Introduction

A remote control vehicle is defined as any mobile device that is controlled by a means that does not restrict its motion with an origin external device. A remote controlled vehicle (RCV) differs from robot as the RCV is always controlled by a human and takes no positive action autonomously.

It is vital that a RCV should be capable of proceeding accurately to a target area; maneuvering within that area to fulfill its mission and returning equally accurately and safely to base.

Conventionally, wireless-controlled robots use RF circuits, which have the drawbacks of limited working range, limited frequency range and limited control.

Use of a mobile phone for robotic control can overcome these limitations. It provides the advantages of robust control, working range as large as the coverage area of the service provider, no interference with other controllers and up to twelve controls.

Although the appearance and capabilities of robots vary vastly, all robots share the features of a mechanical, movable structure under some form of control. The control of robot involves three distinct phases: perception, processing and action.

Generally, the preceptors are sensors mounted on the robot, processing is done by the on-board microcontroller or processor, and the task (action) is performed using motors or with some other actuators.
Project Overview

In this project, the robot is controlled by a mobile phone that makes a call to the mobile phone attached to the robot. In the course of a call, if any button is pressed, a tone corresponding to the button pressed is heard at the other end of the call. This tone is called ‘dual-tone multiple-frequency’ (DTMF) tone.

The robot perceives this DTMF tone with the help of the phone stacked in the robot.

The received tone is processed by the ATmega16 microcontroller with the help of DTMF decoder MT8870. The decoder decodes the DTMF tone into its equivalent binary digit and this binary number is sent to the microcontroller.

The microcontroller is preprogrammed to take a decision for any given input and outputs its decision to motor drivers in order to drive the motors for forward or backward motion or a turn.

The mobile that makes a call to the mobile phone stacked in the robot acts as a remote. So this simple robotic project does not require the construction of receiver and transmitter units.

DTMF signaling is used for telephone signaling over the line in the voice-frequency band to the call switching centre. The version of DTMF used for telephone tone dialing is known as Touch-Tone.
DTMF assigns a specific frequency (consisting of two separate tones) to each key so that it can easily be identified by the electronic circuit. The signal generated by the DTMF encoder is a direct algebraic summation, in real time, of the amplitudes of two sine(cosine)waves of different frequencies, i.e., pressing ‘5’ will send a tone made by adding 1336 Hz and 770 Hz to the other end of the line.

**DTMF Keypad Frequencies (With Sound Clips)**

<table>
<thead>
<tr>
<th></th>
<th>1209 Hz</th>
<th>1336 Hz</th>
<th>1477 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>697 Hz</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>770 Hz</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>852 Hz</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>941 Hz</td>
<td>*</td>
<td>0</td>
<td>#</td>
</tr>
</tbody>
</table>
**Origin of idea**

The idea of controlling the robot was originated in my mind when I called the customer care number from my cell phone. I was asked to dial different number for various options available during the call. Each number was assigned to a specific option. This led me to think that dialing a number during a call can be easily identified by the computer and it uses that signal for the specific command.

Later on, when I studied the tone-based command system in an engineering subject, the concept of remote control through tones was very much clear.

This led to the conclusion that any mechanical device can be operated to move through the cell phone.

The launch of 3G technology in Indian mobile communication industry encouraged me to put on the camera for video streaming at a low price.

The objective was to develop a robot whose movement can be controlled wirelessly without any range limitation.
Components used

- IC1 - MT8870 DTMF decoder
- IC2 - ATmega16 AVR microcontroller
- IC3 - L293D motor driver
- IC4 - 74LS04 NOT gate
- D1 - 1N4007 rectifier diode

Resistors (all ¼-watt, ±5% carbon):
- R1, R2 - 100-kilo-ohm
- R3 - 330-kilo-ohm
- R4-R8 - 10-kilo-ohm

Capacitors:
- C1 - 0.47μF ceramic disk
- C2, C3, C5, C6 - 22pF ceramic disk
- C4 - 0.1μF ceramic disk

