**CHAPTER 1**

**INTRODUCTION**

A gesture is a form of non-verbal communication in which visible bodily actions communicate particular messages, either in place of speech or together and in parallel with words. Gestures include movement of the hands, face, or other parts of the body. Gestures differ from physical non-verbal communication that does not communicate specific messages, such as purely expressive displays or displays of joint attention.

The movement of gestures can be used to interact with technology like computers, using touch or multi-touch popularized by the iPhone, physical movement detection and visual motion capture, used in video game consoles.

Human Computer Interaction(HCI) plays a vital role in day to day activities. Use of computer vision has taken HCI to an altogether different level. In this project, simple potentiometers are used for tracking hand movements to control mouse pointer and left click. The features to be detected include hand movement and click of button. Our objective is to implement all the mouse tasks without the use of mouse.

For detecting a hand gesture, first the hand needs to be recognized. Then its features can be detected using different logic. Finally the features extracted are used for making gestures recognized by a system as input. In this project, the designed systemis capable of identifying four different gestures, one for each hand movement. To make sure that only one of the three required tasks is performed, at an instant.

The ability to track a person's movements and determine what gestures they may be performing can be achieved through various tools. Although there is a large amount of research done in image/video based gesture recognition, there is some variation within the tools and environments used between implementations.

**Wired gloves:** These can provide input to the computer about the position and rotation of the hands using magnetic or inertial tracking devices. Furthermore, some gloves can detect finger bending with a high degree of accuracy (5-10 degrees), or even provide hectic feedback to the user, which is a simulation of the sense of touch. The first commercially available hand-tracking glove-type device was the DataGlove, a glove-type device which could detect hand position, movement and finger bending. This uses fiber optic cables running down the back of the hand. Light pulses are created and when the fingers are bent, light leaks through small cracks and the loss is registered, giving an approximation of the hand pose.

**Depth-aware cameras:** Using specialized cameras such as structured light or time-of-flight cameras, one can generate a depth map of what is being seen through the camera at a short range, and use this data to approximate a 3d representation of what is being seen.

**Controller-based gestures:** These controllers act as an extension of the body so that when gestures are performed, some of their motion can be conveniently captured by software. Mouse gestures are one such example, where the motion of the mouse is correlated to a symbol being drawn by a person's hand, which can study changes in acceleration over time to represent gestures.

Devices such as the LG Electronics Magic Wand, the Loop and the Scoop use Hillcrest Labs' Freespace technology, which uses MEMS accelerometers, gyroscopes and other sensors to translate gestures into cursor movement.

**Mechatronics-based gestures:** These gestures include some mechanical arrangement with body part. Thus with these systems motion can be detected and proper controlling action can beperformed.

Gestures can be used to control interactions within video games to try and make the game player's experience more interactive or immersive.

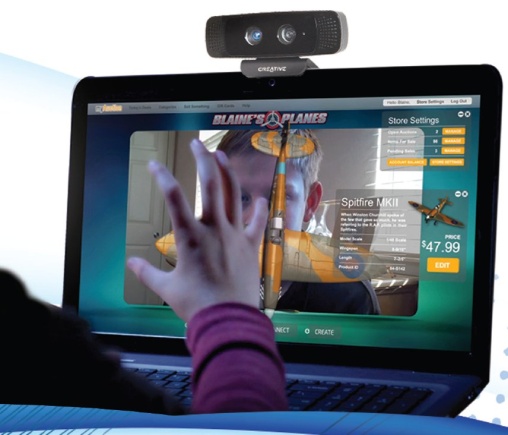
In affective computing, gesture recognition is used in the process of identifying emotional expression through computer systems.

Through the use of gesture recognition, "remote control with the wave of a hand" of various devices is possible. The signal must not only indicate the desired response, but also which device to be controlled

Foregoing the traditional keyboard and mouse setup to interact with a computer, strong gesture recognition could allow users to accomplish frequent or common tasks using hand gestures.

