A project submitted to the Department Of Electrical And Information Engineering in partial fulfilment of the requirements of BSc. Electrical and Electronics Engineering of the University of Nairobi.

Date of Submission:

DECLARATION OF ORIGINALITY

FACULTY/SCHOOL/INSTITUTE: Engineering

DEPARTMENT: Electrical and Information Engineering

COURSE NAME: Bachelor of Science in Electrical and Electronics Engineering

NAME OF STUDENT: KIMEMIA ELVIS WANJAGI

REGISTRATION NUMBER: F17/357962/2010

COLLEGE: Architecture and Engineering

WORK: Wireless Control of DC Motor

1. I understand what plagiarism is and I am aware of the university policy in this regard.

2. I declare that this final year project report is my original work and has not been submitted elsewhere for examination, award of a degree or publication. Where other people's work or my own work has been used, this has properly been acknowledged and referenced in accordance with the University of Nairobi’s requirements.

3. I have not sought or used the services of any professional agencies to produce this work.

4. I have not allowed, and shall not allow anyone to copy my work with the intention of passing it off as his/her own work.

5. I understand that any false claim in respect of this work shall result in disciplinary action, in accordance with University anti-plagiarism policy.

Signature: ..............................................
Date: ........................................................
DEDICATION
This project is dedicated to my parents and family for the moral, financial and technical support and also to those who have guided me throughout my journey of education.
ACKNOWLEDGEMENTS
First and foremost, I thank the almighty God for guiding me throughout my studies till the accomplishment of this project.

To God be the Glory.

I am also grateful to my supervisor, DR. G.N KAMUCHA for his useful guidance and suggestions throughout the project, it has been a great pleasure for me to get an opportunity to work under him.

Special thanks to my dad, Mr. WANJAGI MICHAEL KIMEMIA, for his moral, financial and technical support throughout the project, which has contributed greatly to the provision of knowledge as well as the completion of this project.

I also would like to express deep gratitude to my friends Wamai RuthWangu, Weru Maxwell Mburu, Mutahi Germano Kiruthu, and Oriare Moses for helping me in understanding the project.
# Table of Contents

DECLARATION OF ORIGINALITY ........................................................................... 2

DEDICATION ........................................................................................................... 3

ACKNOWLEDGEMENTS ......................................................................................... 4

Acronyms ............................................................................................................... 8

ABSTRACT .............................................................................................................. 9

CHAPTER 1: INTRODUCTION .................................................................................. 10

1.1 Project Definition ............................................................................................ 10

1.2 Project Overview ............................................................................................. 10

1.3 Project Objectives ........................................................................................... 11

1.3.1 Overall Objective ....................................................................................... 11

1.3.2 Specific Objectives .................................................................................... 11

1.4 Project Justification ......................................................................................... 11

Advantages ............................................................................................................ 12

1.5 Project Applications ......................................................................................... 12

1.6 Project Scope .................................................................................................. 12

1.7 Project Organization ....................................................................................... 13

Chapter 1: Introduction ......................................................................................... 13

Chapter 2: literature review .................................................................................. 13

Chapter 3: Design and Implementation ................................................................. 13

Chapter 4: Results, Analysis and Discussion ....................................................... 13

Chapter 5: Conclusion and Recommendations ................................................... 13

CHAPTER 2: LITERATURE REVIEW ....................................................................... 14

2.1 classification of control units .......................................................................... 14

2.1.1 Manual control units (systems) ................................................................. 14

2.1.2 Wireless control units (systems) ............................................................... 14

2.2 Block diagram description ............................................................................. 16

2.3 Hardware and software components .............................................................. 18

2.3.1 Microcontroller ........................................................................................ 18

2.2 RF module (RX-TX MODULES (434MHz)) .................................................. 25

2.2.1 Description ................................................................................................ 26

2.2.2 HT12D DECODER ................................................................................... 30

2.2.2.1 Pin Diagram ......................................................................................... 30

2.2.3 HT12E ENCODER ................................................................................... 31
2.2.3.1 Pin Diagram........................................................................................................31
2.2.3.2 Pin description ....................................................................................................31
2.3 PWM .........................................................................................................................32
  2.3.1 Frequency: ............................................................................................................33
  2.2.2 Duty Cycle: ............................................................................................................33
2.4 H BRIDGE ..................................................................................................................34
  L293D IC .........................................................................................................................38
  L293D connections ..........................................................................................................38
Figure 1 block diagram for atmega32 ................................................................. 20
Figure 2 atmega 32 circuit diagram ........................................................................ 22
Figure 3 10-pin cable pin ...................................................................................... 23
Figure 4 USBasp .................................................................................................... 24
Figure 5 RF module receiver and transmitter ........................................................ 25
Figure 6 Receiver module ...................................................................................... 26
Figure 7 Transmitter module .................................................................................. 26
Figure 8 RX/TX ...................................................................................................... 26
Figure 9 black diagram for RF module ................................................................... 27
Figure 10 block diagram for remote module .......................................................... 27
Figure 11 block diagram for receiver ..................................................................... 28
Figure 12 RF Module circuit diagram .................................................................... 29
Figure 13 .................................................................................................................. 30
Figure 14 .................................................................................................................. 31
Figure 15 .................................................................................................................. 33
Figure 16 .................................................................................................................. 34
Figure 17 h bridge ................................................................................................. 35
Figure 18 voltage control ...................................................................................... 35
Figure 19 circuit diagram or h-bridge .................................................................... 37
Figure 20 L293D Connections ............................................................................ 38
Acronyms

ALU - Arithmetic Logic Unit
CPU - Central Processing Unit
ROM - Read Only Memory
RAM - Random Access Memory
MOS - Metal Oxide
PCB - Printed Circuit Board
RF - Radio Frequency
PWM - Pulse Width Modulation
MCU - Microcontroller
ABSTRACT
The aim of this project is to design an effective, efficient and low cost microcontroller based control unit that will be used to wirelessly control a DC motor using RF (radio frequencies) at distances ranging from 1-100 metres. The user should be able to wirelessly control the DC Motor.

