VOICE-ACTIVATED REMOTE CONTROL FOR TELEVISION

By

SAREH BAYAT
(6155)

FINAL PROJECT REPORT

Submitted to the Electrical & Electronic Engineering Department
In Partial Fulfilment of the Requirements
For the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronic Engineering)

University Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh Perak
Darul Ridzuan

CERTIFICATION OF APPROVAL

VOICE-ACTIVATED REMOTE CONTROL FOR TELEVISION

By

Sareh Bayat

A Project Dissertation Submitted to the
Electrical & Electronic Engineering Department
Universiti Teknologi PETRONAS
in Partial Fulfillment of the Requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronic Engineering)

Approved:

Dr.Mumtaj Begam Kasim Rawthar
Project Supervisor

University Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh Perak
Darul Ridzuan

June2007

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein has not been undertaken or done by unspecified sources or persons anywhere else.

S. Bayat

Sareh Bayat

ABSTRACT

The Voice-activated remote control can be applied for any home appliances which work with a remote control. This could be so helpful for those who are constantly misplacing remote controls or are too tired after coming home from a long day of work and most importantly persons with disabilities, particularly people with paralysis, quadriplegia or paraplegia. This project is aimed to come up with a device which can communicate with a target appliance (which works with remote control) through voice commands. The device should accept voice command and on other hand the target device should correspond with a correct execution instantaneously. This project is targeted for electronic gadgets in general and a television set in particular. In addition to the traditional design of IR remote controls, a microphone, microcontroller and voice recognition board (using HM2007 voice recognition chip) will be added in the design. Although the project scope will only focus on controlling a television set, this project can be modified for a numbers of applications; such as door opener, VCR programmer, air conditioner and etc.

ACKNOWLEDGEMENTS

My advance appreciation goes to my supervisor Dr.Mumtaj Begam Kasim Rawthar for her great support and guidance. His patience and cooperation helped me a lot through this project.

I would like to thank my parents for their moral support .Their support has always helped me to ease the pressure and become more concentrated on the matter.

Also I would like to thank my other lecturers, who have given their sincere opinions toward this project.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

HMM	Hidden Markov Modeling
DARPA	Defense Advanced Research Project Agency
ASR	Automatic Speech Recognition
LPC	Linear Predictive Coefficients
PIC	Programmable Integrated Circuit
CPU	Central Processing Unit
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LSD	Least Significant Digit
MSD	Most Significant Digit

CHAPTER 1

INTRODUCTION

1.1 Background of Study

The most common type of communication between people is verbally. Even other means of communication, such as writing or sign language, can not compete with voice communication. Speaking is the easiest way to communicate that from long ago people were dreaming about speaking through devices to perform their demands. Voice activation is a technology which can help to make that dream come true. "The study of automatic speech recognition and transcription began in the 1936 with AT&T's Bell Labs. At that time, most research was funded and performed by Universities and the U.S. Government (primarily by the Military and DARPA – Defense Advanced Research Project Agency)" [1].

The key feature of voice-activated technology is the user interface. It is based upon speech synthesis and recognition and operates through voice commands given by the user. To control and command an appliance (computer, VCR, TV security system, etc.) by speaking to it, will make it easier, while increasing the efficiency and effectiveness of working with that device. At its most basic level speech recognition allows the user to perform parallel tasks, (i.e. hands and eyes are busy elsewhere) while continuing to work with the computer or appliance. In the near future, speech recognition will become the method of choice for controlling appliances, toys, tools, computers and robotics. There is a huge commercial market waiting for this technology to mature [9].

A voice-activated remote control, allows users to give voice commands to most home appliances, such as TV, VCR/DVD, air conditioners and Hi-Fi stereos. This will address to help those who want to enjoy the comfort and technical advancements, or more importantly old and disabled people who can not do their daily jobs without others help. The voice-activated remote control could be helpful to provide comfort and ease the work load of this kind of people.

1.2 Problem Statement

As people get used to automated industries, they become more impatient and always seek for the easiest and fastest ways to perform their jobs. Old people as well as people with disabilities have difficulties in living their life independently. Most of them use wheel chairs or they have a constant care from someone. Voice-activated devices can be a good help for them to do some of the simple jobs such as turning a TV on, by themselves to be dependent from help.

A remote control works when a button is pressed by the user. Controlling devices using voice commands would help to work with them much easier and faster. In general this project is targeted for any electronic gadgets. Figure 1.1 illustrates the project with the specific target device (television).

TV remote control includes more than hundreds of buttons, but for the sake of this project, the voice activated buttons are limited to six main keys which are common in all TV remote controls. These buttons are power (on/off), volume up, volume down, channel up, channel down and mute.

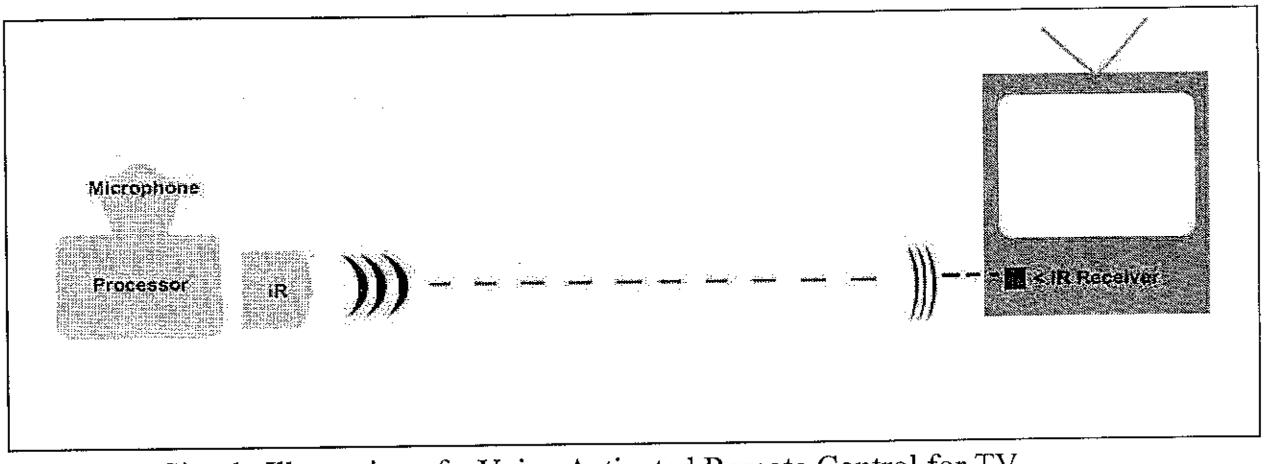


Figure 1.1 Simple Illustration of a Voice-Activated Remote Control for TV.

1.3 Objectives and Scope of Study

The main objective of this project is to provide a remote control for television set which can be activated with voice. Followings are some highlights on the scope of study of this project:

- The recognition process should be accurate and sensitive. Recognizing the correct command is required.
- The response duration of the device should be fast enough. The execution of the voice commands should appear instantaneous to the user.
- The out coming product should be easy to work with.

CHAPTER 2

LITERATURE REVIEW AND THEORY

2.1 Background of Remote Control

The idea for the television remote began with Eugene F. McDonald Jr., the late founder-president of Zenith Electronics Corporation (then known as Zenith Radio Corporation). He believed that viewers would appreciate being able to 'tune out annoying commercials'. The first TV remote control, called "Lazy Bones", was developed in 1950 by Zenith Corporation. Viewers could turn the television on and off, and switch channels. Pressing a button on the remote rotated the tuner clockwise or counterclockwise depending on changing the channel to a higher or lower number. But unfortunately, it was connected to the television by a long cable; heading to the kitchen to replenish supplies was somewhat hazardous [1, 2].

To rectify this problem, in 1955 a young Zenith engineer named Eugene Polley hit on the idea of using light to control the television. He invented the "Flash Matic", which represented the industry's first wireless TV remote. It came with a specially equipped television that had photocells embedded in each corner of its cabinet. The viewer used a highly directional flashlight to activate the four control functions, which turned the picture and sound on and off and changed channels by turning the tuner dial clockwise and counterclockwise. Unfortunately, the photocells reacted to sunlight, and if the sun shone on the TV, the tuner could start rotating. Thus the commander directed his engineers to develop a better remote control [1, 2].

Zenith's engineer, Dr Robert Adler, came up with a solution using "ultra-sonic", that is, high frequency sound beyond the rang of human hearing. Zenith patented Adler's invention in 1956 and marketed it the same year as the Space Commander. The remote was only working with Zenith televisions. The transmitter of this remote used no batteries; it was built around aluminum rods that were light in weight. Pressing a button on remote, struck one end and the rods emitted distinctive high-frequency sounds. The first such remote control used four rods: one for channel up, one for channel down, one for sound on and off, and one for power on and off. Each rod was called a 'vibrator element' which varied in length so emitting different noises. Adler's invention had two advantages over modern remotes: it needed no batteries and did not have to be pointed at the television. Figure 2.1 illustrates one type of zenith space commander remote control [1, 2 and 3].

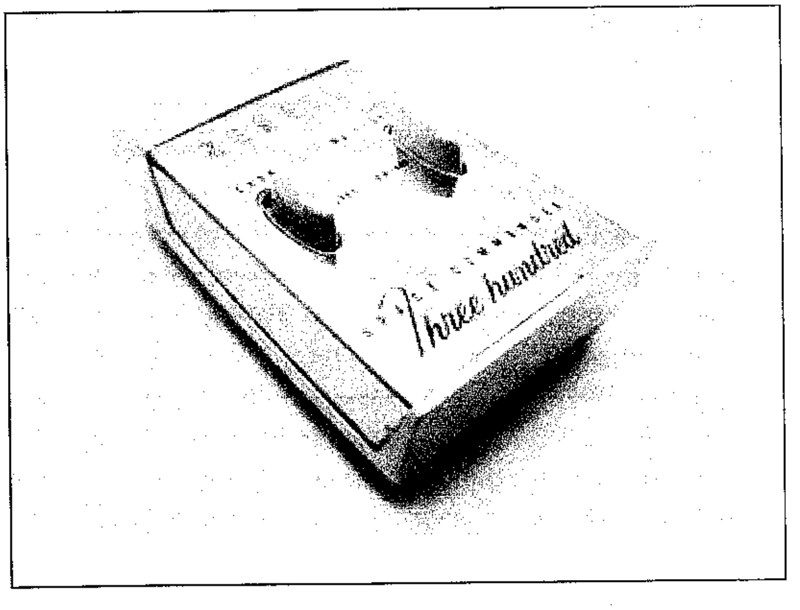


Figure 2.1 Zenith Space Command Remote Control [3]

In the early 1960s, solid-state circuitry (i.e., transistors) began to replace vacuum tubes. Hand-held, battery-powered control units could now be designed to generate the inaudible sound electronically. In this modified form, Dr. Adler's ultrasonic remote control invention lasted through the early 1980s, a quarter century from its inception. Table 2.1 summarizes the development milestone of the remote control [4].

Table 2.1 Remote Control Development Milestone [4]

Year	Achievements
1893	A remote control was described by Nikola Tesla, in his patent, U.S. Patent 613809.
1950	The first TV remote control, called "Lazy Bones," was developed by Zenith.
1955	Zenith engineer Eugene Polley invented the "Flash Matic," the industry's first wireless TV remote.
1956	Robert Adler's"Zenith Space Command," remote control went into production.
1980	By the early 1980s, the industry moved to infrared.

2.2 Today's Remote Controls

By the early 1980s, the industry moved to infrared, or IR remote technology. The IR remote works by using a low frequency light beam. The frequency is so low that it is invisible to human eyes, but can be detected by a receiver in the TV. Today, remote control is a standard feature on other consumer electronic products, including VCRs, cable and satellite boxes, digital video disc players and home audio receivers. And the most sophisticated TV sets have remotes with as many as 50 buttons.

2.2.1 How Does a Remote Control Work?

The remote control waits for the user to press a key, and then translates that into infrared signals that are received by television. The printed circuit board of the remote control can be seen when the cover is unscrewed. Figure 2.2 illustrates a typical remote control circuit board and also the rubbery buttons. The components on the circuit board are typical for most remote controls. The circuit board usually contains an integrated circuit (known as a chip) which is the processor of the system, also some capacitors, resistors, transistors and diodes [5].

