International Journal on Recent Researches in Science, Engineering & Technology (IJRRSET)
A Journal Established in early 2000 as National journal and upgraded to International journal in 2013 and is in existence for the last 10 years. It is run by Retired Professors from NIT, Trichy. Journal Indexed in JIR, DIIF and SJIF.
Research Paper
Available online at: www.jrrset.com
UGC Approved Journal No: 45483
Received on: 10.11.2017 Published on: 30.11.2017 Pages : 22-29

ISSN (Print) : 2347-6729
ISSN (Online) : 2348-3105
Volume 5, Issue 11
Novmber 2017.
JIR IF : 2.54
DIIF IF : 1.46
SJIF IF : 1.329

DESIGN AND SIMULATION OF SPEECH RECOGNITION FOR VEHICLE CONTROL

Aryamol Sudhakaran, Karthika V S and Smith P S
Assistant Professor, Department of Electrical and Electronics Engineering, Sri Vellappally Natesan College of Engineering, Mavelikara, Kerala, India

Abstract
The main goal is to implement "hearing" sensor and also the Speech Recognition Technology to the Mobile robot such that it is capable to interact with human through Spoken Natural Language (NL). We have chosen Mobile Robot, because this type of robot is getting popular as a service robot in the social context, where the main challenge is to interact with human. Uncertainty is a major problem for navigation systems and interaction with humans in a natural way would be a means of overcoming difficulties with localization. It has numerous applications in the field of aerospace and medical fields. Military applications include setting radio frequency, commanding an auto – pilot system and setting steer – point coordinates. A voice navigated toy car is created which is able to distinguish at least 3 words with more than 90% accuracy in real time and respond accordingly to the control signals. The processor section and the toy car section are designed to be having wireless communication (IR Sensor).

Keywords: Natural language, speech recognition, wireless communication

1. Introduction

Observer is a system that assigns labels to events occurring in the environment. If the labels belong to sets without a metric distance it is said that the result of the observation is a classification and the labels belong to one of several sets. If, on the contrary, the sets are related by a metric, it is said that the result is estimation and the labels belong to a metric space. According to these definitions, the goal of this work is to devise an observer that describes air pressure waves using the labels contained by some written language. Because these labels are not related by a metric, the desired process is a classification.

If an efficient speech recognizer is produced, a very natural human-machine interface would be obtained. By natural one means something that is intuitive and easy to use by a person, a method that does not require special tools or machines but only the natural capabilities that every human possesses [1]. Such a system could be used by any person able to speak and will allow an even broader use of machines, specifically computers. This potentiality promises huge economical rewards to those who learn to master the techniques needed to solve the problem, and explains the surge of interest in the field during the last 10 years. If an efficient speech recognition machine is enhanced by natural language systems and speech producing techniques, it would be possible to produce computational applications that do not require a keyboard and a screen. This would allow incredible

miniaturization of known systems facilitating the creation of small intelligent devices that can interact with a user through the use of speech. An example of this type of machines is the Carnegie Mellon University JANUS system that does real time speech recognition and language translation between English, Japanese and German for customers of different countries to interact without worrying about their language differences. The economical consequences of such a device would be gigantic. Phonemes and written words follow cultural conventions. The speech recognizer does not create its own classifications and has to follow the cultural rules that define the target language. This implies that a speech recognizer must be taught to follow those cultural conventions. The speech recognizer cannot fully self-organize. It has to be raised- in a society.

A speech recognition system, sampling a stream of speech at 8 kHz with 8 bit precision, receives a stream of information at 64 Kbits per second as input. After processing this stream, written words come out at a rate of more or less 60 bits per second. This implies an enormous reduction in the amount of information while preserving almost all of the relevant information. A speech recognizer has to be very efficient in order to achieve this compression rate (more than 1000:1). In order to improve its efficiency, a recognizer must use as much a priori knowledge as possible. It is important to understand that there are different levels of a priori knowledge. The topmost level is constituted by a priori knowledge that holds true at any instant of time. The lowermost extreme is formed by a priori knowledge that only holds valid within specific contexts. In the specific case of a speech recognizer, the physical properties of the human vocal tract remain the same no matter the utterance, and a priori knowledge derived from these properties is always valid [1]. On the other extreme, all the a priori knowledge collected about the manner in which a specific person utters words is only valid when analysing the utterances of that person. Based on this fact, speech recognizers are normally divided into two stages.

