DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

# VEHICLE MOTION CONTROL USING WIRELESS COMMUNICATION

by
Remzi TÜKENMEZ

September, 2006
İZMİR

# VEHICLE MOTION CONTROL USING WIRELESS COMMUNICATION

**A Thesis Submitted to the**

**Graduate School of Natural and Applied Sciences of Dokuz Eylül University**

**In Partial Fulfillment of the Requirements for the Degree of Master of Science**

**in Electrical & Electronics Engineering,**

**Electrical & Electronics Engineering Program**

**by**

**Remzi TÜKENMEZ**

**September, 2006**

**İZMİR**

**M.Sc THESIS EXAMINATION RESULT FORM**

We have read the thesis entitled **"VEHICLE MOTION CONTROL USING WIRELESS COMMUNICATION"** completed by **REMZİ TÜKENMEZ** under supervision of **PROF. DR. MUSTAFA GÜNDÜZALP** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

PROF. DR. MUSTAFA GÜNDÜZALP

Supervisor

(Jury Member) (Jury Member)

Prof.Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

## ACKNOWLEDGMENTS

I am greatly indebted to my thesis supervisor, Prof. Dr. Mustafa GÜNDÜZALP for kindly providing guidance throughout the development of this study. His comments have been of greatest help at all times.

I thank Mehmet BEKLERGÜL and my friends Aydın KOYUNCU, Serkan ESER, Metin TAŞTAN and Kamil ÇETİN.

Finally, I thank my parents for their understanding and never ending support throughout my life.

Remzi TÜKENMEZ

**VEHICLE MOTION CONTROL USING WIRELESS COMMUNICATION**


## ABSTRACT

In this thesis, using the video information taken with a camera having wireless ethernet transmitter, a remotely controlled vehicle is designed. It is possible to control the vehicle up to 100 meters remotely. The circuits designed for the vehicle, are microcontroller boards and wireless communication RF transceiver modules.

Afterwards, a computer interface which communicates with the transmitter circuit via the serial port of computer is prepared and through this interface the vehicle is able to be controlled remotely.

The camera having wireless ethernet transmitter is assembled on the vehicle and in this way transmitting the video information to a computer was provided. In this system, using the IP address given to the camera, the captured video data can be watched with the camera software or by using internet browser.

A palette system composed of triger belts which are driven by two DC motors provides the motion system.



**Keywords:** Computer software, Microcontroller, Transceiver module, Serial port, DC motor, Triger belt.

# ARAÇ HAREKET KONTROLÜNDE KABLOSUZ HABERLEŞMENİN KULLANILMASI

## ÖZ

Bu tezde, üzerinde bulunan kameradan alınan görüntünün bilgisayarda izlenmesi ile uzaktan kumanda edilen bir araç yapılmıştır.

Araç yaklaşık 100 metre mesafe de kablosuz kumanda edilebilmektedir. Burada mikrodenetleyici kullanılan iki adet devre tasarlanmıştır. Bu devrelere kablosuz haberleşme için kullanılmak üzere RF alıcıverici modüller monte edilmiştir.

Daha sonra, bilgisayar seri portu üzerinden verici devre ile haberleşen bilgisayar kontrol programı hazırlanmıştır. Bu program vasıtasıyla aracın bilgisayar ile uzaktan kumandalı hareketi sağlanmıştır.

Üzerinde kablosuz ethernet verici modülü bulunan bir kamera araç üzerine monte edilmiştir ve görüntünün bilgisayara iletilmesi sağlanmıştır. Bu sistemde, kameraya verilen IP adresi kullanılarak alınan görüntü, kameranın kendi yazılımıyla yada internet explorer programıyla izlenebilir.

Aracın mekanik hareketinde ise, sağ ve sol ön tarafta bulunan iki adet DC motor ile sürülen triger kayışlarından oluşan paletli sistem kullanılmıştır.

**Anahtar kelimeler:** Bilgisayar yazılımı, mikrodenetleyici, alıcı-verici modül, seri port, DC motor, Triger kayışı.

**CONTENTS**

**Page**

# CHAPTER ONE

# INTRODUCTION

Nowadays, remote controlled vehicle systems are used in many applications. These vehicles work in many areas (civil or military). Below, some examples of the usage of vehicle are given.

- Movement of objects (furniture, packet etc.)
- Usage in some dangerous places for people
- In automatic wheelchair systems
- Bomb destruction
- Bomb placement

In this thesis, our main purpose is to produce a vehicle which can be controlled by computer remotely. A camera having wireless transmitter has been assembled on the vehicle. In this manner, movement of the vehicle can be watched using the program provided with the camera or an internet browser through a computer.

Figure 1.1 IP3137 wireless network camera is shown.

First of all, required specifications are determined. The metal body of the vehicle is prepared and motors, triger belts, gearboxes, wheels and accumulators are assembled on it. In this manner, the vehicle got ready to move. For vehicle motion, a palette system composed of triger belts which driven by two 12 volt DC (direct current) motors which are at both side of the vehicle is used.

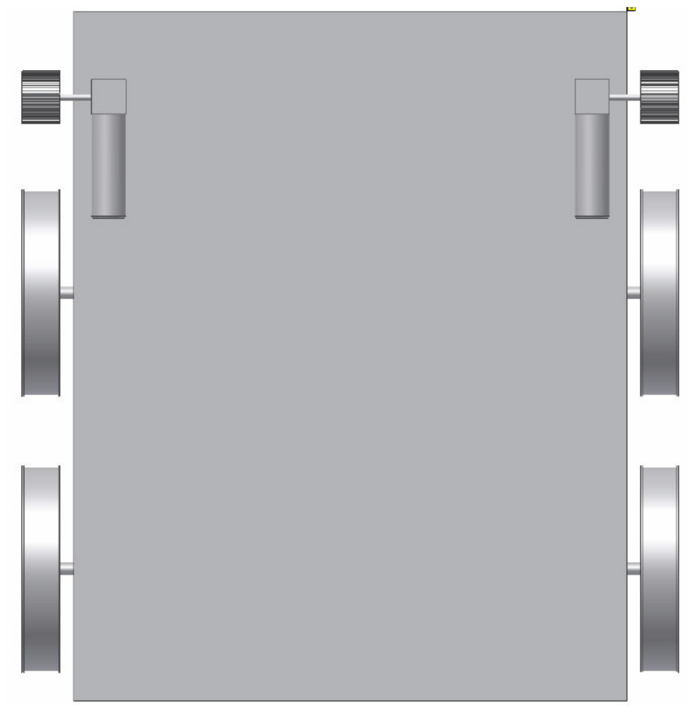


Figure 1.2 The metal body of the designed vehicle.

Figure 1.3 Motor, gearbox and triger gear of the vehicle.

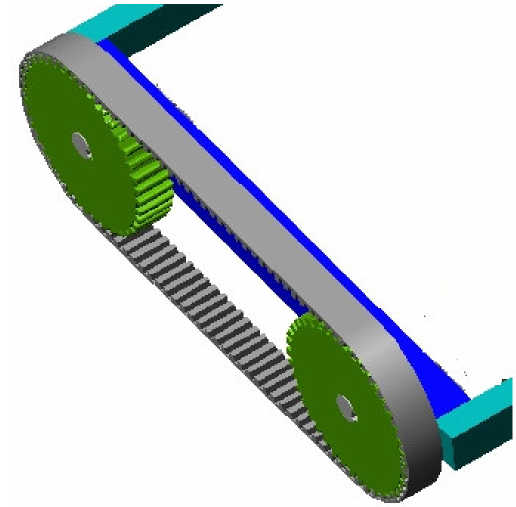


Figure 1.4 Triger belt and wheels are shown.

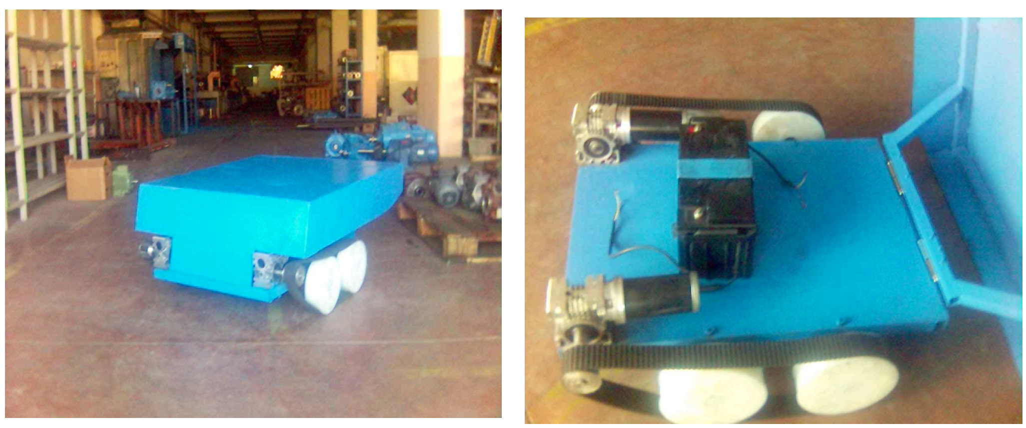


Figure 1.5 Two pictures of the vehicle are shown.

Secondly, the electronic control circuits are designed and assembled. Microcontrollers and wireless communication RF transceiver modules are used in the control circuits. In order to control two motors, driver and power circuits are designed.

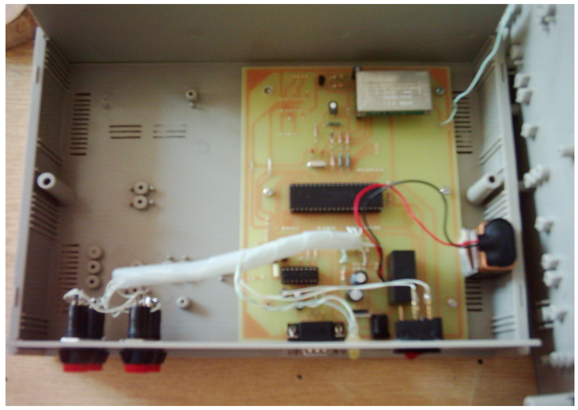


Figure 1.6 The electronic control circuit is prepared.

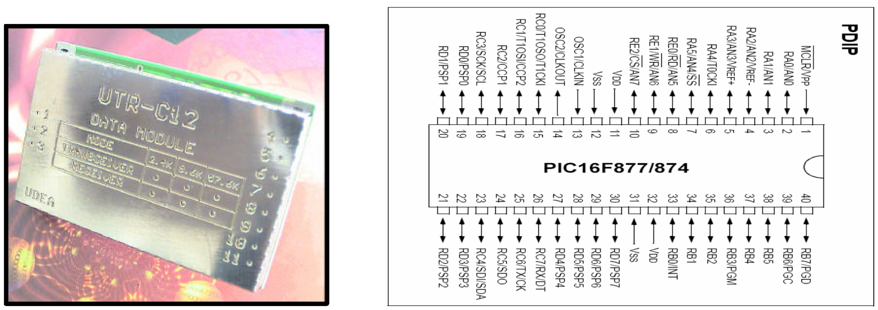



Figure 1.7 UTR-C12U transceiver module and PIC 16F877 microcontroller are shown.

Finally, a control program was written. The vehicle can be controlled both using the control program and the buttons on the transmitter circuit remotely. In addition to this, the vehicle is also controled using the buttons connected to the vehicle via cable.

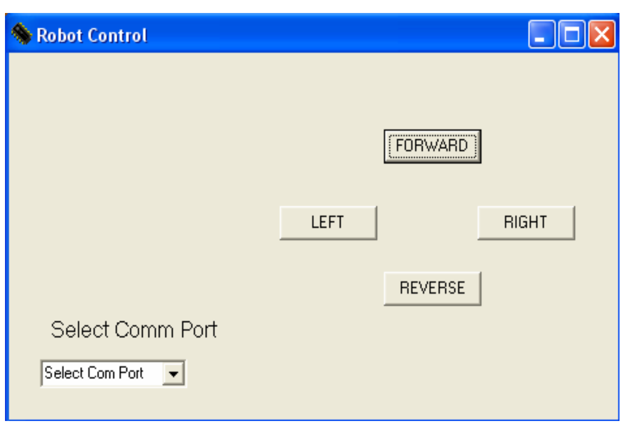



Figure 1.8 The control program of the vehicle is prepared.

# CHAPTER TWO

# WIRELESS COMMUNICATION, DC MOTOR, H-BRIDGE AND RS232 STANDARTS

In this chapter, detailed information about the main parts and methods about the vehicle designed and some other vehicles such as wireless communication, DC motor construction, H-bridge motor driver circuit and RS232 serial port communication standarts will be given.

## 2.1 Examples of Some Vehicles

When some vehicles designed earlier have been examined, it has been observed that they mostly use palette system for motion. In this way, vehicles can be designed more robustly compared to the ones with wheels. Figure 2.1 shows example of some vehicles.


Figure 2.1 Some examples of vehicles are shown.

## 2.2 Wireless Communication Using RF

Companies look for any advantage to remain competitive in today's high-speed business climate. They work to streamline their supply chain management systems, knowing that even a few minutes in order processing time can make a huge difference in getting product to end user. And that can be the difference between who gets the job and who doesn't - even who gets to stay in business and who doesn't. One of the ways to save minutes that turn into competitive advantages is to save seconds on highly repetitive tasks by using fast, accurate radio frequency (RF) technology. More and more companies are evaluating RF technology because they have learned that wirelessly transmitting data can provide tremendous time and cost savings.

RF is the wireless transmission of data by digital radio signals at a particular frequency. It maintains a two-way, online radio connection between a mobile terminal and the host computer. The mobile terminal, which can be portable, even worn by the worker, or mounted on a forklift truck, collects and displays data at the point of activity. The host computer can be a PC, a minicomputer or a much larger mainframe. The end result is a seamless flow of information to and from the host, allowing workers to go wherever they need to go to get their job done without fear of being out of touch with the data they need. RF improves the timeliness of information, and therefore the value of information, especially in time-sensitive operating environments like cross-dock, make-to-order manufacturing and just-in-time replenishment.

A basic RF system consists of up to three components:

- A mobile RF terminal;
- A base station (transmitter/receiver); and
- A network controller.

The mobile terminal forms the link (interface) between the user and the RF system. It collects the data to be sent, receives instructions or data from the host, and allows the user to view the data or messages on its display screen. The terminal also has a radio transmitter/receiver and antenna to provide communication with the rest of the system. The base station has a system antenna and acts as a bridge between wireless and wired networks. It is connected to a controller, (controller can be a separate device or included in the base station), which in turn is connected to the host. The "controller" receives and processes information it gets from the host computer and passes this information to the mobile terminals via the base station.

### *2.2.1 RF Adventages*

The advantages of a RF communication system are many. Start with the simple fact that if it is wireless, you don't have to lay cable all over your facility. Cable is expensive, less flexible than RF coverage and is prone to damage. For new facilities, implementing a wireless infrastructure may be more cost effective than running cable through industrial environments, especially if the space configuration may change to support different storage space allocation or flexible manufacturing stations.

Accessibility is a key benefit. If workers are within range of the system and they always should be if a proper site survey is performed they are always in touch with their data. This advantage cannot be overstated. To always have your data literally at your fingertips whenever needed means there is no break in productivity and no empty or "deadhead" trips to a stationary terminal, docking station or dispatch location to receive pick or putaway instructions. Critical decisions can be made and action taken immediately at the point of activity. Less wasted time means you can do significantly more, faster, without adding additional employees.

Other general advantages of real-time RF communication include a significant improvement in order accuracy (>99%), the elimination of paperwork, replacement of time-consuming batch processing by rapid real-time data processing, prompt response times and improved service levels. Complementing a real-time data collection system with automated data entry by bar code scanning or another automatic data collection technology improves the accuracy of information and eliminates the need for redundant data entry, which provides another set of time- and cost-saving advantages.

Many points in the supply chain can realize important advantages of accurate, real-time data that RF provides. Here are some examples of RF applied to a few common environments.

### *2.2.2 RF Transmitter*

The block diagrams in Figure 2.2 shown typical functional components in AM, FM, and PM transmitters. They all start with an RF oscillator, a modulator (AM, FM, or PM), and a power amplifier coupled to an antenna or coaxial cable via an impedance matching circuit (which always includes some filtering functionality). It is instructive to compare the FM and PM block diagrams to those of the transmitter sections of modern transceivers for digital communications where, of course, the "digital" in digital communications is just another baseband waveform (Young, 2004, chap. 6).

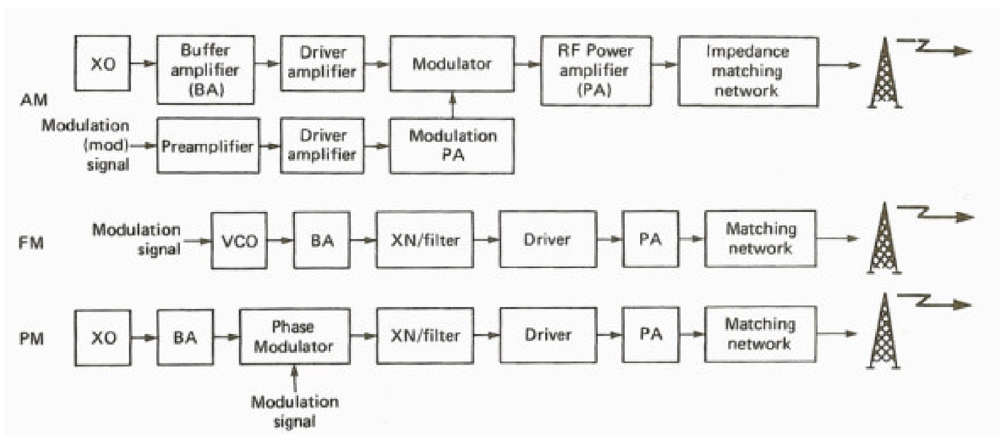



Figure 2.2 General block diagrams for transmitters of amplitude, frequency, and phase modulation.

### *2.2.3 RF Receiver*

Receiver system topics included are as follows (Young, 2004, chap.7):

- Information bandwidth
- Multiplexing
- Distortion
- Noise, noise figure, and S/N
- Modulatıon index, power, and bandwidth for AM transmissions
- Superheterodyning (mixing), image response, and RFI, including adjacent channel rejection
- Selectivity, sensitivity, and threshold
- Dynamic range and AGC, including delayed AGC
- AM demodulation

Once the requirements are established for a given system, the circuit performance parameters can be defined and the circuits designed. Some circuit design and analysis techniques needed for receivers have already been developed.

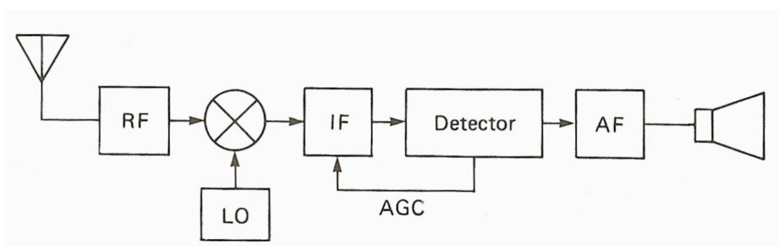



Figure 2.3 AM receiver block diagram.