To achieve this, the digital circuit will be interfaced to a microcontroller, Atmega 32. An electronic technique called Pulse Width Modulation (PWM) is used to generate High and Low pulses (duty cycles) that vary and thus control the speed of the motor. The generation of this pulses is made possible by using a microcontroller (Atmega 32) which in turn sets the speed ranges as per each cycle. H-Bridge was also used to achieve direction control (clockwise and counterclockwise direction) and an RF module, a small electronic circuit used to transmit, receive radio wave where transmitter transmits the signal while receiver receives the signals that have same range of frequencies.

This project is practical in the economic view and hence gives a reliable, durable, accurate and most efficient way of a DC motor control.
CHAPTER 1: INTRODUCTION

1.1 Project Definition
The wireless control of a DC Motor involves the design and implementation of a microcontroller based control unit to use RF (radio frequency) to wirelessly control a DC Motor. DC motors have played a vital role in the development of industrial power transmission systems. It was the first practical device to convert electrical power into mechanical power.

Inherently straightforward operating characteristics, flexible performance and high efficiency encouraged the widespread use of DC motors in many types of industrial drive applications. With the advancement in the field of wireless communication technology has thus encouraged their use in other fields such as military drones, surveillance systems, toy cars among others.

1.2 Project Overview
A device that converts electrical energy into mechanical energy is a motor. The DC motor therefore utilizes a DC supply to produce a mechanical output. And some of the advantages of a DC motor over conventionally AC motor are;

- DC motors have a high efficiency.
- DC motors have speed torque characteristics that can be varied to almost any useful form.
- DC motors have higher controller efficiency.
- DC motors have a better overload and peak voltage characteristics.

And thus as a result of the above advantages they have a wide range of applications, in places where constant speed is to be maintained at varying loads, e.g. conveyor belts, cranes, mixers, elevators are few applications where DC motors are used.

A modern trend in the field of automation is to use wireless supervision and feedback processes. This fact became the reason behind the decision to design and build for my undergraduate project, Wireless control of DC motor using RF.

Compared to other wireless communication protocols such as Wi-Fi (IEEE 802.11), GPRS GSM networks, FM modules and Bluetooth, RF is cheap and reliable network with data delivery guaranteed, and also able to communicate over large distances up to 100m.

The first step in the designing process was to be able to switch direction of motor rotation (clockwise and anti-clockwise rotation) and vary speed of motor (clockwise and counterclockwise). And in order to achieve the desirable effects we used an H-bridge configuration which reverses the voltage on motor leads. Pulse Width Modulation was employed so as to drive the DC motor at the desirable speed, which was measured by observing, in set periods of time, the pulsation produced by the pulse generator of motor.
The above was achieved by utilizing open software of which digital inputs were used during pulse measurements, digital outputs were used for current alteration. A C/C++ code was produced that was to enumerate the pulses and to transform the measurement into revolutions per minute. Also to make the system user friendly a Human Machine Interface program was made so it would be possible to supervise the motor’s speed performance, as well as create a control unit to change those parameters.

Finally a PCB board was developed and aiming to integrate the H-bridge, pulse generator, RF module and input/output of microcontroller.

1.3 Project Objectives
1.3.1 Overall Objective
To design and implement a microcontroller based control unit to wirelessly control a DC Motor using RF.

1.3.2 Specific Objectives

- To design and implement a microcontroller based control unit using receiver/transmitters, encoder/decoder, Atmega 32 (MCU) among other system components.
- To design and implement an H-Bridge to control the clockwise and counterclockwise direction and the speed of DC Motor.
- To develop a software control program/code using C/C++ language of the microcontroller in Atmel Studio 6.0.
- Finally to design and implement the complete control unit for the DC Motor.
- To vary the speed and change direction of the DC motor wirelessly using RF (radio frequency) wave.

1.4 Project Justification
The successful design and implementation of the Wireless DC Motor control will enable the wireless supervision of robots and machines that utilize DC Motors. It develops a combination of mechanical engineering, electronics, programming, controls, and motors, while also providing us with a chance of hands on experience with design and development. It can be justified in the many practical applications and broad scopes of engineering in details. As discussed earlier our main focus of justification is its entertainment, security and military value. Entertainment wise it will be is an extremely durable and fun project. It’s easy to use and appeals to all age groups making it very marketable. Military wise, it could change everything about our armies. Instead of shipping human lives into dangerous and life threatening situations we could send a machine to keep the peace. It’s safer, risk free and will save lives, something no one can put a price on. As for security we could remotely control CCTV cameras.
Advantages

- Speed and direction control from a remote place
- The WDCM is easy to operate.
- The system has high sensitivity and not much sensitive to the environmental changes
- The system is reliable and inexpensive. The control unit can control a DC Motor over a long distance, also the hardware and software components required in the system implementation are locally and readily available.
- No line of sight is required and not sensitive to light.