- The Feature Extractor (FE) block generates a sequence of feature vectors, a trajectory in some feature space that represents the input speech signal. The FE block is the one designed to use the human vocal tract knowledge to compress the information contained by the utterance. Since it is based on a priori knowledge that is always true, it does not change with time.
- The Recognizer performs the trajectory recognition and generates the correct output word. Since this stage uses information about the specific ways a user produce utterances, it must adapt to the user.

2. System Analysis and Developmental Workflow

The block diagram below shows the development workflow and it consists of the following steps:
1. Speech acquisition: Using an on chip ADC & TIMER module
2. Speech analysis: Sampled sound signals are passed thru multiple Band Pass Filters.
3. Finger print analysis: Euclidean distance formulae for comparing two data's.
4. Output interface operation: H-bridge, PIC & PWM section for DC motor interface

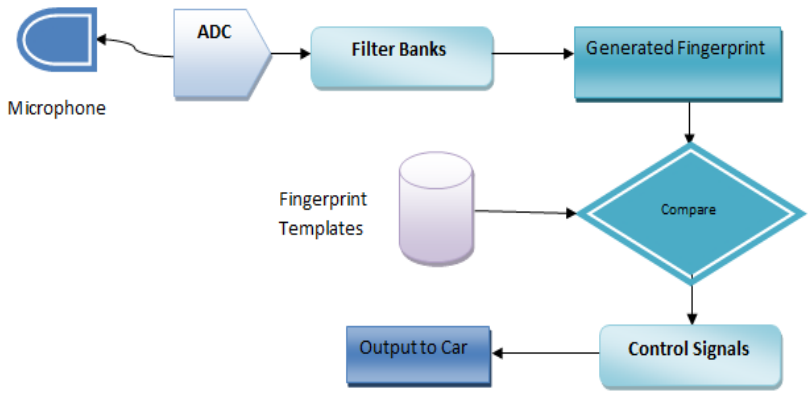

Fig. 1. Voltage and frequency control

A. ACQUIRING THE SPEECH SIGNALS

One of the most significant blocks of speech recognition systems are preamplifiers and signal conditioning circuits. Preamplifier circuits are placed to reduce the effects of noise and interference. It is used to boost the signal strength without significantly degrading the SNR ratio. There are two stages for the microphone analog circuit: The sampling frequency (F_S) chosen to be 4 KHz. The hardware circuit consists of a high pass filter, an amplifier and a low pass filter.

1. Microphone bias supply & High Pass Filter\ DC blocker

The first stage, an RC high pass filter uses a 0.1µF capacitor and a 10KΩ resistor. The cut-off frequency (F_L), $F_L=1/(2\pi R_2C_4)$

This is designed to be approx. 159.15Hz. It is near to the frequency 150 Hz which is the lower limit of the human voice spectrum. This also helps to cut off the surrounding noise frequencies of the range 50 Hz. The 10k resistor R2 also provides the DC-bias supply, approx. 2.5v for the electret microphone or the condenser microphone[2].The above mentioned R_2C_4 networks also block the microphone DC bias supply from entering the amplifier stage.

2. Gain stage

A two-stage amplifier is needed to obtain the desired voltage swing compatible with ADC. The gain of each stage is $-R_6/R_4$, i.e. 10KΩ/1KΩ which is -10, thus providing the total gain of -100, the negative sign implies that the amplifier is an inverting type. The input of the amplifier was coupled to a DC voltage of around 2.5V. This was necessary to center the amplified signal in the A/D range for the MCU. We implemented this with a voltage divider with R_3R_4 network, 2 x 10kΩ resistors.

3. Anti-aliasing filer (Low Pass Filter)

The last stage comprises of a simple Low Pass Filter or in this context an anti-aliasing filter. The stage comprises of a Low Pass RC network, R_9C_5 a 0.01uF capacitor and a 5.6kΩ resistor,so that the cut off frequency will be $F_L=1.5xF_{max}=3KHz$.

Where, Fmax= Fs/2 , so $FL=1/2\pi R_9C_5$, so our -3db point will be at 3Khz, thereby ensuring no higher frequencies greater than F_{max} enter ADC, and cause aliasing.