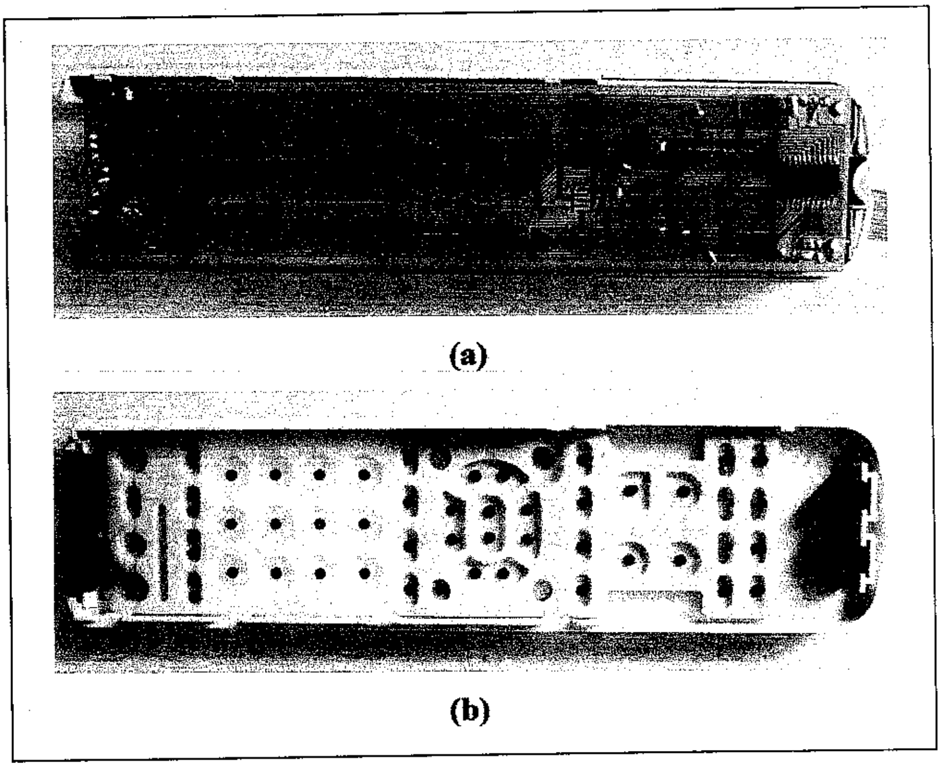


Figure 2.2 (a) Remote Control Circuit Board. (b) Back Side of the Buttons

The circuit board is a thin piece of fiber glass that has thin copper "wires" etched onto its surface. Electronic parts are assembled on printed circuit boards. As it is can be seen in figure 2.2 (a), a set of contact points is provided for each button. The voltage across these two contact points is 3V which is provided by the remote control's batteries. The buttons themselves are made of a thin rubbery sheet (Figure 2.2 (b)). For each button there is a black conductive disk. When the disk touches the contacts on the printed circuit board, it connects them and the chip can sense that connection.

Thus the basic operation of the remote goes like this; when a button is pressed, a specific connection is completed. "The chip senses that connection and knows what button is being pressed and produces a Morse-code-line signal specific to that button. The transistors amplify the signal and send them to the LED, which translates the signal into infrared light" [5]. Then the receiver in TV detects IR signals and reacts appropriately.

2.3 History of Voice Recognition

The technology of Automatic Speech (voice) Recognition (ASR) and transcription has improved enormously over the past few years. "The study of automatic speech recognition and transcription began in the 1936 with AT&T's Bell Labs. At that time, most research was funded and performed by Universities and the U.S. Government (primarily by the Military and DARPA – Defense Advanced Research Project Agency). It was not until the early 1980's when the technology reached the commercial market" [6]. Some of these early technologies are summarized in table 2.2.

Table 2.2 Timeline of Speech and Voice Recognition [7]

Year	Invention
1936	• AT&T's Bell Labs produced the first electronic speech synthesizer called the Voder (Dudley, Riesz and Watkins). This machine was demonstrated in the 1939 World Fairs by experts that used a keyboard and foot pedals to play the machine and emit speech.
Early 1970's	• The Hidden Markov Modeling (HMM) approach to speech & voice recognition was invented by Lenny Baum of Princeton University and shared with several ARPA (Advanced Research Projects Agency) contractors including IBM.
1984	• Speech Works, the leading provider of over-the-telephone automated speech recognition (ASR) solutions, was founded.
1996	• Charles Schwab is the first company to devote resources towards developing up a speech recognition IVR system with Nuance. The program, Voice Broker, allows for up to 360 simultaneous customers to call in and get quotes on stock and options. • BellSouth launches the world's first voice portal, called Val and later Info by Voice.
2003	• Scan Soft Ships Dragon Naturally Speaking 7 Medical, Lowers Healthcare Costs through Highly Accurate Speech Recognition. • Scan Soft Closes Acquisition of Speech Works International, Inc. • Scan Soft closes deal to distribute and support IBM via Voice Desktop Products.

As it can be verified from the table, most speech recognition systems available today are programs that use personal computers equipped with compatible sound cards. These programs reside in the memory of the computer and operate continuously in the background of computer operating systems. As a result, the speech recognition program is allowed to be used with other programs like Word or Excel. There is a noticeable slow-down in the operation and function of the computer when the voice recognition program is enabled. From a commercial aspect, the disadvantage in this approach is the necessity of a computer. While these speech programs are impressive, it is not economically viable for manufacturers to add full blown computer systems to control a television or VCR [8].

Human's hearing system has the ability to listen for granted. For instance we are capable of listening to one person's speech among several people at a party. We sub-consciously filter out the extemporaneous conversations and sound. This filtering ability is beyond the capabilities of today's speech recognition systems. Speech recognition is not speech understanding. Understanding the meaning of words is a higher intellectual function.

2.4 Definition of Voice Recognition

Voice recognition (speech recognition) is "the technology by which sounds, words or phrases spoken by humans are converted into electrical signals and these signals are transformed into coding patterns to which meaning has been assigned" [9]. Speech recognition is classified into two categories, speaker dependent and speaker independent.

2.4.1 Speaker Dependent

Speaker dependent systems are trained by the individual who will be using the system. These systems are capable of achieving a high command count and better than 95% accuracy for word recognition. The drawback to this approach is that the system responds accurately only to the individual who trained the system. This is the most common approach employed in software for personal computers [8, 10].

2.4.2 Speaker Independent

Speaker independent is a system trained to respond to a word regardless of who speaks. Therefore the system must respond to a large variety of speech patterns, inflections and enunciation's of the target word. The command word count is usually lower than that in speaker dependent. However, high accuracy can still be maintained within processing limits. Industrial requirements more often need speaker independent voice systems, such as the AT&T system used in the telephone systems [8, 10].

Speech recognition systems have another constraint concerning the style of speech they can recognize. There are three styles of speech: isolated, connected and continuous.

- **Isolated** speech recognition systems can just handle words that are spoken separately. This is the most common speech recognition systems available today. The user must pause between each word or spoken command.

- **Connected** is a half way point between isolated word and continuous speech recognition. It allows users to speak multiple words.

- **Continuous** is the natural conversational speech we are accustomed to in everyday life. "It is extremely difficult for a recognizer to shift through the text as the words tend to merge together. For instance, "Hi, how are you doing?" sounds like "Hi,howyadoin." Continuous speech recognition systems are on the market and are under continual development" [8, 10].

2.5 Different Approaches for Voice Recognition

The most common approaches to voice recognition can be divided into two classes: "template matching" and "feature analysis".

2.5.1 Template Matching

Template matching is the simplest technique and has the highest accuracy when used properly, but it also suffers from some limitations. A collection of prototypical speech patterns (the templates) are stored as reference patterns representing the dictionary of candidate words. Recognition is then carried out by matching an unknown spoken utterance with each of these reference templates and selecting the category of the best matching pattern. Usually templates for entire words are constructed. This has the advantage that errors due to segmentation or classification of smaller acoustically more variable units such as phonemes can be avoided. In turn, each word must have a template preparation and matching become prohibitively expensive or impractical as the vocabulary size increases beyond a few hundred words. Nonetheless, many important lessons have been learnt from template-based recognition that continues to be applied in most recognition systems today. Speaker dependent systems are based on this type of voice recognition approach [11].

2.5.2 Feature Analysis

A more general form of voice recognition is available through feature analysis and this technique usually leads to "speaker-independent" voice recognition. Instead of trying to find an exact or near-exact match between the actual voice input and a previously stored voice template, this method first processes the voice input using "Fourier transforms" or "linear predictive coding (LPC)", then attempts to find characteristic similarities between the expected inputs and the actual digitized voice input. These similarities will be present for a wide range of speakers, and so the system does not need to be trained by each new user. The types of speech differences that the speaker-independent method can deal with, but which pattern matching would fail to handle, include accents, and varying speed of delivery, pitch, volume, and inflection. Speaker-independent speech recognition has proven to be very difficult, with some of the greatest hurdles being the variety of accents and inflections used by speakers of different nationalities. Recognition accuracy for speaker-independent systems is somewhat less than for speaker-dependent systems, usually between 90 and 95 percent [12].

2.6 HM2007 Voice Recognition System

HM2007 is a single chip CMOS voice recognition LSI circuit with on-chip analog front end, voice analysis, recognition process and system control functions. The pin configuration of the HM2007 chip is available in Appendix B. A voice recognition system can be composed of external microphone, keyboard, 64K SRAM and some other components. Combined with the microprocessor, an intelligent recognition system can be built [13].Some applications are listed below;

- Command and control of appliances and equipment
- Telephone assistance systems
- Data entry
- Speech controlled toys
- Speech and voice recognition security systems

The HM2007 voice recognition chip provides the options of recognizing either forty (0.96 second) words or twenty (1.92 second) words, all depending on the style of speech recognition as described before. When isolated words are used for speech recognition the maximum number of word or commands that can be identified is forty. And when user speaks multiple words for a command (Connected) the number of commands reduces to twenty. For memory, the circuit uses an 8K X 8 static RAM [13].

The chip has two operational modes; it can be run either in manual mode or CPU mode depending on how the circuit is set up. The CPU mode is designed to allow the chip to work under a host computer. This is an attractive approach to speech recognition for computers because the speech recognition chip operates as a co-processor to the main CPU. The jobs of listening and recognition don't occupying any of the computer's CPU time. When the HM2007 recognizes a command it can signal an interrupt to the host CPU and then relay the command code. The HM2007 chip can be cascaded to provide a larger word recognition library [13]. The CPU mode voice recognition circuit schematic is available in appendix C for more details.

In the other hand the manual mode allows to build a stand alone speech recognition board that doesn't require a host computer and may be integrated into other devices to utilize speech control.

CHAPTER 3

METHODOLOGY

The approach for this project has been modified and the final block diagram is as shown in figure 3.1.

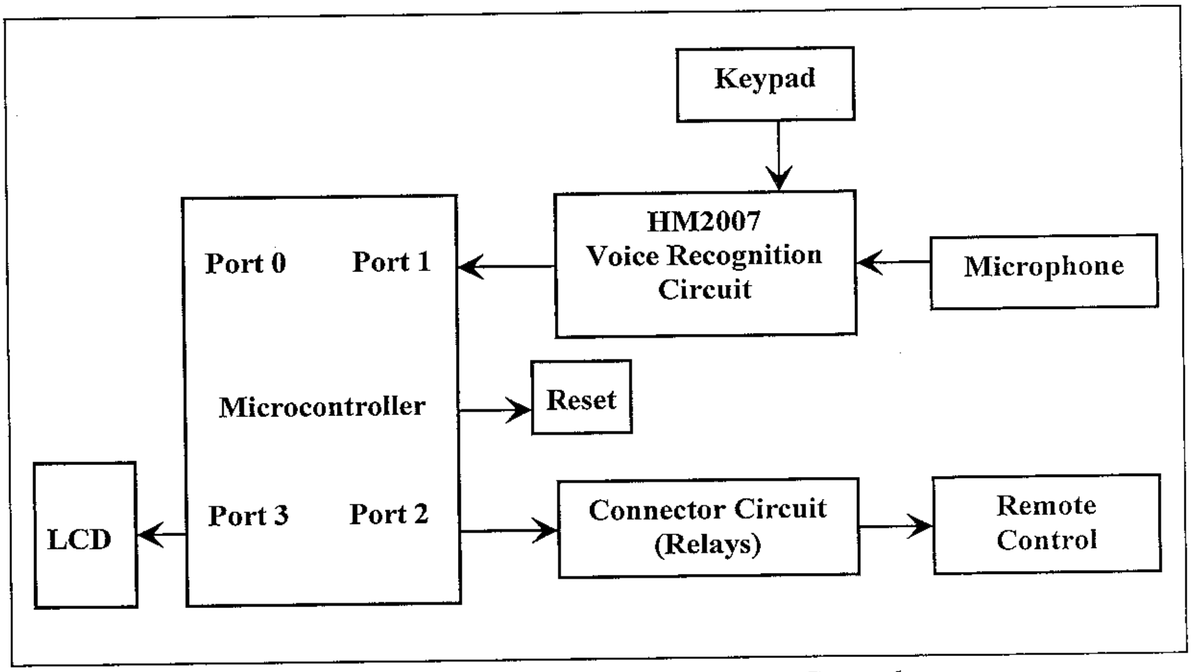


Figure 3.1 Block Diagram of the Voice-Activated Remote Control

As the figure illustrates, direction of the arrows shows the I/O relationship between different sections of the project. Microphone is the input of the whole system and the final output is the remote control. However LCD is another output which displays all the commands and errors defined for the remote control. Thus when a user says a command, the voice recognition system recognizes the command and will send it to the microprocessor. The processor activates the corresponding relay and displays the command on LCD.