## 2.3 Principles of Direct-Current Machines

The DC motor is the most versatile of all rotating electrical machines. Its speed may be easily adjusted in very fine increments ranging from standstill to rated speed and above, and if not properly controlled, may reach speeds high enough to cause destruction by centrifugal force. Direct-current motors can develop rated torque at all speeds from standstill (locked-rotor) to rated speed, and the torque that it can develop at standstill is many times greater than the torque developed by an AC motor of equal power and speed ratings (Hubert, 2002, chap. 10).

Direct-current motors are used in a wide variety of industrial drives, such as robots, machine tools, petrochemical, pulp, paper and steel mills, oil drilling rigs, and mining. They are also used extensively in automotive systems and railroads.

### *2.3.1 Commutation*

The rectangular-shaped voltage wave generated within a DC armature coil is changed to a unidirectional voltage in the bad circuit by means of a mechanical rectifier, called a commutator, mounted on the armature shaft. This is illustrated in Figure 2.4 for an elementary two-pole DC machine with one armature coil and a two-bar commutator. Connections to the external terminals are made via small stationary blocks of graphite, called brushes that are pressed against the commutator by springs.

As shown in Figure 2.4, the generated voltage within the armature coil changes direction every 180 electrical degrees of rotation, but the voltage in the external circuit remains in the same direction. The rotating commutator and stationary brushes constitute a rotary switch that provides a switching action, called commutation that switches the internal alternating voltage and current of an AC generator to direct voltage and direct current in the external circuit.

When the coil is rotating through the neutral plane, as shown in Figures 2.4(a) and (c), it is shorted by the brushes. Since the coil sides are not cutting flux, however, no armature voltage is generated, and no short-circuit current occurs.

A practical machine has many coils distributed around the armature, and the coils pass through the neutral plane one at a time. As one coil moves into the neutral plane another moves out, producing an essentially constant voltage.

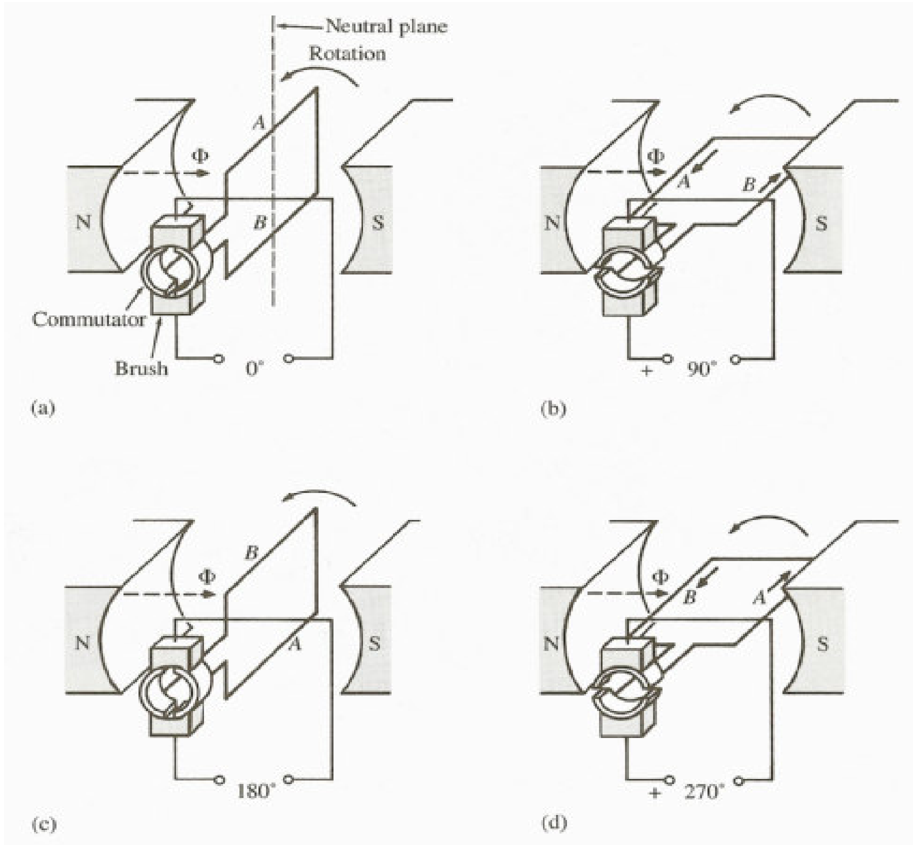



Figure 2.4 Sketches showing the commutation process illustrated with a one-coil armature.

### *2.3.2 Construction*

A cutaway view of a DC machine is shown in Figure 2.5. The shunt field coils and series field coils are wound around the same pole iron and provide the specific machine characteristics. Some DC machines omit the shunt field coils and use permanent magnets for the pole iron; in such cases, the series field would be wound around the magnets.

Small poles, called interpoles or commutating poles are located between the main field poles. The interpoles are used to minimize sparking between the brush and the commutator, which would otherwise occur when the machine is loaded. The inter pole iron and main-pole iron are bolted to a yoke or frame of cast steel. Graphite or metal-graphite brushes provide the connection between the rotating commutator and the external bad; they are designed to slide freely in metallic brush-boxes, and are connected to the box by a short flexible copper conductor called a pigtail or shunt.

The commutator is composed of alternate sections of copper bars and mica separators, clamped together with mica-insulated vee-rings. The number of commutator bars is determined by the number of coils in the armature, the number of poles, and the type of winding.

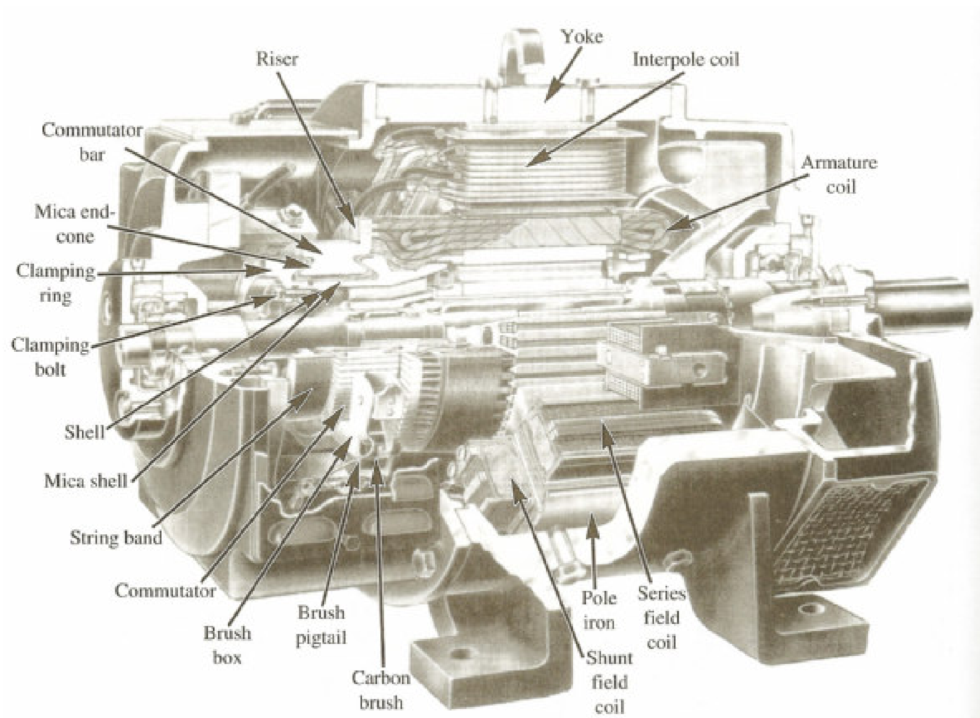



Figure 2.5 Cutaway view of a DC machine.

### *2.3.3 Reversing the Direction of Rotation of a Dc Motor*

The direction of rotation of a direct-current motor may be reversed by reversing either the current in the armature or the polarity of the field, but not both. Figures 2.6(a) and (b) illustrate how reversing the armature current reverses the direction of rotation. Figures 2.6(c) and (d) illustrate how reversing the polarity of the field reverses the direction of rotation. The direction of rotation in each case may be determined by the flux bunching rule, as shown on the inset in each figure.

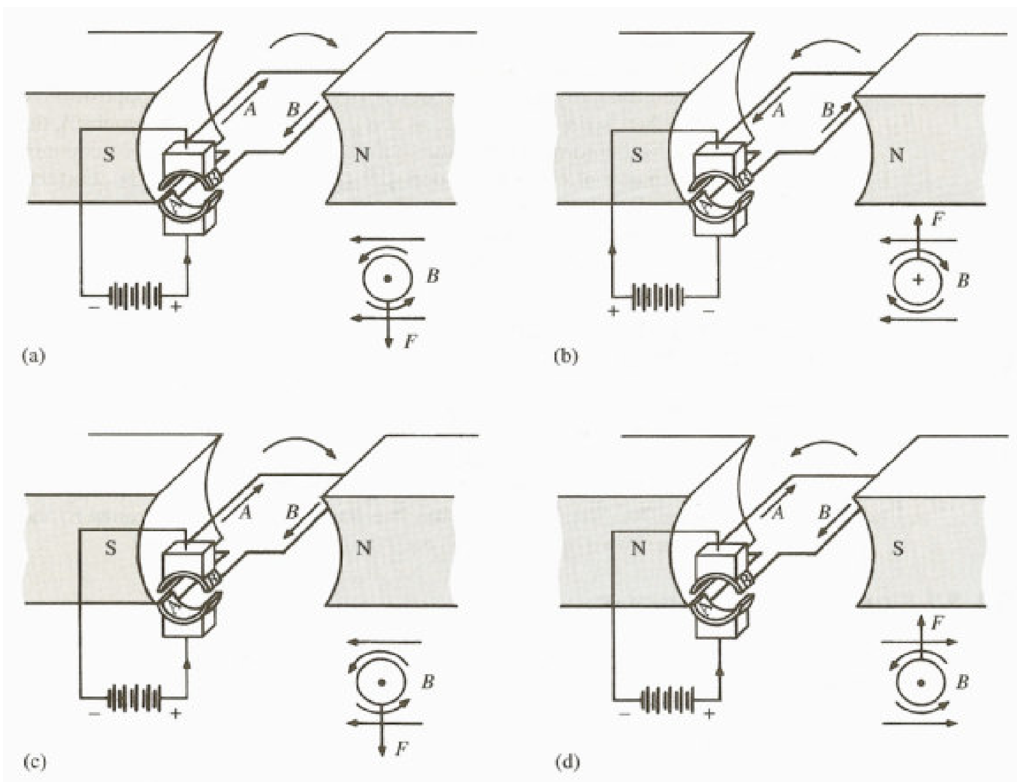



Figure 2.6 Reversing a direct-current motor by (a) and (b) reversing the armature current; (c) and (d) reversing the polarity of the field.

### *2.3.4 Controllers for Dc Motors*

Direct-current motors above 3/4 hp are started with reduced voltage. The starters may be divided into two general classifications: definite-time starters and load-sensitive starters. A definite-time starter will remove the starting resistance from the armature circuit after a preset time has elapsed. Thus, regardless of the bad on the shaft or whether or not the motor is running, the armature will be connected across full voltage when the timing period runs out. A load-sensitive starter will adjust motor acceleration to the bad. Heavy loads will be given longer starting time and lighter loads shorter starting time. If the motor does not start, the starting resistance will not be cut out.

## 2.4 H-Bridge

Therefore, for high power applications, you would want to do several things to minimize power dissipation in your circuit. You would use N-channel MOSFETs with low Rdson values. It is desirable to maximize the gate drive voltage in order to minimize the Rdson of your MOSFET. You would also ensure that when you turn your MOSFET on, the gate drive voltage would have a fast rise time to minimize the amount of time that the MOSFET is in its active region (the fall time of the gate drive voltage is also important when turning a MOSFET off). Figure 2.7 shows the block diagram of H-Bridge.

This brings us to the concept of an H-bridge. The H-bridge is a configuration of four switching devices that allows you to change the direction of current flow through a load. This is done by selecting which pair of switches is on and which pair is off. This is particularly useful for controlling bi-directional motors.

We'll assume that our load is a motor for this design, although many types of loads could be controlled with the circuit that will be defined here. An H-bridge requires that there be two switches located between the system voltage supply and the load, and two switches located between the load and the system ground return. To reduce power dissipation, it is desirable to use N channel MOSFETs for each of the four switches. This creates a significant problem. Too fully turn on an N channel

MOSFET, the gate drive voltage should be about 12V above the voltage at that MOSFET's source. In an H-bridge, the high side switches will have their source pins at the system voltage level when they are on.

Figure 2.8 shows the logic states required for changing the current flow in an H-bridge. In many loads, the amount of current passing through the load is required for control, as well as the current direction. With a motor as your load, this translates to controlling both motor speed and direction. To control speed you can turn on the appropriate high side switch, and then pulse width modulate (PWM) the low side switch.

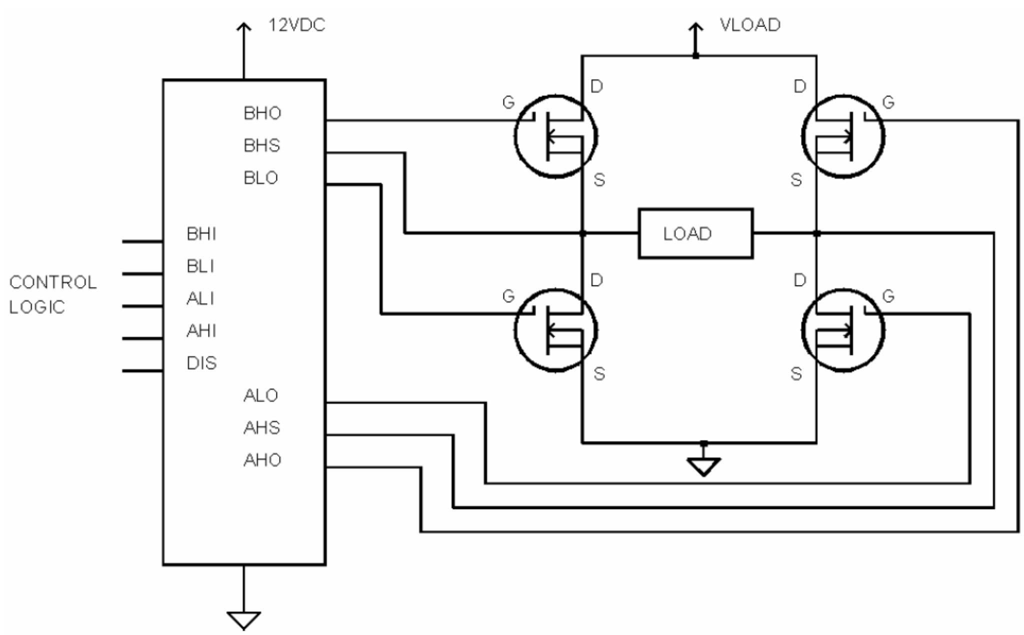



Figure 2.7 Block diagram of H-bridge.

There are protective gate resistors and zener diodes on each MOSFET which will slow the rise and fall times of the gate drive voltage but they do not have much effect on the performance of this particular circuit.

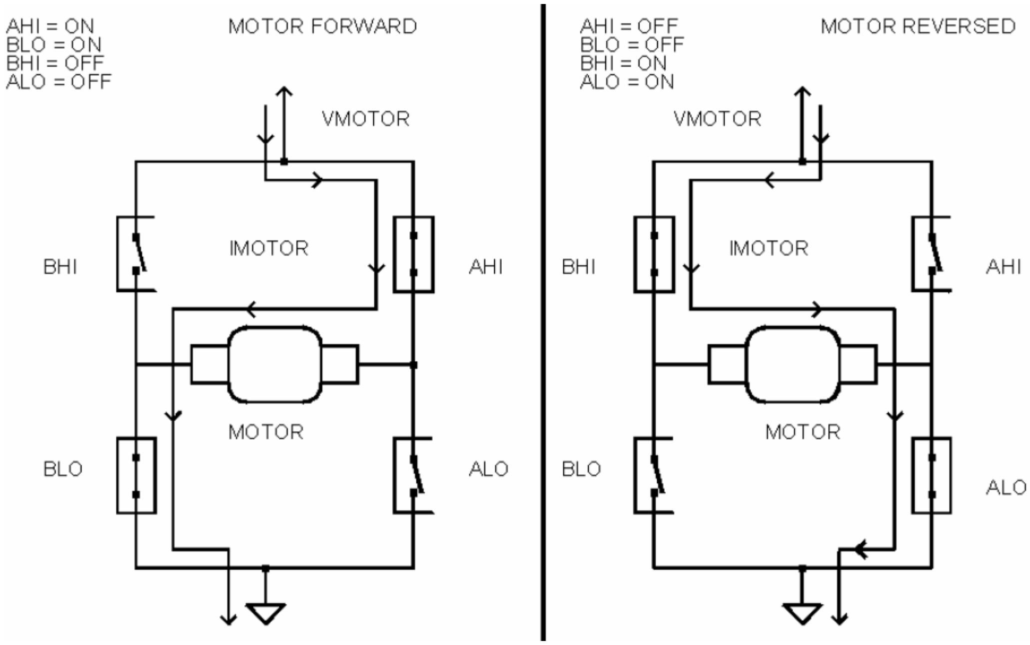



Figure 2.8 Current flow through an H-Bridge.

### 2.5 RS232 Standarts

Successful communications occur only when the equipment at both ends of a path uses the same values for the signal voltages, frequencies, timing, and meaning of key lines. The RS- 232 is a very common standard for communicating between a terminal, printer, test instrument, or similar device, and its associated communications interface for the link. The interface standard defines some things with great precision but also leaves the user many choices. Handshaking signals allow each device to indicate to the other whether it is ready or not. The functions of the RS-232 interface are implemented in special interface ICs that minimize the time-consuming, critical attention to detail activity by the microprocessor within each system. Other special ICs act as the physical interface to the communications link, providing the necessary signal levels and protection against incorrect installation (Schweber, 2002, chap. 17).

There are many standards in use. Each serves a different application need: some are low-cost, some are high-speed, some are simple but not flexible, and others are complex but very versatile. Electronic Industries Association (EIA) standard RS-232

(RS means “recommended standard”) is one of the most common low-to-moderate performance standards in use. RS-232 has three levels of specification: some things are defined with exacting precision, for some other technical issues it provides several different options, and for other points it does not define any specifics. The standard has been updated periodically during its long existence to reflect technical developments and to clarify some ambiguities and inconsistencies. We wili study this standard because it is so common and because many of the issues it addresses arise in other standards besides RS-232.

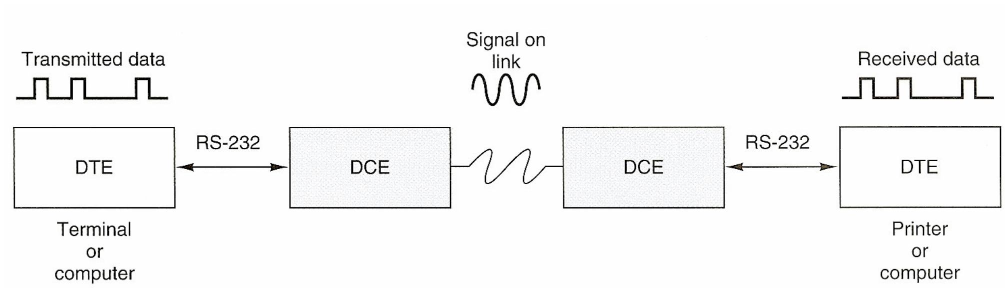



Figure 2.9 RS-232 used in DTE to DCE—Iink—DCE to DTE situation.