1.5 Project Applications

DC Motor possess excellent torque speed characteristics and offer a wide range of speed control, and due to this the demand for DC motors will be undiminished.

- Commercial wireless applications such as door announcers, gate control, remote activation.
- Consumer products including electronic toys, home security, gate and garage door openers, intercom, fire and safety systems and irrigation controllers
- Bottle filling systems, conveyer application.
- Automotive companies employing RF for wireless remote control, remote keyless entry and safety applications.
- Department of defense as will be applied in controlling their drones.
- For remotely controlling wireless CCTV cameras.

1.6 Project Scope

This project covers DC Motor control hardware and software design and implementation. The software system entailed developing a program for microcontroller using Atmel Studio 6.0 platform and Proteus simulation software which simulates real time circuit. The hardware system involved the design and construction of a properly working microcontroller based control unit.
1.7 Project Organization
This project entails five chapters as mentioned below:

Chapter1: Introduction
This gives the full description of the project, justification, objectives, scope of work, methodology and project report organization.

Chapter2: literature review
This chapter gives a detailed description of hardware and software components required in the design and implementation.

Chapter3: Design and Implementation
This gives a compressive description of the technical aspects of the wireless DC motor control, its design and implementation; project description, system program code, system flow chart, hardware and software design.

Chapter4: Results, Analysis and Discussion
This chapter gives the results, data analysis and discussion of system performance and challenges encountered.

Chapter5: Conclusion and Recommendations
This is the final chapter and covers the conclusion and recommendations after completion of the final year project. It covers assessment of whether the objectives and scope were achieved and highlights area for future development, bibliography and appendices.
CHAPTER 2: LITERATURE REVIEW

2.1 classification of control units

Electronic control units can be classified into two main categories:

- Wireless control unit (system)
- Manual control units (systems)

2.1.1 Manual control units (systems)

In this systems the control of our systems can be obtained in the following ways:

- Engaging personnel to manually control the system at its location.
- Having to manually operate the control unit, physically.

And thus it comes at a great expense such as time consuming and expensive since more personnel will be involved.

2.1.2 Wireless control units (systems)

This system use the transfer of information between two or more points which are not directly connected through an electrical conductor. Controlling the different parameters of a DC motor such as direction (clockwise and counterclockwise) and speed control, is made possible at a distance. Such as a few meters away. This technologies include:

1. Radio

It’s the radiation (wireless transmission) of electromagnetic signals through the atmosphere or free space.

Advantages

- They are fairly inexpensive
- Greater efficiency, and the ability to remove signal variations and noise

Disadvantages

- Bandwidth would depend on the actual IR/RF devices being used.
- Radio Frequency devices, however, need to be operated in accordance with the FCC.
- Interference could be an issue, RF due to other RF emitting devices.

2. Free space optical

Free-space optical communication (FSO) is an optical communication technology that uses light propagating in free space to wirelessly transmit data for telecommunications or computer networking. "Free space" means the light beams travel through the open air or outer space. This contrasts with other communication technologies that use light beams traveling through transmission lines such as optical fiber or dielectric "light pipes".

The technology is useful where physical connections are impractical due to high costs or other considerations. For example free space optical links are used in cities between office
buildings which are not wired for networking, where the cost of running cable through the building and under the street would be prohibitive.

3. **Sonic**

Sonic, especially *ultrasonic* short range communication involves the transmission and reception of sound.

**Advantages**
- Relatively accurate
- Inexpensive

**Disadvantages**
- Sensitive to temperature and pressure, and also sensitive to other sensors with the same frequency.
- Its accuracy is dependent on the distance.

4. **Electromagnetic induction**

Electromagnetic induction short range communication and power. This has been used in biomedical situations such as pacemakers, as well as for short-range RFID tags.

5. **Infrared**

IR data transmission is also employed in short-range communication among computer peripherals and personal digital assistants.

**Advantages**
- Many things are controlled by infrared. Sensors are invisible to the naked eye and are very reliable.
- Are relatively accurate.

**Disadvantages**
- Most infrared sensors must be lined up or they will not work.
- Strong infrared radiation in certain industry high-heat settings may be hazardous to the eyes, resulting in damage or blindness to the user. Since the radiation is invisible, special IR-proof goggles must be worn in such places.
- Sensitive to blockage or obstacles
- Interference could be an issue IR mainly due to ambient light or any obstruction in the light path.

6. **Bluetooth**
2.2 Block diagram description
The basic block diagram of RF based microcontroller based control unit.

![Block Diagram]

*Figure 1*
Figure 2
2.3 Hardware and software components

USBasp, Breadboard, RF module 433MHz, I293d h-bridge IC, HT12E/HT12D encoder and decoder, Atmega 32 microcontroller, connecting wire, DC Motor, push buttons, resistors, capacitors, LEDs, personal computer running atmelStudio.

2.3.1 Microcontroller

A microcontroller is a compact microcomputer designed to govern the operation of embedded systems in motor vehicles, robots, office machines, complex medical devices, mobile radio transceivers, vending machines, home appliances, and various other devices. A typical microcontroller includes a processor, memory, and peripherals.

The simplest microcontrollers facilitate the operation of the electromechanical systems found in everyday convenience items. The most sophisticated microcontrollers perform critical functions in aircraft, spacecraft, ocean-going vessels, life-support systems, and robots of all kinds.