This project has two main parts and each part has its own divisions.

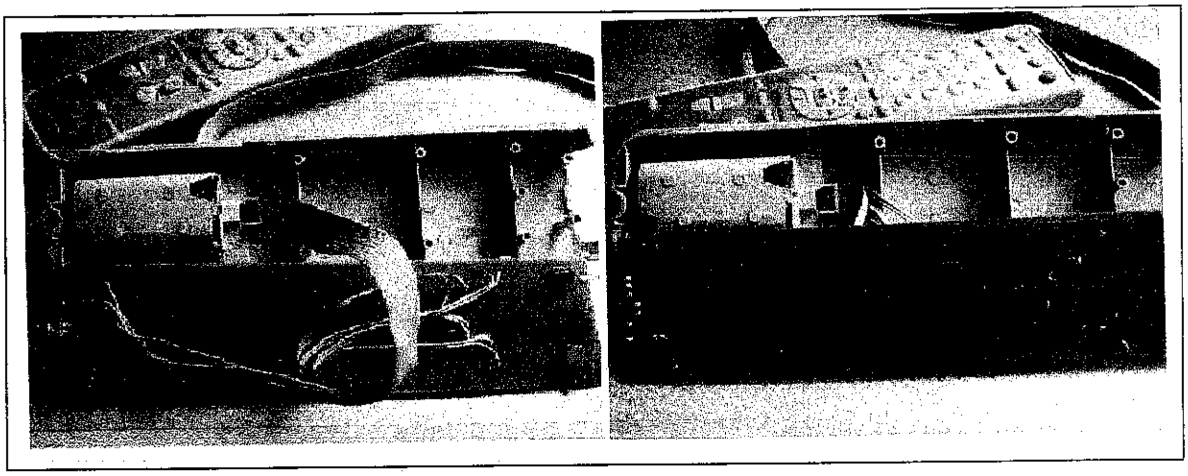
.Hardware Setting up HM2007 voice recognition circuit and building the connector circuit (relays) with all the connection between the microcontroller, HM2007 voice recognition circuit and remote control.

Software Programming the microcontroller to activate the remote control and display commands on the LCD screen

3.1 Remote Control (IR system)

The target device is television, thus a simple TV remote control is suitable for the purpose of this project. The control should be activated with the user's voice instead of pressing any key on it. Pressing a button on a remote control will complete a certain connection that leads into sending signals via infrared LED. For each voice command a relay is used to make the connection without pressing a button.

Two holes have been made beside each side of the contact points on the circuit board to solder wires. These two wires will be a connection between the relay and the contacts on the circuit board so when a relay is switched on, the connection has been made and it seems like a key is pressed. Figure 3.2 illustrate the wires connected to the remote control circuit board.



(a) **(b)**

Figure 3.2 (a) Back View and **(b)** Front view of the remote control circuit board.

The holes have been made using a CNC machine. In overall twelve holes have been made to activate six keys on the remote control. These six keys have been chosen since they are typical on any remote control for TV.

3.2 Voice Recognition System (HM2007)

HM2007 has been chosen since its circuit is much simpler and is less expensive than others. As it was described before, the HM 2007 chip has two operational modes; Manual mode and CPU mode. Manual mode is chosen for this project since this mode has the ability to build a stand alone speech recognition board that doesn't need a host computer. The circuit diagram for manual mode is shown in figure 3.3.

HM2007 circuit displays the output on two 7-segments. Thus each combination of two digits represents one output of the system. These outputs will be defined for the system and then in terms of words, will be displayed on LCD.

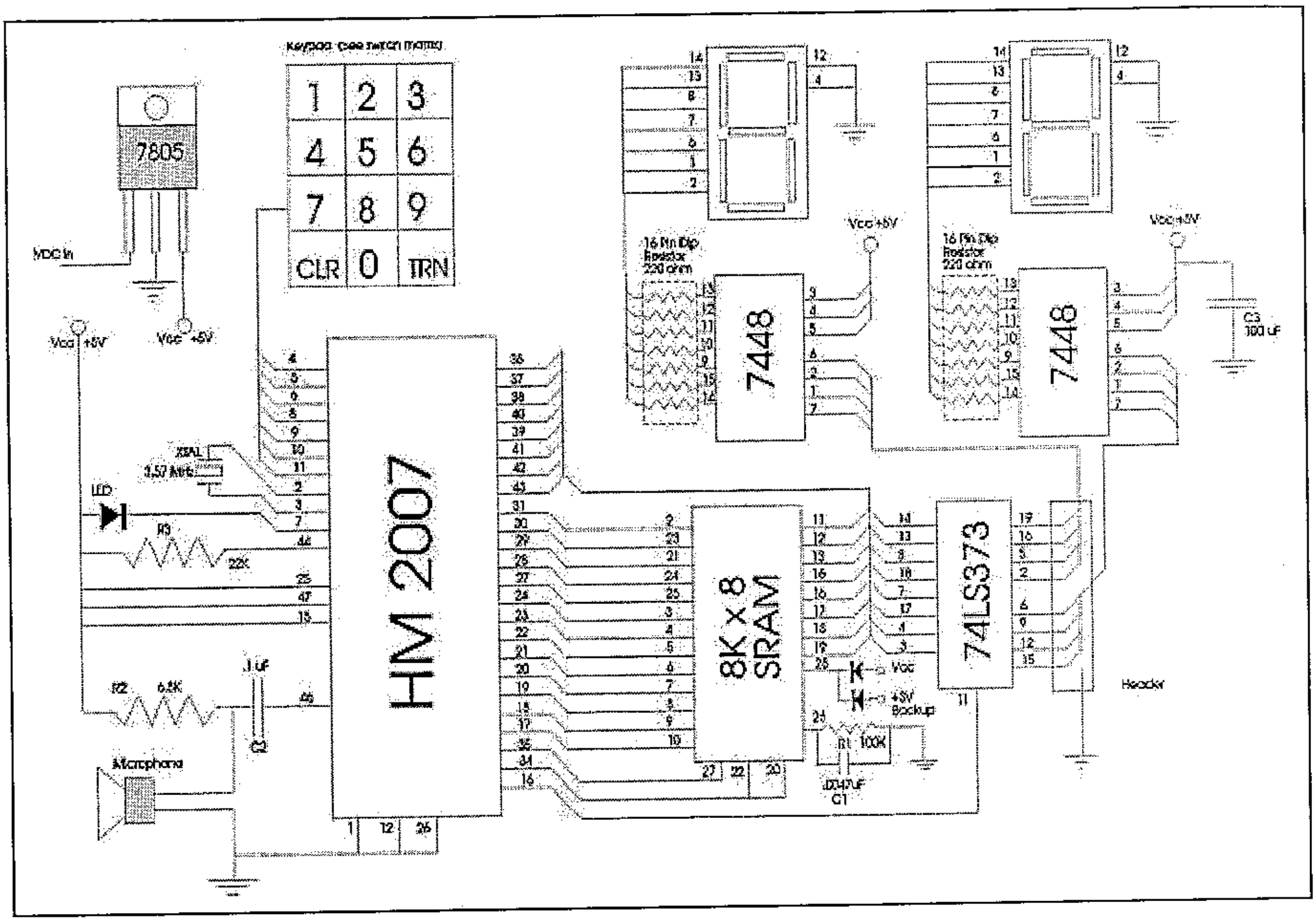


Figure 3.3 The Voice Recognition (Manual mode of HM2007) [8]

The supply voltage for the circuit is 9V. The voice input to the chip is an analog signal from the microphone of the system. To record a command, the chip stores the analog signal pattern and amplitude, and saves it in the 8kx8 SRAM. In recognition mode, the chip compares the user- inputted analog signal from the microphone with those stored in the SRAM and if it recognizes a command, an output of the command identifier will be sent to the microprocessor through the D0 to D7 ports of the chip. Whenever recording and recognition process is successful, the ready pin on the chip will be set to low and then back to high. An LED is placed in the ready pin to notify the user of a successful operation [8, 10, and 13]. Additional 3V lithium battery is used to backup the memory not to lose any information recorded to the system.

3.2.1 Recording and Clearing a Command

To record a command in a specific memory slot, the user should press a 2-digit number, which is the command identifier, on the keypad and then press the 'train' or '#' button. Any number can be used from 00-40 for a maximum length of one second word and 00-20 for maximum length of almost two seconds. If more than two digits are entered, only the last two digits are valid. When number key is pressed, the number of key will be encoded to D-bus. When the function key "#" is pressed after entering digits, it signals the chip to listen for a training word.

To clear a command, the user will press the command identifier and then press the 'clear' button. In this case, the K-bus and S-bus of the voice chip will be used as inputs to accept the input from the keypad and the D-bus will only be used as outputs to the microprocessor. To clear the entire patterns the number key (command identifier) 99 should be pressed. The D-bus output is the binary form of the memory identifier. The first four bits represent the first digit, while the last four represent the last digit [10, 13]. An illustration of the D-bus output is shown in Table 3.1.

Table 3.1 Content of D-bus Output [13]

Digital Display	D7	D6	D5	D4	D3	D2	D1	D0	Description
00	0	0	0	0	0	0	0	0	Circuit power on
55	0	1	0	1	0	1	0	1	Voice too long
66	0	1	1	0	0	1	1	0	Voice too short
77	0	1	1	1	0	1	1	1	Not Match
99	1	1	1	1	1	1	1	1	Clear all pattern

3.2.2 Recognition of the Command

When a recorded word (command) is repeated into the microphone, the number of the word should be displayed on the digital display (Table 3.1). For instance if the word "ON" was trained as word number 1. Saying the word "ON" into the microphone will cause the number 5 to be displayed on the digital display

3.3 Microcontroller

The heart of the whole system is the microcontroller. It should be programmed in a way that any output from the command identifier of the voice recognition chip be executed correctly. It means that if the voice recognition chip recognizes a command, microcontroller should activate the corresponding button on the remote control. Microcontroller links the voice recognition output to remote control.

The microcontroller selected for this project is AT89C52 which is a low-power, high performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable. The pin configuration is available in appendix D. As it is shown in figure 3.1, port 3 and port 2 are the outputs and port 1 is selected to be the input from voice recognition circuit. The program is written using Assembly language. The algorithm of the program is shown in figure 3.4. When voice recognition system recognizes a voice command input, microcontroller starts to compare the output of HM2007 (port D0-D7) with existing functions that has been stored previously. Then microcontroller finds the correspondent function for output command of HM2007 and this means that it activates the key on the remote and display a proper statement on the LCD screen.