Miscellaneous:
- XTAL1 - 3.57MHz crystal
- XTAL2 - 12MHz crystal
- S1 - Push-to-on switch

- M1, M2 - 6V, 50-rpm geared DC motor
- Battery - 6V, 4.5Ah battery
- Two metal clamps
- Wooden sheet
- IC holder
- Screws
- Connecting leads
- MultiMeter
- Drilling machine
- Soldering kit
- Wheels
- PCB of required circuit
- Two cell phones
- Headset of one cell phone
- Tool kit
Cell phone operated land rover
DTMF Decoder

About MT8870

The M-8870 is a full DTMF Receiver that integrates both band split filter and decoder functions into a single 18-pin DIP or SOIC package.

Manufactured using CMOS process technology, the M-8870 offers low power consumption (35 mW max) and precise data handling. Its filter section uses switched capacitor technology for both the high and low group filters and for dial tone rejection. Its decoder uses digital counting techniques to detect and decode all 16 DTMF tone pairs into a 4-bit code.

External component count is minimized by provision of an on-chip differential input amplifier, clock generator, and latched tri-state interface bus. Minimal external components required include a low-cost 3.579545 MHz color burst crystal, a timing resistor, and a timing capacitor.

The M-8870-02 provides a “power-down” option which, when enabled, drops consumption to less than 0.5 mW. The M-8870-02 can also inhibit the decoding of fourth column digits.

Application

MT8870 DTMF decoder is used to decode telephone tones.
Pin Diagram of MT8870

* Connect to Ground
<table>
<thead>
<tr>
<th>Pin</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IN+</td>
<td>Non-inverting input. Connections to the front-end differential amplifier.</td>
</tr>
<tr>
<td>2</td>
<td>IN-</td>
<td>Inverting input.</td>
</tr>
<tr>
<td>3</td>
<td>GS</td>
<td>Gain select. Gives access to output of front-end amplifier for connection of feedback resistor.</td>
</tr>
<tr>
<td>4</td>
<td>VREF</td>
<td>Reference voltage output (nominally VDD/2). May be used to bias the inputs at mid-rail.</td>
</tr>
<tr>
<td>5</td>
<td>INH*</td>
<td>Inhibits detection of tones representing keys A, B, C, and D.</td>
</tr>
<tr>
<td>6</td>
<td>PD*</td>
<td>Power down. Logic high powers down the device and inhibits the oscillator. Internal pulldown.</td>
</tr>
<tr>
<td>7</td>
<td>OSC1</td>
<td>Clock input. 3.579545 MHz crystal connected between these pins completes the internal oscillator.</td>
</tr>
<tr>
<td>8</td>
<td>OSC2</td>
<td>Clock output.</td>
</tr>
<tr>
<td>9</td>
<td>VSS</td>
<td>Negative power supply (normally connected to 0 V).</td>
</tr>
<tr>
<td>10</td>
<td>OE</td>
<td>Tri-statable output enable (input). Logic high enables the outputs Q1 - Q4. Internal pullup.</td>
</tr>
<tr>
<td>11</td>
<td>Q1, Q2, Q3, Q4</td>
<td>Tri-statable data outputs. When enabled by OE, provides the code corresponding to the last valid tone pair received (see Table 5).</td>
</tr>
<tr>
<td>15</td>
<td>StD</td>
<td>Delayed steering output. Presents a logic high when a received tone pair has been registered and the output latch is updated. Returns to logic low when the voltage on St/GT falls below VTTSH.</td>
</tr>
<tr>
<td>16</td>
<td>EST</td>
<td>Early steering output. Presents a logic high immediately when the digital algorithm detects a recognizable tone pair (signal condition). Any momentary loss of signal condition will cause EST to return to a logic low.</td>
</tr>
<tr>
<td>17</td>
<td>St/GT</td>
<td>Steering input/guard time output (bidirectional). A voltage greater than VTTH detected at St causes the device to register the detected tone pair and update the output latch. A voltage less than VTTH frees the device to accept a new tone pair. The GT output acts to reset the external steering time constant, and its state is a function of EST and the voltage on St. (See Figure 7).</td>
</tr>
<tr>
<td>18</td>
<td>VDD</td>
<td>Positive power supply. (Normally connected to +5V.)</td>
</tr>
</tbody>
</table>