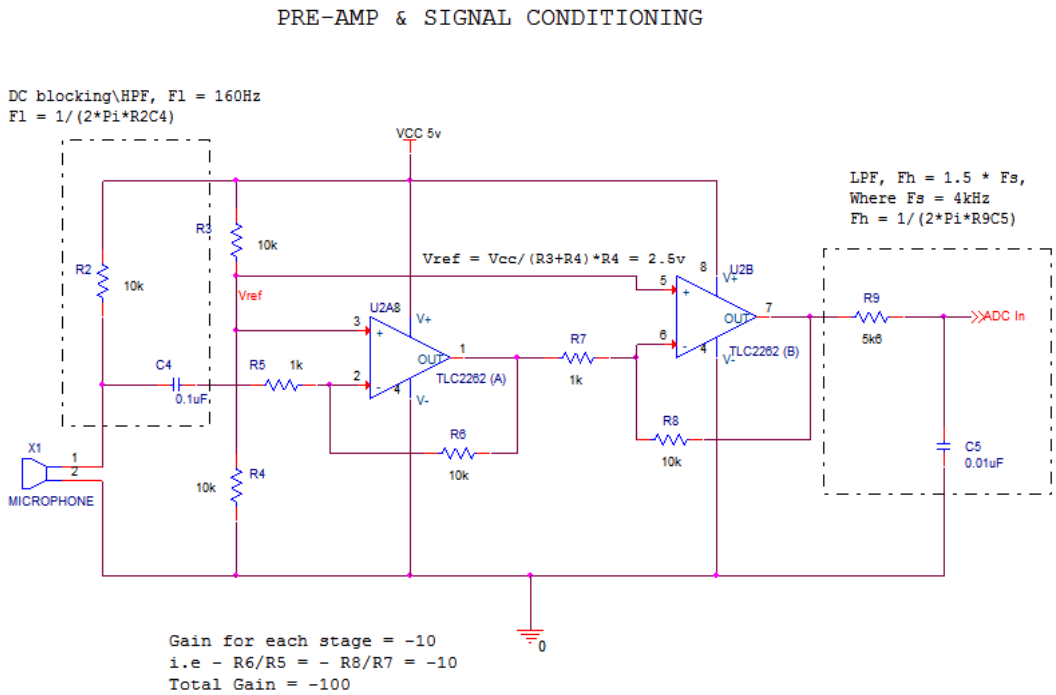

Fig. 2. Amplifier and Signal Conditioning Circuit Diagram

B. ANALYZING THE ACQUIRED SPEECH USING DIGITAL FILTER

We use digital filters to implement the feature extraction of the acquired speech signals. A digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. In our project we need digital filters because the processing is done in a microcontroller which is where the algorithm is implemented. Program Instructions (software) running on the microprocessor implement the digital filter by performing the necessary mathematical operations on the numbers received from the ADC of the DSPIC we use. Since digital filters use a sampling process and discrete-time processing, they experience latency (the difference in time between the input and the response), which is almost irrelevant in analog filters. A variety of mathematical techniques may be employed to analyze the behaviour of a given digital filter. Many of these analysis techniques may also be employed in designs, and often form the basis of a filter specification. Digital filters are often described and implemented in terms of the difference equation that defines how the output signal is related to the input signal. Here, we use IIR (Infinite Impulse Response) filters because they have an impulse response function that is non-zero over an infinite length of time. In digital IIR filters, the output feedback is immediately apparent in the equations defining the output.

Design of digital IIR filters is heavily dependent on that of their analog counterparts because there are plenty of resources, works and straightforward design methods concerning analog feedback filter design while there are hardly any for digital IIR filters. As a result, usually, when a digital IIR filter is going to be implemented, an analog filter (e.g. Chebyshev filter, Butterworth filter, Elliptic filter) is first designed and then is converted to a digital filter by applying discretization techniques such as Bilinear transform or Impulse invariance. We use Bilinear Transformation because it is one of the fastest. The filter cut – off is determined by the MATLAB analysis and the coefficients are also generated by MATLAB. These coefficients are programmed on to the processor for generating and analysing the fingerprints.