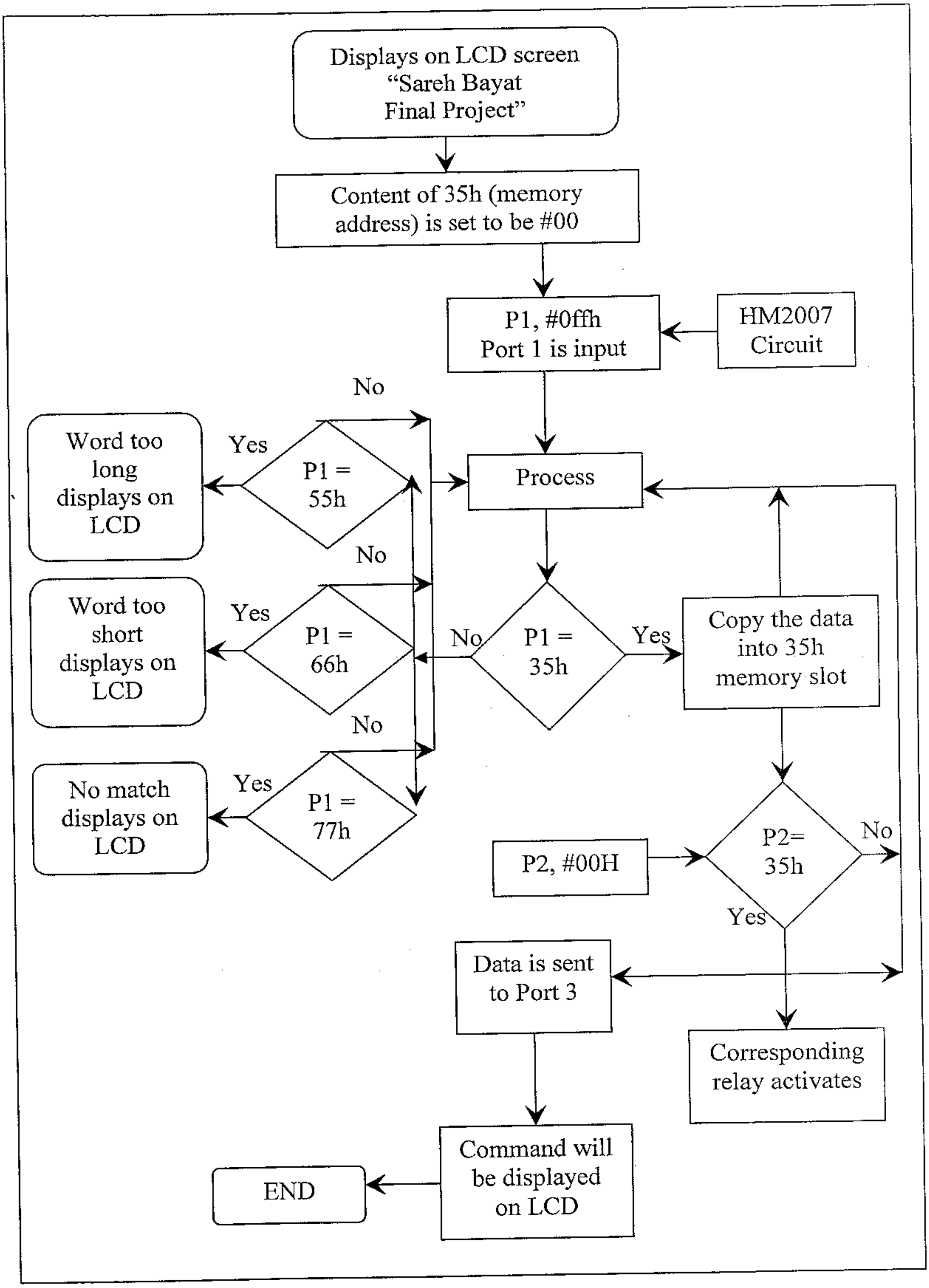


Figure 3.4 Program Flow Chart

The program is available in appendix E for more details on the algorithm of programming. The commands in the program follow a specific order and so it is needed that the voice recognition system be trained as the order of commands stated in the program. The order of commands is as follows:

01: Volume up
02: Volume down
03: Power
04: Channel up
05: Channel down
06: Mute

Besides the above mentioned functions, a Stop function is defined in the program with 09 as the command identifier. It means that during training session a phrase should be recorded as 09 for the stop function. The command will be used whenever there is a Channel up/down and Volume up/down. Whenever one of these commands has been repeated in a microphone by the user, the corresponding relay switches on and off until the user calls the recorded phrase to recall the stop function. It means that the channel or volume goes up or down till it reaches as it is desired by user. Thus another word "Stop, Break, Pause or any other desired phrase" should be trained.

A reset key is provided to reset the whole circuit if something goes wrong within the system or the screen freezes. Figure 3.5 illustrates the circuit schematic of the part that contains the microcontroller. All the connections between the microcontroller and the other parts of the whole system have been shown. As it was mentioned before port 1 is considered as an input from HM2007 voice recognition circuit. Port 2 is the output to relays. Along the way from port 2 to relays there is an octal Bus Transmitter/Receiver (74LS245). Setting Enable pin to low and the Direction Input (DIR) pin to high, bus A data is transmitted to bus B.

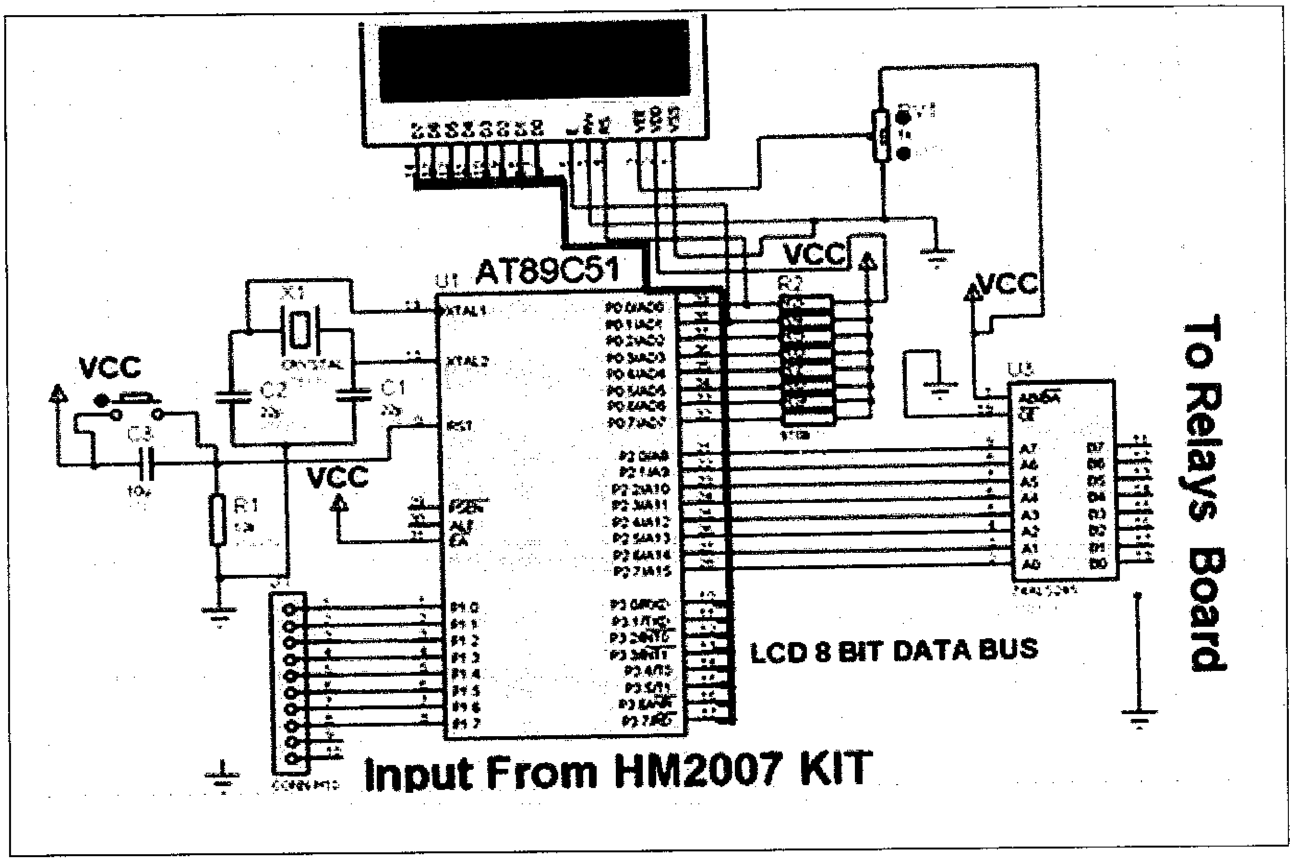


Figure 3.5 Microcontroller and LCD schematic

The microcontroller is programmed in such a way that a relay will be switched on and then off right after. It is similar like the time a button on a remote control is pressed and then released. This type of programming is just for the buttons with one action, like Power button. It is just needed to be pressed once to turn on or off the TV. However for the other buttons such as Channel up/down or Volume up/down, in a way it can be summed up as scroll buttons, the programming is different. When a scroll command is called the relay will be on and it will be switched on and off till the time the user says Stop. Thus a Stop command should be added during voice training of the system.

3.4 Relays (Connector Circuit)

A relay is an electrically operated switch. Current flowing through the coil of the relay creates a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and they are double throw (changeover) switches. As it was described previously relays are used to replace the manual activation of remote control's buttons. Figure 3.6 is the relay's circuit. The coil of a relay passes a relatively large current, typically 30mA for a 12V relay. Most ICs (chips) cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil.

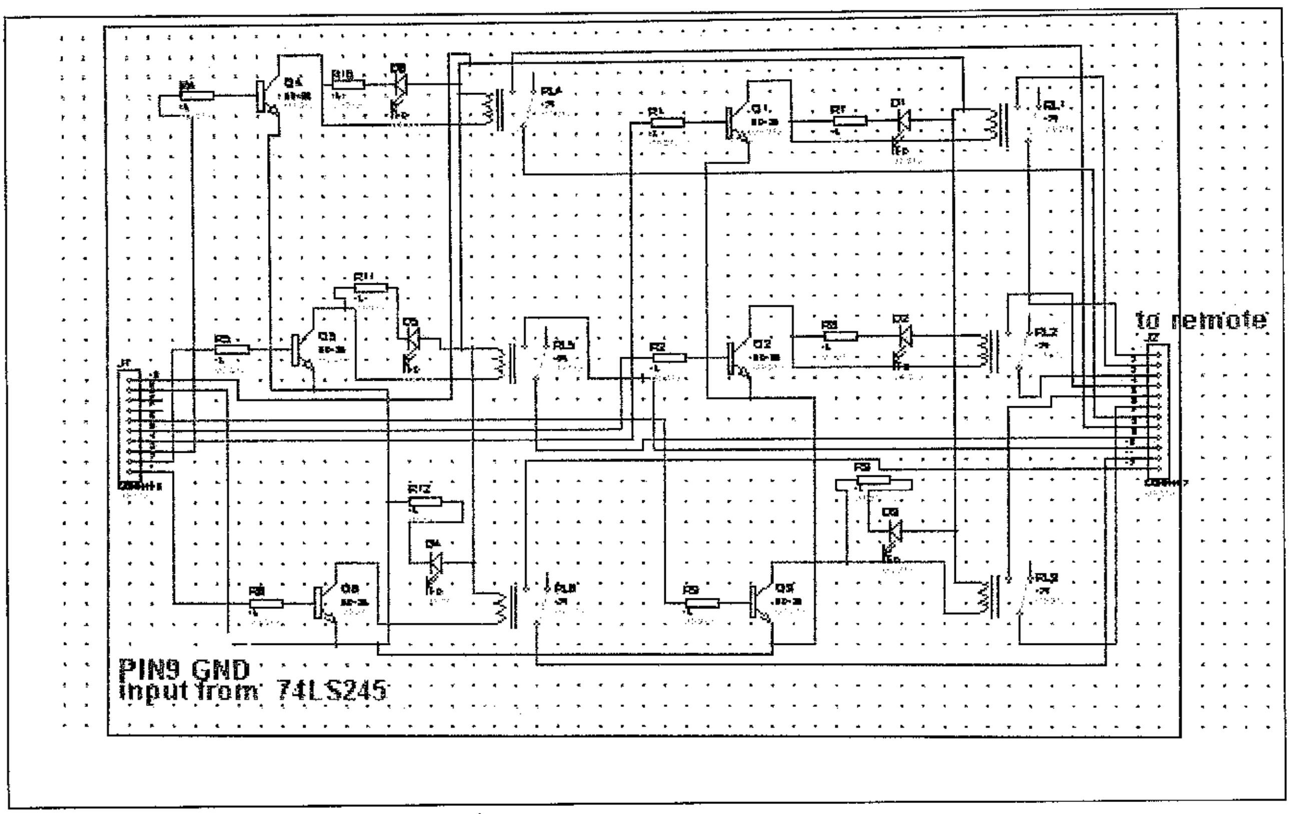


Figure 3.6 Relays Circuit Schematic

As the figure shows beside each relay there is LED to indicate the status of the relay. When the LED turns on, it indicates that the relay has switched on.

CHAPTER 4

RESULTS AND DISCUSSION

This remote control (prototype) is equipped with six voice commands. Calling commands depends on the phrase that has been recorded by the user. The prototype is illustrated in figure 4.1. As the figure shows, three main sections are connected to each other using connector. On the left hand side is the HM 2007 voice recognition circuit, and the circuit in middle contains microcontroller and its outputs, relays and LCD.