* -02 only. Connect to VSS for -01 version.
<table>
<thead>
<tr>
<th>$F_{\text{HIGH}}$</th>
<th>Key (ref.)</th>
<th>$Q_1$</th>
<th>$Q_2$</th>
<th>$Q_3$</th>
<th>$Q_4$</th>
<th>OE</th>
<th>FLow</th>
</tr>
</thead>
<tbody>
<tr>
<td>1209</td>
<td>1</td>
<td>H</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1336</td>
<td>2</td>
<td>H</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1477</td>
<td>3</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1209</td>
<td>4</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1336</td>
<td>5</td>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1477</td>
<td>6</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1209</td>
<td>7</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1336</td>
<td>8</td>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1477</td>
<td>9</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1209</td>
<td>10</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1336</td>
<td>D</td>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1477</td>
<td>#</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1209</td>
<td>A</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1336</td>
<td>B</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1477</td>
<td>C</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1209</td>
<td>D</td>
<td>H</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

$L = \text{logic low, } H = \text{logic high, } Z = \text{high impedance}$
ATmega16 AVR Microcontroller
Device Architecture

Flash, EEPROM, and SRAM are all integrated onto a single chip, removing the need for external memory in most applications. Some devices have a parallel external bus option to allow adding additional data (or code) memory, or memory-mapped devices. All devices have serial interfaces, which can be used to connect larger serial EEPROMs or flash chips.

ATmega16 Microcontroller Features

- **AVR – High-performance and Low-power RISC Architecture**
  - Lead-free, RoHS-compliant package
  - 131 powerful instructions -- most single clock cycle execution
  - 32 x 8 general-purpose working registers
  - Fully static operation
  - Up to 16 MIPS throughput at 16 MHz
  - On-chip 2-cycle Multiplier

- **Non-volatile Program and Data Memories**
  - 16 KBytes of in-system self programmable Flash (10,000 Write/Erase Cycles)
  - Optional Boot Code section with independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - 512 Bytes EEPROM (100,000 Write/Erase Cycles)
  - 1 KByte internal SRAM
  - Programming Lock for software security
**JTAG Interface (IEEE 1149.1 Compliant)**

- Boundary-scan capabilities according to the JTAG standard
- Extensive On-chip Debug support
- Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG interface

**Peripheral Features**

- Two 8-bit Timer/Counters with separate Prescalers and Compare Modes
- One 16-bit Timer/Counter with separate Prescaler, Compare Mode and Capture Mode
- Real-time Counter with separate oscillator
- Four PWM channels
- 8-channel, 10-bit ADC
- Byte-oriented Two-wire Serial interface
- Programmable Serial USART
- Master/Slave SPI Serial Interface
- On-chip Analog Comparator
- Programmable Watchdog Timer with Separate On-chip Oscillator

**Special Microcontroller Features**

- Power-on Reset and programmable Brown-out Detection
- Internal Calibrated RC Oscillator
- External and Internal Interrupt Sources
- 32 Programmable I/O lines
- Operating Voltage: 4.5-5.5V
- Industrial temperature range (-40°C to 85°C)
L293D Motor Driver IC

Introduction

The L293D is a quadruple half H-bridge bidirectional motor driver IC that can drive current of up to 600mA with voltage range of 4.5 to 36 volts. It is suitable to drive small DC-Geared motors, bipolar stepper motor etc.

Specifications

- Supply Voltage Range 4.5V to 36V
- 600-mA Output current capability per driver
- Separate Input-logic supply
- It can drive small DC-geared motors, bipolar stepper motor.
- Pulsed Current 1.2-A Per Driver
- Thermal Shutdown
- Internal ESD Protection
- High-Noise-Immunity Inputs

Applications

- DC and stepper motor drives
- Position and velocity servomechanisms
This chip enables you to take 4 outputs and turn them into 2 that can be reversed.