**CHAPTER 2**

**MARKET SURVEY**

**1.) SOFTKINETIC** 

**SoftKinetic** is a Belgian company which develops gesture recognition hardware and software for real-time range imaging (3D) cameras (such as time-of-flight cameras).

SoftKinetic technology has been applied to interactive digital signage and advergaming, interactive television and physical therapy. SoftKinetic's gesture recognition software platform, named *iisu*, can recognize and distinguish or isolate different scenic elements, can identify and track the body parts of a user, and can adapt the user's shape, posture, and movements to an existing physical model, and vice versa. iisu is compatible with all major real-time range imaging cameras, and the middleware guards developers from the particularities of the hardware.

**2.) Virtual Keypad**



A **virtual keyboard** is a software component that allows a user to enter characters. A virtual keyboard can usually be operated with multiple input devices, which may include a touchscreen, an actual Computer keyboard and a computer mouse.

**3.) KINECT**

****

**Kinect** (codenamed in development as ***Project Natal***) is a line of motion sensing input devices by Microsoft for Xbox 360 and Xbox One video game consoles and Windows PCs. Based around a webcam-style add-on peripheral, it enables users to control and interact with their console/computer without the need for a game controller, through a natural user interface using gestures and spoken commands. The first-generation Kinect was first introduced in November 2010 in an attempt to broaden Xbox 360's audience beyond its typical gamer base. A version for Windows was released on February 1, 2012. Kinect competes with several motion controllers on other home consoles, such as Wii Remote Plus for Wii, PlayStation Move/PlayStation Eye for PlayStation 3, and PlayStation Camera for PlayStation 4.

**4.) LEAP MOTION**



**Leap Motion, Inc.** is an American company that manufactures and markets a computer hardware sensor device that supports hand and finger motions as input, analogous to a mouse, but requiring no hand contact or touching.

The Leap Motion controller is a small USB peripheral device which is designed to be placed on a physical desktop, facing upward. Using two monochromatic IR cameras and three infrared LEDs, the device observes a roughly hemispherical area, to a distance of about 1 meter (3 feet). The LEDs generate a 3D pattern of dots of IR light and the cameras generate almost 300 frames per second of reflected data, which is then sent through a USB cable to the host computer, where it is analyzed by the Leap Motion controller software using "complex math" in a way that has not been disclosed by the company, in some way synthesizing 3D position data by comparing the 2D frames generated by the two cameras.

[](http://en.wikipedia.org/wiki/File:Leap_Motion_Controller.JPG)

[http://bits.wikimedia.org/static-1.23wmf20/skins/common/images/magnify-clip.png](http://en.wikipedia.org/wiki/File:Leap_Motion_Controller.JPG)

Leap Motion Controller

The smaller observation area and higher resolution of the device differentiates the product from the Kinect, which is more suitable for whole-body tracking in a space the size of a living room

**CHAPTER 3**

**LITERATURE SURVEY**

Gone are the days when people wanted to own a TV with remote control — now almost everything can be controlled from far away — cars, TVs, other electronic engineering, and even apps installed on our computers. But I can’t remember any case when a computer could be controlled remotely.

Thus the compatibility and comfort of human is first and foremost thing when it comes to design of electronic design for humans. This lead us to think designing a hand gesture controlled game which is almost completely user friendly and can easily be operated by everyone. There are various functions which can be developed and built in hand gesture operation to make it more efficient, but our main aim was to design model that could be implemented in low cost. This device can be used to play games by controlling keyboard buttons virtually by hand gesture. Thus this project can serve as a function of computer control.

