Design and Implementation of an Unmanned Ground Vehicle

Shreyas H, Thirumalesh H S
Department of Electrical and Electronics Engineering, SJCE, Mysore, India
Email: shreyas9693@gmail.com, hsthirumalesh@gmail.com

Abstract

There has been an increase in the rate of human crimes in today’s world prompting safety measures to be taken without risking human lives. Robots can help us achieve this goal. In this paper we present the design of an efficient and economical robot that can perform both spying and rescue operations. This robot is an Unmanned Ground Vehicle (UGV) that can monitor enemy attacks and intruders secretly as if it were a first person to map an unknown location. These robots also find application in factories to move parts around on the floor. A UGV is basically a locomotive robot that can be controlled without human presence in the vehicle. This is done via wireless networking. The robot is controlled through a dual tone multi frequency (DTMF) call made to a mobile phone placed on the robot from the operator. The HT9170B DTMF Decoder IC is used to decode the dial tone into binary sequence. The decoded signals are sent to the ATMEGA8 microcontroller which generates appropriate output signals that are predefined in the microcontroller by a program. Since the robot is controlled through mobile network, the range over which it can be controlled is significantly large. The mobile also ensures real time video transfer from the robot’s surroundings to the operator’s display unit, thereby reducing the risks involved in direct human interaction with hostile environments.

Index Terms—DTMF, UGV, video transmission.

I. Introduction

The deployment of autonomous vehicles is rapidly becoming possible in today’s world because of technological advances in networking and in miniaturization of electromechanical systems. Robots can coordinate their actions through communication networks, and can perform challenging tasks such as search and recovery operations, manipulation in hazardous environments, exploration, surveillance, and also environmental monitoring for pollution detection and estimation [1]. An unmanned ground vehicle (UGV) is basically any piece of mechanized equipment that moves across the surface of the ground without an onboard human presence [2]. The potential advantages of employing such robots in espionage are numerous. They can spy on people in ways people can’t move and from views humans can’t reach. They can perform tasks faster than humans with much more consistency and accuracy. They can also capture moments that are too fast for the human eye.
In this paper, the design and implementation of a UGV which leverages the already existing dual tone multi frequency (DTMF) technology to establish a secure, fast and reliable connection with the operator’s display unit is presented. The robot is controlled through a DTMF call made to the mobile phone placed on the robot from the operator. The HT9170B DTMF Decoder IC decodes the dial tune into binary sequence. These decoded signals are sent to the ATMEGA8 microcontroller which generates appropriate output signals that are predefined in the microcontroller by a program. The mobile is also used to transfer video in real time from the robot’s surroundings to the operator’s display unit.

The rest of this paper is organized as follows: Section II contains the hardware description of the robot, Section III contains the software description. The testing of the robot and results obtained are presented in Section IV and the conclusion is presented in Section V.

II. Hardware Description

The hardware comprises of:
1. The vehicle
2. An ATmega8 microcontroller
3. An Android mobile
4. A DTMF decoder

1. Vehicle:

The vehicle comprises of a fiber body chassis which houses two geared DC motors rated at 3V each. The two wheels are coupled to the shaft of the motor. A picture of the robot is shown in Figure 1. There is an arrangement made to place the android mobile on top of the vehicle. The agility of the vehicle is improved by making it as compact and light as possible. The motors are powered by a 5V regulated supply from a 9V DC rechargeable battery. The CD7805 IC regulates 9V DC to 5V DC [3]. The L293DNE IC is used to drive the motors by providing sufficient current for their operation [4].

Figure 1: Robot
2. ATmega8 Microcontroller:

The microcontroller used is ATmega8A-PU, an 8-bit microcontroller from Atmel [5]. It controls the robot’s overall movement. Based on the program written by the user onto this microcontroller, appropriate control signals are sent to the motors causing the robot to move as per the given command. The Atmega8 microcontroller has three PORTS, namely B, C and D. We utilize PORTB to send control output signals to the motors. PORTC receives external signals from the decoder in the form of a binary sequence. An onboard clock of frequency 4 MHz is used to provide clock pulses to the microcontroller.

3. Android Mobile:

The mobile is placed on a panel provided on top of the robot. It serves two purposes:
- It receives DTMF tone from the operator’s mobile and sends them to the DTMF decoder.
- It records the video and sends it to the user’s laptop using an android application called IP Webcam.

The photo resolution and the video resolution depend upon the quality of the camera in the phone that is placed on the robot.

4. DTMF Decoder:

The decoder module used is HT9170B [6]. The input to this decoder module is the DTMF tone. It converts the DTMF tone to its corresponding 4-bit binary sequence. As shown in the DTMF frequency matrix in Figure 2, when a key is pressed, a particular set of two frequencies superimpose over one another. For example, when ‘1’ is pressed frequencies of 697 Hz and 1209 Hz superimpose over one another and an audio tone unique to that number is generated. The decoder module recognizes the corresponding key pressed on the mobile on receiving the DTMF tone and produces a 4-bit binary sequence output.