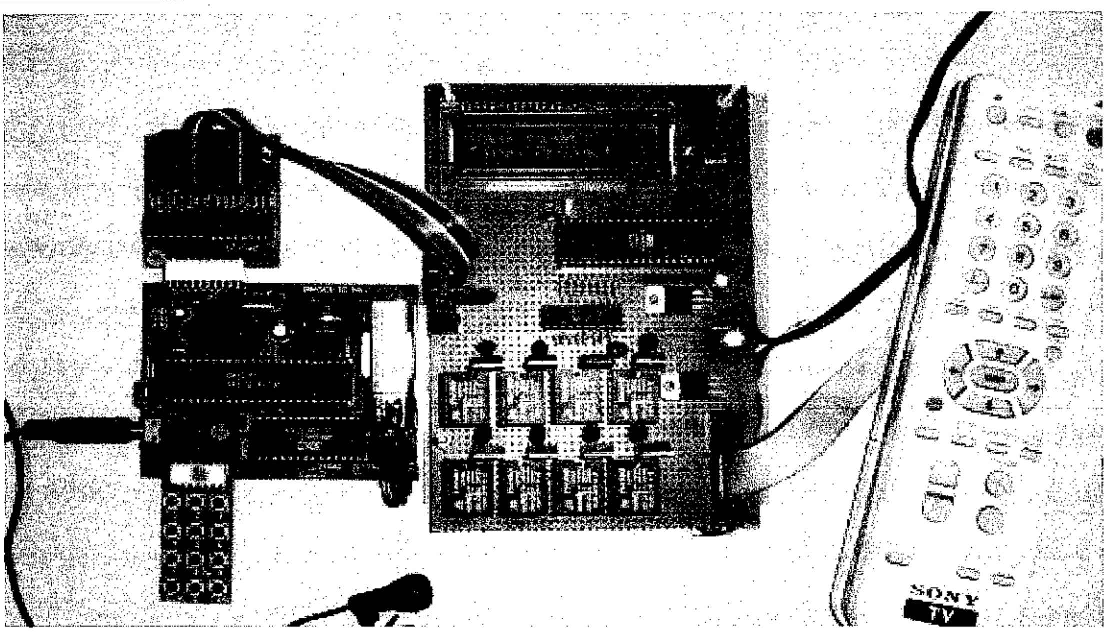


Figure 4.1 Project Prototype (Product)

In following sections the behavior of the circuit for all six voice commands will be demonstrated and described in details.

4.1 Changing Volume

In order to increase/decrease the volume of TV, the user should press the button continuously until it reaches the desired level. Here instead of pressing any button the user speak the phrase which has been recorded previously and the corresponding relay gets activated. Relay's LED turns on and off, indicating that the volume is increasing or decreasing. Calling the word recorded for stop function (any word stop, break, pause or any other preferred words) will stop the volume from increasing or decreasing. Figure 4.2 shows the behavior of the circuit when the volume up command has been called. As it can be seen the 7-segment displays show 01 which is the command identifier. Also the respective LED turns on and off until the stop function is called by the user which has been displayed on the LCD to show that volume is increasing and call stop when the desired level has been reached.

The behavior of the circuit during decreasing the volume is shown in Figure 4.3. The 7-segment displays show 02 which is the command identifier as it was defined in the program burnt on the microcontroller. Like the previous command the respective LED turns on and off indicating that the volume is decreasing. As it can be verified the LCD displays volume down and call stop to give instruction to the user. Thus when the volume is stopped from decreasing the led turns off and the microcontroller waits for the next input command.

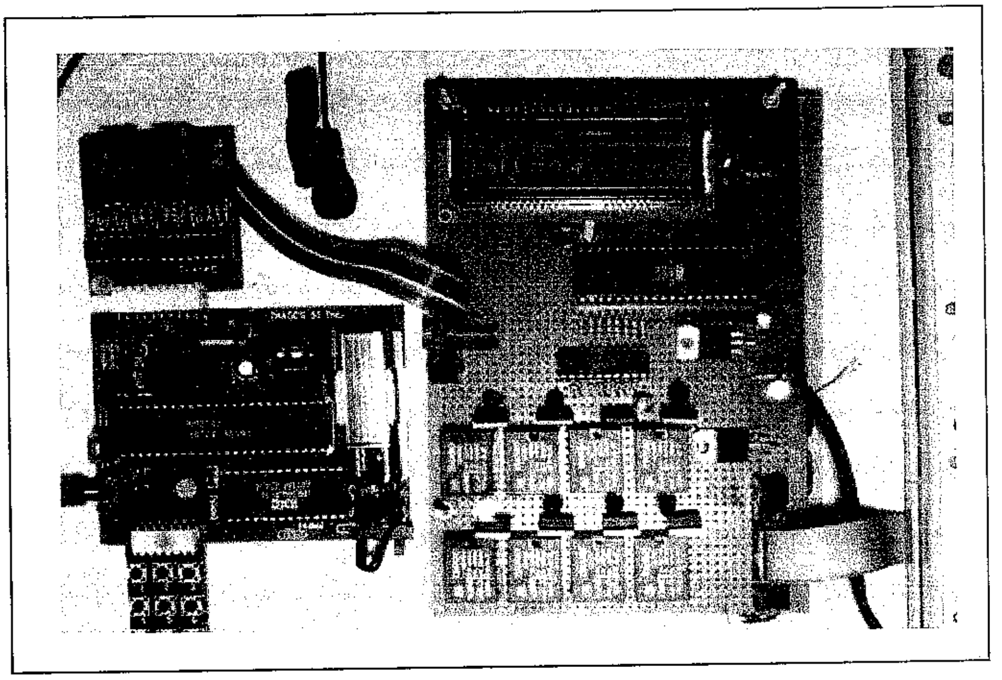


Figure 4.2 Circuit Behaviors of Volume up Command

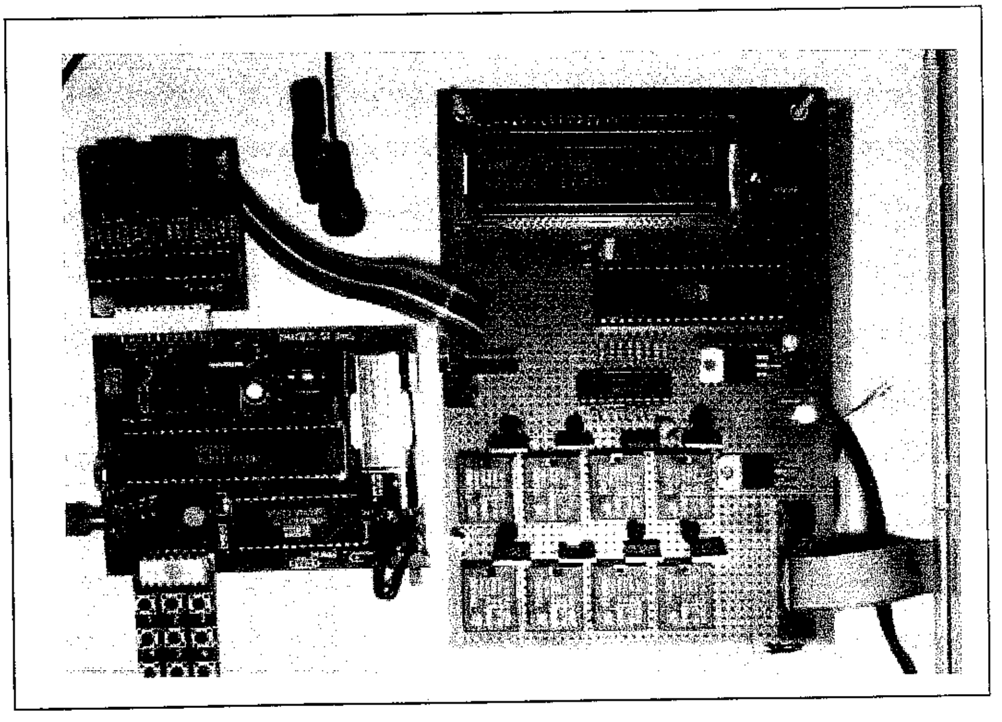


Figure 4.3 Circuit Behaviors of Volume down Command

4.2 Power ON/OFF

The TV is turned on when the power button is pressed once on the remote control. If the button is pressed again, the television turns off. Here instead of pushing the button calling the recorded phrase for power is just needed. Once the user speaks the phrase recorded for power the corresponding relay switches on and off. Thus the indicator (LED) turns on and off right after. As it was described before each input command resides in the memory slot until a new input command is received from HM2007 circuit. Since turning off the TV uses the same command identifier, the user should call the stop function first and then the phrase recorded for power. Thus if power is needed to be called twice right after each other, calling the stop function in between is necessary. As the Figure 4.4 shows, the command identifier is 03 as expected. The power command is displayed on the LCD and corresponding LED turns on and off right after.

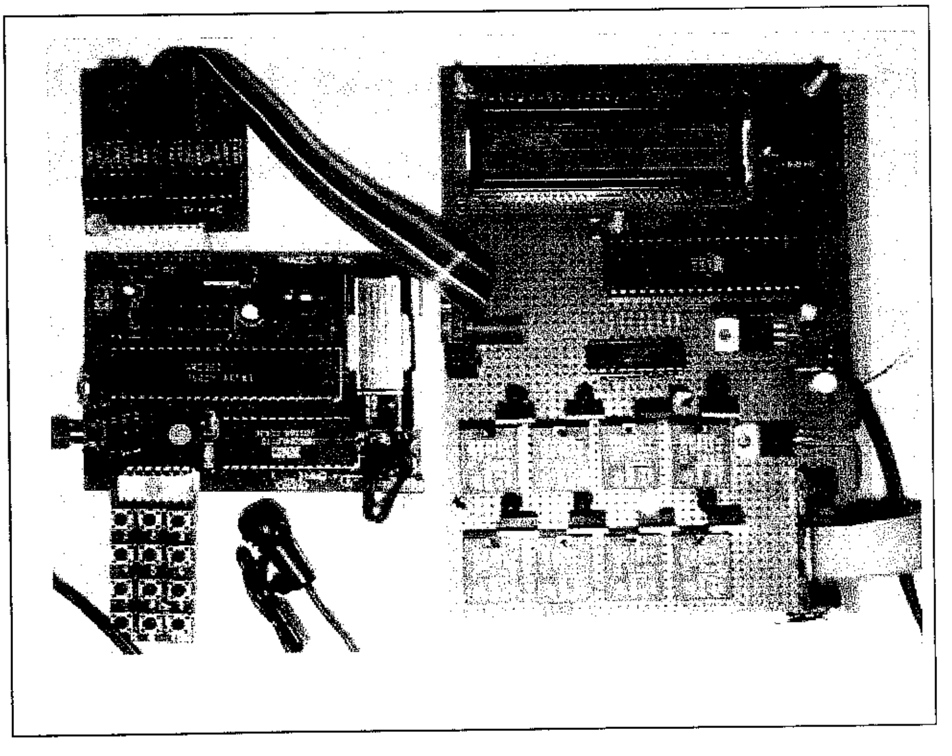


Figure 4.4 Circuit Behaviors of Power Command

4.3 Changing Channel

The changing channel procedure is the same as the volume which has been described in 4.2. Figures 4.5 and 4.6 illustrate the circuit behavior for changing the channel to higher level and changing to lower level respectively. When the user says the recorded phrase for either changing the channel to higher/lower level the corresponding relay as well as the LED turns on and off until the stop function is called. As the figures confirm the command identifiers are 04 and 05 for channel up and channel down functions respectively. The command has been displayed on LCD which is asked from the user to call stop.