**CHAPTER 4**

**IDENTIFICATION OF PROJECT:**

The Hand Gesture controlled game functions as stated below:

* Obtaining data from potentiometer sensor
* Encode the Data
* Transmit the data using RF module (wireless)
* Receives data and decode
* Controlling according to data

**CHAPTER 5**

**OBJECTIVE OF PROJECT**

The main objective of to make **Hand Gesture Project** since the compatibility and comfort of human is first and foremost thing when it comes to design of electronic design for humans and to make it more reliable. Compared to traditional methods whose performance and tracking ability depend on the surface the mouse moves and need mouse pad or any special surface. Again the function of the keyboard will be done **wirelessly** which will reduce the hassle of devices and thus will miniaturize the respective electronic device

**CHAPTER 6.1**

**PROJECT SPECIFICATIONS**

This Hand Gesture controlled Game executes following steps:

* Data Acquisition and Transmitter System
* Receiver System
* Computer Interface system

This controller base keyboard control which uses 8-bit AVR RISC-based microcontroller ATmega 328 P which is compatible with Arduino Open Source.

It combines 32KB ISP flash memory with read-while-write capabilities with 6-channel 10-bit A/D converter.

The device operates between 1.8-5.5 volts. At transmitter side this is taken from Battery(9V). At receiver side power is supplied through USB by PC.

For hand movement potentiometer sensors 100K and push buttons are used.

For wireless data transmission RF modules 434MHz are used which uses ASK modulation technique.

It also has LEDs for On/Off indication, Reset Switches and 16MHz crystal oscillator.

**CHAPTER 6.2**

**BLOCK DIAGRAM OF HAND GESTURE TRANSMITTER**

**CRYSTAL OSCILLATOR 16MHz**

**RESET CIRCUIT**

**RADIO FREQUENCY TRANSMITTER 433.92MHz**

**PUSH BUTTON SENSOR**

**REGULATOR 7805**

**POTENTIOMETER SENSOR-2**

**POTENTIOMETER SENSOR-1**

**LED INDICATION ON/OFF**

**POWER SUPPLY (9V BATTERY)**

**MICROCONTROLLER**

**ATmega 328P**

**Fig. 6.2.1**

**BLOCK DIAGRAM OF HAND GESTURE RECEIVER**

**USB**

**SIGNAL AMPLIFIER CC Configuration**

**CONDITIONING CIRCUIT**

**CRYSTAL OSCILLATOR 16MHz**

**RADIO FREQUENCY RECEIVER 433.92MHz.**

**DATA PINS D+ D-**

**RESET CIRCUIT**

**VOLTAGE CONDITIONINGG**

**VCC**

**GND**

**LED INDICATION ON/OFF**

**MICROCONTROLLER**

**ATmega328P**

**Fig. 6.2.2**

**CHAPTER 6.3**

**DESCRIPTION**

The transmitter block diagram which consists of three systems named as:

1) Sensor Data Acquisition System

2) Power System

3) Transmitter Module

It also consists of heart of the system (microcontroller) ATmega 328P.

**Microcontroller:**

Features:

• High Performance, Low Power AVR® 8-Bit Microcontroller

• Advanced RISC Architecture

– 131 Powerful Instructions – Most Single Clock Cycle Execution

– 32 x 8 General Purpose Working Registers

– 32K Bytes of In-System Self-Programmable Flash progam memory

– 6-channel 10-bit ADC in PDIP Package

– Programmable Serial USART

– Master/Slave SPI Serial Interface

– On-chip Analog Comparator

– 28-pin PDIP

• Operating Voltage:

– 1.8 - 5.5V for ATmega48PA/88PA/168PA/328P

• Speed Grade:

– 0 - 20 MHz @ 1.8 - 5.5V

**Power Supply System:**

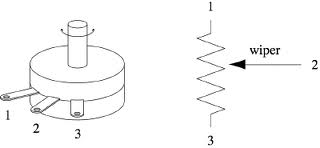
Firstly the regulated power supply i.e. +5 V dc supply by using 7805 regulator IC which is provided to microcontroller.