The following things have had a crucial influence on development and success of the microcontrollers:

1. Powerful and carefully chosen electronics embedded in the microcontrollers can independently or via input/output devices (switches, push buttons, sensors, LCD displays, relays etc.), control various processes and devices such as industrial automation, electric current, temperature, engine performance etc.
2. Very low prices enable them to be embedded in such devices in which, until recent time it was not worthwhile to embed anything. Thanks to that, the world is overwhelmed today with cheap automatic devices and various “smart” appliances.
3. Prior knowledge is hardly needed for programming. It is sufficient to have a PC (software in use is not demanding at all and is easy to learn) and a simple device (called the programmer) used for “loading” ready-to-use programs into the microcontroller.

A microcontroller can be compared to a standalone computer, capable of executing a series of pre-programmed tasks and interaction with other hardware devices. Traditionally they were programmed using assembly language of target device. As a result different manufactures have different assembly languages and thus they can be programmed using high level languages as C/C++.

Advantages of High-level languages over Assembly Language

- It is easier to develop programs using a high-level language.
- Program maintenance is much easier if the program is developed using a high-level language.
- Testing a program developed in a high-level language is much easier.
- High-level languages are more user-friendly and less prone to making errors.
- It is easier to document a program developed using a high-level language.
Disadvantages of High-level languages over Assembly Language

- The length of the code in memory is usually larger when high-level languages are used.
- The programs developed using the assembly language usually run faster than those developed using high-level languages.

Based on the above factors I consider Amega32.

2.3.1.1 Atmega32

The features based on its data sheet are as below;

- High-performance, Low-power.
- Advanced RISC Architecture (reduced instruction set computing)- it’s a strategy based on the insight that a simplified instruction set and able to execute instructions using fewer microprocessor cycles per instruction.
  - 32 × 8 General Purpose Working Registers
  - Fully Static Operation
- High Endurance Non-volatile Memory segments
  - 32Kbytes of In-System Self-programmable Flash program memory
  - 1024Bytes EEPROM
  - 2Kbytes Internal SRAM
  - Data retention: 20 years at 85°C/100 years at 25°C
- In-System Programming by On-chip Boot Program
- True Read-While-Write Operation
  - Programming Lock for Software Security
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Pre-scalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Pre-scaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
  - 8 Single-ended Channels
  - 7 Differential Channels in TQFP Package Only
  - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby
- And Extended Standby
- I/O and Packages
  - 32 Programmable I/O Lines
  - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
  - 4.5V - 5.5V for ATmega32
- Speed Grades
  - 0 - 16MHz for ATmega32

Block diagram of MCU atmega32

Figure 2. Block Diagram

We can go through some of peripherals shown in above figure. PORTA, PORTB, PORTC and PORTD are GPIO (General Purpose Input Output) pins. They are used for digital input and output purpose.
ADC interface is used for Analog to Digital conversion. We will be using Analog signals as temperature sensor output, humidity sensor output as input to our microcontrollers. This peripheral helps us in converting those analog values into digital ones so that we can process them with our digital AVR processing core.

Timers are used for setting the timings of events. They are also used in PWM application. Counters are used for counting some external events. Interrupts are used for signalling the microcontroller of some event and processing the instructions associated with that event. Interrupts may be hardware or software interrupt. Software interrupts are generated from the peripherals like USART, ADC, TIMERS, and SPI.

USART, SPI and TWI are used for communicating with other devices which have similar interface. These are serial interface with different speed. SPI is the fastest among all. SPI is 4 wire serial interface while USART is three wire and TWI (I2C) is a two wire serial interface.

The AVR memories

The AVR architecture makes use of four different types of memories:

Flash

Flash memory, or just flash, is erasable, non-volatile storage (it doesn’t require any refreshes to keep storing data) used for program storage. When you have compiled a program and you upload it to the device, this is where it’s stored. When your program runs, it is run directly from the flash memory (it is not copied into RAM as most PCs would do).

SRAM

Static RAM (SRAM) is used for transient program state (such as variables) as well as the program stack and any allocations made by the program. When the program starts, all global variables are initialized in SRAM by a special routine generated automatically by the compiler. The AVR architecture does not inherently or automatically reset memory, so without explicit resets by the program (usually automatic) the memory contents is persistent across resets.

EEPROM

EEPROM (electrically erasable, programmable, read-only memory) is used for non-volatile persistent storage. It is a good place to write configuration values (such as baud rates, unique IDs, etc.), to keep track of counters over a long term, and to keep static data that isn’t needed frequently. Accessing EEPROM memory requires special instructions and is quite slow compared to SRAM, and can be risky if the device loses power (the data can be corrupted). Due to the limited number of writes that EEPROM can handle (approximately 100,000) before failure, care should be taken not to write unnecessarily.
Fuses

In addition to the Flash, SRAM, and EEPROM, AVRs have a few bytes (usually 3) of fuse memory. This memory is used to store “fuse bytes”, which are boot-time configuration values used to initialize the microcontroller itself. On systems that have 3 bytes, such as the Atmega 32, they are called “low”, “high”, and “extended” (or “l”, “h”, and “e”), and they each store individual bits or bit values of configuration settings.