![Figure 2: DTMF Frequency Matrix](image-url)
III. Software Description

Integrated Development Environment (IDE) AVR Studio 4 has been used to write the C-codes and convert them to HEX files which are later burned onto the Atmega8 microcontroller using the AVR programmer - Khazama. An android application called IP Webcam is used for real time video transfer from the mobile phone on the robot to the operator’s display unit. In order for this application to work, both the laptop and the mobile phone must be connected to the same network. This application provides the user with numerous modes to view the real time video such as Java and Flash. Options are also provided to enhance the stream quality and zoom in for a closer view of the surroundings. The night vision option can be used in places where there is insufficient light.

This application is first installed on the operator’s mobile phone. The IP address provided in the application is entered on the browser of the laptop, thus opening a window which enables the user to view the real time video provided by the mobile phone placed on the robot. Then the user can choose to view the video as he/she pleases. Figure 3 shows a flowchart which represents the interaction between the robot and the software.

![Flowchart - Interaction between the robot and the software](image)

Figure 3: Flowchart - Interaction between the robot and the software
IV. Testing and Results

The C-program is written such that the robot initially waits for the signal sent to it by the operator on pressing a certain key on the mobile phone. The mobile phone and the operator’s laptop are connected to the same Wi-Fi network. The operator can watch the live video delivered by the robot on his/her laptop’s browser with the help of an android application called IP Webcam. The operator has to open the link http://<local ip address>:port id on a browser. On opening the link, the android webcam server is seen. Then the operator has to select the way in which he/she wants to view video. Figures 4 and 5 represent screenshots taken on the operator’s laptop from the real time video delivered by the mobile phone on the robot in the presence of light and absence of light (night vision) respectively.

![Figure 4: Screenshot taken in the presence of light](image1)

![Figure 5: Screenshot taken in the absence of light (night vision)](image2)

On pressing a certain key from the operator’s mobile, the corresponding DTMF tone is sent to the android mobile. The DTMF tone as heard from the android mobile is sent to the DTMF MT8870 module via a 3.5mm headjack. The table shown in Figure 6 lists the actions performed by the robot on pressing the following keys on the mobile phone.

<table>
<thead>
<tr>
<th>Key Pressed</th>
<th>4-bit Binary Sequence</th>
<th>Operation performed by the robot</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0010</td>
<td>Move Forward</td>
</tr>
<tr>
<td>4</td>
<td>0100</td>
<td>Turn Left</td>
</tr>
<tr>
<td>6</td>
<td>0110</td>
<td>Turn Right</td>
</tr>
<tr>
<td>8</td>
<td>1000</td>
<td>Move Backwards</td>
</tr>
<tr>
<td>5</td>
<td>0101</td>
<td>Stop</td>
</tr>
</tbody>
</table>

![Figure 6: Operations performed by the robot according to the key pressed](image3)

The robot can be controlled at any point of time as the video displayed on the laptop’s browser is real-time. In order to test the robot, both the laptop and the mobile phone on the robot were
connected to a Wi-Fi network and the range of video transmission was found to be roughly 30 meters in radius. The working of the robot is represented in the form of a block diagram in Figure 7.

![Block Diagram - UGV Control](image)

Figure 7: Block Diagram - UGV Control

The android mobile that is on placed on the robot is connected to the HT9170B DTMF decoder via a 3.5mm audio jack. Once a key is pressed on the operator’s mobile a unique DTMF tone is generated which is decoded into a binary sequence by the HT9170B DTMF decoder. The microcontroller comprehends the binary sequence and causes the robot to operate accordingly. The mobile is configured to operate in auto-answer mode. Referring to the application circuit of the Decoder IC HT9170B in Figure 8, D0,D1,D2,D3 are it’s outputs which are connected to the pins 23(PC0), 24(PC1), 25(PC2), 26(PC3) of the ATmega8 microcontroller shown in Figure 9.

![Application Circuit - IC HT9170B](image)

![Pinout diagram - ATmega8A-P](image)

Figure 8: Application Circuit - IC HT9170B  
Figure 9: Pinout diagram - ATmega8A-PU
There is a negligible delay in the real time video delivered by the mobile on the robot. This can be adjusted by increasing or decreasing the stream quality. However, sometimes there is a delay of 4 seconds in the audio. This happens due to browser buffering and depends on the browser used. It can be resolved either by using the Flash renderer option provided by IP Webcam or by using a different browser.

V. Conclusion

This paper presents the design and implementation of a DTMF-based UGV. The vehicle was found to be capable of traveling over flat surfaces. It was also observed during test runs that the control signals for the UGV were received from the operator’s mobile using DTMF signals as expected. A slight delay was observed in the response time of the UGV due to the execution speed of the algorithm in the microcontroller. The video output from the mobile placed on the robot was observed to be real-time as expected, but with a negligible delay.

Future enhancements/modifications include: incorporating a wide area network (WAN) so that the range of the real-time video transfer can be extended and improving the mechanics of the robot so that it can travel over rough terrain.

References