RS-232 is designed as the specification standard between two types of equipment (Figure 2.9). One is the data terminal equipment (DTE), which is the device that is generating the data to be sent or using the data that are received. For a variety of technical reasons, as we will see, the DTE is not electrically compatible with the data link. A special interface device called the data communications equipment (DCE) takes signals to and from the DTE and makes them compatible with the physical link. The physical and electrical connection interface between DTE and DCE is defined by RS-232, so that the DTE and DCE can plug directly into each other. At the other end of the communications link another DCE is connected, and it interfaces to the far-end DTE using the same RS-232 standard.

There does not even have to be a traditional “DTE to DCE-communications link-DCE to DTE” connection: many systems use the DTE as one active end of the communications system and the DCE as the other, as when a computer (DTE) is connected to its peripheral device: for example, a printer functioning as a DCE even though it is not really data communications equipment. Although the designations

DCE and DTE are somewhat outdated in light of how the standard is used, these labels are still used in much RS-232 documentation.

The RS-232 standard defines a scheme for asynchronous communications, where there is a specified timing between data bits but no fixed timing between the characters that the bits form. In contrast, synchronous communications requires specific timing between bits and characters, and the clock for recovering data bits must be synchronized with or derived from the bit stream. Asynchronous systems are simpler but transmit data at much lower rates and are therefore less efficient than synchronous systems.

Communication defined by RS-232 is serial. There is a single wire, or link, for each direction of data flow, and the bits of the message are sent in sequence, one at a time. Cornpared to parallel systems where multiple lines carry many data bits simultaneously, serial communication requires less circuitry at either end of the data link, is easier to manage, and needs only a single physical link. Of course, serial systems have lower throughput than parallel systems, but in practice a parallel link is practical only over short distances of a few feet.

Although the RS-232 standard defines a serial system with just a single wire for each direction, the standard allows for other signal wires between the DTE and DCE as well. These additional signals are used to manage the data interface and to indicate communications status at any time to both the DTE device and the DCE device.

Finally, the phrase serial ASCII format is often used in conjunction with RS-232. The formal standard itself does not specify at all what bit patterns should be used to represent the actual information being transmitted. In contrast, a straight binary format can express values up to 255 in an eight-bit field. However, unless noted otherwise, we will assume that the RS-232 data is in ASCII format.

RS-232 is primarily a signal voltage and timing, DTE-to-DCE interface standard. Just as it does not specify how the data should be represented by the bits, it does not define the overall message format and protocol. These definitions are left to the developers of the communications system. This means that RS-232 provides a great deal of flexibility, but it also can have problems when signal levels and timing agree but message format, content, and protocol differ between the two DTE devices that it interconnects in a complete system.

There are three areas where RS-232 defines a communications standard: signal voltages, the use of signal lines, and signal and bit timing. These are the lowest levels of a complete communications interface protocol. By specifying these, RS-232 develops a basic building block for more complex protocols and message interchange.

### *2.5.1 Signal Levels*

To represent a binary 1 or 0, two signal voltage spans are required (Figure 2.10). Any voltage between + 3 and + 25 V is a binary 0 (also known as a space, from historical use of mechanical teletypes), while a binary 1 (known as a mark) is any voltage from —3 to —25 V.

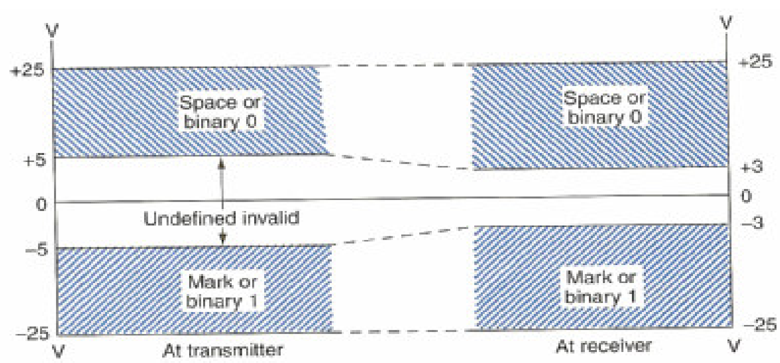



Figure 2.10 RS-232 specification voltage levels at the source and the receiver.

The voltage span from —3 to + 3 V is undefined (neither 1 nor 0) and should not exist in any system using the RS-232 standard. The signal transmitter sends any voltage between +5 and +25 V to indicate 0; the receiver decides that it has received 0 when it sees a voltage between + 3 and + 25 V. Note that the specification requires that the transmitted signal be either —5 to —25 V or + 5 to + 25 V but the received signal specification is relaxed to -3 and + 3 V, respectively, to allow for line voltage drop.

In practice, the system has greatest resistance to noise or voltage loss in wires when the largest voltage possible is used at the transmitter. This noise margin is critical to reliable system operation. When a 25-V signal is sent but is corrupted by 10 V of noise or has 10 V of loss, it still arrives at 15 V, which is a valid binary 0. In contrast, when a +3-V signal is used, there is no margin: any voltage decrease due to noise or line loss puts the signal in the undefined region.

For these reasons, RS-232 systems use a voltage as high as possible. The amount of noise margin needed depends on how much noise and line loss are realistically expected. For a very short interconnection, RS-232 levels of ± 15 V or even ±5 V are sufficient and are easier to generate with most common digital logic circuitry and power supplies. If the conditions are more difficult, the larger voltages are used for additional “insurance.” Modern systems, which often have just a 1.8-, 3.3-, or 5-V supply, tend to use lower voltages.

Although RS-232 requires bipolar (plus and minus) voltages, most electronic equipment in use operates from a unipolar (single) supply voltage. Some early systems “cheated” and used zero volts in place of the negative supply (a violation of the RS-232 standard), and the systems worked—somewhat. Performance was unreliable and inconsistent. Fortunately, IC manufacturers have developed power-supply ICs and RS-232 interface ICs that convert positive voltage to negative, using circuits known as charge pumps and dc/dc converters. Now, nearly all RS-232 interfaces, even those operating from a unipolar source, provide bipolar voltages.

### *2.5.2 Signal Lines*

Twenty-five signal lines are defined by the RS-232 standard. This does not mean that every DTE-to-DCE interface requires 25 signal wires—many RS-232 interfaces use just a few. The lines are divided into four groups: data, control, timing, and special secondary functions. The signal lines, arranged by line number and by function, are listed in Figures 2.11 and 2.12. RS-232 specifies a signal ground, used for all signal wires, plus a chassis ground for protection and to make sure that the DTE and DCE chassis are at the same electrical potential. In practice, the signal ground and chassis ground are often the same wire, or the chassis ground is omitted, which are reaily violations of the standard’s requirements; fortunately, this usually works, but it can cause operational (or even safety) problems.

The data lines are the most important signals. Received data and transmitted data lines permit full-duplex communication between the DTE and DCE. To make sure that there is no confusion on the direction of data flow on the received data and transmitted data lines, the RS-232 standard specifies data flow directions from the perspective of the DTE: data go from the DTE to the DCE on the transmit data line, and data go from the DCE to the DTE on the received data time. From the perspective of the DCE, then, data is transmitted on what is called the received data line and received on what is called the transmitted data line. This may seem confusing but it really avoids the more serious problem that will occur if both units try to transmit on the same line (signals clash) or receive on the same wire (there is no signal present: there are two listeners on the same line and no signal source).

| Line pin | Description |
|---|---|
| 1 | Protective ground |
| 2 | Transmitted data |
| 3 | Received data |
| 4 | Request to send |
| 5 | Clear to send |
| 6 | Data set ready |
| 7 | Signal ground (common return) |
| 8 | Received line signal detector (data carrier detect) |
| 9 | Reserved for data set testing |
| 10 | Reserved for data set testing |
| 11 | Unassigned |
| 12 | Secondary received line signal detector |
| 13 | Secondary clear to send |
| 14 | Secondary transmitted data |
| 15 | Transmission signal element timing (DCE source) |
| 16 | Secondary received data |
| 17 | Receiver signal element timing (DOE source) |
| 18 | Unassigned |
| 19 | Secondary request to send |
| 20 | Data terminal ready |
| 21 | Signal quality detector |
| 22 | Ring indicator |
| 23 | Data signal rate selector (DTE/DCE source) |
| 24 | Transmit signal element timing (DTE source) |
| 25 | Unassigned |

Figure 2.11 Function of RS-232 lines, by pin number.

| | Line pin | Description | Gnd | Data | | Control | | Timing | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | From DCE | To DCE | From DCE | To DCE | From DCE | To DCE |
| Data | 1 | Protective ground | × | | | | | | |
| | 7 | Signal ground/common return | × | | | | | | |
| Control | 2 | Transmitted data | | | × | | | | |
| | 3 | Received data | | × | | | | | |
| | 4 | Request to send (RTS) | | | | | × | | |
| | 5 | Clear to send (CTS) | | | | × | | | |
| | 6 | Data set ready (DSR) | | | | × | | | |
| | 20 | Data terminal ready (DTR) | | | | | × | | |
| | 22 | Ring detector | | | | × | | | |
| | 8 | Received line signal detector or data carrier detect (DCD) | | | | × | | | |
| | 21 | Signal quality detector | | | | × | | | |
| | 23 | Data signal rate selector (DTE) | | | | | × | | |
| | 23 | Data signal rate selector (DCE) | | | | × | | | |
| Timing | 24 | Transmitter signal element timing (DTE) | | | | | | | × |
| | 15 | Transmitter signal element timing (DCE) | | | | | | × | |
| | 17 | Receiver signal element timing (DCE) | | | | | | × | |
| Secondary group | 14 | Secondary transmitted data | | | × | | | | |
| | 16 | Secondary received data | | × | | | | | |
| | 19 | Secondary request to send | | | | | × | | |
| | 13 | Secondary clear to send | | | | × | | | |
| | 12 | Secondary received line signal detector | | | | × | | | |

Figure 2.12 RS-232 lines, organized by functional grouping together with direction of signal flow.

The control lines are next in importance (Figure 2.13). These lines are used for hand shaking, so either the DTE or the DCE can signal to the other that there is data to be transmitted and can indicate to the other if either DTE or DCE is ready to accept new data. Four control lines are used most frequently:

- Request to Send (RTS) from the DTE signals the DCE that the DTE has new data it would like to transfer.
- Clear to Send (CTS) from the DCE indicates to the DTE that the DCE can accept the new data.
- Data Set Ready (DSR) lets the DCE tell the DTE that DCE can accept new data.
- Data Terminal Ready (DTR) from the DTE is another indication that the DTE is ready.

Note that in the RS-232 standard the handshake lines are asserted or active (binary value = 1) when they are in the space zone of + 3 to +25 V, which also corresponds to a binary 0 for data. This sometimes causes confusion during troubleshooting, since an asserted control line is said to be in the binary 1 state, yet its signal voltage is the same as a data bit that is binary 0. As long as everyone follows the standard, however, there is no operating problem.

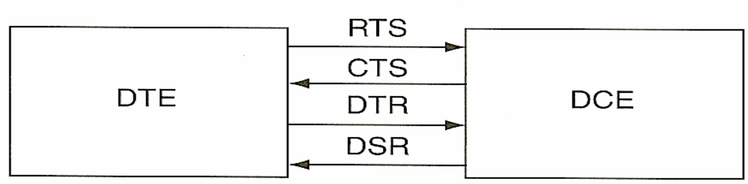



Figure 2.13 Four key control (handshaking) lines for DTE/DCE interfacing.

### *2.5.3 Signal and Bit Timing*

One more element is needed to complete the RS-232 picture: how the actual data bits are presented in the asynchronous environment. We will look at it with the specific values of RS-232 operation (see Figure 2.14).

Between data bits, when no signal is present, the RS-232 transmitted data line is at the marking level, also called the "idle" state. When there is data to send, the line goes to the space state for one bit period called the start bit. This transition, indicating that a new series of data bits will follow is sensed by circuitry in the RS-232 receiver as a warning to "wake up" and look for the incoming data bits.

The start bit is followed by five, six, seven, or eight data bits, depending on the de sign of the system (seven and eight bits are most common). The LSB of the data bits is sent first. After the last data bit, there is an optional parity bit for error detection for systems that are using parity. The entire sequence of data and parity bits is terminated by a stop bit (or bits), where the data line goes back to the marking state for 1, 1½, or 2 bit periods. This indicates that the entire sequence of data bits is complete.

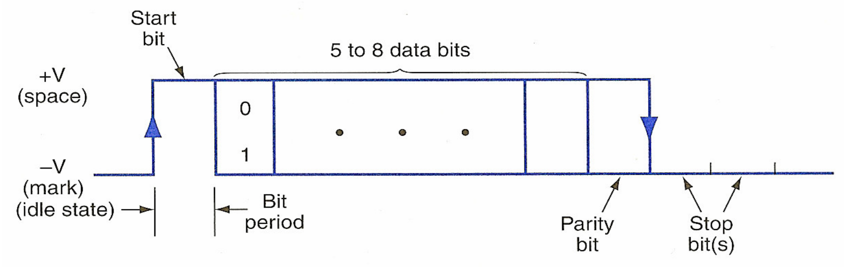



Figure 2.14 RS-232 signaling, with mark and space values for start, data, parity, and stop bits.

Let's look at how the letter B (1000010) looks in RS-232 format, using ASCII representation, along with seven bits, no parity, and one stop bit format (Figure 2.15a): 1 0100001 0 (spaces between bits are for printed clarity; also, the LSB of the character is sent first). For the digit 8, in ASCII with seven bits, no parity, and two stop bits, the pattern is 1 0001110 00 (Figure 2.15b). For the same example with even parity, it is 1 0001110 1 00 (Figure 2.15c).

From this we see that the RS-232 interface allows choices: number of data bits (five, six, seven, or eight), parity (none, even, or odd), and number of stop bits (1, 1½, or 2). The transmitter and receiver must be set to the same choices, or else all bits may be received perfectly but make no sense. For example, if the transmitter is set up for even parity and 1 stop bit while the receiver expects no parity and 2 stop bits, there will be confusion and apparent errors ("apparent" because the bits are received as sent, but misinterpreted).

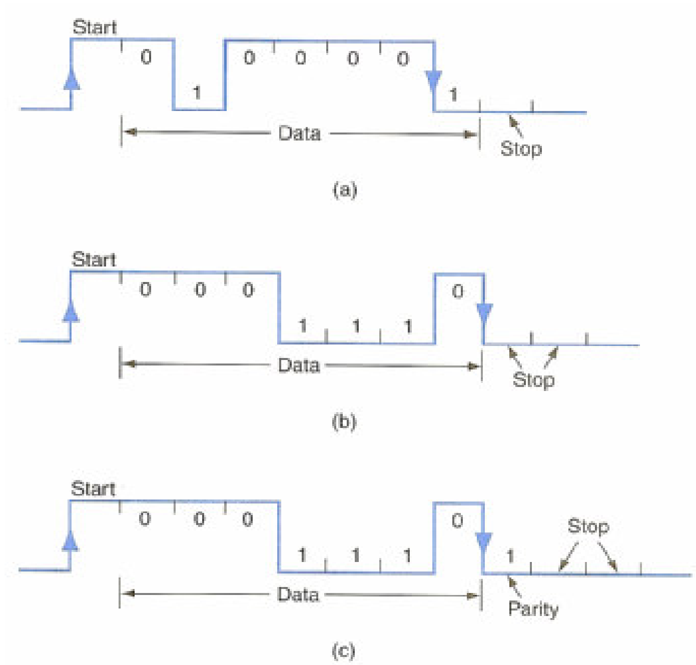



Figure 2.15 Three ASCII RS-232 examples: (a) seven-bit letter B, no parity, one stop bit; (b) 7-bit digit 8, no parity, two stop bits; (c) seven-bit digit 8, even parity, two stop bits.

### *2.5.4 Bit Periods and Timing*

The RS-232 standard allows many baud values, up to a maximum of 20,000 baud (normally, 1 baud = 1 bit/s). Common rates are 110, 300, 600, 1200, 2400, 4800, 9600, 14,400, and 19,200 bits/s. The 300-bit/s rate was used for now obsolete mechanical teletypes or very low-bandwidth, noisy channels, while most electronic systems run at the higher values of 1200, 2400, or 4800 baud and above. A single bit period is the inverse of the baud value; for example, the bit period is 0.83 millisecond at 1200 baud. Although 19,200 baud is the fastest standard value within the 20,000-baud maximum value specified by RS-232, some systems communicate at up to twice that value (38.4 kilobaud).

Indication to stop is sent before the buffer is 100% full, so that any characters already in the process of being sent will be completed, and to provide a small cushion in case the DTE does not respond immediately to the change in CTS status.

Then, as the DCE processes it's received characters and empties the buffer by transmitting aver the link, it signals once again to the DTE that it can accept new character bits.

### *2.5.5 RS-232 ICs*

The simplistic way to generate the necessary start, data, parity, and stop bits for an RS-232 interface is to have the system microprocessor control one digital output bit of a multiple bit port. The microprocessor must also generate the necessary handshake lines on other out put bits of this port, sense incoming handshake signals on bit input lines, and receive in formation bits on an input line. However, such an approach is very inefficient. The microprocessor and software would have to devote a great deal of time to servicing the RS- 232 port, and perform the following operations to transmit:

- Determine what the data bits are, as well as add the start bit ahead of them.
- Calculate the parity bit (even or odd), if there is one.
- Add a stop bit (1, 1 or 2) after the data bits.
- Manage the handshake lines to generate an RTS signal, continually monitor CTS, and send out the bits only when the CTS line indicates that it is OK.

For receiving data, the task is equally complex:

- Check the status of handshake lines for incoming data, and indicate to the remote device if it is OK to send (if the microprocessor can accept new bits).
- Receive the bits, strip off the start and stop bits.
- Strip off the parity bit (if any), calculate the parity of the received data bits, and check to see if the parity bit as received agrees with the parity bit as calculated.

All these activities, for transmit and receive, would have to be done with precise timing at the specified baud value. Any deviation from the proper timing would cause an error: transmitted bits would be sent out with the wrong timing, while received bits would be misinterpreted because they could not be checked at the right instants or the microprocessor would not be there to receive them. It is impractical to use the microprocessor and soft ware directly to manage the RS-232 interface (or any other communications format) for any but the very slowest speeds of 100 to 300 baud.

Instead, a special IC is used which relieves the microprocessor from dealing directly with the RS-232 interface. The universal asynchronous receiver/transmitter (UART) IC is designed to provide the interface functions between the microprocessor and the RS-232 signal lines.

# CHAPTER THREE

# DESIGN OF THE VEHICLE

It is supposed to give brief information how to design and how to choose compatible tools in this chapter.

## 3.1 Mechanical Design of the Vehicle

Many of design aspects have been considered in mechanical design. Firstly, it has been dediced how the vehicle motion would be, and then started making mechanical parts with this consideration.

Taking the carrying capacity and the size into consideration, mechanical parts of the vehicle has been selected. In this chapter, technical aspects of the mechanical parts will be discussed.

### *3.1.1 Design of the Metal Frame*

As having been aware of carriying capacity and the load, steel frames were used. Assembling the frames together, a window frame was made. Afterwards, steel plates on these frames were put and applied welding process to put them together as shown in Figure 3.1.

Further detailed information will be given in application part.