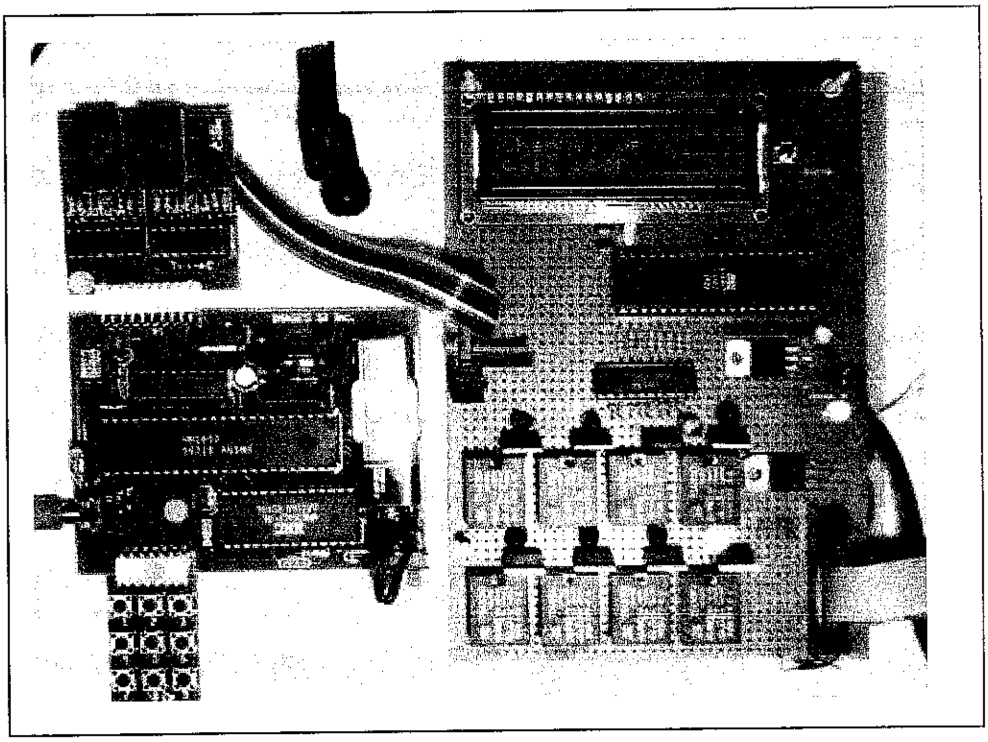


Figure 4.5 Circuit Behaviors of Channel up Command

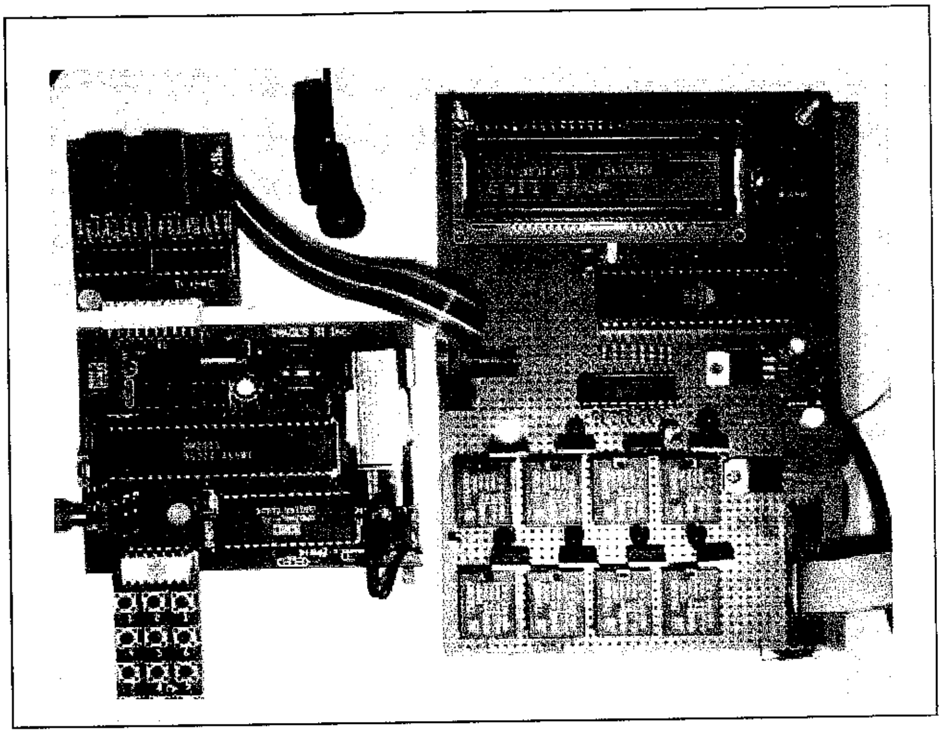


Figure 4.6 Circuit Behaviors of Channel down Command

4.4 Muting Volume

Mute button is the same as power button. Since changing the mute requires the same command identifier, the user should call the stop function first and then the phrase recorded for power. As the Figure 4.7 shows 06 is the command identifier for muting function and the command is displayed on LCD.

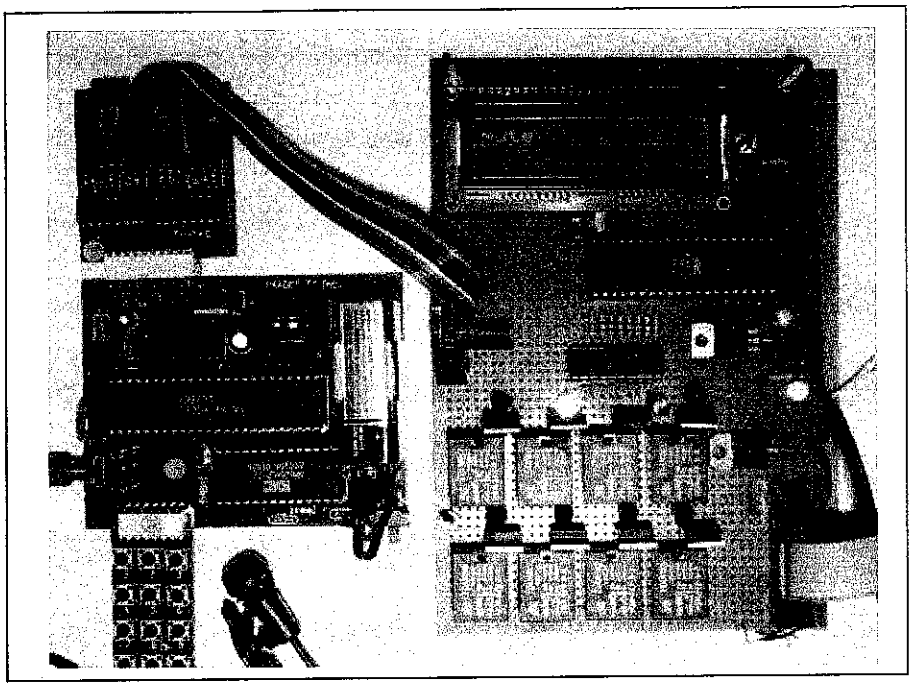


Figure 4.7 Circuit Behaviors of Mute Command

4.5 Errors Display

Sometimes the command said by the user can not be matched with any of the recorded words either because it is not properly pronounced or a word is called which has not been recorded before, so the system will respond with the command identifier displayed on 7-segment as 77. This command identifier according to HM2007 data sheet means that no match is found for the input. In order to clarify this for the users to correct themselves, no match will be displayed on the LCD screen as showed in Figure 4.8.

Also during recording session since HM2007 chip has two options of choosing words with duration of max 0.96 seconds or 1.92 seconds, two more errors may occur. Word chosen for the command is either too long or too short for the limited duration allocated for this matter. Figures 4.9 (a) and (b) illustrate the displays on the LCD screen for these errors.

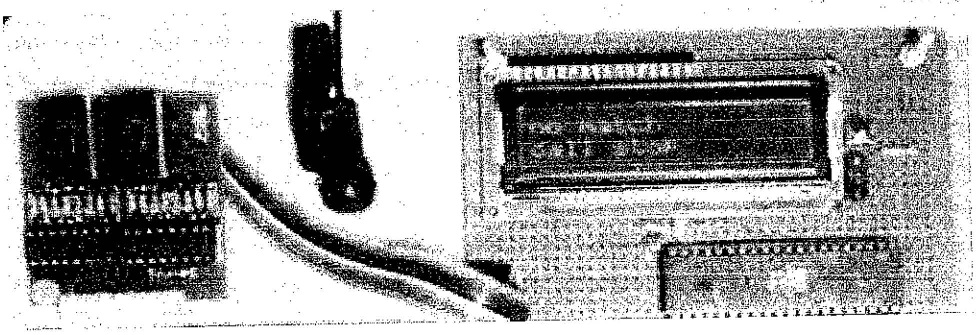
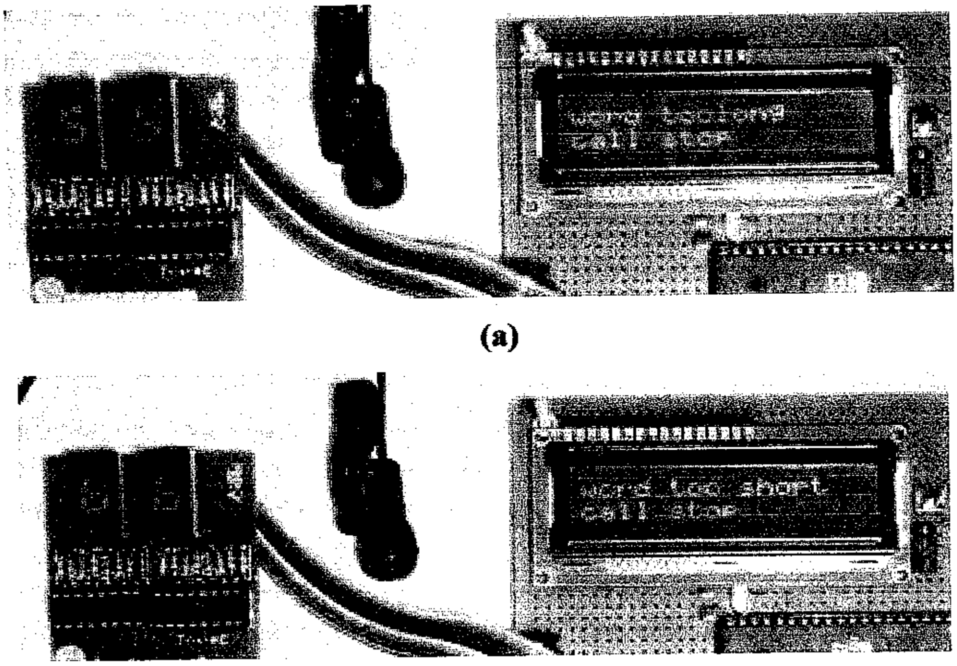


Figure 4.8 No Match Error Display



(a)

(b)

Figure 4.9 (a) Word too Long, and **(b)** Word too Short errors display

4.6 Selecting Suitable Target Phrases

When choosing the target words, it is better not to confuse the speech recognition circuit by selecting couple of homonyms. Homonyms are words that sound alike and because of their like sounding nature the can confuse the system. Also Stress and excitement alters ones voice. This affects the accuracy of the circuit's recognition. These factors should be kept in mind to achieve the high accuracy possible from the circuit.

Microcontroller is programmed in such way that recognizes the LSD from the HM2007 output as the command identifier. Thus for better performance the four word spaces are chosen so that they all have the same Least Significant Digit (LSD). Doing this the words can be recognized by just decoding the least significant digit on the digital display.

Word spaces 01, 11, 21 and 31 are allocated to the first target word. The Most Significant Digit (MSD) is dropped by the interfacing circuits. By decoding the LSD number, in this case 1 of "X1" (where X is any number) we can recognize the target word. We can continue in this manner until all the words are recorded.

It's good to use different people when training the target word to make it more speaker independence. This will enable the system to recognize different voices, intonations and accents of the target word. To have the most robust and accurate system possible, training target words using one voice with different intonations and accents of the target word.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This project provides an interesting experience in learning several subjects combined together such as voice recognition system, IR remote controls and programming a microcontroller. The conventional TV remote control has been modified in such away that it allows user to simply speak desired command such as turning on/off, volume up/down and etc, through the microphone instead of pressing any buttons. This project is targeted for electronic gadgets in general and a television set in particular. Thus the voice-activated remote control can be applied for any home appliances which work with a remote control. This project could be so helpful for the old people as well as people with disabilities who have difficulties in living their life independently.

5.2 Recommendation.

Some recommendation can be made in order to improve the project in different aspects:

1. The ordinary remote control used in this project can be switched to a universal remote control. In this way by speaking through the remote not only the television can be controlled but also other devices such as VCR, air condition, stereo, etc can be controlled simultaneously.

2. Increasing the transmit range by adding a set of Bluetooth receiver/transmitter to the system. However the Bluetooth class should be chosen carefully in order not to cause interference with any neighboring Bluetooth devices.

3. Using more robust system for the voice recognition system, in order to get better overall performance.

REFERENCES

[1] Corporate History. "**Five Decades of Channel Surfing: History of the TV Remote Control**". < http://www.zenith.com/sub_about/about_corp_history.html>. Retrieved on April 13, 2007.

[2] Philip Downey. **"Dead medium: Obsolete TV-type remote control techniques".** December 18th, 2000. < http://www.deadmedia.org/notes/49/497.html>. Retrieved on January 30, 2007.

[3] History of Buttons. "**Zenith Space Command**". Creative Commons Attribution, Revised March 28th, 2007. <http://www.historyofthebutton.com>. Retrieved on April 13, 2007.

[4] Phil Ament. "**Remote Control**". Troy MI: ©1997-2007 The Great Idea Finder, Revised December 5, 2006. <http://www.ideafinder.com/history/inventions>. Retrieved on January 30, 2007.

[5] Marshall Brain. **"Inside a TV Remote Control"**. <http://electronics.howstuffworks.com/inside-rc.htm>. Retrieved on January 30, 2007.

[6] Garfinkel (1998). "**History of Speech and Voice Recognition and Transcription Software**", Dragon Systems. <http://www.dragon-medicaltranscription.com/history_speech_recognition.html >. Retrieved on September 15, 2006.

