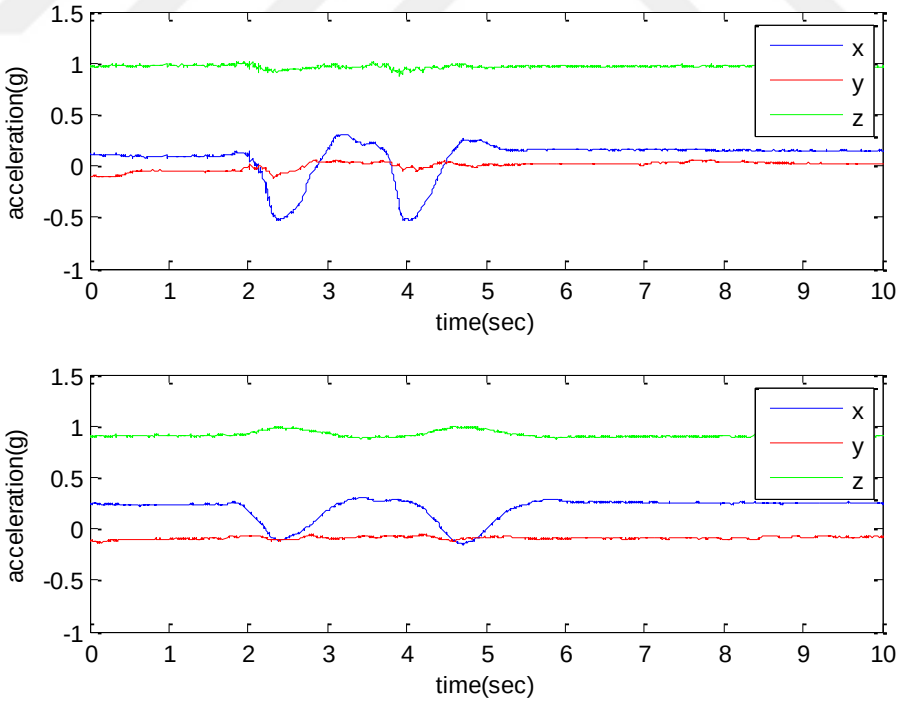


Figure 5.8 Head right movement

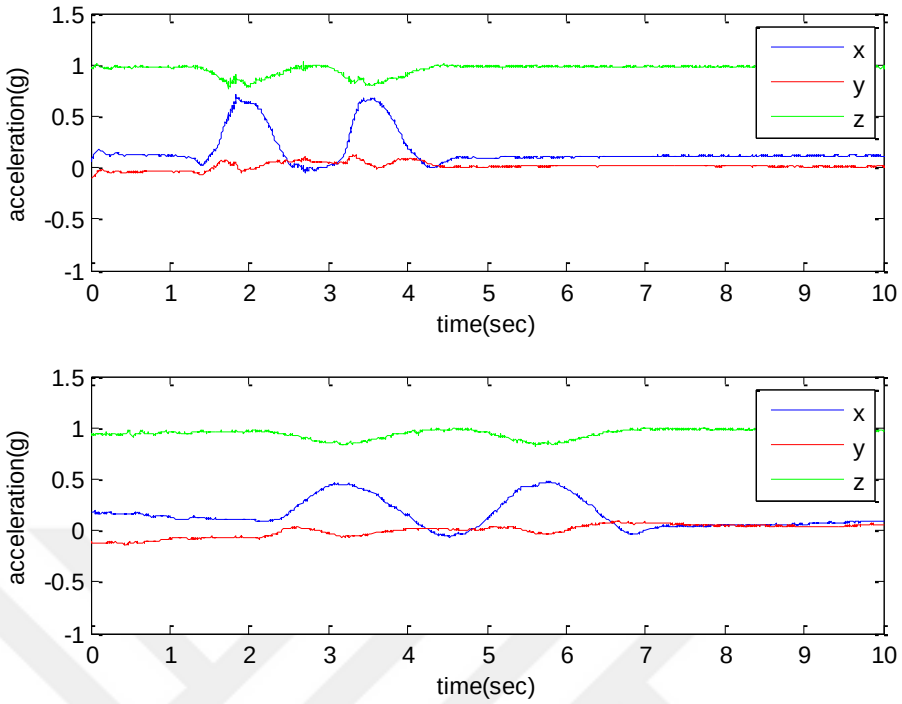


Figure 5.9 Head left movement

In thesis, some characteristics in accelerometer signals help to determine type of user's head movement. These are;