![Figure 4 atmega 32 circuit diagram](image)

2.3.1.2 USBasp

USB based programmer for the AVR, and uses AVRdude for burning hex files into AVR microcontroller. In order to program any microcontroller you need the .HEX file. It is nothing but the machine code for the microcontroller. This file is generated by the corresponding assembler software, which converts programming code into machine code. Programming code can be produced by third party cross compiler software or can be handwritten.

Using the USBasp is an inexpensive and easy method therefore connecting it up to allow for it to program an AVR isn’t that difficult. We will now go over the wiring connections necessary for the USBASP to transfer data from a computer to an AVR chip. One end connects into the computer. This allows for us to transfer the compiled program from the computer to the USBASP. The other end of the USBASP normally gets connected either to a 6-pin or a 10-pin cable, which can then get hooked up easily to a breadboard through header pins.

Both 6-pin and 10-pin cables are common, so knowing the pin out of these is essential to connecting them.

The 10-pin cable pin out is shown below
Regardless of whether the 6-pin cable or 10-pin cable is used, and it really doesn't matter, there are 6 pins we are really wiring up and these are the MISO, SCK, RST, VTG, MOSI, and GND connections.

Basically, we're connecting the USBASP to the AVR chip's MISO, SCK, RST, VTG, MOSI, and GND connections that we are programming. This allows direct communication between the USBASP and the target AVR chip.

Below is an explanation of each pin and their function.

- **MOSI** - (Master Out Slave In) - it allows the master device to send data to slave or target device.
- **MISO** - (Master In Slave Out) - it allows slave device/target to send information to master device.
- **SCK** – (serial clock) - this mutual clock shared between master and slave device for synchronized communication.
- **Reset** - (target AVR MCU Reset) - The reset pin for the AVR chip being programmed must be put in active low in order for programming to occur.
- **VCC** - (Power) - The master and slave device both need power in order to operate.
- **GND** - (Common Ground) - The master and slave device must share a common power ground for operation.

With this setup, the USBASP is the master device and the AVR chip we are programming is the target, or slave device. The best way to think of it is the master device is doing the programming, while the slave device is the subordinate device that is need of being programmed. In order for a master device to program a slave device, they need have communication amongst each other. The master device needs to be able to write data to the slave device's flash memory and be able to receive data from the slave data. This is done through the MOSI and MISO pins. The MOSI pin stands for Master Out Slave In. This is the pin by which the master device outputs, or writes, data to the target AVR which is being
programmed. MISO stands for Master In Slave Out. This is the pin by which the slave device (AVR being programmed) can send information to the master device. Both these communication pathways are essential to program a device. So the MOSI and MISO pins establish communication amongst the master device and the slave device it is to program.

The SCK pin is the clock. It is essential because in order for the master and slave device to communicate, they need to have a time signal to communicate data in synchrony. The common clock signal shared between the master and slave device allow for efficient communication.

The RST pin is an essential connection because it must be put to an active low connection in order for programming to occur between the master and slave device. It is normally held high, but for programming to occur, it must be put low. It is an active low pin. When the RST pin is put low, the master slave can communicate on the SCK, MISO, and MOSI lines.

The remaining 2 connections, VCC and GND, are the simplest; they are for power. The AVR chip being programmed and the master device doing the programming both need to power in order to operate.

These are the only 6 essential connections that need to be made from the USBASP Programmer to the target AVR chip.

*Figure 6 USBasp*
2.2 RF module (RX-TX MODULES (434MHz))
Since our circuit utilizes the RF module (Tx/Rx) for making the wireless remote, this could be used to drive an output from a distant place. RF module uses radio frequency to send signals. These frequencies are transmitted at a particular frequency and band rate. A receiver can only receive these signals only if it is configured for that frequency.

A four channel encoder/decoder pair will also be used in this system. The input signal at the transmitter side, are taken through four switches while output are monitored on a microcontroller considering to each input switch.

This circuit will be used for designing the Remote Appliance Control System, and outputs from receiver will drive the corresponding DC motors.

![RF module receiver and transmitter](image)

**Figure 7 RF module receiver and transmitter**

The RF module, as the name suggests, operates at Radio Frequency. The corresponding frequency range varies between 30 kHz & 300 GHz. In this RF system, the digital data is represented as variations in the amplitude of carrier wave. This kind of modulation is known as Amplitude Shift Keying (ASK).

Transmission through RF is better than IR (infrared) because of several reasons.
- Firstly, signals through RF can travel through larger distances making it suitable for long range applications.
- Also, while IR mostly operates in line-of-sight mode, RF signals can travel even when there is an obstruction between transmitter & receiver.
- Next, RF transmission is more strong and reliable than IR transmission.
- RF communication uses a specific frequency unlike IR signals which are affected by other IR emitting sources.

This RF module comprises of an **RF Transmitter** and an **RF Receiver**. The transmitter/receiver (Tx/Rx) pair operates at a frequency of 434 MHz. An RF transmitter receives serial data and transmits it wirelessly through RF through its antenna connected at pin4. The transmission occurs at the rate of 1Kbps - 10Kbps. The transmitted data is received...
by an RF receiver operating at the same frequency as that of the transmitter. The RF module is often used along with a pair of encoder/decoder. The encoder is used for encoding parallel data for transmission feed while reception is decoded by a decoder.