**Benefits:**

You can control 2 motors in both directions instead of 4 in only one direction.

**Disadvantages:**

There is a 1.5V voltage drop within the L293D driver chip.

If you run both motors and servos on the same circuit, your servos will always get 1.5V more than the motors - and typically you would want it the other way around!
NOT GATE

In digital logic, an inverter or NOT gate is a logic gate which implements logical negation. The truth table is shown on the right.

This represents perfect switching behavior, which is the defining assumption in Digital electronics. In practice, actual devices have electrical characteristics that must be carefully considered when designing inverters. In fact, the non-ideal transition region behavior of a CMOS inverter makes it useful in analog electronics as a class A amplifier (e.g., as the output stage of an operational amplifier).

An inverter circuit outputs a voltage representing the opposite logic-level to its input. Inverters can be constructed using a single NMOS transistor or a single PMOS transistor coupled with a resistor. Since this 'resistive-drain' approach uses only a single type of transistor, it can be fabricated at low cost. However, because current flows through the resistor in one of the two states, the resistive-drain configuration is disadvantaged for power consumption and processing speed. Alternately, inverters can be constructed using two complimentary transistors in a CMOS configuration. This configuration greatly reduces power consumption since one of the transistors is always off in both logic states. Processing speed can also be improved due to the
relatively low resistance compared to the NMOS-only or PMOS-only type devices. Inverters can also be constructed with Bipolar Junction Transistors (BJT) in either a resistor-transistor logic (RTL) or a transistor-transistor logic (TTL) configuration.

Digital electronics circuits operate at fixed voltage levels corresponding to a logical 0 or 1 (see Binary). An inverter circuit serves as the basic logic gate to swap between those two voltage levels. Implementation determines the actual voltage, but common levels include (0, +5V) for TTL circuits.

**Digital building block**

This schematic diagram shows the arrangement of NOT gates within a standard 4049 CMOS hex inverting buffer.
The digital inverter is considered the base building block for all digital electronics. Memory (1 bit register) is built as a latch by feeding the output of two serial inverters together. Multiplexers, decoders, state machines, and other sophisticated digital devices all rely on the basic inverter.

The Hex Inverter is an integrated circuit that contains six (hexa-) inverters. For example, the 7404 TTL chip which has 14 pins and the 4049 CMOS chip which has 16 pins, 2 of which are used for power/referencing, and 12 of which are used by the inputs and outputs of the six inverters (the 4049 has 2 pins with no connection).

**Performance measurement**

Digital inverter quality is often measured using the Voltage Transfer Curve, which is a plot of input vs. output voltage. From such a graph, device parameters including noise tolerance, gain, and operating logic-levels can be obtained.

![Voltage Transfer Curve for a 20 μm Inverter](image)

Ideally, the voltage transfer curve (VTC) appears as an inverted step-function - this would indicate precise switching between *on* and *off* - but in real devices, a gradual transition region exists. The VTC indicates that for low input voltage, the circuit outputs high voltage; for high input, the output tapers off towards 0 volts. The slope of this transition region is a measure of quality - steep (close to -Infinity) slopes yield precise switching.
DIODE

In electronics, a diode is a two-terminal electronic component that conducts electric current in only one direction. The term usually refers to a semiconductor diode, the most common type today, which is a crystal of semiconductor connected to two electrical terminals, a P-N junction. A vacuum tube diode, now little used, is a vacuum tube with two electrodes; a plate and a cathode.

The most common function of a diode is to allow an electric current in one direction (called the diode's forward direction) while blocking current in the opposite direction (the reverse direction). Thus, the diode can be thought of as an electronic version of a check valve. This unidirectional behavior is called rectification, and is used to convert alternating current to direct current, and remove modulation from radio signals in radio receivers.

However, diodes can have more complicated behavior than this simple on-off action, due to their complex non-linear electrical characteristics, which can be tailored by varying the construction of their P-N junction. These are exploited in special purpose diodes that perform many different functions. Diodes are used to regulate voltage (Zener diodes), electronically tune radio and TV receivers (varactor diodes), generate radio frequency oscillations (tunnel diodes), and produce light (light emitting diodes).