[7] Garfinkel (1998). "**A Timeline & History of Voice Recognition Software**", Dragon Systems. <http://www.dragon-medicaltranscription.com>. Retrieved on September 15, 2006.

[8] John Lovine. "**Speech Recognition**", < http://www.talkingelectronics.com/John/SpeechRecognition.html >. Retrieved on September 20, 2006.

[9] Adams, Russ. Sourcebook of Automatic Identification and Data Collection, Van Nostrand Reinhold, New York, 1990

[10] Images Scientific Instruments Inc. "**Build a Speech Recognition Circuit**", <http://www.imagesco.com/articles/hm2007/SpeechRecognitionTutorial01.html>, Retrieved on August 10, 2006.

[11] Readings in **SPEECH RECOGNITION**, Alex Waibel & Kai – fu Lee, Ch4 pg113, Template-based approach, Morgan Kaufman Publishers, Inc. San Mateo, California.

[12] Jim Baumann (Fall 1993). "**Virtual Reality and Telepresence Course**", Voice Recognition,<http://www.hitl.washington.edu/scivw/EVE/I.D.2.d.VoiceRecognition.html>, Retrieved on August 15, 2006.

[13] HM2007 Speech Recognition chip datasheet

APPENDICES

APPENDIX A-1 PROJECT MILESTONE SEMESTER 1

No.	Detail/Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Selection of Project Topic															
	• Topic assigned to students															
2	Preliminary Research Work															
	• Background															
	• Objectives															
	• Literature reviews and theory															
3	Submission of Preliminary Report				●											
4	Project work															
	• Reference/Literature															
	• Collecting information about components availability															
5	Submission Progressive Report								●							
6	Continue with studies															
	• Understanding HM2007 Voice Recognition Chip															
	• Learning about Microcontroller															
	• Organizing all the researches to use in reports															
7	Submission of Draft Report												●			
8	Submission of interim report														●	
9	Oral Presentation															●

Process ● Suggested Milestone

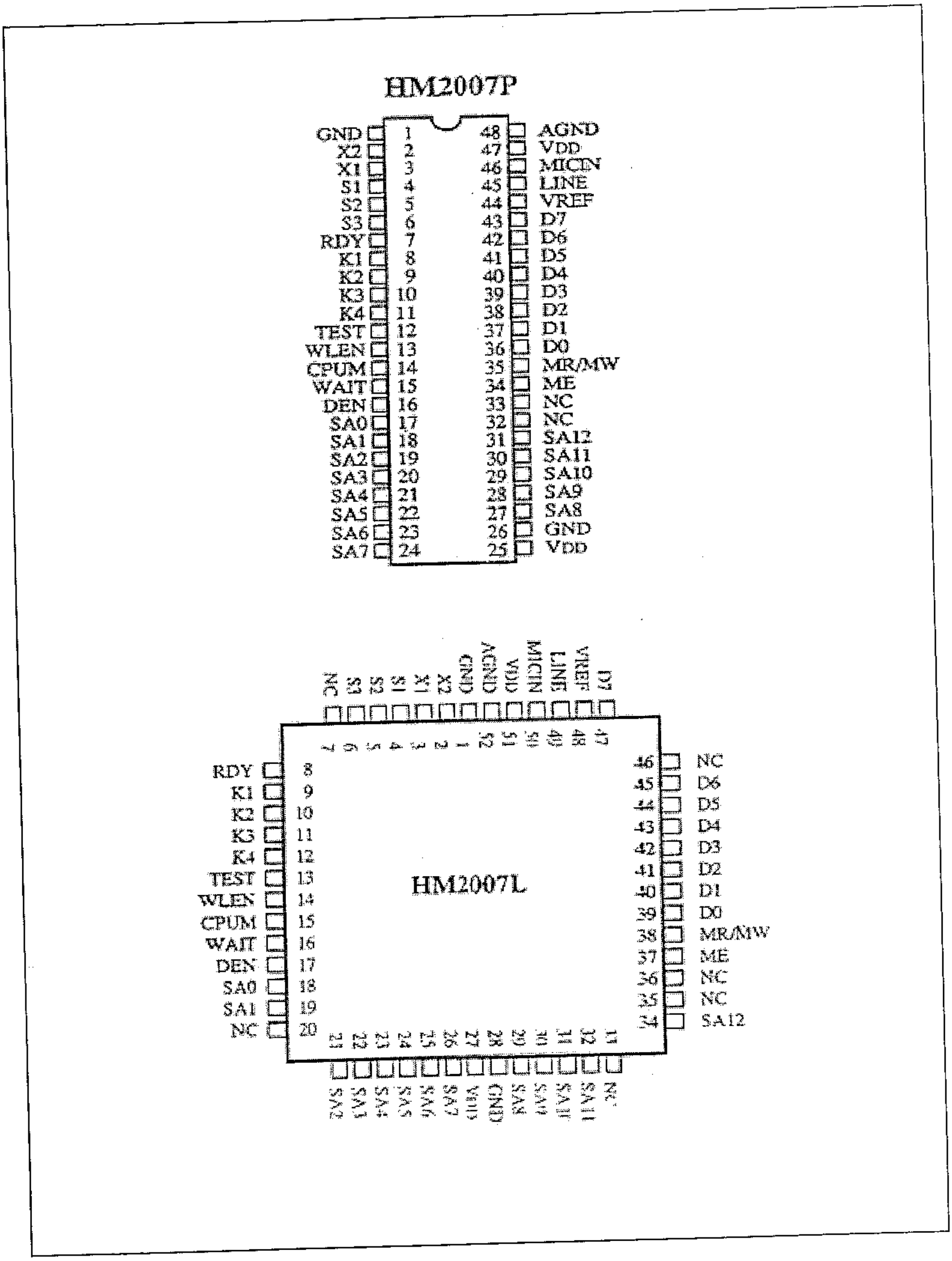
APPENDIX A-2 PROJECT MILESTONE SEMESTER 2

No.	Detail/Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	19
1	Project Work Continue	■	■	■												
	• Practical/Laboratory Work															
2	Submission of Progress Report 1				●											
3	Project Work Continue				■	■	■	■								
	• Practical/Laboratory Work															
4	Submission of Progress Report 2								●							
5	Project Work Continue								■	■	■	■				
	• Practical/Laboratory Work															
6	Submission of Dissertation Final Draft												●			
7	Submission of Final Report (Soft Cover)														●	
8	Oral Presentation															●
9	Submission of Final Report (Hard Cover)															●