i. which axis has the highest variation and main axis is detected
ii. It is determined the variations in other axis and these must be rather small than main axis
iii. For each head movement, total twelve threshold values are obtained for main and auxiliary axis.
iv. It is controlled reaching time for each threshold.

The acceleration threshold values have determined by comparing daily activities. With this purpose, the data collected for talking, looking someone, walking, stand up and sit down. The maximum variations in accelerometer signals were detected. First threshold in main axis were taken as higher value than threshold values in daily activities. The others were chosen randomly using other samples in a period. Furthermore, time thresholds obtained by observing the time between the amplitude thresholds in slowest movement. For each movement, threshold values can be seen from Table 5.3, 5.4, 5.5 and 5,6.

Table 5.3 Head front movement thresholds

Head Front Movement	X axis	Y axis	Z axis	Time
Threshold 1	-0.2g<X<0.2g	Y<-0.5g	Z<0.9g	S1 is saved when y axis is smaller than -0.5g
Threshold 2	-0.2g<X<0.2g	Y>-0.4g	Z>0.9g	(S2 saved in threshold 2) – (S1) <0.5sec
Threshold 3	-0.2g<X<0.2g	Y<-0.5g	Z<0.9g	(S3 saved in threshold 3) – (S2) <0.5sec
Threshold 4	-0.2g<X<0.2g	Y>-0.4g	Z>0.9g	(S4 saved in threshold 4) – (S3) <0.5sec

Table 5.4 Head back movement thresholds

Head Back Movement	X axis	Y axis	Z axis	Time
Threshold 1	-0.2g<X<0.2g	Y>0.5g	Z<0.7g	S1 is saved when y axis is greater than 0.5g
Threshold 2	-0.2g<X<0.2g	Y<0.4g	0.8g<Z	(S2 saved in threshold 2) – (Time 1) <0.5sec
Threshold 3	-0.2g<X<0.2g	Y>0.5g	Z<0.7g	(S3 saved in threshold 3) – (S2) <0.5sec
Threshold 4	-0.2g<X<0.2g	Y<0.4g	0.8g<Z	(S4 saved in threshold 4) – (S3) <0.5sec

Table 5.5 Head right movement thresholds

Head Right Movement	X axis	Y axis	Z axis	Time
Threshold 1	X<-0.4g	-0.2g<Y<0.2g	1.1g<Z	S1 is saved when X axis is smaller than -0.4g
Threshold 2	X>-0.3g	-0.2g<Y<0.2g	Z<1.1g	(S2 saved in threshold 2) – (S1) <0.6sec
Threshold 3	X<-0.4g	-0.2g<Y<0.2g	1.1g<Z	(S3 saved in threshold 3) – (S2) <0.6sec
Threshold 4	X>-0.3g	-0.2g<Y<0.2g	Z<1.1g	(S4 saved in threshold 4) – (S3) <0.6sec

Table 5.6 Head left movement thresholds

Head Left Movement	X axis	Y axis	Z axis	Time
Threshold 1	X>0.4g	-0.2g<Y<0.2g	Z<0.9	S1 is saved when X axis is greater than 0.4g
Threshold 2	X<0.4g	-0.2g<Y<0.2g	0.9g<Z	(S2 saved in threshold 2) – (S1) <0.6sec
Threshold 3	X>0.4g	-0.2g<Y<0.2g	Z<0.9	(S3 saved in threshold 3) – (S2) <0.6sec
Threshold 4	X<0.4g	-0.2g<Y<0.2g	0.9g<Z	(S4 saved in threshold 4) – (S3) <0.6sec

As it can be seen from the figures and tables, y axis is main axis for front head and back head shaking, other axes are auxiliary axis. The variations for y axis along these motions equal to each other as magnitude. However, directions are opposite. Other axes have not significant variations like y axis, but these are used to test and verify algorithms. For right and left shaking of head, x axis is more powerful than the others. Directions of these are opposite like front and back head movements. Also, y and z axes are used as auxiliary. In Figure 5.10, general block diagram for detection

of algorithms is given. As it can be seen from figure, all axes and time thresholds are used as effectively.