Figure 8 Receiver module

Figure 9 Transmitter module

2.2.1 Description
This radio frequency (RF) transmission system employs Amplitude Shift Keying (ASK) with transmitter/receiver (Tx/Rx) pair operating at 434 MHz; the transmitter module takes serial input and transmits these signals through RF. The transmitted signals are received by the receiver module placed away from the source of transmission. The system allows one way communication between two nodes, namely, transmission and reception. The RF module has been used in conjunction with a set of four channel encoder/decoder ICs. Here HT12E & HT12D have been used as encoder and decoder respectively. The encoder converts the parallel inputs (from the remote switches) into serial set of signals. These signals are serially transferred through RF to the reception point. The decoder is used after the RF receiver to decode the serial format and retrieve the original signals as outputs. These outputs can be observed on corresponding DC motor.

Figure 10 RX/TX
The Encoder IC (HT12E) receives parallel data in form of address bits and control bits. This control signals from remote switches along with 8 address bits constitute a set of 12 parallel signals. It encodes the parallel signals into serial bits.

The serial data is fed to the RF transmitter as shown below.

---

Figure 11 block diagram for RF module

Figure 12 block diagram for remote module
Transmitter upon receiving serial data from encoder IC (HT12E) transmits it wirelessly to RF receiver. Upon receiving these signals the receiver sends them to a decoder IC (HT12D). The decoder then retrieves the original parallel format from received serial data.

When no signal is received at data pin of HT12D, it remains in standby mode and consumes very less current (less than 1μA) for a voltage of 5V. When signal is received by receiver, it is given to DIN pin (pin14) of HT12D. On reception of signal, oscillator of HT12D gets activated. IC HT12D then decodes the serial data and checks the address bits three times. If these bits match with the local address pins (pins 1-8) of HT12D, then it puts the data bits on its data pins (pins 10-13) and makes the VT pin high.

An LED is connected to VT pin (pin17) of the decoder. This LED works as an indicator to indicate a valid transmission. The corresponding output is thus generated at the data pins of decoder IC. A signal is sent by lowering any or all the pins 10-13 of HT12E and corresponding signal is received at receiver’s end (at HT12D).

Address bits are configured by using the first 8 pins of both encoder and decoder ICs. To send a particular signal, address bits must be same at encoder and decoder ICs. By configuring the address bits properly, a single RF transmitter can also be used to control different RF receivers of same frequency.

**NB:** To summarize, on each transmission, 12 bits of data is transmitted consisting of 8 address bits and 4 data bits. The signal is received at receiver’s end which is then fed into decoder IC. If address bits get matched, decoder converts it into parallel data and the corresponding data bits get lowered which could be then used to drive the DC motor. The outputs from this system can either be used in negative logic or NOT gates (like 74LS04) can be incorporated at data pins.
Figure 14 RF Module circuit diagram
2.2.2 HT12D DECODER

HT12D is a decoder integrated circuit that belongs to \(2^{12}\) series of decoders. This series of decoders are mainly used for remote control system applications, like burglar alarm, car door controller, security system etc.

It is mainly provided to interface RF circuits. They are paired with \(2^{12}\) series of encoders. The chosen pair of encoder/decoder should have same number of addresses and data format. In simple terms, HT12D converts the serial input into parallel outputs. It decodes the serial addresses and data received by, say, an RF receiver, into parallel data and sends them to output data pins. The serial input data is compared with the local addresses three times continuously. The input data code is decoded when no error or unmatched codes are found. A valid transmission is indicated by a high signal at VT pin. HT12D is capable of decoding 12 bits, of which 8 are address bits and 4 are data bits. The data on 4 bit latch type output pins remain unchanged until new is received.

2.2.2.1 Pin Diagram

![Figure 15]

Table 1

<table>
<thead>
<tr>
<th>Pin number</th>
<th>function</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8 bit address pins for input</td>
<td>A0</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>A1</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>A2</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>A3</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>A4</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>A5</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>A6</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>A7</td>
</tr>
<tr>
<td>9</td>
<td>GROUND(0V)</td>
<td>GROUND</td>
</tr>
<tr>
<td>10</td>
<td>4 bit data pins for output</td>
<td>D0</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>D1</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>D2</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>D3</td>
</tr>
<tr>
<td>14</td>
<td>Serial data input</td>
<td>INPUT</td>
</tr>
<tr>
<td>15</td>
<td>Oscillator output</td>
<td>OSC 2</td>
</tr>
<tr>
<td>16</td>
<td>Oscillator input</td>
<td>OSC 1</td>
</tr>
<tr>
<td>17</td>
<td>Valid transmission</td>
<td>VT</td>
</tr>
<tr>
<td>18</td>
<td>Supply voltage</td>
<td>Vcc</td>
</tr>
</tbody>
</table>
2.2.3 HT12E ENCODER

HT12E is an encoder integrated circuit of $2^{12}$ series of encoders. They are paired with $2^{12}$ series of decoders for use in remote control system applications. It is mainly used in interfacing RF circuits. The chosen pair of encoder/decoder should have same number of addresses and data format. Simply put, HT12E converts the parallel inputs into serial output. It encodes the 12 bit parallel data into serial for transmission through an RF transmitter. These 12 bits are divided into 8 address bits and 4 data bits. HT12E has a transmission enable pin which is active low. When a trigger signal is received on TE pin, the programmed addresses/data are transmitted together with the header bits via an RF or an infrared transmission medium. HT12E begins a 4-word transmission cycle upon receipt of a transmission enable. This cycle is repeated as long as TE is kept low. As soon as TE returns to high, the encoder output completes its final cycle and then stops.