3. Simulation, Results and Discussion

A. ALGORITHM IMPLEMENTATION

We used Tor Aamodt algorithm to implement the digital IIR filters. The coefficients are generated using MATLAB and the cut off of the filters are determined by the MATLAB analysis. We need to recognise five words: Go, Left, Right, Backwards and Halt. The frequency components associated with each of these words are analysed and it was concluded that to distinctly differentiate between these words minimum 4 filters are required. The following spectrum shows the amplitude plot the word analysed with respect to time:

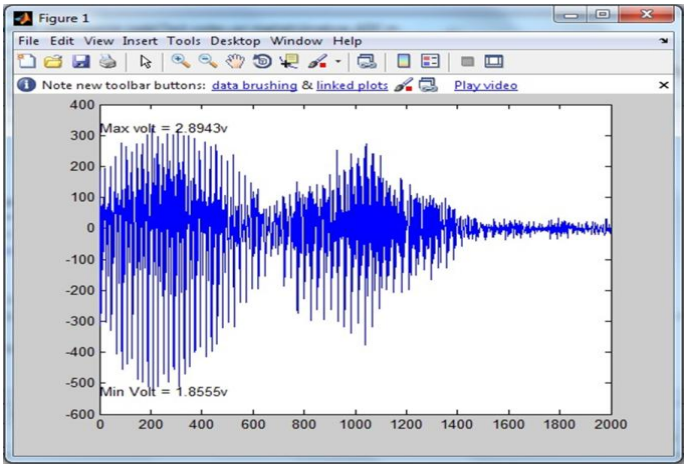

Fig. 3. Amplitude plot

Then the cut – off ranges of the filters are determined based on the Simulation results shown below:

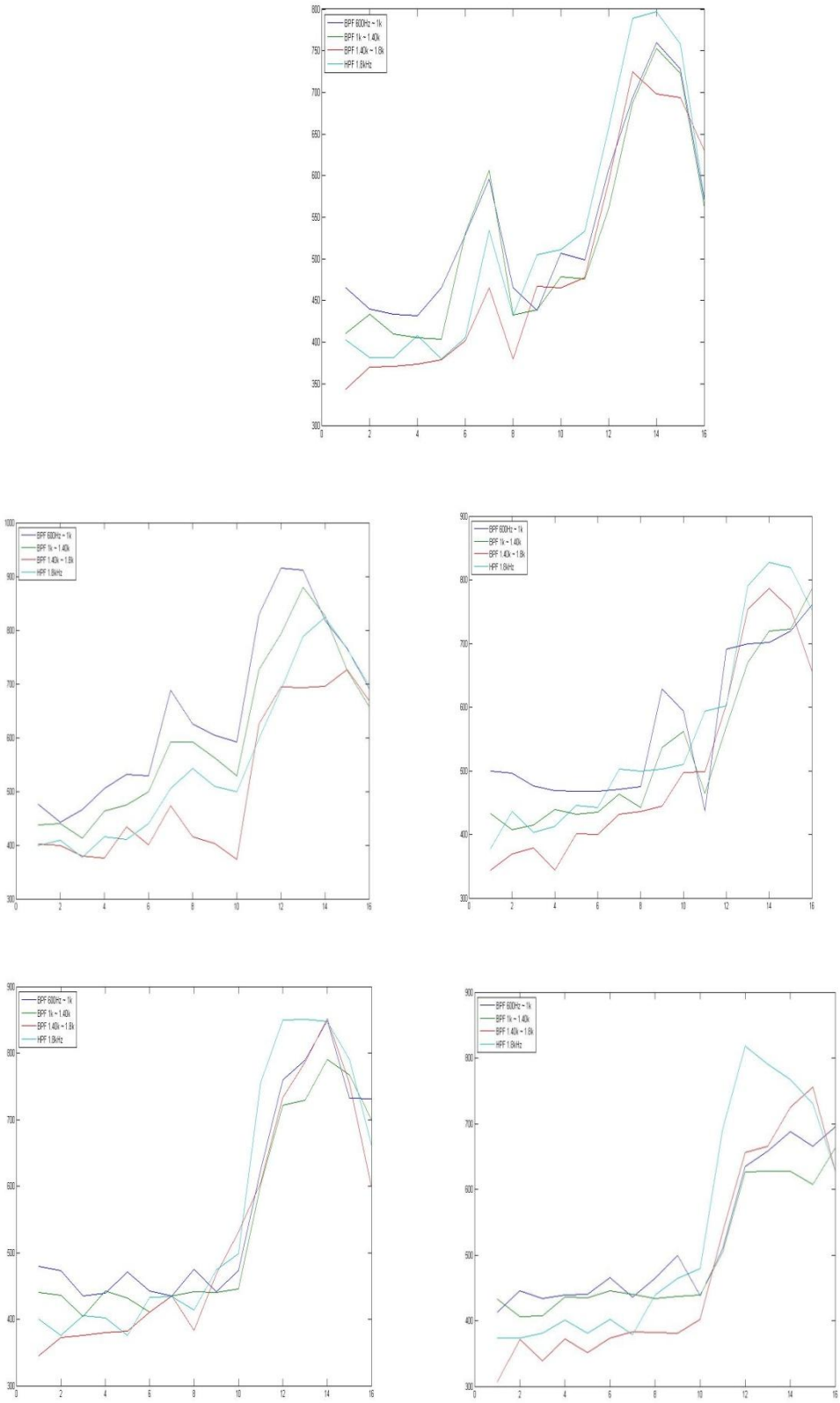

Fig.4. Cut off ranges(for LEFT, RIRHT, START, HALT, BACKWARDS)

So four digital filters [5] are programmed on to the processor having the ranges: 3 bandpass filters:

- 600 Hz to 1k
- 1k to 40 k
- 1.4 k to 1.8 k

It also consists of a high pass filter of cut – off of 1.8 kHz. In the design part, a low pass filter of 600 Hz cut – off was also designed but was avoided in the actual implementation due to higher dominance [3].

Thus the fingerprints of words are determined. Then we use Template approach to compare the real time signal with the templates. We have 2 stages: Training and Testing. In training phase, the templates are stored and in the testing phase, the algorithm is employed to determine whether there is a match found. Now when the match is found the corresponding signal is send via the RF Module and is received by the PIC in the trolley car.

B. OTA TROLLEY CAR

The Over The Air, commanding can be accomplished via any wireless communication techniques, such as Bluetooth, ZigBee or RF. For cost effectiveness we decided to stick with RF Transceiver, which supports Serial-communication at speed of maximum 2400 Baud.
The trolley car interface consists of a RX02 ASK receiver (Rx section of an RF Transceiver), PIC 16F73 for processing (an 8-bit mid range CPU from microchip®), L293D for motor interface (Quadruple half-bridge). Two motors were used for control. One for motion control and one for direction control. The interface circuit is simulated using Proteus and virtual RX terminal is used for giving the control signal. The different commands used here for the trolley are F – Go, B – Backwards, S – Halt, R – Right and L – Left. The received information is analysed by the PIC 16F73 and is given to L29D H – Bridge Motor driver [6] which gives corresponding signals to the two dc motors or servo.

The *'Activity'* LED indicates that a command is being received\processed, also serves a visual debugger. It actually 'blinks' while a data is received over micro-controller UART via RF transceiver. The Left\Right LEDs are just turn indicators.