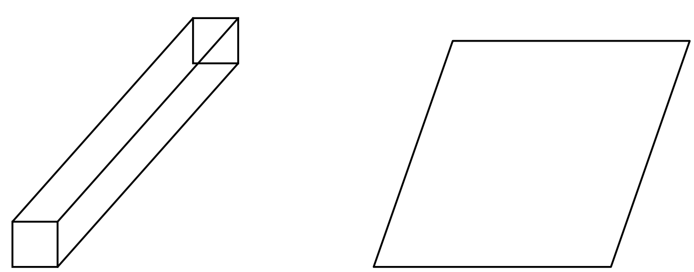


Figure 3.1 Steel frame and plate are desinged.

### *3.1.2 Selection Motor and Gearbox*

Two motors were used so that the vehicle could turn left and right side. The motors are compatible with 12 V DC and output power is 190 Watt. This power is sufficient to meet the required motion. The speed of the motors is 1400 cycle/minute.

To decrease the period speed and to gain more torque gearboxes were used as shown in Figure 3.2.

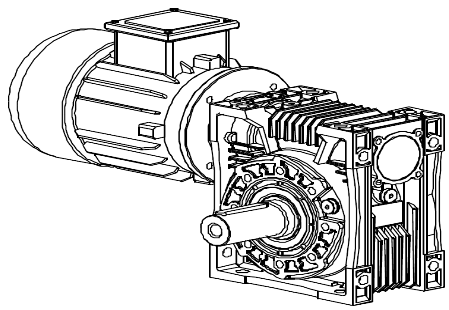


Figure 3.2 Motor and gearbox are assembled.

Gearboxes are called TSR40r and the cycly proportion is 1/16. In this manner, by decreasing the speed, the torque is increased.

### *3.1.3 Selection Wheel and Triger Belt*

The wheels and belts which are appropriate for motion of the vehicle were selected. Triger belts were used to make palet system. In figure 3.3 an example of belts is shown.

Figure 3.3 Example pictures of triger (time) belt.

After that, two triger gears T10 50 were connected to the gearbox in order to transfer motion from motor to triger belt. In figure 3.4, two pictures of triger gear are shown. There are some advantages of triger gear which are listed below.

- Increasing its life by making surface hard covered.
- It is light. So, load is too small for axle and gearbox.
- Corrosion problem does not exist.

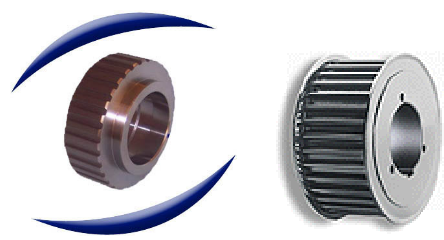


Figure 3.4 Triger gear is shown.

Beause of the fact that polyemid material is strong and easy to process, whells which are made of polyemid material have been used. Two pictures of raw polyemid bar material are shown in figure 3.5.

Figure 3.5 Polyemid bars.

## 3.2 Design of Electronic Circuits

Firstly, a driver circuit for controlling of motors was designed and then, two main boards for switching and using communication wireless system are considered. Microcontroller and transceiver modules are choosen. Finally, the circuits are designed. In this chapter, design of circuits and some information about electronic equipments will be given.

### *3.2.1 Design of Power Circuit*

Three different circuits are required. These are transmitter circuit, receiver circuit and motor power circuit.

- 220 V AC / 12 V DC adaptor for transmitter circuit.
- 12 V DC / 7 Ah acumulator for receiver circuit.
- 12 V DC / 65 Ah acumulator for motor power circuit.

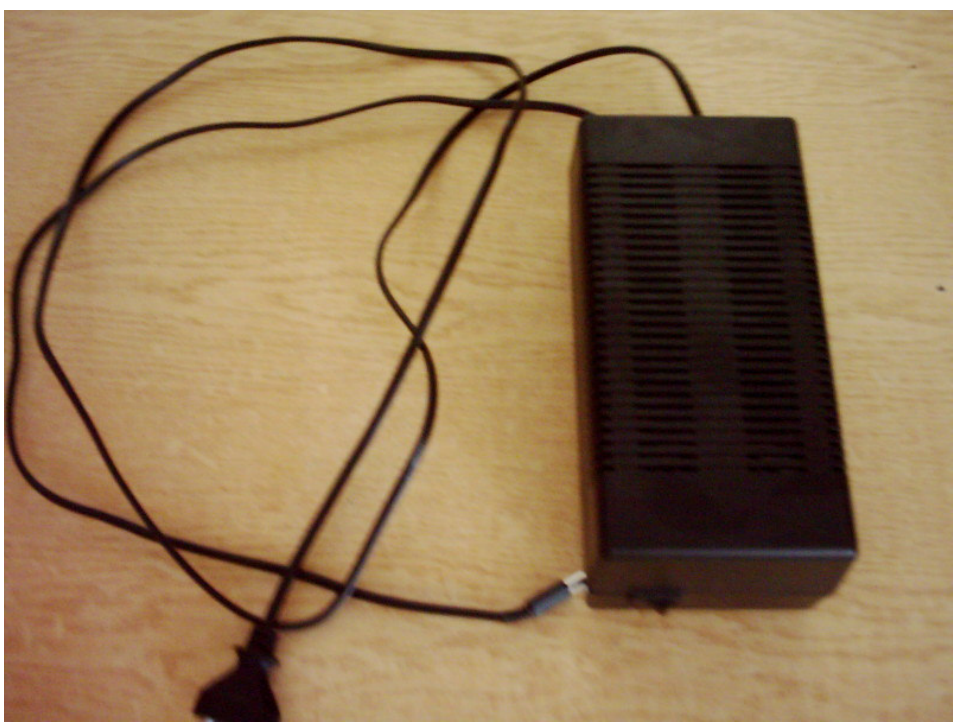


Figure 3.6 The power supply.

### *3.2.2 Selection of Microcontroller*

Two main boards (transmitter and receiver circuit) are choosen. One of them is connected to computer and the other one is on the vehicle. PWM switching signals are used for motion control of motors. So, a microcontroller for these specifications is selected. This IC is PIC 16F877 produced by Microchip Inc. Basic features of PIC 16F877 are given below. This IC's pin diagram is shown in figure 3.7.

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
  Up to 368 x 8 bytes of Data Memory (RAM)
  Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
  Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins

- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 0.6 mA typical @ 3V, 4 MHz
  - 20 μA typical @ 3V, 32 kHz
  - < 1 μA typical standby current

**Peripheral Features:**

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. Resolution is 12.5 ns
  - Compare is 16-bit, max. Resolution is 200 ns
  - PWM max. Resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI (Master mode) and I2C (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

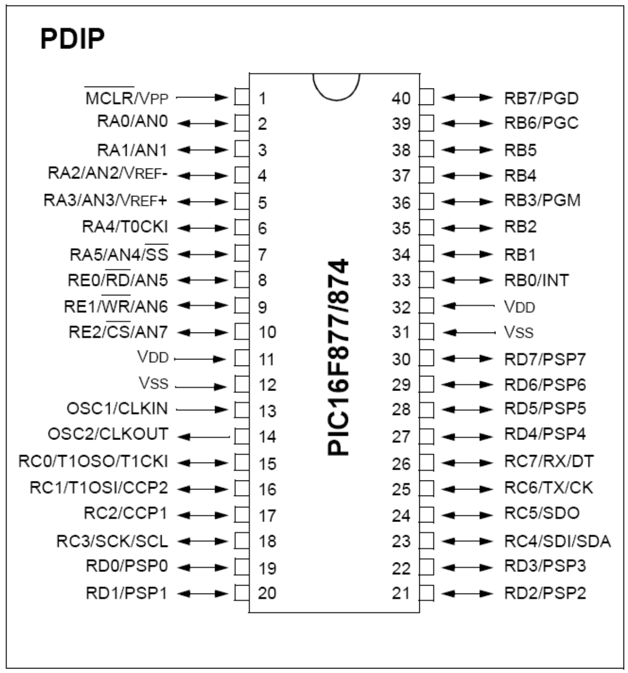



Figure 3.7 PIC 16F877 pins.

### *3.2.3 Selection of Transceiver Module*

Using two UTR-C12U transceiver modules, wireless communication between transmitter circuit and receiver circuit are realized. There is a specification of serial port communication at these modules. Pinout of module is shown in figure 3.8.

- 434 MHz or 868 MHz UHF band. Compatible with European EN 300 220 standard.
- High frequency stability
- Low current consumption, ideal for mobile application.

| Pin No | Pin-Name | Input/Output | Description | |
|---|---|---|---|---|
| 1,3,4 | GND | | Connection to GND. | |
| 2 | ANT | | Antenna connection. | |
| 5 | DIO | I/O | Data Input/Output | |
| 6 | DCLK | - | Clock Output | UTR-C12M |
| | NC | | - | UTR-C12U |
| 7 | NC | - | - | |
| 8 | NC | - | - | |
| 9 | T/R | I | Transmitter/Receiver selection | UTR-C12M/UTR-C12U |
| 10 | CH ½ | I | Channel selection | |
| 11 | +3V | | The power supply terminal | use regulated voltage source. |

Figure 3.8 UTR-C12U transceiver module pins.

| | Min. | Typ. | Max | Unit | Not |
|---|---|---|---|---|---|
| Voltage supply | 2,7 | 3 | 3,3 | Vdc | use regulated voltage source . ±100 mV |
| Supply current TX mod | | 23.4 | | mA | |
| Supply current RX mod | | 16.9 | | mA | |
| Logic "0" DI volt | 0 | | 0.3Vcc | Vdc | |
| Logic "1" DI volt | 0.7Vcc | | Vcc | Vdc | |
| Logic "0" DO volt | 0 | | 0.4 | Vdc | |
| Logic "1" DO volt | 2.5 | | Vcc | Vdc | |
| Working Temperature | -20 | | +55 | °C | ETSI 300 220 |
| Storage Temperature | -50 | | +150 | °C | |
| Dimensions | 42 X 28 X 8 mm | | | | |

Figure 3.9 Technical features of the UTR-C12U.

### *3.2.4 Design of Power Circuit of Motor*

Properties of motor are 12 volt direct current (DC), 190 watt power, and 1400 cycle/minute. To drive the motors having the specification above, a power circuit in the shape of H-bridge using IRFP064N MOSFET (Metal Oxide Semiconductor Field Efect Transistor) is designed.

Technical specifications and pinout about this compenent are given below.

- Dynamic dv/dt rating
- Repetitive avalanche rated
- Ultra-low on resistance
- Very low thermal resistance
- Isolated central mounting hole
- Fast switching

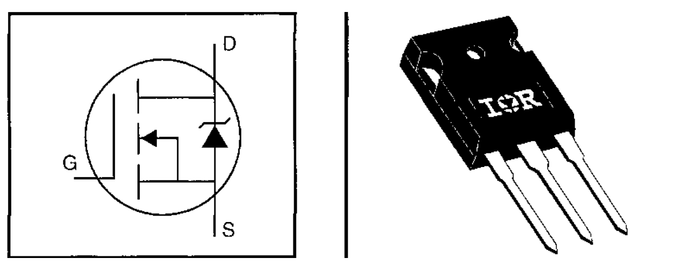



Figure 3.10 IRFP064N pins.

### *3.2.5 Design of Driver Circuit*

PWM signal is used for motion control of motors. A driver circuit is designed for coding the PWM signals.

Receiver circuit operates with 5 volt. However, driver circuit uses 12 volt. So, 6N138 optocoupler integrated circuit on the input part of driver circuit is designed. Figure 3.11 shows pinout of 6N138.

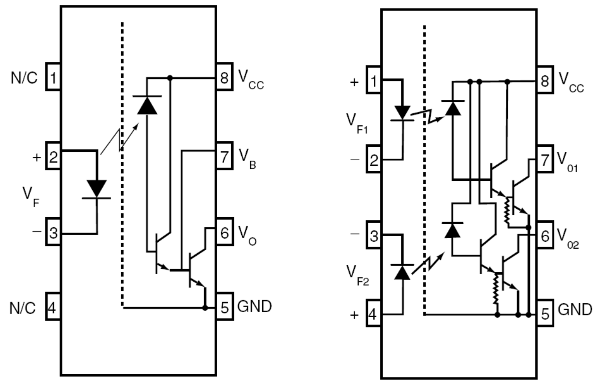



Figure 3.11 6N138 pins.

IR2113 integrated circuit for H-bridge in the driver circuit is used. Some technical information about IR2113 is given below and in figure 3.12 shows pinout of IR2113.

- Floating channel designed for bootstrap operation
- Fully operational to +500V or +600V
- Tolerant to negative transient voltage dV/dt immune
- Gate drive supply range from 10 to 20V
- Undervoltage lockout for both channels
- 3.3V logic compatible
  Separate logic supply range from 3.3V to 20V
  Logic and power ground ±5V offset
- CMOS Schmitt-triggered inputs with pull-down
- Cycle by cycle edge-triggered shutdown logic
- Matched propagation delay for both channels
- Outputs in phase with inputs

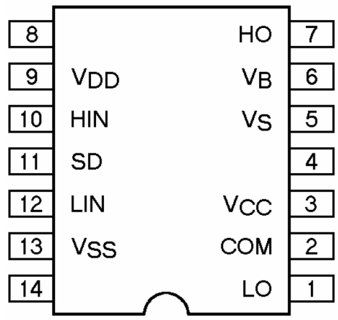



Figure 3.12 IR2113 integrated circuit is pinout.

There are two PWM signals which control the speed of the motors out of receiver circuit and four outputs to control the direction of the motors.These six signals were reduced four PWM signals using an AND logic gate so as to drive motors.

### *3.2.6 Preparing Computer Interface Circuit*

Transmitter circuit uses serial port for communication. Devices which use serial cables for their communication are split into two categories. These are DCE (Data Communications Equipment) and DTE (Data Terminal Equipment.) Data Communications Equipment is devices such as your modem, TA adapter, plotter etc while Data Terminal Equipment is your Computer or Terminal.

The electrical specifications of the serial port are contained in the EIA (Electronics Industry Association) RS232C standard. It states many parameters such as –

- A “space” (logic 0) will be between +3 and +25 volts.
- A “mark” (logic 1) will be between -3 and -25 volts.
- The region between +3 and -3 volts is defined.
- An open circuit voltage should never exceed 25 volts.
- A short circuit current should not exceed 500 mA.

Serial Ports come in two "sizes", there are the D-Type 25 pin connector and the D-Type 9 pin connector both of which are male on the back of the PC, and thus you will require a female connector on your device. Below is a table of pin connections for the 9 pin and 25 pin D-Type connectors.

Table 3.1 D tip 9 pin and D tip 25 pin connectors.

| D type 25 pin no | D type 9 pin no | abbreviation | Full name |
|---|---|---|---|
| Pin 2 | Pin 3 | TD | Transmit data |
| Pin 3 | Pin 2 | RD | Receive data |
| Pin 4 | Pin 7 | RTS | Request to send |
| Pin 5 | Pin 8 | CTS | Clear to send |
| Pin 6 | Pin 6 | DSR | Data set ready |
| Pin 7 | Pin 5 | SG | Signal ground |
| Pin 8 | Pin 1 | CD | Carrier detect |
| Pin 20 | Pin 4 | DTR | Data terminal ready |
| Pin 22 | Pin 9 | RI | Ring indicator |

Table 3.2 Pin functions.

| abbreviation | Full name | Function |
|---|---|---|
| TD | Transmit data | Serial data output (TXD) |
| RD | Receive data | Serial data input (RXD) |
| RTS | Request to send | This line informs the Modem that the UART is ready to exchange data. |
| CTS | Clear to send | This line indicates that the Modem is ready to exchange data. |
| DSR | Data set ready | This tells the UART that the modem is ready to establish a link. |
| CD | Carrier detect | When the modem detects a "Carrier" from the modem at the other end of the phone line, this Line becomes active. |
| DTR | Data terminal ready | This is the opposite to DSR. This tells the Modem that the UART is ready to link. |
| RI | Ring indicator | Goes active when modem detects a ringing signal from the PSTN. |

The UART is a sophisticated IC that provides management of the RS-232 interface bits and signals. However, it is not designed for actual physical connection to the data link, since any signal wire connected outside its original circuit board or chassis is subject to all types of abnormal conditions or miswiring. The complete RS-232 interface requires two types of ICs: a UART type, which provides management of the interface, and a more rugged physical interface IC. The physical interface IC is designed to withstand, without damage, voltages and currents that would damage an ordinary IC such as the UART. The UART IC is usually a +5-V component, while the RS-232 interface requires signals up to ±25 V and may be accidentally connected to higher voltages.

The use of a UART combined with a line driver/receiver pair is standard in RS-232 communications. The UART provides the interface management, while the line driver and receiver provide the higher voltages and appropriate interface to the physical link. In addition, by separating the UART functions from the physical interface, the communications system can switch to another type of physical link or to a more sophisticated UART device.

In addition, figure 3.13 shows pin diagram of MAX232 integrated circuit.

- Supply Voltage (VCC) ............................................-0.3V to +6V
- Input Voltages
  TIN..............................................................-0.3V to (VCC - 0.3V)
  RIN …………………… ......................................................±30V
  RIN ……………......................................................................±25V
  TOUT …………………………… .....................................±15V
  TOUT ………….................................................................±13.2V
- Output Voltages
  TOUT.......................................................................................±15V
  ROUT.........................................................-0.3V to (VCC + 0.3V)
- Driver/Receiver Output Short Circuited to GND.........Continuous
- Continuous Power Dissipation (TA = +70°C)

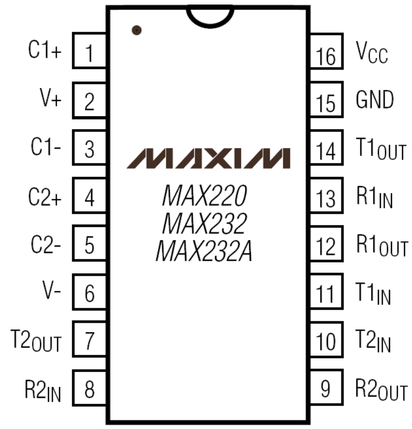



Figure 3.13 MAX232 pins.

## 3.3 Technical Features of IP3137 Wireless Network Camera

In this thesis, a wireless network camera has been used. Below, some features of the camera are given.

- High performance MPEG4 compression engine
- Audio capability with built-in microphone
- Built-in web server
- 10/100 Mbps ethernet
- 802.11g wireless LAN with WEP & WPA support
- 4X digital zoom
- Easy homepage customization
- Built-in intelligent motion detection
- 16-channel recording software
- UPnP/DDNS support

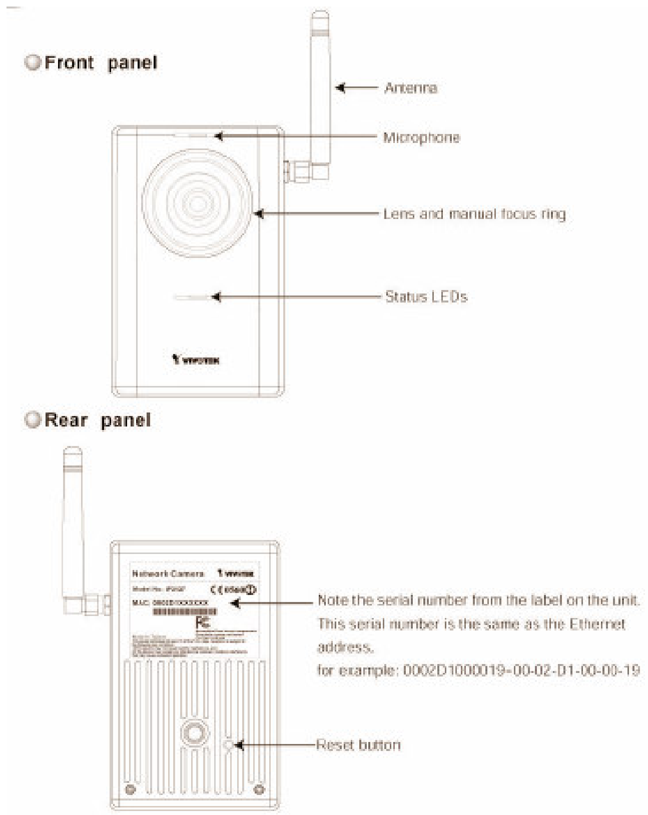



Figure 3.14 Physical description of the IP3137 is shown.