2.2.3.1 Pin Diagram

![Figure 16](image)

2.2.3.2 Pin description

Table 2

<table>
<thead>
<tr>
<th>Pin number</th>
<th>Function</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8 bit address pins for input</td>
<td>A0</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>A1</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>A2</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>A3</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>A4</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>A5</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>A6</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>A7</td>
</tr>
<tr>
<td>9</td>
<td>GROUND(0V)</td>
<td>GROUND</td>
</tr>
<tr>
<td>10</td>
<td>4 bit data for input</td>
<td>D0</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>D1</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>D2</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>D3</td>
</tr>
<tr>
<td>14</td>
<td>Transmission enable</td>
<td>TE</td>
</tr>
<tr>
<td>15</td>
<td>Oscillator output</td>
<td>OSC 2</td>
</tr>
<tr>
<td>16</td>
<td>Oscillator input</td>
<td>OSC 1</td>
</tr>
<tr>
<td>17</td>
<td>Valid transmission</td>
<td>VT</td>
</tr>
<tr>
<td>18</td>
<td>Supply voltage</td>
<td>Vcc</td>
</tr>
</tbody>
</table>
2.3 PWM

A method, which is extensively used in motor controller, is the pulse width modulation (PWM). PWM switching technique is a best method to control the speed of DC motor as compared to any other method. The duty cycle can be varied to get the variable output voltage.

The Pulse-Width-Modulation (PWM) in microcontroller is used to control duty cycle of DC motor drive. PWM is an entirely different approach to controlling the speed of a DC motor. Power is supplied to the motor in square wave of constant voltage but varying pulse-width or duty cycle. Duty cycle refers to the percentage of one cycle during which duty cycle of a continuous train of pulses. Since the frequency is held constant while the on-off time is varied, the duty cycle of PWM is determined by the pulse width. Thus the power increases duty cycle in PWM.

The expression of duty cycle is determined by:

\[
\%\text{Duty cycle} = \frac{\text{on time}}{\text{on time} + \text{off time}} \times 100
\]

Since PWM is a method of transmitting information on a series of pulses. The data that is being transmitted is encoded on the width of these pulses to control the amount of power being sent to a load. PWM is very handy tool, you can use it for power delivery, voltage regulation and amplification and audio effects. In this document we will go through the basic understanding of PWM, register configuration for different modes of PWM in atmega32. Analog voltage and current can be used to control devices directly like speed of a DC Motor.

In a simple analog controller, a knob is connected to a variable resistor. As you turn the knob, the resistance goes up or down. As that happens, the current flowing through the resistor increases or decreases.

Analog circuits can get very hot; the power dissipated is proportional to the voltage across the active elements multiplied by the current through them. Analog circuitry can also be sensitive to noise. By controlling analog circuits digitally, system costs and power consumption can be drastically reduced.

What’s more, many microcontrollers already include on-chip PWM controllers, making implementation easy, for an example atmega32 comes with 3 timers all can be configured to use for PWM.

NB: As intuitive and simple as analog control may seem, it is not always economically attractive or otherwise practical.
2.3.1 Frequency:
Using the switch example, the frequency would be how fast the switch was turned on and off. If the frequency is too low (switch is changed slowly), then the motor will run at full speed when the switch is on, and completely stop when the switch is off. But if the frequency is too high, the switch may mechanically fail. In reality there is no switch, but rather an electronic board named an H-Bridge that switches the motor on and off. So in electrical terms; if the frequency is too low, the time constant of the motor has enough time to fully switch between on and off. Similarly the upper limit on the frequency is the limit that the H-Bridge board will support, analogous to the mechanical switch.

*The maximum frequency of this H-Bridge Board is 500 kHz, but the recommended frequency of the PWM for this board is 31.25 kHz.*

2.2.2 Duty Cycle:
The duty cycle is analogous to how long the upper switch (switch1) remains on as a percentage of the total switching time. In essence it is an average of how much power is being delivered to the motor. Duty cycle gives the proportional speed control of the motor. Effectively, these duty cycles would run the motor at –, – and – of full speed respectively.

An example of –, – and – duty cycles.

![Duty Cycle Diagram](image)

*Figure 17*
2.4 H BRIDGE

An H-Bridge is an electronic power circuit that allows motor speed and direction to be controlled. Often motors are controlled from microcontroller to accomplish a mechanical goal. The microcontroller provides the instructions to the motors, but does not provide the power required to drive the motors. An H-bridge circuit inputs the microcontroller instructions and amplifies them to drive a mechanical motor. The H-bridge takes in the small electrical signal and translates it into high power output for the mechanical motor.

*Basically current is amplified which takes a low-current signal from MCU and gives a proportional higher current signal to which can control and drive a motor. In most cases, a transistor can as a switch and perform this task.*

Most DC Motors can rotate in two directions depending on how the battery is connected to the motor. Both the DC motor and the battery are two terminal devices that have positive and negative terminals. In order to run the motor in the forward direction, connect the positive motor wire to the positive battery wire and negative to negative. However, to run the motor in reverse just switch the connections; connect the positive battery wire to the negative motor wire, and the negative battery wire to the positive motor wire. An H-Bridge circuit allows a large DC motor to be run in both direction with a low level logic input signal.