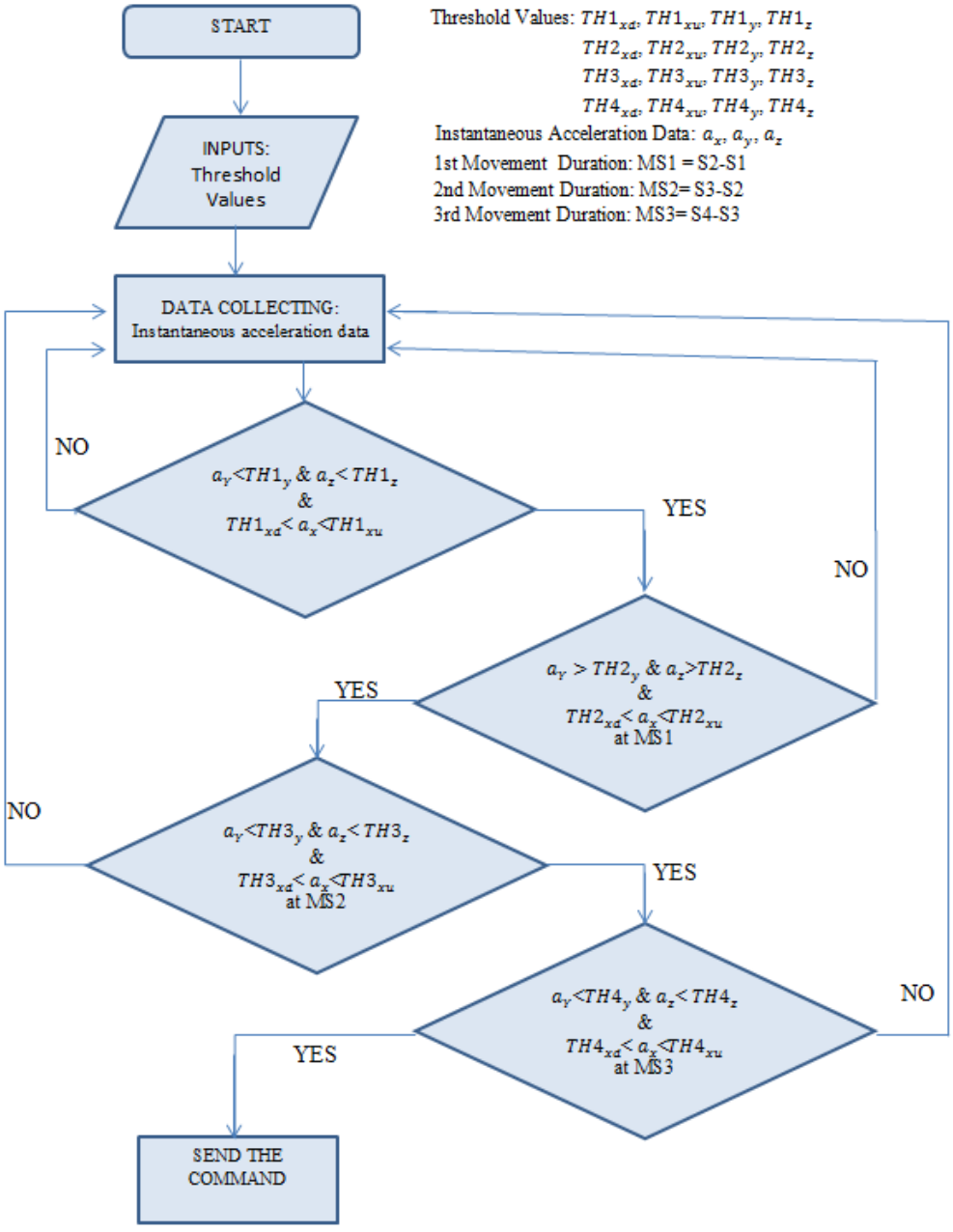


Figure 5.10 Head front movement algorithm

Back movement is using same algorithm shown in Figure 5.10. Notifying that in the control statement, the direction of instantaneous acceleration data is in the opposite direction. Therefore, the control statement should be $a_Y > TH1_y$ and $a_z > TH1_z$ in the first cycle of back movement. It should be $a_Y < TH2_y$ and $a_z < TH2_z$ in the second cycle.

Right and left movements are using the same algorithm. However, the direction of movements is completely different than front and back movements. These movements occur along the x axis. In the first cycle of right movement the system controls $a_x < TH1_x$ and $a_z < TH1_z$ and it controls $a_x > TH2_x$ and $a_z > TH2_z$ for the second cycle. In the first cycle of left movement the system controls $a_x > TH1_x$ and $a_z > TH1_z$ and it controls $a_x < TH2_x$ and $a_z < TH2_z$ for the second cycle.

CHAPTER SIX
CONCLUSIONS AND FUTURE WORK

In this thesis, a voice and gesture operated remote control system was developed for usage of television. This system is suitable for hand disabled individuals who are capable to giving clear, short voice commands and making head movements. The speaker dependent voice commands are recognized by a voice recognition module and the gesture are recognized by mems based accelerometer which is placed on glasses. The given commands are delivered to television via infrared transmitter. The small size and light weight of the system allow the device to be more mobile and portable.