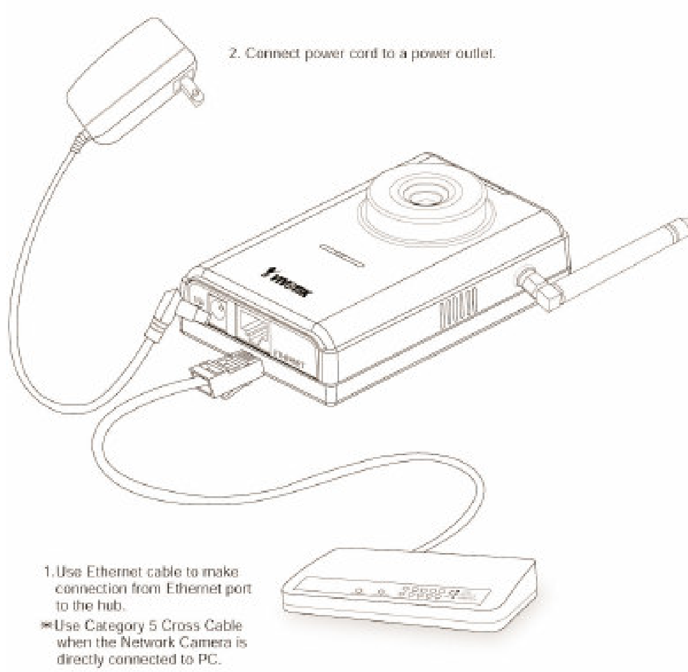



Figure 3.15 Install the camera in ethernet network.

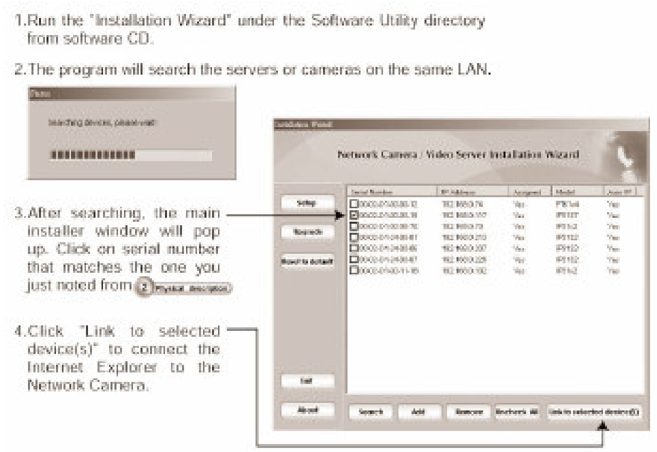



Figure 3.16 Assing IP address.

The video of the camera can be watched using internet browser.

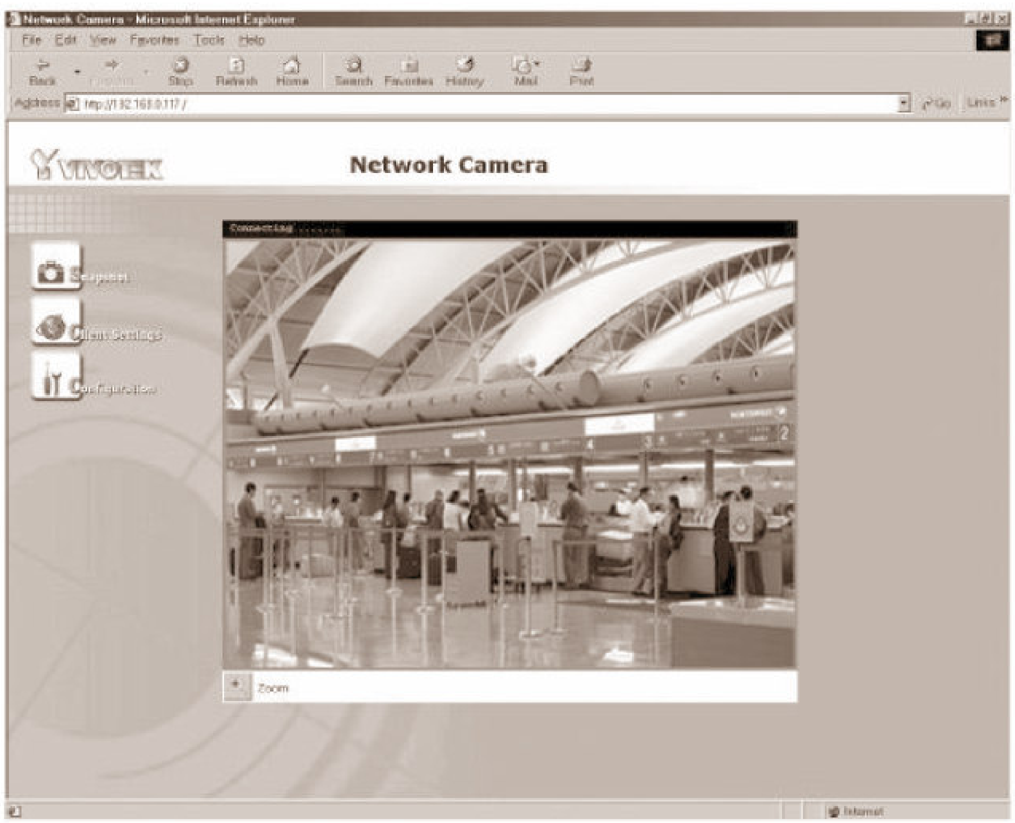



Figure 3.17 Usage of the camera with internet browser.

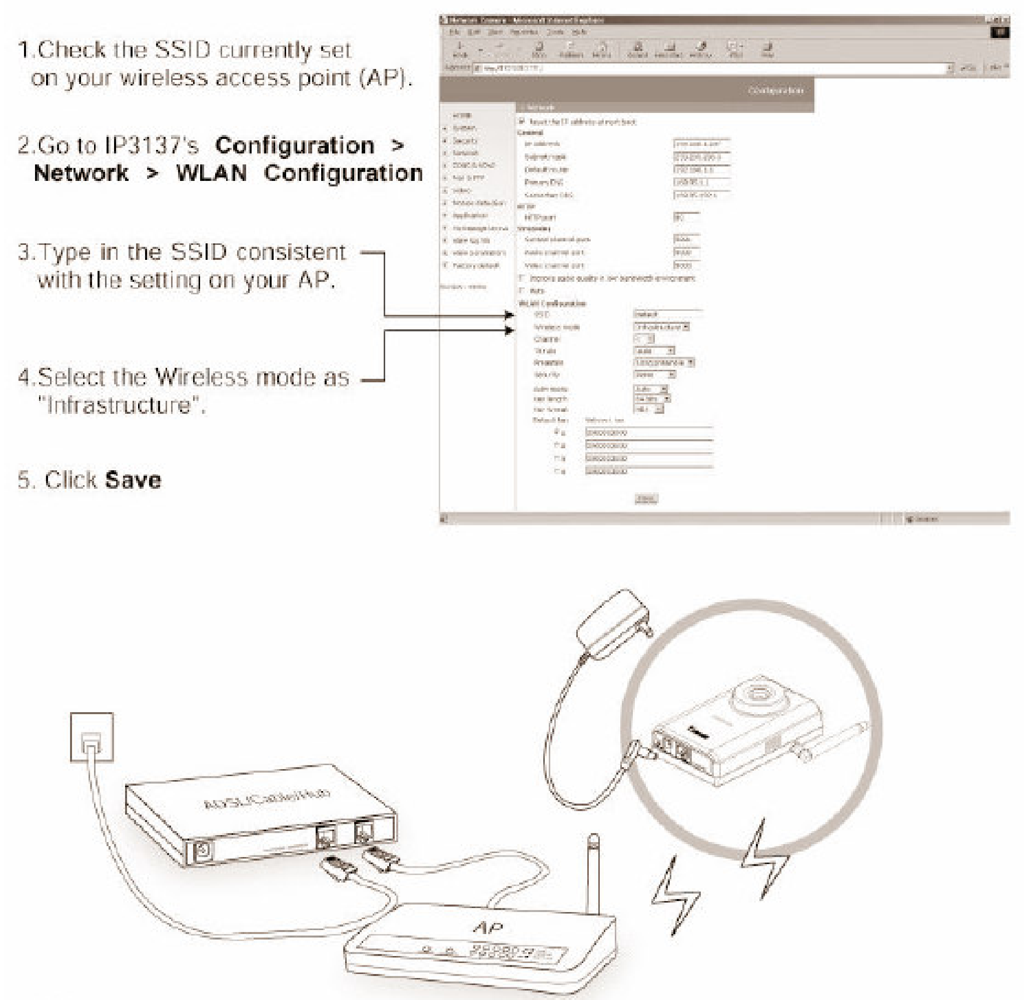



Figure 3.18 Set up wireless configuration for the network camera.

## 3.4 Design of Computer Software

Transmitter control circuit is uses serial port of computer. According to that, the computer software is prepared. C++ programming language was selected as a computer tool for this purpose.

An interface which can be activated by using mouse or direction keys of keyboard has been built in order to control the vehicle via the serial port of the computer.

## CHAPTER FOUR

## PRACTICAL CONSIDIRATIONS IN THE DESIGN OF VEHICLE

This chapter aims to give detailed knowledge about realization of the vehicle. The processes followed will be explained step by step until the vehicle reaches its final form from beginning. In addition, connection schemes of the electronic equipment will be shown.

### 4.1 Manufacturing of the Vehicle's Mechanical Part

Mechanical parts of the vehicle were manufactured in three phases which are manufacturing of metallic frame, motor, gearbox, triger gear, wheel, and triger belt.

#### *4.1.1 Manufacturing of Metallic Frame*

Steel profile and 4mm steel sheet were used for manufacturing of the vehicle's metallic part. Firstly a frame was formed by welding steel profiles like in the following figure.

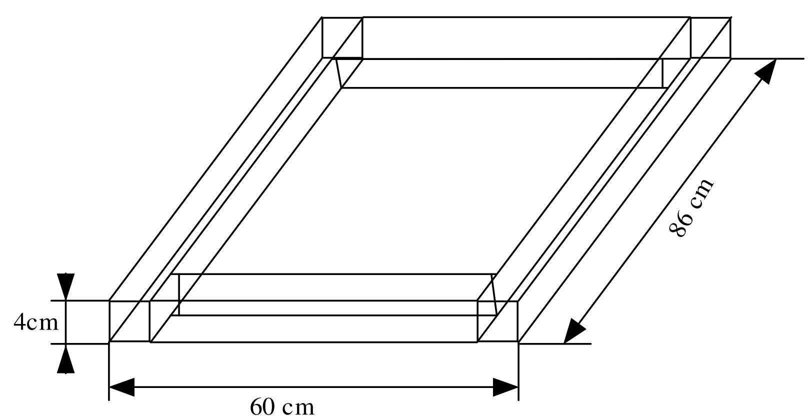



Figure 4.1 Steel frame was attached.

After that, steel sheet was placed on the steel frame like in the following figure.

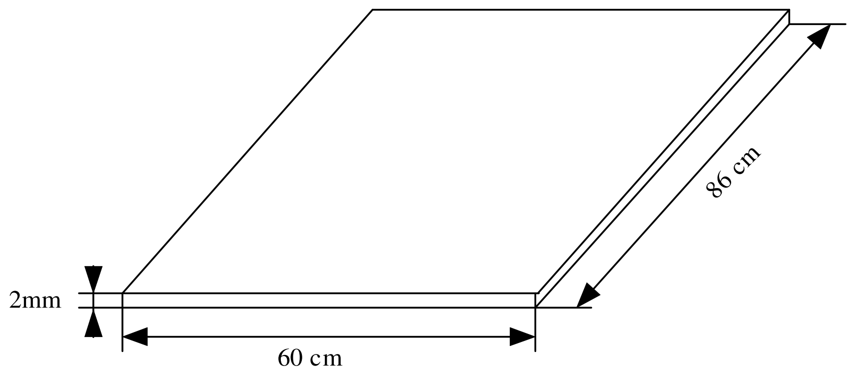



Figure 4.2 Steel sheet was prepared.

The steel fringe was welded to the steel frame. Thus, the vehicle's metallic body was finished. Metallic body is shown in the following figure.

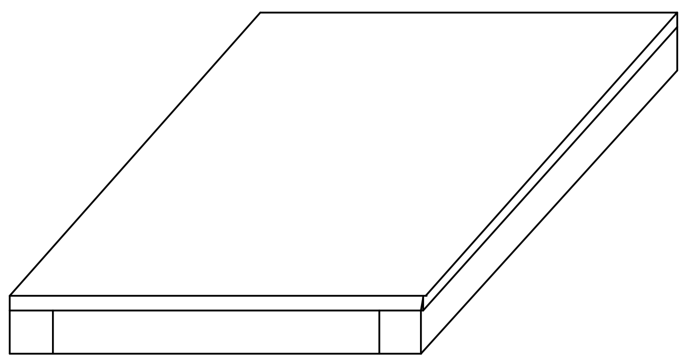


Figure 4.3 Metal body was assembled.

### *4.1.2 Assembling of Motor and Gearbox*

Motor's label values are shown in the following:

- Source voltage: 12 Volt DC
- Power: 190 Watt
- Rotation: 1440 period/minute

Two motors meeting the required specifications are obtained. However, speed of the motors is too high to design a powerful vehicle. Because of this reason, two gearboxes attached on the tips of the motors are used. The speed of the vehicle is calculated and suitable gearboxes are choosen. The label values of gearboxes are shown in the following list.

- Model: S40
- Transfer rate: 1/16
- Rotation: 87.5 rpm

Motor and gearbox which were assembled are shown in the following figure.

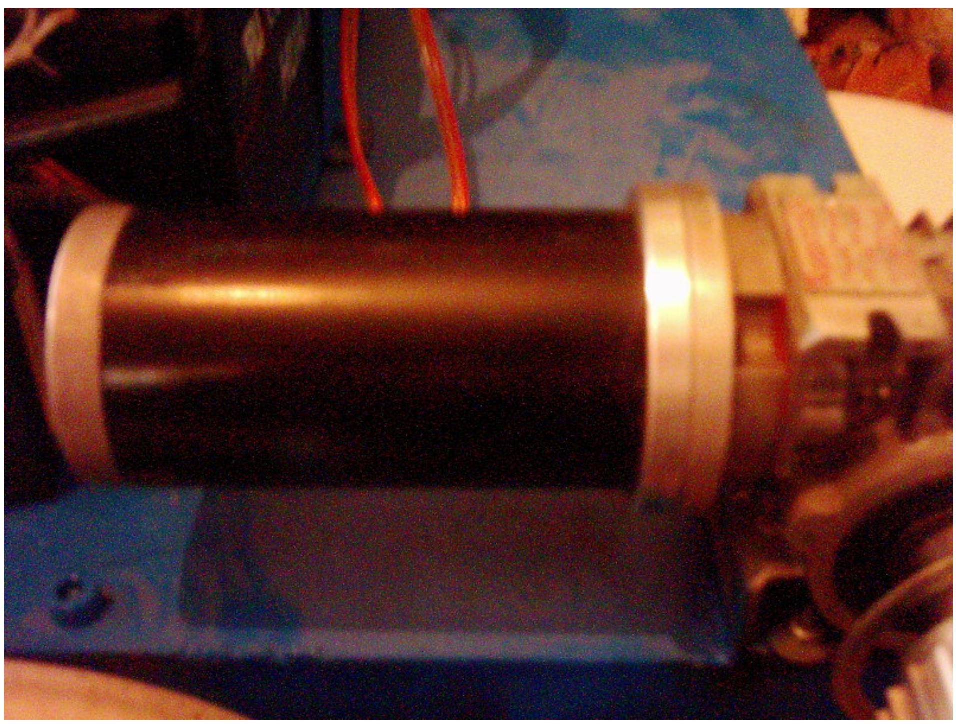


Figure 4.4  Motor and gearbox.

Triger belt is used for the movement of the vehicle. So, two triger gears are used to transmit the movement taken from the gearbox shaft to the triger belt. The gears are assembled to gearbox shaft. It is fastened together by a segment.

The block composed of gearbox, motor and triger gear are assembled on the metallic body as shown in figure 4.5.

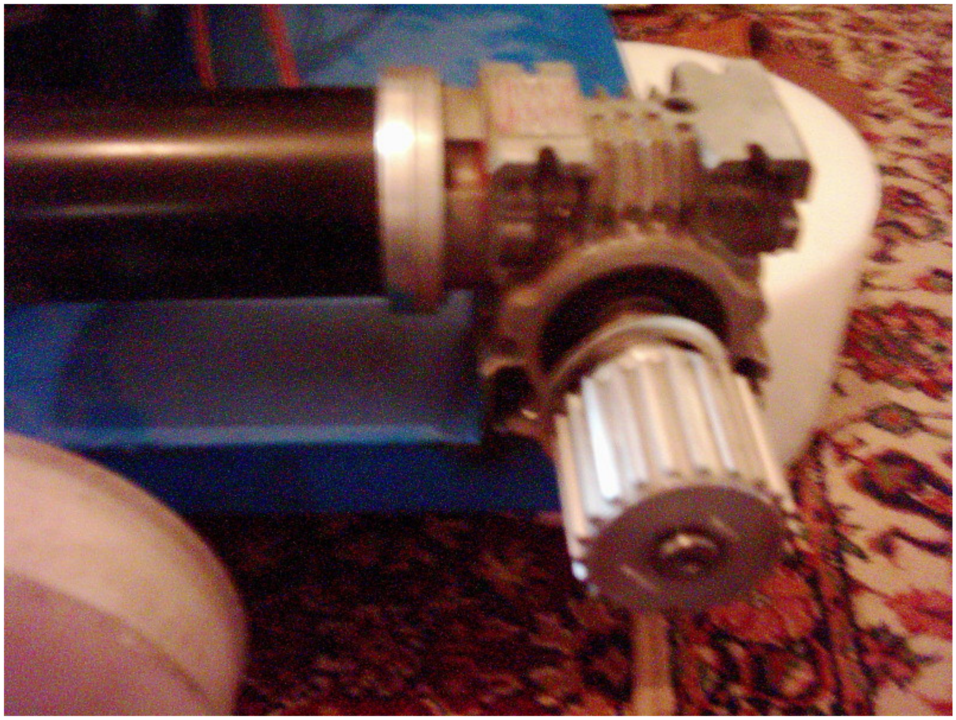


Figure 4.5 Block of motor, gearbox and triger gear.