Diodes were the first semiconductor electronic devices. The discovery of crystals' rectifying abilities was made by German physicist Ferdinand Braun in 1874. The first semiconductor diodes, called cat's whisker diodes were made of crystals of minerals such as galena. Today most diodes are made of silicon, but other semiconductors such as germanium are sometimes used.
Figure 5: I–V characteristics of a P-N junction diode (not to scale).

Figure 7: Typical diode packages in same alignment as diode symbol. Thin bar depicts the cathode.

The 1N4007 is a diode with forward current rating of 1 ampere, and a reverse voltage rating of 1,000 volts.
Team is analyzing the circuit to check any defect during implementation.
**Circuit description**

Figure shows the block diagram of the microcontroller-based mobile phone operated land rover. The important components of this rover are a DTMF decoder, microcontroller and motor driver.

An MT8870 series DTMF decoder is used here. All types of the MT8870 series use digital counting techniques to detect and decode all the 16 DTMF tone pairs into a 4-bit code output. The built-in dial tone rejection circuit eliminates the need for pre-filtering. When the input signal given at pin 2 (IN-) in single-ended input configuration is recognized to be effective, the correct 4-bit decode signal of the DTMF tone is transferred to Q1 (pin 11) through Q4 (pin 14) outputs.

Table II shows the DTMF data output table of MT8870. Q1 through Q4 outputs of the DTMF decoder (IC1) are connected to port pins PA0 through PA3 of ATmega16 microcontroller (IC2) after inversion by N1 through N4, respectively. The ATmega16 is a low-power, 8-bit, CMOS microcontroller based on the AVR enhanced RISC architecture. It provides the following features:

- 16 kB of in-system programmable Flash program memory with read-while-write capabilities
- 512 bytes of EEPROM, 1kB SRAM,
- 32 general-purpose input/output (I/O) lines
- 32 general-purpose working registers

All the 32 registers are directly connected to the arithmetic logic unit, allowing two independent registers to be accessed in one
single instruction executed in one clock cycle. The resulting architecture is more code-efficient. Outputs from port pins PD0 through PD3 and PD7 of the microcontroller are fed to inputs IN1 through IN4 and enable pins (EN1 and EN2) of motor driver L293D, respectively, to drive two geared DC motors. Switch S1 is used for manual reset.

The microcontroller output is not sufficient to drive the DC motors, so current drivers are required for motor rotation. The L293D is a quad, high-current, half-H driver designed to provide bidirectional drive currents of up to 600 mA at voltages from 4.5V to 36V. It makes it easier to drive the DC motors.

The L293D consists of four drivers. Pins IN1 through IN4 and OUT1 through OUT4 are input and output pins, respectively, of driver 1 through driver 4. Drivers 1 and 2, and drivers 3 and 4 are enabled by enable pin 1 (EN1) and pin 9 (EN2), respectively. When enable input EN1 (pin1) is high, drivers 1 and 2 are enabled and the outputs corresponding to their inputs are active. Similarly, enable input EN2 (pin 9) enables drivers 3 and 4.
PCB layout of the circuit
Construction

When constructing any robot, one major mechanical constraint is the number of motors being used. You can have either a two-wheel drive or a four-wheel drive. Though four-wheel drive is more complex than two-wheel drive, it provides more torque and good control. Two-wheel drive, on the other hand, is very easy to construct. The chassis used in this model is a 10×18cm² sheet made up of parax. Motors are fixed to the bottom of this sheet and the circuit is affixed firmly on top of the sheet. A cell phone is also mounted on the sheet as shown in the picture.

In the four-wheel drive system, the two motors on a side are controlled in parallel. So a single L293D driver IC can drive the rover. For this robot, beads affixed with glue act as support wheels. Further applications this land rover can be further improved to serve specific purposes. It requires four controls to roam around. The remaining eight controls can be configured to serve other purposes, with some modifications in the source program of the microcontroller.