Features:

* Input voltage: 7-25V
* Output voltage: min 4.8- max5.
* Output current: up to 1.5A
* Operating Junction Temp: 0-125C

**Sensor Data Acquisition System:**

This system consists of 2 potentiometers and one push button. One potentiometer for X co-ordinate of mouse pointer and other for Y co-ordinate of mouse pointer. Push button is employed for left click operation. The potentiometers are connected to analog input pins of AVR microcontroller since there is no need of external ADC. Push button is connected to digital input pins.

****

****

**Fig. 6.3.1**

**Transmitter Module:**

The data is taken by controller. Controller collects the data and encodes the data. The ADC data is of 10 bit and USART logic sends 8 bits of data. So this data is converted and extra check bits are added.This data is given to RF transmitter Module. They have reasonable range and works very good. Typical data rate is 1000 bits per seconds.

The PINs are as follows:

TX

1) Antenna

2) Vcc (Positive Supply)

3) DATA (Data Input)

4) GND

The receiver block diagram which consists of

1)Receiver Module

2)Signal Amplifier

3)USB interface

**Receiver Module:**

The receiver receives the data sent by transmitter module.The RF module does not requires two devices to be in line of sight and can cross many obstacles. The data is then amplified using Transistor in Common Collector mode.The pin configuration of RF receiver is as below:

1) Antenna

2) GND

3) GND

4) Vcc (Positive Supply)

5) Vcc (Positive Supply)

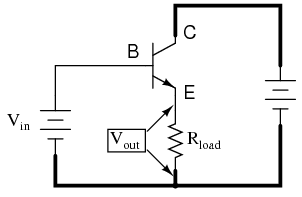
6) DATA

7) DATA

8) GND

**Signal Amplifier:**

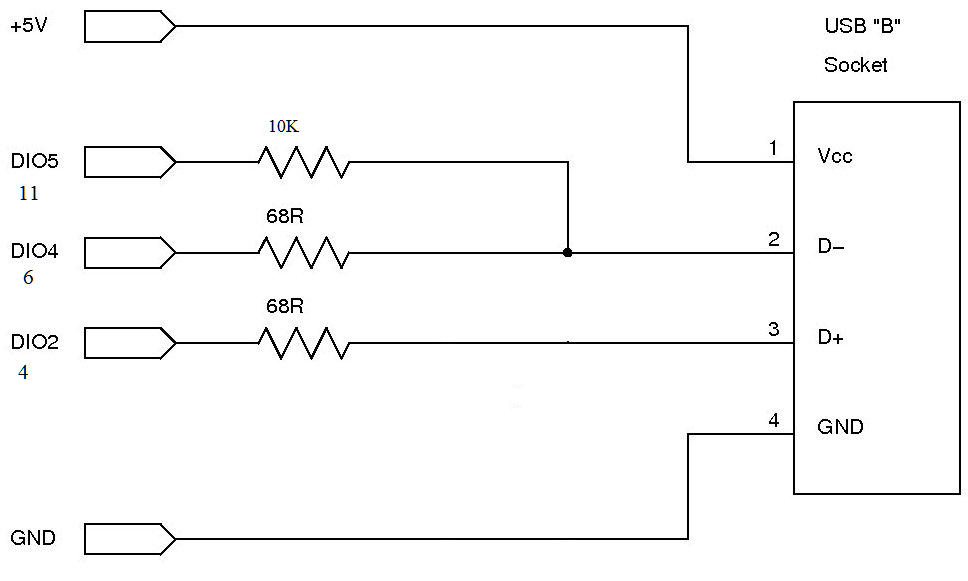
To ensure signal detection the data is amplified using transistor in common collector Configuration. The input is applied to base terminal of transistor. Supply voltage i.e. Vcc is connected to collector terminal of transistor. The output is taken across load resistor (Rload). The output voltage is nearly same as that of input voltage but current is amplified by hfe.

 **Fig. 6.3.2**

**USB interface:**

To interface USB directly with ATmega 328 we need some circuitary. We used 2 diodes in series to Vcc pin of USB to lower down the voltage level to 3.6V. Since ATmega 328 can work on minimum 1.8V.