### *4.1.3 Assembling of Wheels and Belt*

The vehicle has the same palette system as a tank and moves using the motors at both sides. By processing polyemid bars, four wheels are obtained.

Firstly, four polyemid bars are bought from the market. They are shrinked at wanted size. After, they are pierced. After then canals are carved their surrounds for triger belt. The final state and physical size of wheels are shown in the following figure.

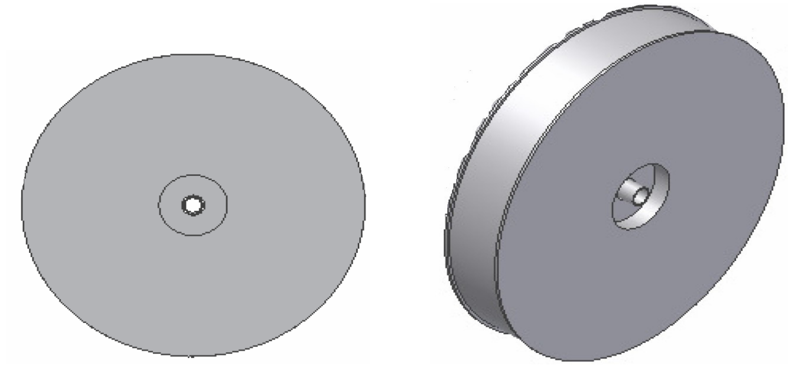


Figure 4.6 Using wheels are shown.

These wheels and a mil are attached together and rulmans are joined to the mil's point. The inner diameter of the rulmans is 25 mm and the outer diameter of the rulmans is 62mm.

Figure 4.7 Connection of wheels.

Constructed mechanism is attached with screw on the metalic body and then, two belts are provided according to triger gear and wheel sizes.

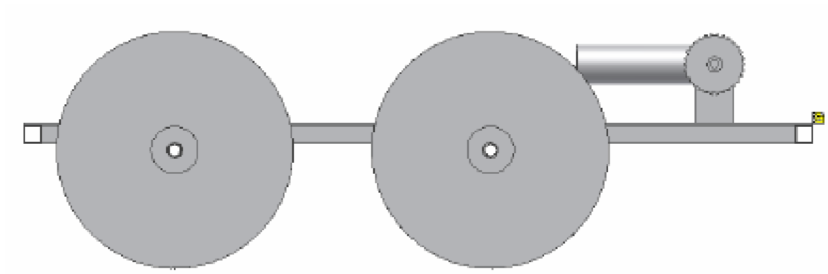


Figure 4.8 Assembling wheels to metallic body.

This belt's physical specifications are shown in the following figure.

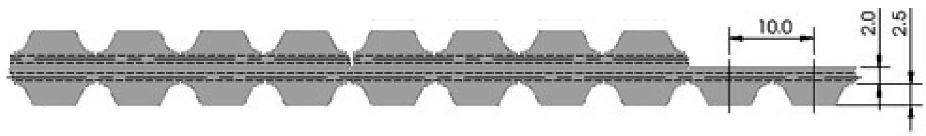



Figure 4.9 Triger belts specifications.

After that, triger belts are put on prepared triger gear and wheels and the vehicle's metalic part is finished. The vehicle can be driven by applying 12 Volt DC on the motors. The vehicle's whole metalic part is shown in the following figure.

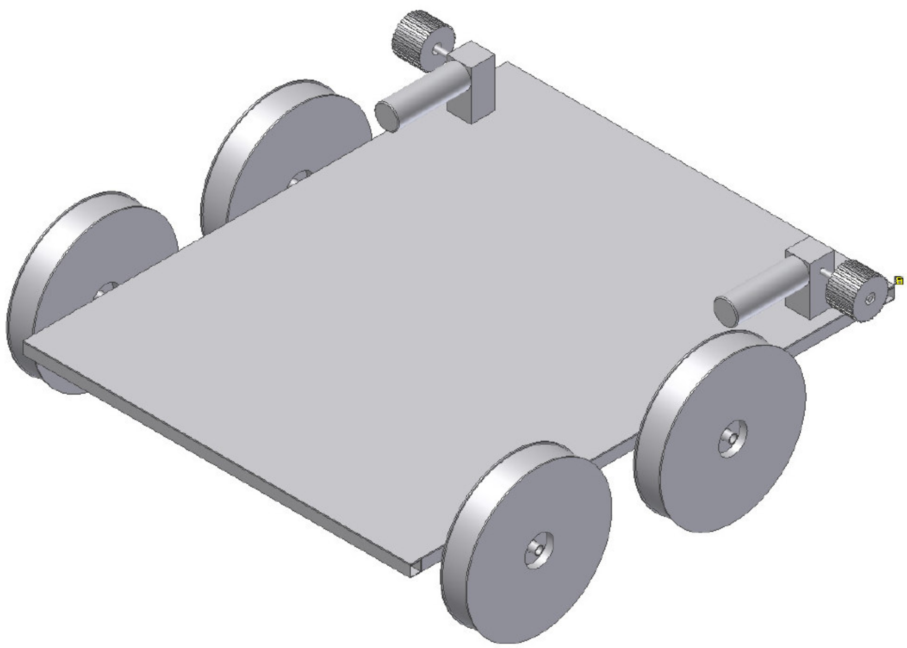


Figure 4.10 Mechanical part of vehicle is shown.

## 4.2 Manufacturing of Electronic Part

In this section, the electronic circuits of the designed vehicle will be discussed and then, the power, the control and the motor circuits will be explained.

### *4.2.1 Manufacturing of Power Circuit*

In this project, two electronic control circuits are used. First part is receiver circuit on the vehicle and the other one is transmitter circuit which is connected to computer. Furthermore, two different power circuits are designed.

Transmitter circuit needs 12 Volt DC source voltages. That suitable 220 Volt AC / 12 Volt DC adaptor circuit is prepared.

Receiver circuit is worked by 12 Volt DC. 12 Volt DC / 7 Ah accumulator was assembled on the vehicle. Sealed lead battery is shown in the figure 4.11.

Figure 4.11 12 volt DC / 7 Ah sealed lead battery.

Second accumulator is placed on the vehicle to drive motors. Sealed lead battery which is 12 Volt DC / 65 Ah, is shown in the following figure 4.12.


Figure 4.12 12 volt DC / 65 Ah sealed lead battery.

### *4.2.2 Manufacturing of Microcontroller Circuit*

In the circuit, PIC16F877 microcontroller is used. The circuit's minimum connection scheme which needs to be provided to work PIC16F877 is shown in the figure 4.13.

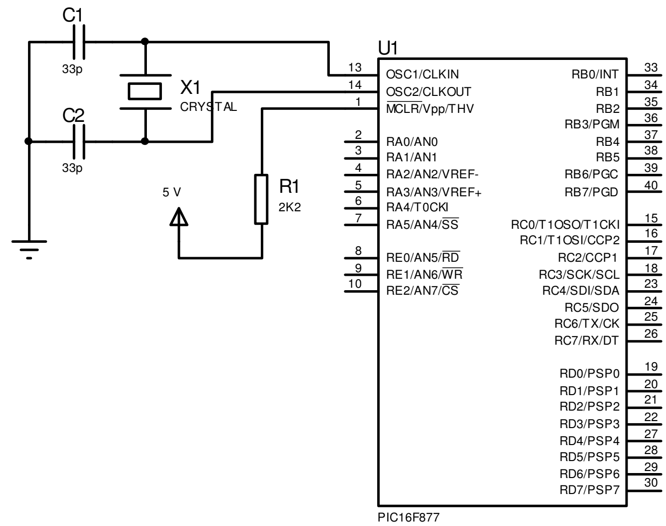



Figure 4.13 Crystal and reset circuit scheme of PIC 16F877.

The microcontroller circuit connections to receiver and transmitter circuits are shown in the figure 4.14 and figure 4.15.

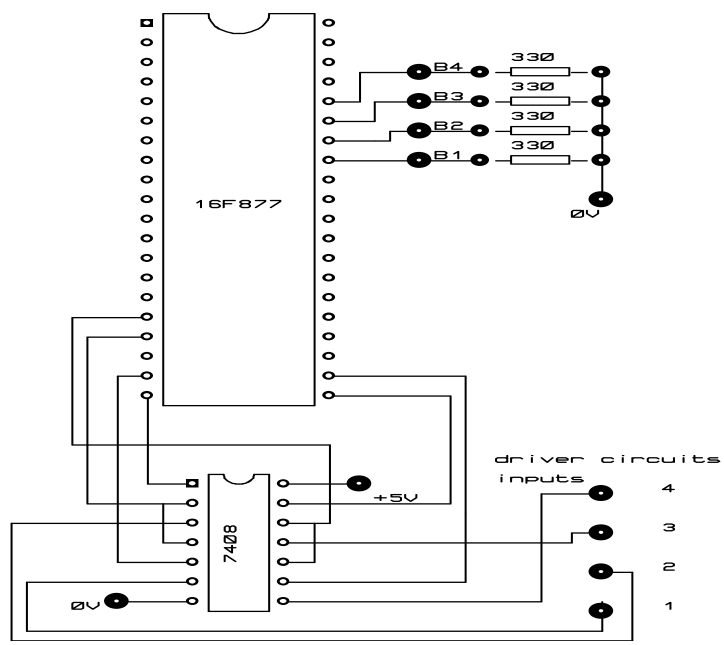



Figure 4.14 Receiver circuit interface.

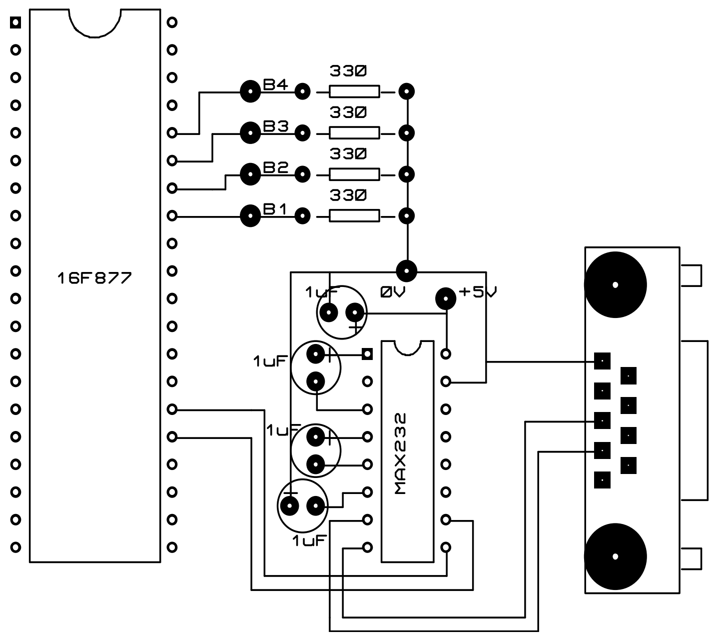



Figure 4.15 Transmitter circuit interface.

### *4.2.3 Assembling of Transceiver Module*

As given by the technical information in the design section, it will be seen that the transceiver module's source voltage is 3 Volt. The signal for selection and data pins is worked between 0 and 3 Volt. Taking the voltage levels into consideration, a interface circuit is designed. Because PIC16F877 needs 5 V but the transceiver module runs with 3 V, some equipment should be used for convenience. Suitable circuit schemes are shown in the figure 4.16 and 4.17.

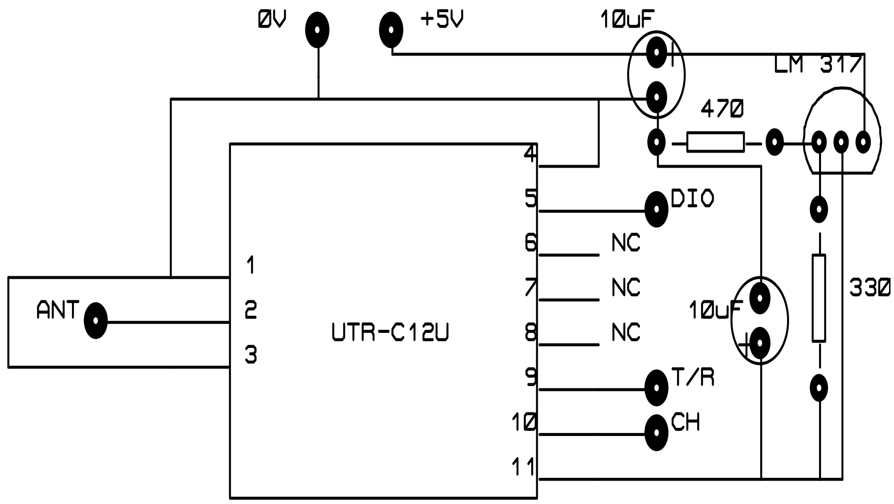



Figure 4.16 Power circuit of UTR-C12U connections.

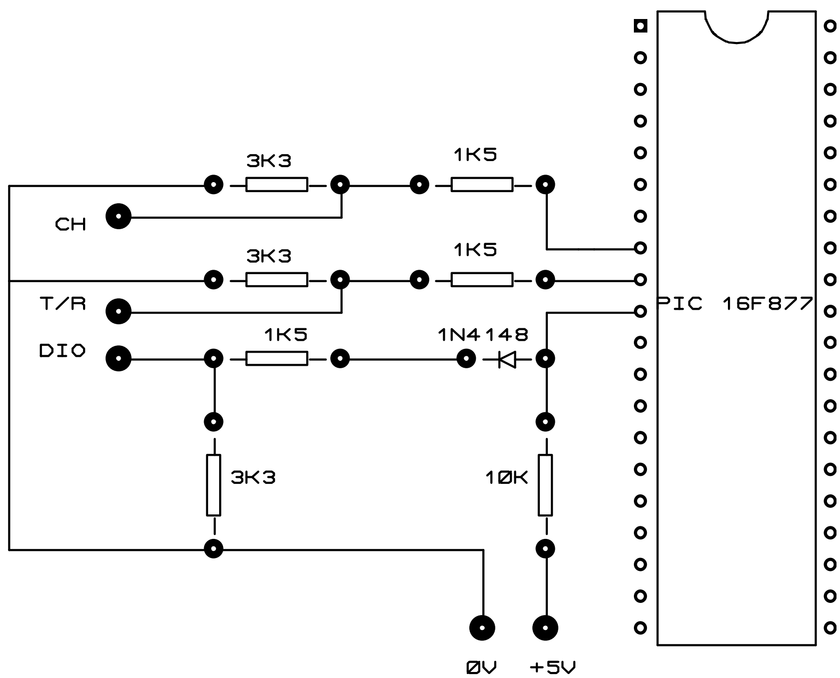



Figure 4.17 PIC16F877 and UTR-C12U connections.

### *4.2.4 Assembling of Motor's Power Circuit*

H-Bridge composed of four MOSFETs is used for the power circuit of the motors. Taking switching signal and the motor current into consideration, IRP064N MOSFET were selected and used. H Bridge's connection scheme is shown in the figure 4.18.

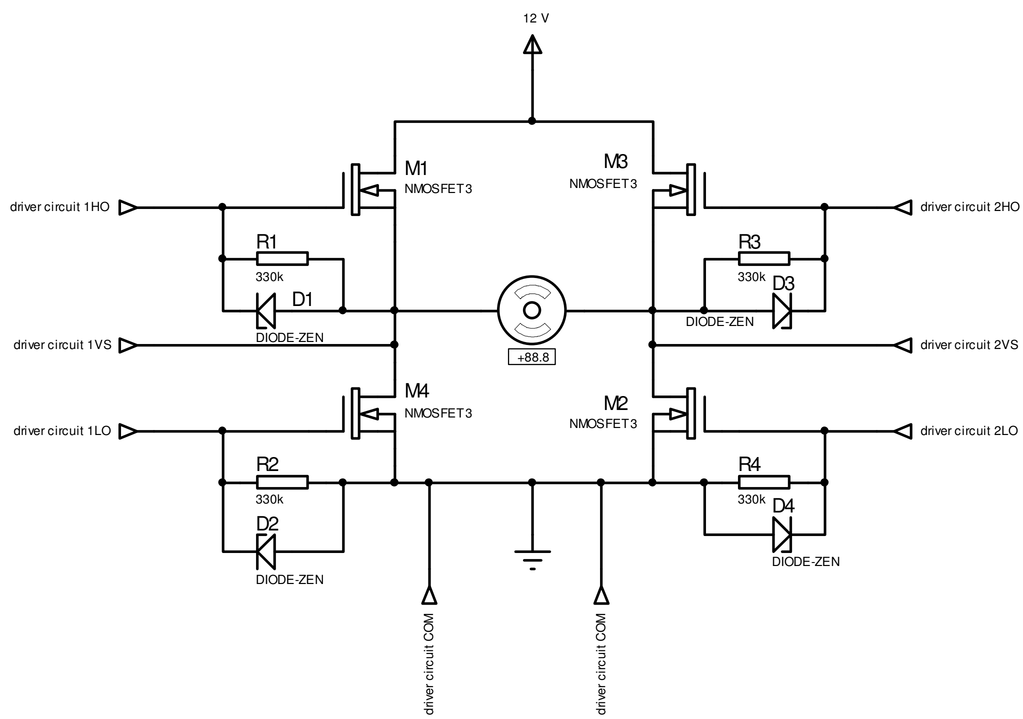



Figure 4.18 Motor's power circuit scheme.

### *4.2.5 Assembling of Driver Circuit*

Switching signal out of PIC16F877 on the receiver circuit is connected the driver circuit whose out is connected to motor power circuit. MOSFETs are triggered by this circuit. Circuit's connecting schemes are shown in the figure 4.19.

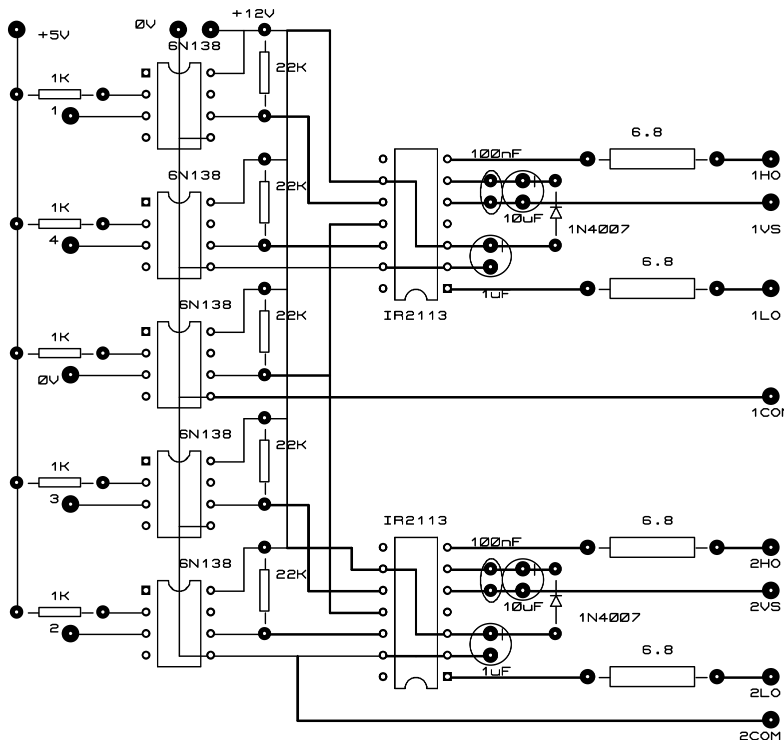
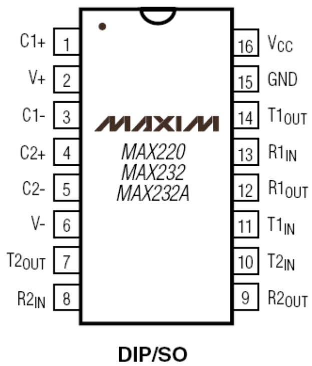



Figure 4.19 Motor's driver circuit scheme.