Two wheel drive system employed to construct the robot finally.
**Construction process**

- A wooden/mica sheet of required dimension is used as the chassis of the land rover.

- The land rover runs on three wheels, among which both the rear wheels are motor driven and the front wheel is free to rotate in any direction.

- Drilling machine is employed to make holes of specific diameter at specific location on sheet so that wheels can be joined to the chassis.

- The motor driven wheels are joined to the chassis with the help of the metal clamp.

- These metal clamps support the motors to be hanged below the chassis. Clamps and the front wheel are joined to the chassis with the help of screw.

- The motor driven wheels were put along the motor shaft as shown in picture.

- Battery and the PCB are placed above the chassis. The battery and PCB is fixed firmly with help of nuts-and-bolts so that it does not change its position. PCB should be placed above the battery and there must be cushion in between battery and the PCB so that the PCB's base remains safe.
This mica sheet was used as the main frame or chassis of the land rover.

These metal clamps were to support the wheels and motor to the main chassis.
These are DC motors which are capable of running in forward or reverse direction. Maximum rpm is 50.

Wheels are directly connected to the shaft of the motor. Hence they can be moved in either direction.
MOHIT AND SHERKHAN

Busy in drilling holes in the chassis in mechanical workshop.

SHER KHAN

Joining the wheels and clamp to the chassis for building the main frame of the land rover.
View of the main frame structure from different directions.
A small soldering kit is used for completing the soldering work on the PCB. Four IC’s, one diode, eight resistors, four capacitors, two crystal oscillators and one reset switch were placed on PCB through the process of soldering.

Four connecting leads were soldered on specific input and output points. There are two input and two output points on the PCB.

One of the input points was for power supply through the 6V battery used. Other input point was for connecting the cell phone to circuit through headset.

The output points were connected to the dc motors through the connecting leads.

The four IC’s were not directly soldered on the PCB instead the IC’s holders were soldered and IC’s were placed on the holders later.

MultiMeter was used in various situations time to time to check out the electrical faults/soldering works and for path checking of PCB during development of the electrical and electronics circuitry.

Small wires were employed to complete the PCB circuitry which is termed as jumpers.
Mr. Gupta is directing the team member on certain technical aspects.
Cell phone operated land rover

ANSHUL
Performing soldering on the PCB

ANSHUL
MOHIT
SHER KHAN

Placing components on the PCB
**Problems faced**

Although the concept and design of the project seemed perfect, there were some problems faced while actual implementation. Some of the obstacles were easily corrected while in some cases components were replaced. These conditions led to do some R&D work to the team member.

**Connecting headset of the cell phone to DTMF decoder IC input:**

There were several types of headset cord are available in the market, the right one had to be chosen from them. Several ways to break up the cords and connect them to the input of IC were tried & some was newly developed by us.

**Solution:**

Finally the earplugs of the headset were removed and resulting set of wires were connected in an appropriate manner to the decoder IC’s input. There were two wires, one was the ring and other was the tip of the headset jack. With the help of the MultiMeter tip and ring were identified.
**Defects in the developed PCB:**

When the supply was given into the PCB, one circuit track was instantly damaged along with some burning fumes.

Solution:

Initially it was advised to develop a new PCB, but sooner with the help of the MultiMeter the defects were identified. There were two short circuit locations of tracks which were corrected by use of sharp blade. The tracks were cut down to avoid the short circuit. The damaged track was recreated by use of a normal wire.

**Damage of one wheel driving DC motor:**

During the initial operation of the robot it was found that one of the DC motor was not running in reverse direction. The motor was checked by opening its outer case and it was found that geared part of the motor is damaged.

Solution:

Many methods were tried to correct the defect but success was not achieved. Finally a new DC motor was purchased and employed in the project.
The marked portions show the defective location on the PCB. These defects were corrected unconventionally.

The two circles show the locations of short circuit of the tracks.