The test results are hopeful. Voice operated system can do almost every action which a normal TV user wants to do. It can understand twenty speaker dependent voice commands. The system increases quality of TV experiences even for healthy people. The proposed movement recognition algorithm gives encouraging results on healthy and hand disabled individuals. Unfortunately, the voice and gesture operated remote control system has not been tried on people who are confined to bed. The efforts are kept on going to test this system on confined to bed individuals.

Consequently as a future work, this project can be adapted to work on mobile phones which have internal accelerometer, and infrared transmitter. Moreover, voice recognition systems easily interfere with environmental noise like voice of other people and the TV itself. This project can be expanded with additional filters before voice recognition module.

REFERENCES

Bureau of Labor Statistics (2016). *America time survey 2015 results.* Retrieved August 1, 2016 from http://www.bls.gov/news.release/pdf/atus.pdf

Clark, J.A. (1979). A spoken word recognition system for aiding the severely paralyzed. *Proceedings of the Compcon Spring, 79, 19-22*.

Craig, A., Tran, Y., McIsaac, P., & Boord, P. (2004). The efficacy and benefits of environmental control systems for the severely disabled. *Medical Science Monitor, 11(1), 32-39*

Fisher, D.E., & Fisher, M.J. (1997). *Tube the invention of television (1st ed.),* Berkeley: Counterpoint.

Freeman, W., & Weissman, C. (1994). Television control by hand gestures. *IEEE International workshop on automatic face and gesture recognition.*

Invensence (2013), *MPU-6000/MPU-6050 product specification*. Retrieved December 12, 2015, from https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf

InvenSense Inc., (2010). *ITG-3200 product specification revision 1.4*. Retrieved April 6, 2016, from https://www.sparkfun.com/datasheets/Sensors/Gyro/PS-ITG-3200-00-01.4.pdf.