### *4.2.6 Assembling of Computer Connecting Circuit*

Transmitter circuit is communicated with computer's serial port. However, computer's serial port and PIC16F877's input voltages are different. So, an interface circuit is designed at between computer's serial port and PIC16F877 using a MAX232 integrated circuit. Its standard connection is shown in the following figure.




| CAPACITANCE (μF) | | | | | |
|---|---|---|---|---|---|
| DEVICE | C1 | C2 | C3 | C4 | C5 |
| MAX220 | 4.7 | 4.7 | 10 | 10 | 4.7 |
| MAX232 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| MAX232A | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |

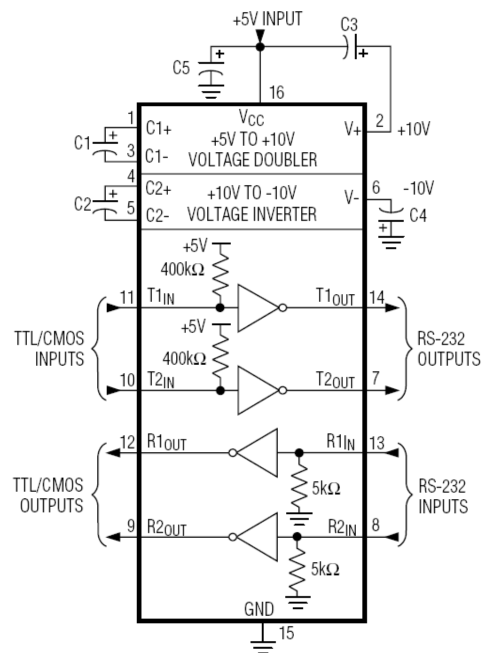



Figure 4.20 Minumum connection scheme of MAX232.

### *4.2.7 Connecting All Circuits*

Connections of the circuits designed will be shown in this chapter. Transmitter and receiver circuit's connection are shown in the figure 4.21 and 4.22.

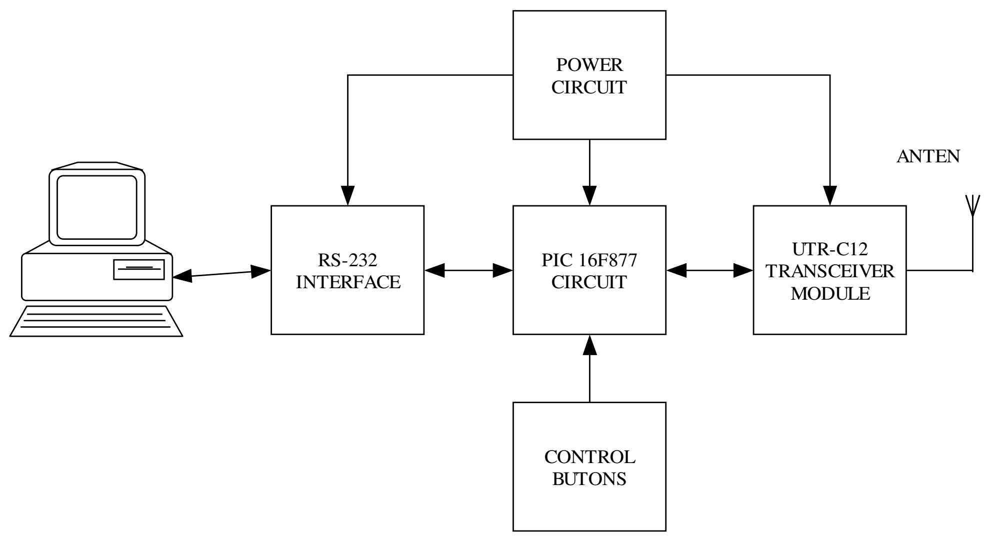



Figure 4.21 Block scheme of transmitter circuit.

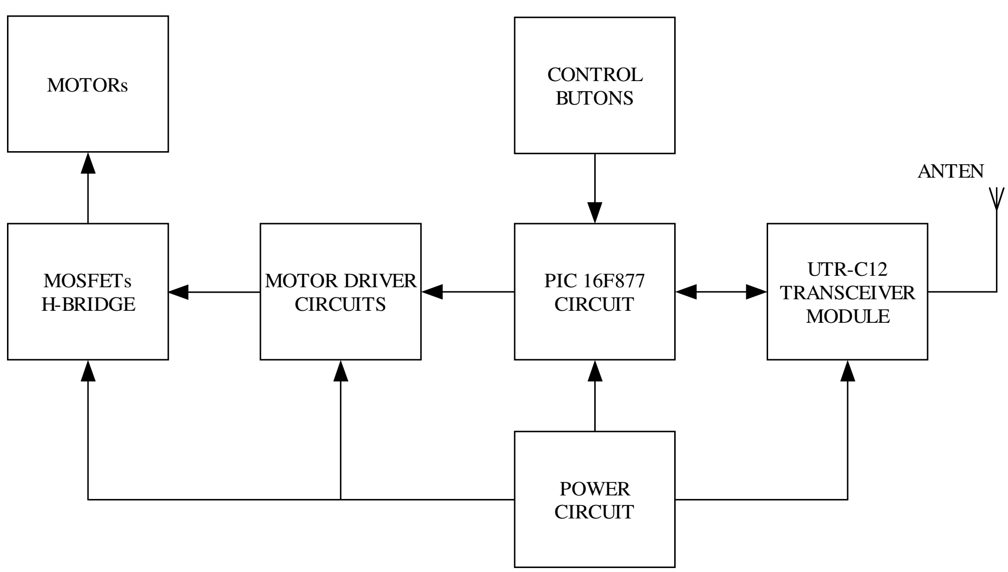



Figure 4.22 Block scheme of receiver circuit.

Circuits are assembled according to this connection block diagrams by using suitable equipments. Transmitter and receiver circuit’s connection schemes are shown in the figure 4.23 and 4.24.

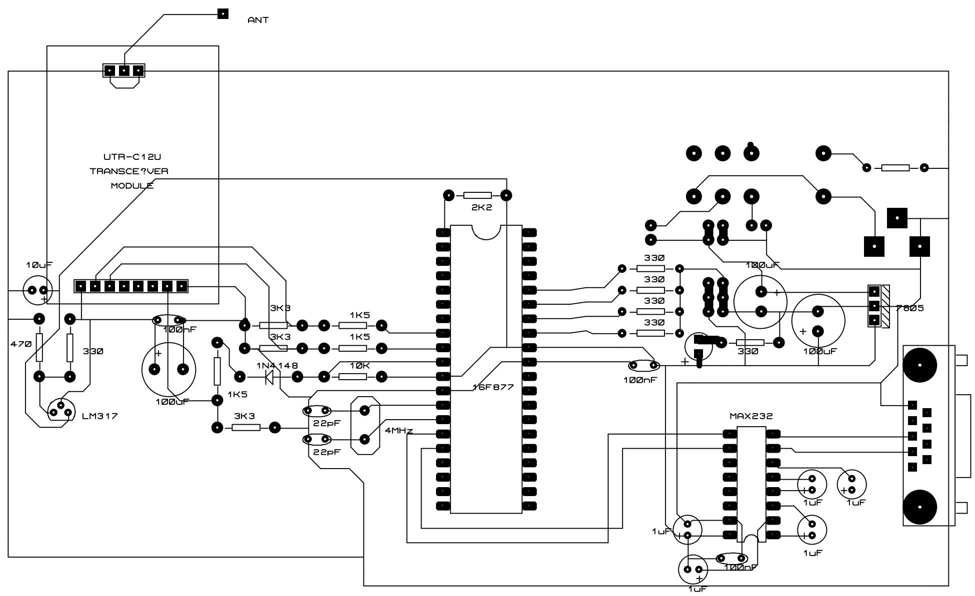



Figure 4.23 Transmitter circuit connection diagram.

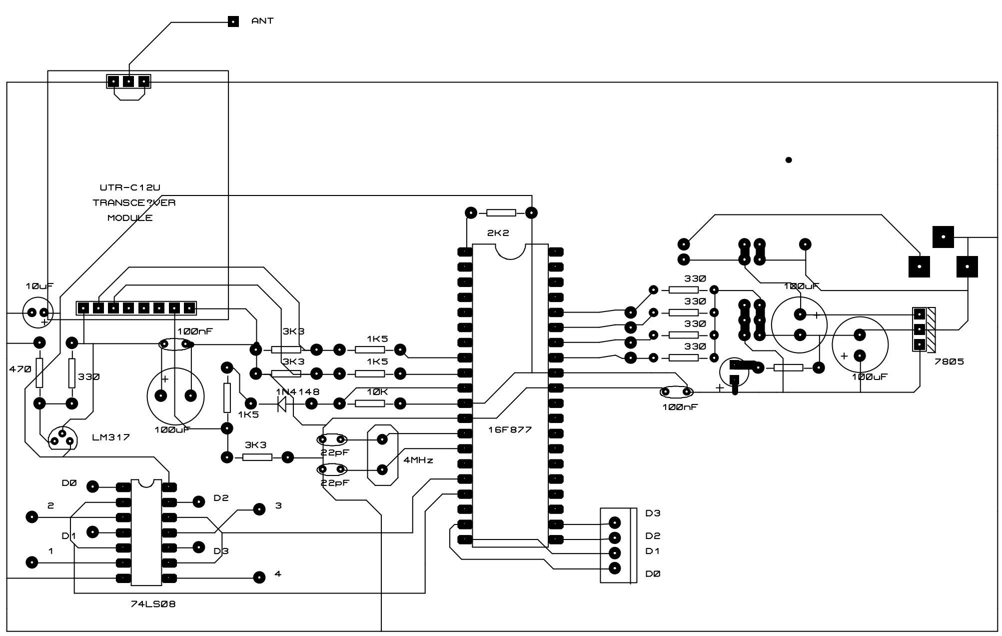



Figure 4.24 Receiver circuit connection diagram.

Printed circuit board shown in the figure 4.25 are used for both receiver and transmitter circuits.

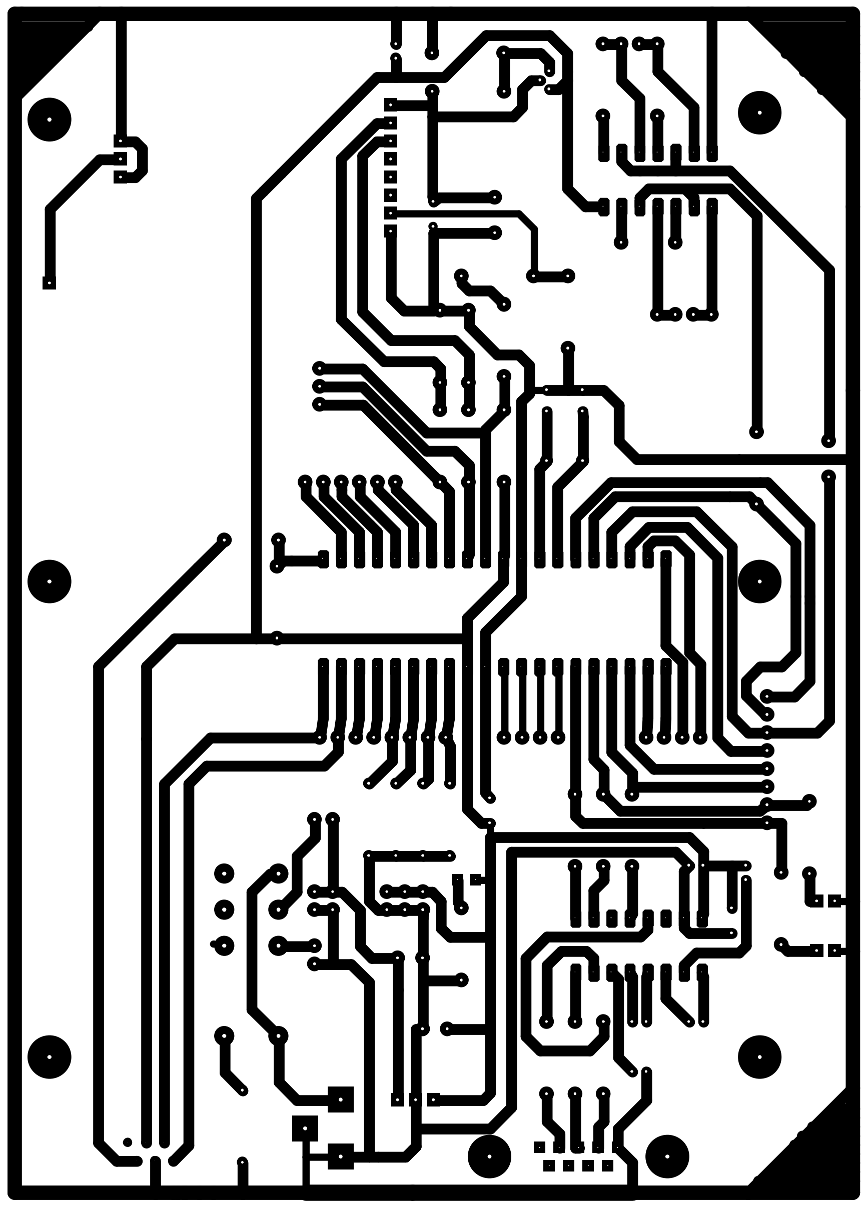


Figure 4.25 Printed circuit board of transmitter and receiver circuit.

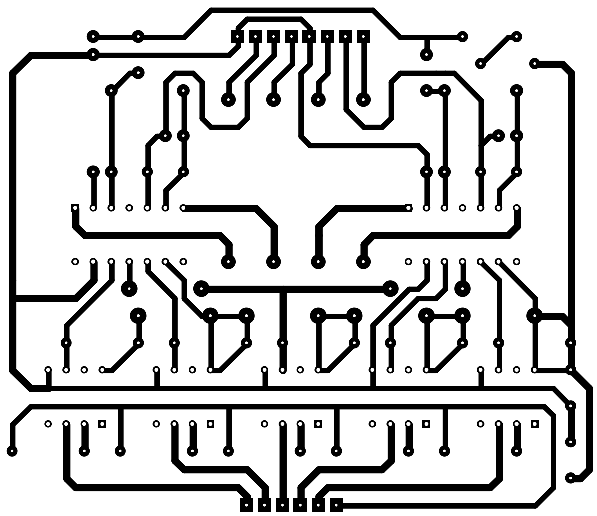


Figure 4.26 Printed circuit board of driver circuit.

Designed circuits are placed in a box. The boxes containing the circuits are shown in the figure 4.27.

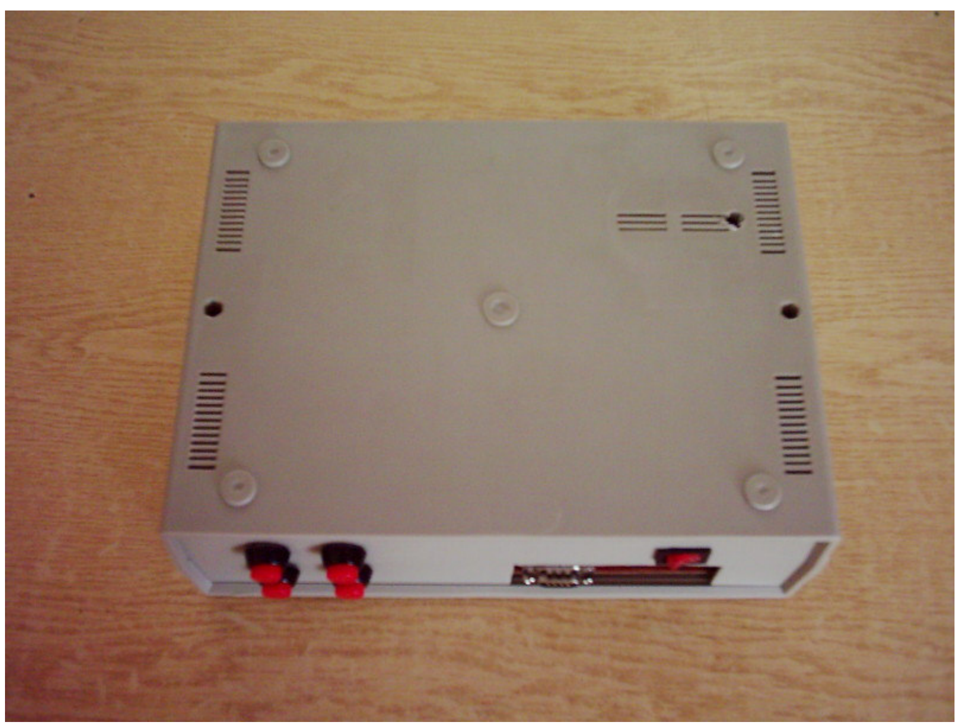


Figure 4.27 Transmitter circuit is shown in the box.

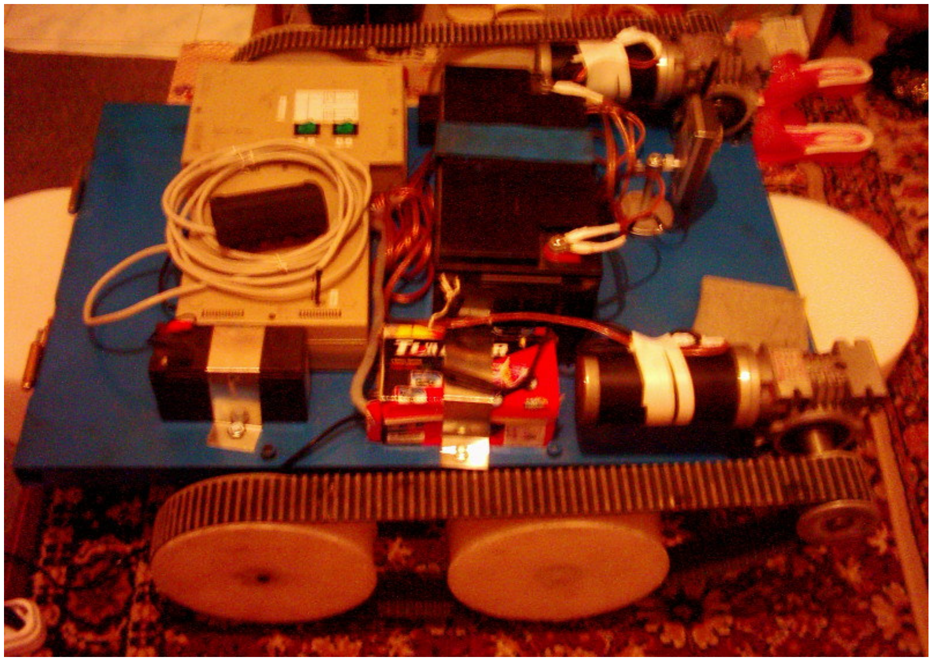


Figure 4.28 Picture of vehicle.