■ Process ● Suggested Milestone

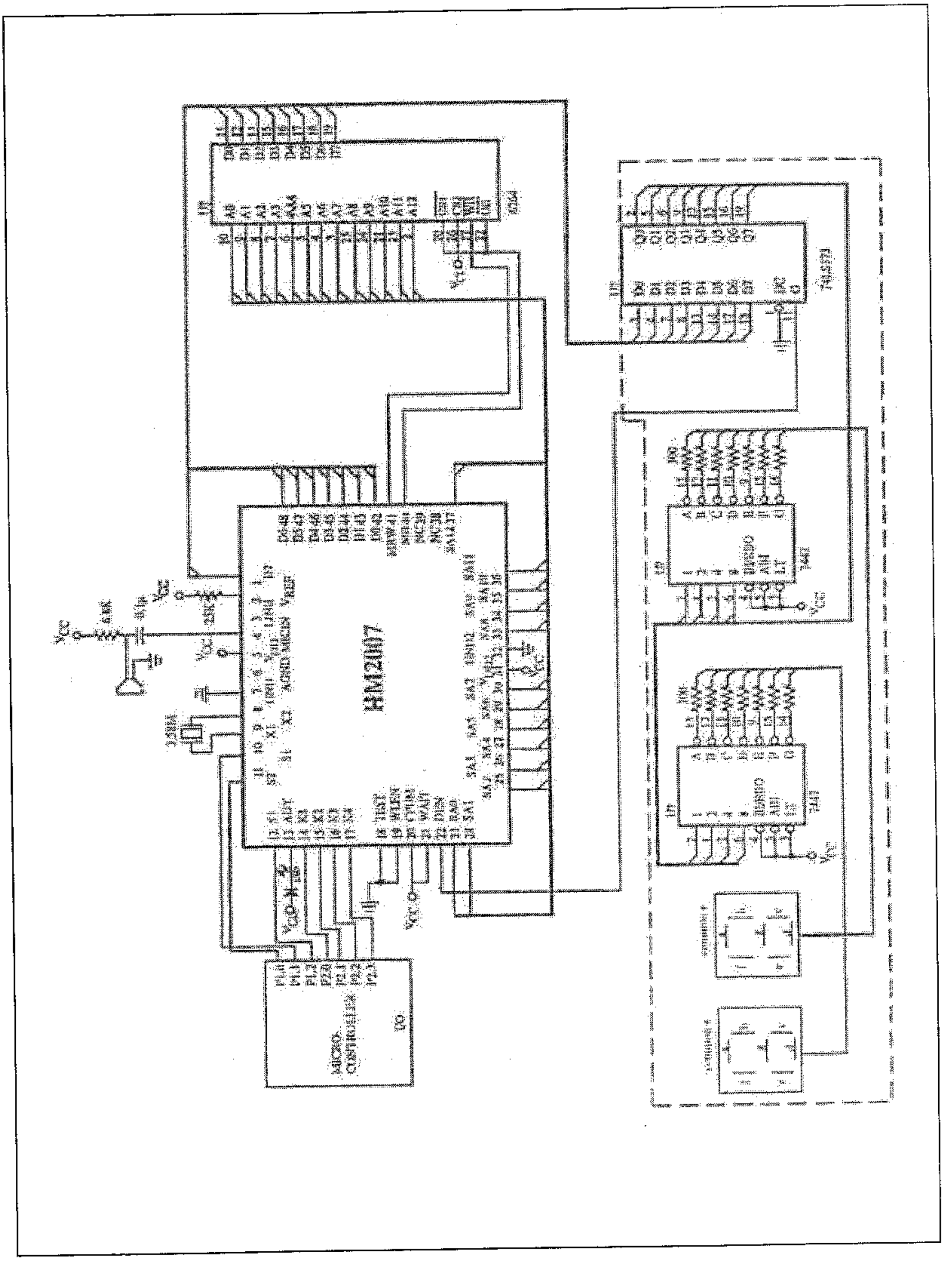
APPENDIX B

HM2007 PIN CONFIGURATIONS



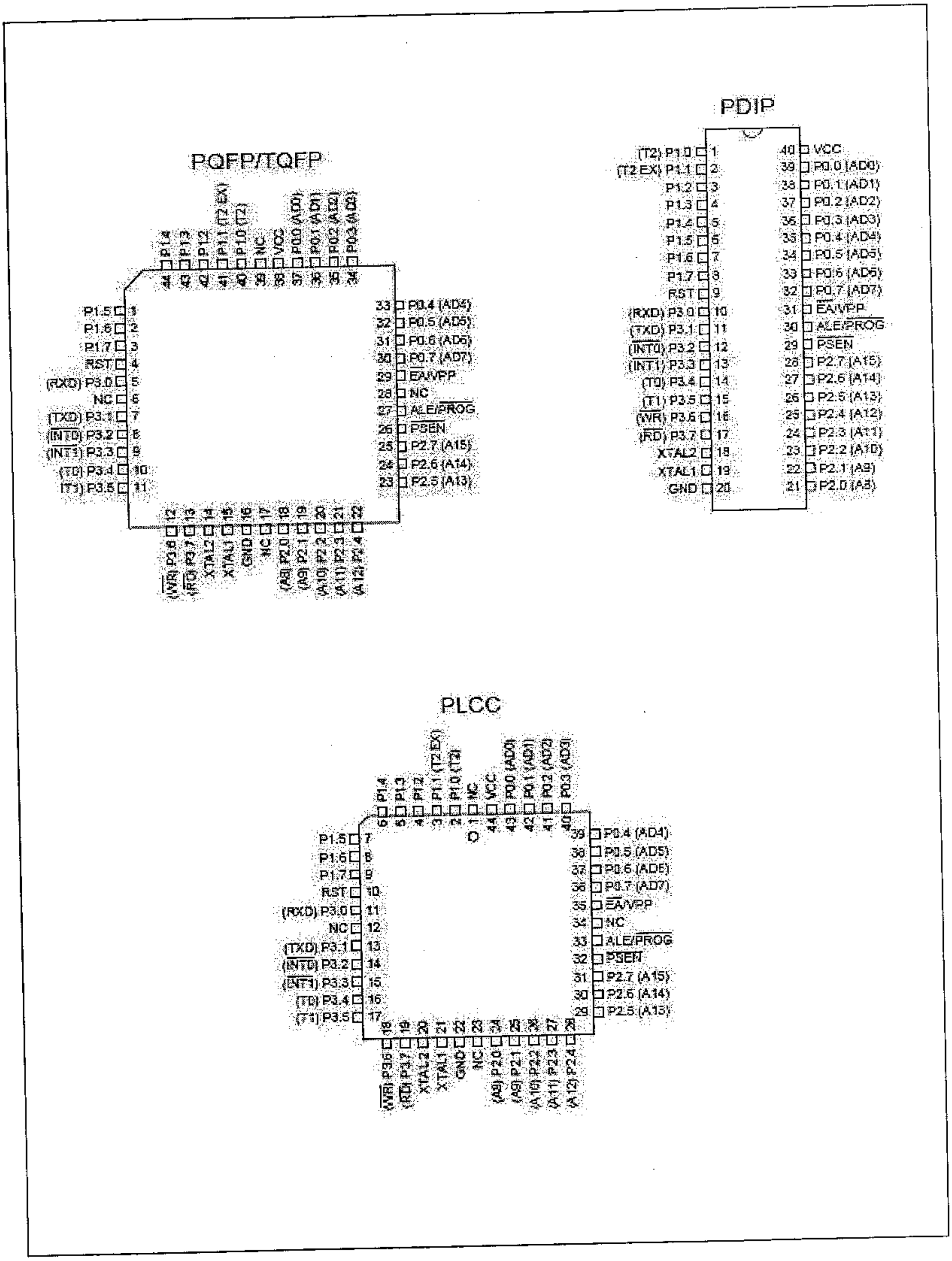
APPENDIX C

HM2007 CPU MODE



APPENDIX D

AT89C52 PIN CONFIGURATIONS



APPENDIX E

ASSEMBLY CODES

```
        org     00h
        jmp     main
        org     30h

   main:        LCALL   SETUP
                mov     35h,#00
                mov     p1,#0ffh
                mov     P2,#00H
                LCALL   ID
                LCALL   TI1
process:        mov     a,p1
                cjne    a,35h,dff1
                jmp     process
   dff1:        mov     r1,a
                mov     35h,a
                cjne    a,#55h,xq1
                lcall   long
                lcall   ti1
                jmp     process
    xq1:        cjne    a,#66h,xq2
                lcall   short
                lcall   ti1
                jmp     process
    xq2:        cjne    a,#77h,xq3
                lcall   nomatch
                lcall   ti1
                jmp     process
    xq3:        mov     a,r1
                anl     a,#00001111b
                cjne    a,#0fh,start
                jmp     process
  start:        lcall   check
                jmp     process
  check:        cjne    a,#1,zs1
                lcall   vup
                jmp     zs8
    zs1:        cjne    a,#2 ,zs2
                lcall   vdown
                jmp     zs8
    zs2:        cjne    a,#3 ,zs3
                lcall   power
                jmp     zs8
    zs3:        cjne    a,#4 ,zs4
                lcall   chaup
                jmp     zs8
    zs4:        cjne    a,#5,zs5
                lcall   chadown
                jmp     zs8
    zs5:        cjne    a,#6,zs8
                lcall   mute
                jmp     zs8
```

```
zs8:      nop
          ret
vup:      lcall  vup1
          setb   p2.0
          lcall  stop
          clr    p2.0
          ret
vdown:    lcall  vdown1
          setb   p2.1
          lcall  stopdown
          clr    p2.1
          ret
power:    lcall  power1
          setb   p2.2
          lcall  ti1
          clr    p2.2
          ret
chaup:    lcall  chaup1
          setb   p2.3
          lcall  stopch
          clr    p2.3
          ret
chadown:  lcall chadown1
          setb   p2.4
          lcall  stopchdown
          clr    p2.4
          ret
mute:     lcall   mute1
          setb    p2.5
          lcall   ti1
          clr     p2.5
          ret
stop:     mov   a,p1
          mov   r1,a
          cjne  a,#55h,xq11
          lcall long
          lcall stp
          lcall ti1
          jmp    stop
xq11:     cjne  a,#66h,xq12
          lcall  short
          lcall  stp
          lcall  ti1
          jmp   stop
xq12:     cjne  a,#77h,xq13
          lcall nomatch
          lcall  stp
          lcall  ti1
          jmp    stop
xq13:     mov   a,r1
          anl   a,#00001111b
          cjne  a,#01h,start1
          clr    p2.0
          lcall ti1
          setb  p2.0
          lcall ti1
          jmp   stop
```

```
start1:     cjne  a,#9,ds11
            jmp   ds2
ds11:       jmp   stop
ds2:        nop
            ret
stopdown:   mov   a,p1
            mov   r1,a
            cjne  a,#55h,xq111
            lcall long
            lcall stp
            lcall til
            jmp   stopdown
xq111:      cjne  a,#66h,xq112
            lcall short
            lcall stp
            lcall til
            jmp   stopdown
xq112:      cjne  a,#77h,xq113
            lcall nomatch
            lcall stp
            lcall til
            jmp   stopdown
xq113:      mov   a,r1
            anl   a,#00001111b
            cjne  a,#02h,start11
            clr   p2.1
            lcall til
            setb  p2.1
            lcall til
            jmp   stopdown
start11:    cjne  a,#9,ds111
            jmp   ds21
ds111:      jmp   stopdown
ds21:       nop
            ret
stopch:     mov   a,p1
            mov   r1,a
            cjne  a,#55h,xq119
            lcall long
            lcall stp
            lcall til
            jmp   stopch
xq119:      cjne  a,#66h,xq129
            lcall short
            lcall stp
            lcall til
            jmp   stopch
xq129:      cjne  a,#77h,xq139
            lcall nomatch
            lcall stp
            lcall til
            jmp   stopch
xq139:      mov   a,r1
            anl   a,#00001111b
            cjne  a,#04h,start19
            clr   p2.3
            lcall til
```

```
                setb  p2.3
                lcall til
                jmp   stopch
start19:        cjne  a,#9,ds119
                jmp   ds29
ds119:          jmp    stopch
ds29:           nop
                ret
stopchdown:     mov   a,p1
                mov   r1,a
                cjne  a,#55h,xq1111
                lcall long
                lcall stp
                lcall til
                jmp    stopchdown
xq1111:         cjne  a,#66h,xq1112
                lcall  short
                lcall  stp
                lcall  til
                jmp   stopchdown
xq1112:         cjne  a,#77h,xq1113
                lcall nomatch
                lcall  stp
                lcall  til
                jmp    stopchdown
xq1113:         mov   a,r1
                anl   a,#00001111b
                cjne  a,#05h,start111
                clr   p2.4
                lcall til
                setb  p2.4
                lcall til
                jmp   stopchdown
start111:       cjne  a,#9,ds1111
                jmp   ds211
ds1111:         jmp    stopchdown
ds211:          nop
                ret
ID:             mov    a,#01h
                lcall  com
                mov    a,#80h
                lcall  com
                mov    a,#'s'
                lcall  dat
                mov    a,#'a'
                lcall  dat
                mov    a,#'r'
                lcall  dat
                mov    a,#'e'
                lcall  dat
                mov    a,#' '
                lcall  dat
                mov    a,#'b'
                lcall  dat
                mov    a,#'a'
                lcall  dat
                mov    a,#'y'
```

```
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#0c0h
        lcall   com
        mov     a,#'F'
        lcall   dat
        mov     a,#'i'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'P'
        lcall   dat
        mov     a,#'r'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'j'
        lcall   dat
        mov     a,#'e'
        lcall   dat
        mov     a,#'c'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        ret
setup:  mov     a,#38h
        lcall   com
        mov     a,#0ch
        lcall   com
        mov     a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        ret
  com:  mov     p3,a
        clr     p0.0   ;rs
        setb    p0.1
        clr     p0.1
        mov     21,#2
HERE1:  mov     20h,#255
HERE0:  DJNZ    20,HERE0
        DJNZ    21,HERE1
        ret
  dat:  mov     p3,a
        setb    p0.0   ;rs
        setb    p0.1
        clr     p0.1
        MOV     21,#2
```

```
     h1:  MOV    20,#255
    RE0:  DJNZ   20,RE0
          DJNZ   21,h1
          ret
POWER1:   mov    a,#01h
          lcall  com
          mov    a,#80h
          lcall  com
          mov    a,#'P'
          lcall  dat
          mov    a,#'O'
          lcall  dat
          mov    a,#'W'
          lcall  dat
          mov    a,#'E'
          lcall  dat
          mov    a,#'R'
          lcall  dat
          RET
   vup1:  mov    a,#01h
          lcall  com
          mov    a,#80h
          lcall  com
          mov    a,#'v'
          lcall  dat
          mov    a,#'o'
          lcall  dat
          mov    a,#'l'
          lcall  dat
          mov    a,#' '
          lcall  dat
          mov    a,#'u'
          lcall  dat
          mov    a,#'p'
          lcall  dat
          mov    a,#0c0h
          lcall  com
          mov    a,#'c'
          lcall  dat
          mov    a,#'a'
          lcall  dat
          mov    a,#'l'
          lcall  dat
          mov    a,#'l'
          lcall  dat
          mov    a,#' '
          lcall  dat
          mov    a,#'s'
          lcall  dat
          mov    a,#'t'
          lcall  dat
          mov    a,#'o'
          lcall  dat
          mov    a,#'p'
          lcall  dat
          RET
```

```
  stp: mov    a,#0c0h
       lcall  com
       mov    a,#'c'
       lcall  dat
       mov    a,#'a'
       lcall  dat
       mov    a,#'l'
       lcall  dat
       mov    a,#'l'
       lcall  dat
       mov    a,#' '
       lcall  dat
       mov    a,#'s'
       lcall  dat
       mov    a,#'t'
       lcall  dat
       mov    a,#'o'
       lcall  dat
       mov    a,#'p'
       lcall  dat
       ret
vdown1: mov    a,#01h
       lcall  com
       mov    a,#80h
       lcall  com
       mov    a,#'v'
       lcall  dat
       mov    a,#'o'
       lcall  dat
       mov    a,#'l'
       lcall  dat
       mov    a,#' '
       lcall  dat
       mov    a,#'D'
       lcall  dat
       mov    a,#'o'
       lcall  dat
       mov    a,#'w'
       lcall  dat
       mov    a,#'n'
       lcall  dat
       mov    a,#0c0h
       lcall  com
       mov    a,#'c'
       lcall  dat
       mov    a,#'a'
       lcall  dat
       mov    a,#'l'
       lcall  dat
       mov    a,#'l'
       lcall  dat
       mov    a,#' '
       lcall  dat
       mov    a,#'s'
       lcall  dat
       mov    a,#'t'
       lcall  dat
```

```
        mov     a,#'o'
        lcall   dat
        mov     a,#'p'
        lcall   dat
        RET
chaup1: mov     a,#01h
        lcall   com
        mov     a,#80h
        call    com
        mov     a,#'c'
        lcall   dat
        mov     a,#'h'
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'e'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'u'
        lcall   dat
        mov     a,#'p'
        lcall   dat
        mov     a,#0c0h
        lcall   com
        mov     a,#'c'
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'s'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'p'
        lcall   dat
        RET
chadown1: mov   a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        mov     a,#'c'
        lcall   dat
        mov     a,#'h'
```

```
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'e'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'d'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'w'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#0c0h
        lcall   com
        mov     a,#'c'
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'s'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'p'
        lcall   dat
        ret
mute1 : mov     a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        mov     a,#'m'
        lcall   dat
        mov     a,#'u'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'e'
        lcall   dat
        RET
```

```
tI1:    mov   20h,#8
        mov   tmod,#01
her:    mov   tl0,#00
        mov   th0,#00
        setb  tr0
        jnb   tf0,$
        clr   tr0
        clr   tf0
        djnz  20h,her
        ret
long:   mov     a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        mov     a,#'w'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'r'
        lcall   dat
        mov     a,#'d'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'l'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'n'
        lcall   dat
        mov     a,#'g'
        lcall   dat
        RET
short:  mov     a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        mov     a,#'w'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'r'
        lcall   dat
        mov     a,#'d'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'o'
        lcall   dat
```

```
        mov     a,#'o'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'s'
        lcall   dat
        mov     a,#'h'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#'r'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        RET
nomatch: mov     a,#01h
        lcall   com
        mov     a,#80h
        lcall   com
        mov     a,#'n'
        lcall   dat
        mov     a,#'o'
        lcall   dat
        mov     a,#' '
        lcall   dat
        mov     a,#'m'
        lcall   dat
        mov     a,#'a'
        lcall   dat
        mov     a,#'t'
        lcall   dat
        mov     a,#'c'
        lcall   dat
        mov     a,#'h'
        lcall   dat
        RET


        end
```