The H-Bridge electronic structure is explicit in the name of the circuit H -Bridge. The power electronics actually form a letter H configuration, as shown below.
The switches are symbolic of the electronic Power;

![Figure 19 H bridge](image)

**Figure 19 h bridge**

![Figure 20 voltage control](image)

**Figure 20 voltage control**

If it is desired to turn the motor on in the *forward* direction, switches 1 and 4 must be closed to power the motor. If it is desired to turn the motor on in the *reverse* direction, switches 2 and 3 must be closed to power the motor.

This section will explain what the “switches” above actually are in terms of electronic components. The switches are power transistors that have certain properties that allow them to switch high currents based on an input signal. The transistors are used in two
regions of operation; Cut-off mode and Saturation mode which correspond to switch off and switched on respectively.

In the H-Bridge case, to put a transistors into the Cut-off mode, the input signal (Gate Voltage) to the transistors must be grounded. However, to turn on the transistors and put them into saturation mode requires a more complicated process. Transistors are three terminal devices with the terminals being the Base, Collector and Emitter.

This circuit uses the basic concept of transistors as a switch. Transistor with proper biasing can be used as switch, i.e. it can be used to toggle between the two states of a switch on or off.

In this example, I have used an ordinary BC547, which is a general purpose NPN transistor. It is inexpensive and commonly available. However, this transistor isn’t very powerful, and can provide maximum driving current up to 100 mA. This particular driver circuit is still good for driving small dc motors that require less than 100 mA current.

The working principle of this circuit is very simple. The transistor acts as a power switch for the motor. The switch can be controlled through the logic voltage applied at the On/Off terminal. When the On/Off pin is at logic 1 (+5 V), the transistor is turned on and the motor is connected across VCC and ground. The VCC voltage could be greater than 5 V. Therefore, one benefit of this transistor motor driver is that you can control a higher voltage motor (> 5 V) with a 5 V logic output from a microcontroller. The logic 0 at the On/Off terminal turns the transistor off and the motor is stopped.

The 1K resistor connected in series with the base of the transistor limits the base current to a safe level.

The diode connected across the motor terminals is for the protection of the transistor. When the switch is turned off, the collapsing electromagnetic field inside the motor generates a high voltage across the motor terminals with reverse polarity. This voltage could be high enough to damage the transistor permanently. The diode provides a current path back through the motor coil, and hence prevents any high voltage formation. During normal condition the diode is reverse biased and doesn’t affect the switching operation of the transistor.

NB: H-bridges can be built from scratch as shown above using relays, mosfets, FET (field effect transistors) or BJT transistors.
Since the motor is controlled by the 4 switches above. For the speed control explanation that follows only switches 1 and 4 will be considered because speed control is identical in the forward and reverse direction. Say the switches 1 and 4 are turned on, the motor will eventually run at full speed. Similarly if only switch 4 is turned on while switch 1 is off the motor stops. Using this system, how could the motor be run at 1/2 of the full speed? The answer is actually quite simple; turn switch 1 on for half the time and turn it off for the other half. In order to implement this system in reality, one must consider two main factors, namely frequency and duty cycle.

Since my current requirement is not too high and all I need is a single package which does the job of driving a small DC motor in two directions, thus we will use a L293D IC. This has an inbuilt fly-back diode which protects the driving transistors from voltage spikes that occur when motor coil is turned off.
L293D IC
It generally comes as a standard 16 dual-in line package. This can simultaneously control 2 small motors in either direction forward reverse with just four microcontroller pins;

1. Output current capability is limited to 600mA per channel with peak output current limited to 1.2A (non-repetitive). Also note the words "non-repetitive"; if the current output repeatedly reaches 1.2A, it might destroy the drive transistors.
2. Supply voltage can be as large as 36 Volts.
3. L293D has an enable facility which helps you enable the IC output pins. If an enable pin is set to logic high, then state of the inputs match the state of the outputs. If you pull this low, then the outputs will be turned off regardless of the input states.
4. The datasheet also mentions an "over temperature protection" built into the IC. This means an internal sensor senses its internal temperature and stops driving the motors if the temperature crosses a set point
5. Another major feature of L293D is its internal clamp diodes. This fly-back diode helps protect the driver IC from voltage spikes that occur when the motor coil is turned on and off (mostly when turned off)
6. The logical low in the IC is set to 1.5V. This means the pin is set high only if the voltage across the pin crosses 1.5V which makes it suitable for use in high frequency applications like switching applications (upto 5 KHz)
7. Lastly, this integrated circuit not only drives DC motors, but can also be used to drive relay solenoids, stepper motors etc.

L293D connections

Figure 22 L293D Connections

1. Pin1 and Pin9 are "Enable" pins. They should be connected to +5V for the drivers to function. If they pulled low (GND), then the outputs will be turned off regardless of the input states, stopping the motors. This pins are connected to a regulated positive 5 Volts.
2. Pin4, Pin5, Pin12 and Pin13 are ground pins which should ideally be connected to microcontroller's ground.
3. Pin2, Pin7, Pin10 and Pin15 are logic input pins. These are control pins which should be connected to microcontroller pins. Pin2 and Pin7 control the first motor (left); Pin10 and Pin15 control the second motor (right).

4. Pin3, Pin6, Pin11, and Pin14 are output pins. Tie Pin3 and Pin6 to the first motor, Pin11 and Pin14 to second motor.

5. Pin16 powers the IC and it should be connected to regulated +5Volts.

6. Pin8 powers the two motors and should be connected to positive lead of a secondary battery.