The rectangle shows the location of damaged track.
**Completely developed land rover**

Upper view of the land rover

View showing motor assembly of the land rover
Working

In order to control the robot, you need to make a call to the cell phone attached to the robot (through head phone) from any phone, which sends DTMF tunes on pressing the numeric buttons. The cell phone in the robot is kept in ‘auto answer’ mode. (If the mobile does not have the auto answering facility, receive the call by ‘OK’ key on the rover-connected mobile and then made it in hands-free mode.)

So after a ring, the cell phone accepts the call.

Now you may press any button on your mobile to perform actions as listed in Table III. The DTMF tones thus produced are received by the cell phone in the robot. These tones are fed to the circuit by the headset of the cell phone. The MT8870 decodes the received tone and sends the equivalent binary number to the microcontroller. According to the program in the microcontroller, the robot starts moving.

When you press key ‘2’ (binary equivalent 00000010) on your mobile phone, the microcontroller outputs ‘10001001’ binary equivalent.

Port pins PD0, PD3 and PD7 are high. The high output at PD7 of the microcontroller drives the motor driver (L293D). Port pins PD0 and PD3 drive motors M1 and M2 in forward direction (as per Table III). Similarly, motors M1 and M2 move for left turn, right turn, backward motion and stop condition as per Table.
TABLE I
Tones and Assignments in a DTMF System

<table>
<thead>
<tr>
<th>Frequencies</th>
<th>1209 Hz</th>
<th>1336 Hz</th>
<th>1477 Hz</th>
<th>1633 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>697 Hz</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>A</td>
</tr>
<tr>
<td>770 Hz</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>B</td>
</tr>
<tr>
<td>852 Hz</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>C</td>
</tr>
<tr>
<td>941 Hz</td>
<td>*</td>
<td>0</td>
<td>#</td>
<td>D</td>
</tr>
</tbody>
</table>

TABLE II
DTMF Data Output

<table>
<thead>
<tr>
<th>Low group (Hz)</th>
<th>High group (Hz)</th>
<th>Digit</th>
<th>CE</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>697</td>
<td>1209</td>
<td>1</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>697</td>
<td>1336</td>
<td>2</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>697</td>
<td>1477</td>
<td>3</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>770</td>
<td>1209</td>
<td>4</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>770</td>
<td>1336</td>
<td>5</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>770</td>
<td>1477</td>
<td>6</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>852</td>
<td>1209</td>
<td>7</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>852</td>
<td>1336</td>
<td>8</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>852</td>
<td>1477</td>
<td>9</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>941</td>
<td>1336</td>
<td>0</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>941</td>
<td>1209</td>
<td>*</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>941</td>
<td>1477</td>
<td>#</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>697</td>
<td>1633</td>
<td>A</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>770</td>
<td>1633</td>
<td>B</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>852</td>
<td>1633</td>
<td>C</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>941</td>
<td>1633</td>
<td>D</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

TABLE III
Actions Performed Corresponding to the Keys Pressed

<table>
<thead>
<tr>
<th>Number pressed by user</th>
<th>Output of HT9170 DTMF decoder</th>
<th>Input to the microcontroller</th>
<th>Output from microcontroller</th>
<th>Action performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0×02 00000010</td>
<td>0×FD 1111101</td>
<td>0×89 1000100</td>
<td>Forward motion</td>
</tr>
<tr>
<td>4</td>
<td>0×04 00000100</td>
<td>0×FB 1111011</td>
<td>0×85 1000101</td>
<td>Left turn</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Right motor forwarded</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Left motor backwarded</td>
</tr>
<tr>
<td>6</td>
<td>0×06 00000110</td>
<td>0×F9 1111001</td>
<td>0×8A 1000101</td>
<td>Right turn</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Right motor backwarded</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Left motor forwarded</td>
</tr>
<tr>
<td>8</td>
<td>0×08 00001000</td>
<td>0×F7 11110111</td>
<td>0×86 10000110</td>
<td>Backward motion</td>
</tr>
<tr>
<td>5</td>
<td>0×05 00000101</td>
<td>0×FA 11110101</td>
<td>0×00 00000000</td>
<td>Stop</td>
</tr>
</tbody>
</table>
**Application**

Remote control vehicles have various scientific uses including hazardous environments, working in the Deep Ocean, and space exploration. The majority of the probes to the other planets in our solar have been remote control vehicles, although some of the more recent ones were partially autonomous.