## 4.3 Writing of the Computer Software

Program window was generated by using C++. There are buttons on the window which direct the vehicle's route. The buttons here can be controlled by keyboards' route buttons. Besides, they can be controlled by using mouse. Program window is shown in the following figure.

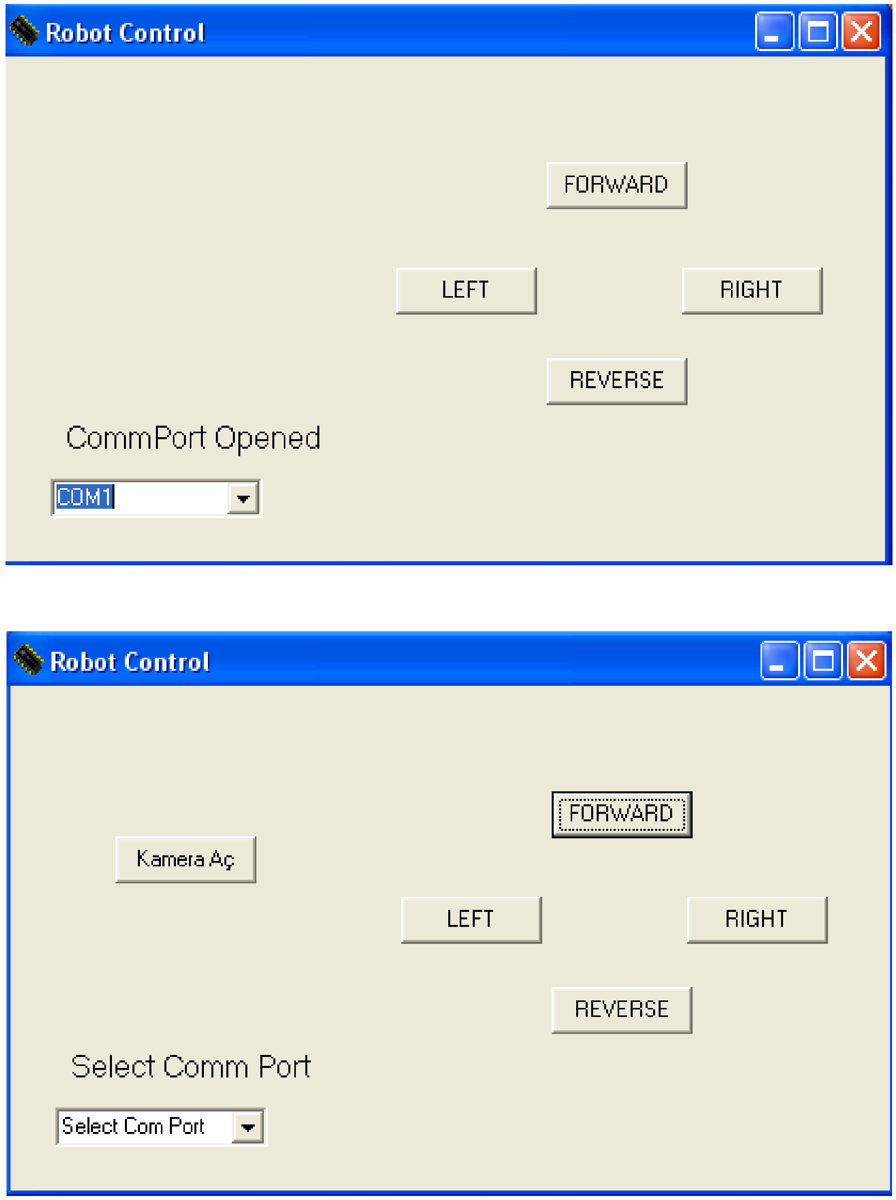



Figure 4.29 Control window of computer program.

# CHAPTER FIVE

# RUNNING OF THE VEHICLE

In this chapter, the vehicle design works will be described. Receiver circuit on the vehicle drives the motors using PWM signals. There are three different ways to control the vehicle remotely. In this chapter, kinds of remote control available are and how they work will be described.

## 5.1 Supplying Motion of Vehicle

When suitable movement signal came to the receiver circuit, how the circuit works will be described. When receiver program in PIC16F877 gets movement signal, it goes to sub-program. There are four sub-programs which are available for forward, reverse, right and left directions.  At these sub-programs, PWM signals are worked at desired frequency and amplitude. At the same time, Port D pins are actived according to movement route. There are two PWM signals for two motors. There are four Port D pins for two motors' four different route. Output signals of PIC16F877 integrated circuit are listed in the following table.

Table 5.1 Switch signal at the output of PIC16F877 integrated circuit.

| PORTD pins | | | | PWM pins | | DIRECTION |
|---|---|---|---|---|---|---|
| D0 | D1 | D2 | D3 | PWM1 | PWM2 | |
| HIGH | LOW | LOW | HIGH | HIGH | HIGH | FORWARD |
| LOW | HIGH | HIGH | LOW | HIGH | HIGH | REVERSE |
| LOW | HIGH | LOW | HIGH | HIGH | HIGH | RIGHT |
| HIGH | LOW | HIGH | LOW | HIGH | HIGH | LEFT |

These six signals are encoded by a logical AND gate similar to the receiver circuit. Four signals at output of integrated circuit are connected to control circuit.

Table 5.2 Input of driver circuit.

| DRIVER CIRCUIT INPUTS | | | | | DIRECTION |
|---|---|---|---|---|---|
| SD | HO1 | LO1 | HO2 | LO2 | |
| LOW | HIGH | LOW | LOW | HIGH | FORWARD |
| LOW | LOW | HIGH | HIGH | LOW | REVERSE |
| HIGH | UNKNOWN | UNKNOWN | UNKNOWN | UNKNOWN | STOP |

These signals are sent to driver circuit and used by driver circuit. IR2113 integrated circuits switch MOSFETs. There are two driver circuits and two power circuits for motors. Switching H-Bridge controls current of motors. MOSFET's switching signals for direction of motors are listed in the table 5.3.

Table 5.3 MOSFET's switching signals.

| MOSFET NO | | | | DIRECTION |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | |
| HIGH | HIGH | LOW | LOW | FORWARD |
| LOW | LOW | HIGH | HIGH | REVERSE |

## 5.2 The Vehicle's Remote Control Ways

The vehicle is controlled with three different ways. At first way, it can be controlled by buttons connected with cable to receiver circuit. These buttons is connected directly to PORTB pins of PIC16F877 integrated circuit. PORT B pins are normally at 0 Volt. When one of the buttons is activated, PORT B pins get 5 Volt. Movement is continued while that button is pushed. Movement routes are shown in the table 5.4 according to PORT B pins of PIC16F877.

Table 5.4 PIC 16F877 PORTB pins according to movement route.

| PORTB pins | | | | DIRECTION |
|---|---|---|---|---|
| B0 | B1 | B2 | B3 | |
| LOW | LOW | LOW | LOW | STOP |
| HIGH | LOW | LOW | LOW | FORWARD |
| LOW | HIGH | LOW | LOW | REVERSE |
| LOW | LOW | HIGH | LOW | RIGHT |
| LOW | LOW | LOW | HIGH | LEFT |

The second way, movement can be controlled by buttons connected to transmitter circuit. The same connection as that table 5.1 is applied to the transmitter circuit. Transmitter circuit perceives movement route by controlling these buttons. Wireless communication is provided by UTR-C12U transceiver module between the receiver and the transmitter circuits. PORT E pins are use in order to established the connection between transceiver module and PIC16F877 integrated circuit. PORT E pins are shown in following table 5.5.

Table 5.5 Port E of the PIC 16F877 and UTR-C12U pins.

| PORTE pins | | | UTR-C12U | | |
|---|---|---|---|---|---|
| E0 = CH | E1 = T/R | E2 = DIO | FUNCTION | | FREQUENCY |
| HIGH | HIGH | SIGNAL | TXon | RXoff | FREQUENCY1 |
| LOW | HIGH | SIGNAL | TXon | RXoff | FREQUENCY2 |
| HIGH | LOW | SIGNAL | TXoff | RXon | FREQUENCY1 |
| LOW | LOW | SIGNAL | TXoff | RXon | FREQUENCY2 |

Communication protocol is generated to protect from peripheral noise. When transmitter circuit has a movement signal, it sends immediately a data to receiver circuit. After that, it sends data appropriate for desired movement. Receiver circuit controls the first coming data. If that data is true, receiver circuit controls the second coming movement data. Communication protocol and movement signals are listed in the following table 5.6. Transmitter circuit works only providing supply voltage in this mode. In this manner, the vehicle’s control is done remotely.

Table 5.6 Communication protocol and movement signal.

| PROTOKOL SIGNALS (8 bit binary) | | SIGNAL | DIRECTION |
|---|---|---|---|
| 0 | 00000000 | | |
| FF | 11111111 | U (01010101) | FORWARD |
| AA | 10101010 | D (01000100) | REVERSE |
| V | 01100101 | R (00100101) | RIGHT |
| E | 01010100 | L (11000100) | LEFT |
| R | 00100101 | | |

The third way is that vehicle can be controlled remotely by computer. The same communication system as that in second way is used between receiver and transmitter circuits in third way too. Transmitter circuit is connected to computer serial port. Serial communication pins of the PIC16F877 integrated circuit are used. These pins are connected to computer serial port.

The software has written by C++ (Cplusplus). When program is activated, an interface is opened. Firstly, the com port which the transmitter circuit attached to is selected and then, the vehicle is remotely controll by using the buttons on the program window. These buttons can be clicked by mouse. Direction buttons on the keyboard can also be used for the same purpose. This program sends directly movement data via serial port to transmitter circuit from computer. Any protocol is not used in here. This coming data is utilized in transmitter circuit. The movement data is sent to receiver circuit by adding the communication protocol prior to it. Thus, the vehicle is controlled by computer.

# CHAPTER SIX

# CONCLUSION

The main purpose of this thesis is to produce a vehicle which can be controlled remotely through a computer. First of all, required specifications are determined which are 100 kg carrying capacity and 0.5 $m^2$ vehicle body. Examing many manufactured vehicles, the required knowledge about their mechanical constructions, motion mechanisms and control circuits are collected.

Afterwards, production of the desired vehicle is initiated. The metal body of the vehicle was prepared and motors, triger belts, gearboxes, wheels and accumulators are assembled on it. In this manner, the vehicle becomes ready to move.

Secondly, the electronic control circuits are designed and assembled. As control unit microcontrollers and for wireless communication RF transceiver modules are used in the control circuits. In order to control two motors, driver and power circuits are designed.

Thirdly, a control program was written. The vehicle can be controled both using the control program and the buttons on the transmitter circuit remotely. In addition to this, the vehicle is also controlled using the buttons connected to the vehicle via cable.

In the last step, a camera having wireless ethernet transmitter was assembled on the vehicle. In this manner, movement of the vehicle can be watched using the program provided with the camera or an internet browser through computer.

After having been finished, the vehicle was tested on different floors and against obstacles. It is possible to use the designed vehicle here in many applications by modifying since it is in a basic form. The vehicle designed here can be manufactured for commercial purposes.

**REFERENCES**

Brindley, K. (1997). *Newnes electronics engineer's pocket book* (3. Ed.). (G. Dinçer & A. Bodur, Çev.). İstanbul, 1998:Bileşim yayıncılık. (Butterworth Heinemann ltd., 1997).

Hubert, C.I.(2002). *Electric machines* (2. Ed.). Upper saddle river, New Jersey Columbus, Ohio:Pearson Education Inc.

International Rectifier, (2006),
www.irf.com/product-info/datasheets/data/ir2110.pdf
www.irf.com/product-info/datasheets/data/irfp064n.pdf

Microchip Coorperation, (2006),
http://www.microchip.com/stellent/idcplgidcplg?IdcService=SS_GET_PAGE&nodeId=1335&dDocName=en010241.

Schildt, H. & Guntle, G. (2002). *Borland C++ builder: the complete reference*. ( S. Tüzel, Çev.). İstanbul 2002: ALFA basım yayım dağıtım ltd. şti. (The McGraw-Hill Companies, 2002).

Schweber, W. (2002). *Electronic communication systems* (4. Ed.). Upper saddle river, New Jersey Columbus, Ohio:Pearson Education Inc.

Soulie J., (07/08/2006),
http://www.cplusplus.com/doc/tutorial/

UDEA, (2006),
http://www.udea.com.tr/Turkce/Urunler/RF_Moduller/RF_Transceivers.htm.

Young, P.H. (2004). *Electronic communication techniques* (5. Ed.). Upper saddle river, New Jersey Columbus, Ohio:Pearson Education Inc.

## APPENDIX: PIC C CODES

- receiver code:

```
#include "RX_TX.h"
#USE FAST_IO (E)
#USE FAST_IO (B)
#byte RCSTA=0x18
#byte RCREG=0x1A
#bit  OERR=RCSTA.1
#bit  FERR=RCSTA.2
#bit  CREN=RCSTA.4

#byte TRISE=0x89
#byte PORTE=0x09
static int a=0;
char timed_getc() {
long timeout;
#define  CH PIN_E0
//   timeout_error=FALSE;
   timeout=0;
   while(!kbhit(UTR)&&(++timeout<500)) // 1/2                              // second

         delay_us(10);

   if(kbhit(UTR))
         return(fgetc(UTR));
   else {
//         timeout_error=TRUE;
         return(0);
   }
}
MOTOR_ILERI()
{
setup_ccp1(CCP_PWM);
setup_ccp2(CCP_PWM);
output_high(PIN_D0);
output_low(PIN_D1);
output_low(PIN_D2);
output_high(PIN_D3);
}
MOTOR_GERI()
{
setup_ccp1(CCP_PWM);
setup_ccp2(CCP_PWM);
output_low(PIN_D0);
output_high(PIN_D1);
output_high(PIN_D2);
output_low(PIN_D3);
}
MOTOR_SAGA()
{
setup_ccp1(CCP_PWM);
setup_ccp2(CCP_PWM);
output_low(PIN_D0);
output_high(PIN_D1);
output_low(PIN_D2);
output_high(PIN_D3);
```

```
}
MOTOR_SOLA()
{
setup_ccp2(CCP_PWM);
setup_ccp1(CCP_PWM);
output_high(PIN_D0);
output_low(PIN_D1);
output_high(PIN_D2);
output_low(PIN_D3);
}
MOTOR_STOP()
{
setup_ccp1(CCP_OFF);
setup_ccp2(CCP_OFF);
output_low(PIN_D0);
output_low(PIN_D1);
output_low(PIN_D2);
output_low(PIN_D3);
}
MOTOR_HIZI(char gelen)
{
static char   hiz[3];
  hiz[a]=gelen;
  a++;
  if(a==3)
  {
  a=0;
  if(hiz[0]=='K')
  {

    set_pwm1_duty(hiz[1]);
    set_pwm2_duty(hiz[2]);
  }
  }
}
void  setup_cpu()
{
  setup_adc_ports(NO_ANALOGS);
  setup_adc(ADC_OFF);
  setup_psp(PSP_DISABLED);
  setup_spi(FALSE);
  setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
  setup_timer_1(T1_DISABLED);
  setup_timer_2(T2_DIV_BY_16,255,1);
  set_tris_a(0xFF);
  set_tris_b(0xFF);
  set_tris_c(0b10111001);
  set_tris_d(0xFF);
  set_tris_e(0b00000100);
  output_low(PIN_E1);
  set_pwm1_duty(230);
  set_pwm2_duty(230);
  setup_ccp1(CCP_OFF);
  setup_ccp2(CCP_OFF);
 //  port_b_pullups (TRUE);
}
void main()
{
```

```
char  c;
int8  Sayac=0;
  setup_cpu();
  output_high(CH);
  output_low(PIN_E1);
  MOTOR_STOP();
  while(TRUE)
  {
    if(input(PIN_B3))
      {
        MOTOR_SOLA();
        Sayac=0;
      }
    else
    {
      if(input(PIN_B1))
      {
        MOTOR_GERI();
        Sayac=0;
      }else
      {
        if(input(PIN_B2))
        {
          MOTOR_SAGA();
          Sayac=0;
        }else
        {
          if(input(PIN_B0))
          {
          MOTOR_ILERI();
          Sayac=0;
          }
        }
      }
    }
    if (OERR || FERR) {CREN=0;OERR=0;CREN=1;c=RCREG;}
    if(kbhit(PC))
    {
      c=fgetc(PC);
      fputc(c,PC);
      fputc(c,UTR);
      output_float(PIN_E2);
    }
    c=timed_getc();
    if(c=='V')
    {
      c=timed_getc();
      if(c=='E')
      {
        c=timed_getc();
        if(c=='R')
        {
          c=timed_getc();
          switch(c)
          {
            case 'U': MOTOR_ILERI();
            break;
            case 'D': MOTOR_GERI();
```

```
            break;
            case 'R': MOTOR_SAGA();
            break;
            case 'L': MOTOR_SOLA();
            break;
            default : MOTOR_HIZI(c);
            break;
           }
           Sayac=0;
           fputc(c,PC);
          }
        }
        }else
        {
          Sayac++;
          if(Sayac > 40)
          {
            MOTOR_STOP();
            Sayac=0;
          }
        }
  }
}
```

- transmitter code:

```
#include "RX_TX.h"
#USE FAST_IO (E)

#byte RCSTA=0x18
#byte RCREG=0x1A
#bit  OERR=RCSTA.1
#bit  FERR=RCSTA.2
#bit  CREN=RCSTA.4

#byte TRISE=0x89
#byte PORTE=0x09
char timed_getc() {
long timeout;
#define  CH PIN_E0
//   timeout_error=FALSE;
   timeout=0;
   while(!kbhit(UTR)&&(++timeout<500)) // 1/2                              // second

         delay_us(10);

   if(kbhit(UTR))
         return(fgetc(UTR));
   else {
//          timeout_error=TRUE;
         return(0);
   }
}
void  Gonder(char veri)
{
        fputc(0xFF,UTR);
        fputc(0x00,UTR);
        fputc(0xAA,UTR);
```

```
        fputc('V',UTR);
        fputc('E',UTR);
        fputc('R',UTR);
        fputc(veri,UTR);
        output_float(PIN_E2);
}
void  setup_cpu()
{
   setup_adc_ports(NO_ANALOGS);
   setup_adc(ADC_OFF);
   setup_psp(PSP_DISABLED);
   setup_spi(FALSE);
   setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
   setup_timer_1(T1_DISABLED);
   setup_timer_2(T2_DISABLED,0,1);
   set_tris_a(0xFF);
   set_tris_b(0xFF);
   set_tris_c(0b10111111);
   set_tris_d(0xFF);
   set_tris_e(0b00000100);
   output_high(PIN_E1);

}
void main()
{
char  c;
   setup_cpu();
   output_high(CH);
   output_high(PIN_E1);
   while(TRUE)
   {
     if(input(PIN_B3))
       {
         Gonder('L');
       }
     else
     {
       if(input(PIN_B1))
       {
         Gonder('D');
       }else
       {
         if(input(PIN_B2))
         {
           Gonder('R');
         }else
         {
           if(input(PIN_B0))
           {
             Gonder('U');
           }
         }
       }
     }
      if (OERR || FERR) {CREN=0;OERR=0;CREN=1;c=RCREG;}
     if(kbhit(PC))
     {
       c=fgetc(PC);
```

```
        fputc(c,PC);
        Gonder(c);
      }
      if(c=timed_getc())
      {
        fputc(c,PC);
      }

    }
  }
```