Ito, E. (1988). Multi-modal interface with voice and head tracking for multiple home appliances. *516 Nanasawa, Atsugi, Kanagawa, 243-012.*

Jiang, H., Han, Z., Scucces, P., Robidoux, S., & Sun, Y. (2000). voice activated environmental control system for persons with disabilities. *Bioengineering Conference, 2000. Proceedings of the IEEE 26th Annual Northeast, 0-7803-6341-8*

Kawarazaki, N., & Yoshidome, T. (2012). Remote control system of home electrical appliances using speech recognition. *Automation Science and Engineering, 2012 IEEE International Conference on, ISSN: 2161-8089*

Kela, J. (2006). Accelerometer-based gesture control for a design environment, *Personal Ubiquitous Computing*, *10 (5*), 285-299.

Lindgren, N. (1965). Machine recognition of human language. *IEEE Spectrum, 2 (3)*, 114-136.

Luplow, W.C., & Taylor, J.I. (2012). A brief history of the tv remote control and a tribute to its coinventors. *IEEE Consumer Electronics Magazine*, *1*, 8-26.

Mangrulkar., J.B., & Shah, J.J. (2013). Design of voice controlled embedded system for tv remote. *International Journal of Computer, Information Technology & Bioinformatics (IJCITB), 2 (4),* 1629-1633.

Murata, Y., Sato, N., & Takayama,T. (2012). A gesture-based remote control for finger disabled people, *Consumer Electronics (GCCE), IEEE 1st Global Conference on Consumer Electronics*, ISBN: 978-1-4673-1501-2, 406-410.

Nakatoh, Y., Kuwano, H., Kanamori, T., & Hoshimi, M. (2007). Speech recognition interface system for digital tv control. *IEEE Transactions on Consumer Electronics*, 49 (3) 765-769.

NXP Semiconductors (2009). *Rapid prototyping for the LPC1768 MCU*. Retrieved September 24, 2015, from http://www.nxp.com/documents/leaflet/LPC1768.pdf

Reddy, Y.N., Singh, T.K., Sajit, S.M., Singh, A.K., & Murthy, M.B.R.(2013). Voice operated tv remote. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2 (4),* 1629-1633.

Rekimoto, J. (1996). Tilting operations for small screen interfaces. *Proceedings of the 9th annual ACM symposium on User Interface Software and Technology*, 167-168.

San Bergmans (n.d). *IR remote control theory*. Retrieved May 3, 2016 from http://www.sbprojects.com/knowledge/ir/index.php

Samsung (2008). *S3F80KB remote controller application notes*. Retrieved May 20, 2016 from http://www.samsung.com/semiconductor/global/file/2011/product/2010/2/19/650291um_s3f80kb_rev121.pdf

Sawada, H., & Hashimoto, S. (2000). Gesture recognition using an accelerometer sensor and its application to musical performance control. *Electronics and Communications, Japan, 3*, 9–17.

Solanki, U., & Desai, N. (2011). Hand gesture based remote control for home appliances : Handmote. *Information and Communication Technologies (WICT), 2011 World Congress on*, ISBN: 978-1-4673-0126-8, 425-429

Sparkfun (n.d.). *Serial communication – rules of serial*. Retrieved June 18, 2016, from https://learn.sparkfun.com/tutorials/serial-communication/rules-of-serial

Sparkfun (n.d). *IR communication- IR communication basics*. Retrieved April 20, 2016 from https://learn.sparkfun.com/tutorials/ir-communication

Sugaya, S., Nishimura, S., Shimizu, T., & Sugiura, Y. (1979). Voice remote controlled tv set. *IEEE CH, 1465,* 4-7.

Valdez, J., & Becker J. (2015). *Understanding the I^2C bus*. Retrieved April 4, 2016 from http://www.ti.com/lit/an/slva704/slva704.pdf

Veear (2015), *EasyVR 2.0 user manual.* Retrieved September 26, 2015, from http://www.veear.eu/files/easyvr_user_manual_3.6.7.pdf

Vine, L., Willitts, M., Farmer, M., & Gunning, C. (2011). *Life opportunities survey, wave one results, 2009/11*. Retrieved August 2, 2016 https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/180891/los_wave_one_200911.pdf

Vishay (2013). *Data formats for IR remote control*. Retrieved May 16, 2016 from http://www.vishay.com/docs/80071/dataform.pdf