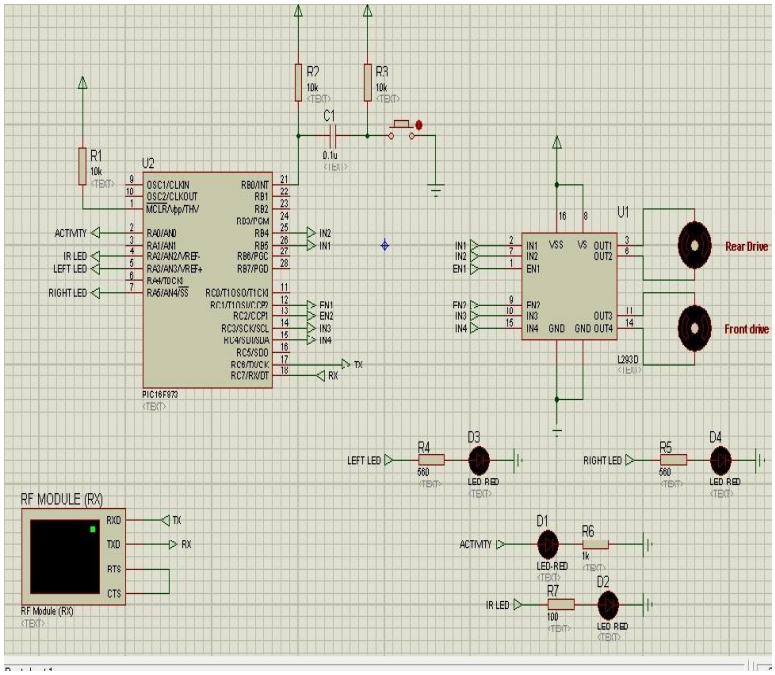

Fig.5. The Trolley Proteus model

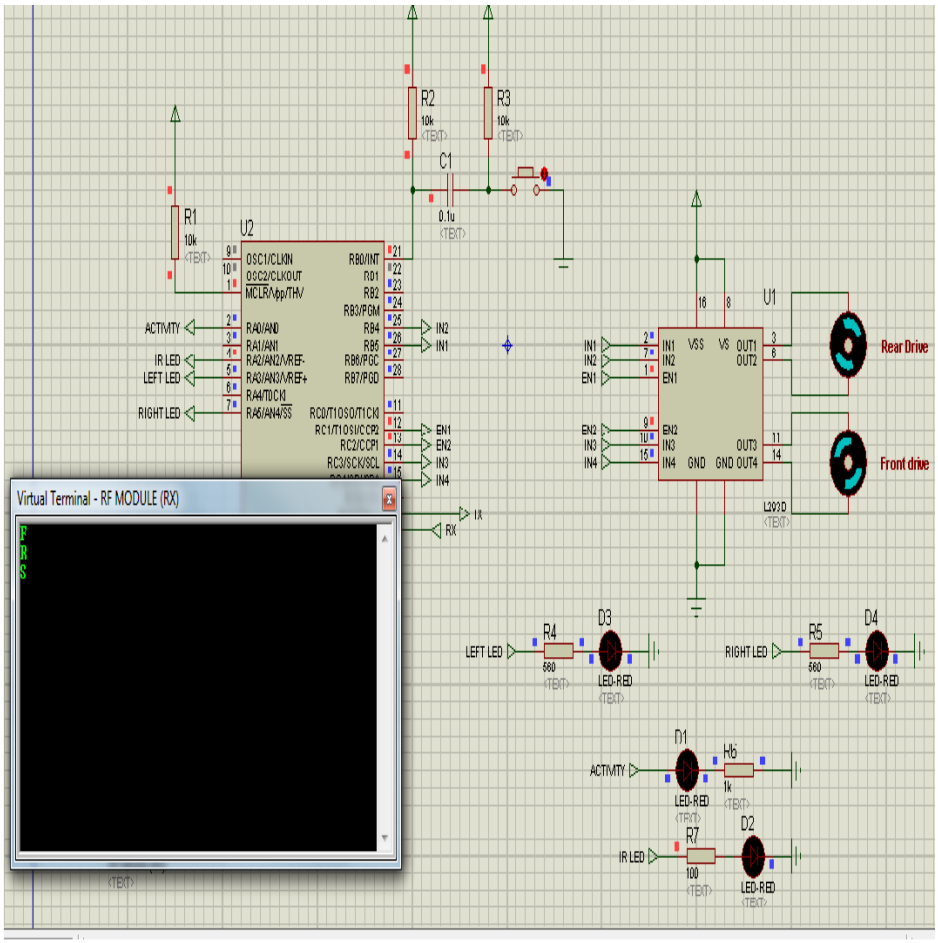

Fig.6. The RX terminal working and the car control output

6. Conclusion

The system is able to recognise efficiently 5 words: Go, Right, Left, Halt and Backwards with more than 75% accuracy. Vehicle steering was effectively coordinated based on the IR sensed control signal based on the recognised word. Both motion control and direction control is realised. There is wireless IR communication [4] with the 2 processing section: one employing the DSPIC and the other in the OTA driven trolley car. The work has these challenges:

- 95% accuracy could not be obtained owing to the following reasons:
  - Signal to noise ratio could not be enhanced as per designed.
  - Number of filters had to be reduced due to memory constraints.
  - Accuracy deteriorated due of lack of real time processing.
- Large number of vocabularies or many similar sounding words makes recognition difficult for the system.
- The system is not only sensitive to noise but also to sensitive to voice tone changes and microphone position.

References

[1] Steven T. Karris, Signals and Systems with MATLAB ® Computing and Simulink ® Modeling, Third Edition, Orchard Publications.
[2] “Digital Processing of Speech Signals” by Lawrence R. Rabiner and Ronald W. Schafer, Signal Processing Series, Pearson Education Ist edition, 1993.
[3] “Speech & Language Processing” by Daniel Jurafsky and James H. Martin, Pearson Education, 2000

[4] Kermit Sigmon, Timothy A. Davis, MATLAB PRIMER, 6th Edition
[5] Rob Williams, Real-Time Systems Development.