The sophistication of these devices has fueled greater debate on the need for manned spaceflight and exploration. The voyager I spacecraft is the first craft of any kind to leave the solar system. The Martian explorer's spirit and opportunity have provided continuous data about the surface of mars since January 3, 2004.

**Military and law enforcement**

Military usage of remotely controlled military vehicles dates back to the first half of 20th century. Soviet red army used remotely controlled teletanks during 1930s in the winter war and early stage of World War II. There were also remotely controlled cutters and experimental remotely controlled planes in the red army.

Remote control vehicles are used in law enforcement and military engagement for some of the same reasons. The exposures to hazards are mitigated to the person who operates the vehicle from a location of relative safety. Remote controlled vehicles are used by many police department bomb-squads to defuse or detonate explosives.
Unmanned aerial vehicles have undergone a dramatic evolution in capability in the past decade. Early UAV’s were capable of reconnaissance missions alone and then only with a limited range. Current UAV’s can hover around possible targets until they are positively identified before releasing their payload of weaponry. Backpack sized UAV’s will provide ground troops with over the horizon surveillance capabilities.

**Search and rescue**

UAV’s will likely play an increased role in search and rescue in various areas. This was demonstrated by the successful use of UAV’s during the 2008 hurricanes that struck Louisiana and Texas.

**Recreation and hobby**

Small scale remote control vehicles have long been popular among hobbyists. These remote controlled vehicles span a wide range in terms of price and sophistication. There are many types of radio controlled vehicles.

These include on-road cars, off-road trucks, boats, airplane, and even helicopters. The “robots” now popular in television shows such as robot war, are a recent extension of this hobby. Remote control submarine also exist.
Further improvement and future scope

**Additional functions**

Since 7 keypad buttons have not assigned any operation, few other operations can be programmed to the robot. It may be any mechanical operation with the addition of few extra structure and moving devices. The programming may get more complex as the complexity of the circuit increases.

**3D motion of the land rover**

The land rover can be developed in such manner that would be capable of climbing the stairs or even a vertical wall. The robot can be made to fly like a small helicopter and being control by the cell phone.

**3-D motion of the video camera**

Video camera can be mounted on the robot with a mechanism of moving the camera in all the possible direction. More than one camera can also be mounted on the robot for receiving picture from all the direction.

**Sensors**

Various sensors can be fixed on the land rover on desired location in order to
Software description

The software is written in ‘C’ language and compiled using Code Vision AVR ‘C’ compiler. The source program is converted into hex code by the compiler. Burn this hex code into ATmega16 AVR microcontroller.

The source program is well commented and easy to understand. First include the register name defined specifically for ATmega16 and also declare the variable. Set port A as the input and port D as the output. The program will run forever by using ‘while’ loop. Under ‘while’ loop, read port A and test the received input using ‘switch’ statement. The corresponding data will output at port D after testing of the received data.

Source program:----- ROBOT.C

```c
#include <mega16.h>
void main(void)
{
    unsigned int k, h;
    DDRA=0x00;
    DDRD=0XFF;
    while (1)
    {
        k =~PINA;
        h=k & 0x0F;
        switch (h)
        {
            case 0x02: //if I/P is 0x02
```

\{ 
  PORTD=0x89; // O/P 0x89 ie Forward
  break;
\}
case 0x08: // if I/P is 0x08
  \{ 
  PORTD=0x86; // O/P 0x86 ie Backward
  break;
\}
case 0x04:
  \{ 
  PORTD=0x85; // Left turn
  break;
\}
case 0x06:
  \{ 
  PORTD=0x8A; // Right turn
  break;
\}
case 0x05:
  \{ 
  PORTD=0x00; // Stop
  break;
\}