Most people link an Arduino to a host using serial communications across a USB connection with custom code running on the host to send information to the Arduino or receive it in return, but giving our Arduino the ability to pretend to be a keyboard or mouse opens up a whole world of possibilities because it means your Arduino can interact with software that was never intended for external control. That could be desktop software such as a game or a web browser: Arduino could "type" into a web form and submit it on your behalf, or act as a custom controller for a game.



**Fig. 6.3.3**

**CHAPTER 7**

**FLOW CHARTS**

**Transmitter System**

START

Set analog reference as EXTERNAL

Select I/O pin 12 as Input

Read analog value from input A0

Shift digital data LEFT by 3 ORing with 2

Save as a1

Shift digital data to right by2 ANDing with 248

ORing data with digital input value

Save as a2

Transmit serially a1

Transmit serially a2

Transmit serially abs a2-a1

Read analog value from input A1

B

A

B

A

Shift digital data LEFT by 3 ORing with 2

Save as a1

Shift digital data to right by2 ANDing with 248

Transmit serially abs a2-a1

Transmit serially a1

ORing data with digital input value

Save as a2

Transmit serially a2

Delay 5ms

STOP

**Fig. 7.1**

**Receiver System**

START

Accept data serially

If exception print Error

If in>0

No

Yes

Save byte in C1

If C1&7==2

No

Yes

Save byte in C2

Save next byte in C3

If C1-C2==C3

No

Save X as C1>>3|C2<<2&31<<5

Save b as C2&7

Save latest 5 values of X in matrix

D

C

D

C

If C1&7==3

No

Save b as C2&7

If C1-C2==C3

Save Y as C1>>3|C2<<2&31<<5

Save byte in C2

Save next byte in C3

No

mx=average of 5 values of x

Save latest 5 values of Y in matrix

my=average of 5 values of y

my<300

No

YES

Press key ‘SPACE’

D

F

E

F

E

D

My>500

mx<340

mx>680

No

No Yes No Yes

Yes

Press key ‘A’

Press key ‘D’

Press key ‘X’

STOP

**Fig. 7.2**

**ALGORITHMS:**

**Transmitter System:**

1. START
2. Set analog reference as EXTERAL.
3. Select digital I/O pin 12 as input.
4. Read analog value from input channel A0.
5. Shift digital data to LEFT by 3 bits, ORing with 2 and save as a1.
6. Shift digital data to right by2 bits and ANDing with 248 (11111000).
7. ORing data with digital input pin 12 value and save as a2.
8. Transmit serially a1
9. Transmit serially a2.
10. Transmit serially abs a2-a1
11. Delay 5 ms and go to step 4
12. STOP.

**Receiver System:**

1. START
2. Accept data serially.
3. If exception print “Error”.
4. If incoming data is not greater than 1 then stay here.
5. Save current byte in C1.
6. If (C1&7) is not equal to 2 then go to step 13.
7. Save byte in variable C2.
8. Save next variable in C3.
9. Check if C1-C2 is equal to C3 or not. If not go to step 13.
10. Extracting original data from C1 and C2. Save in X asC1>>3|C2<<2&31<<5.
11. Save b with (C2&7) i.e. status of key press.
12. Save latest 5 values of X in matrix column1.
13. Check if (C1&7) is equal to 3. If not go to step 20.
14. Save byte in variable C2.
15. Save next variable in C3.
16. Check if C1-C2 is equal to C3 or not. If not go to step 20.
17. Extracting original data from C1 and C2. Save in Y asC1>>3|C2<<2&31<<5.
18. Save b with (C2&7) i.e. status of key press.
19. Save latest 5 values of Y in matrix column2.
20. Store Average of 5 values of X in mx.
21. Store Average of 5 values of Y in my.
22. If my<300 then press key ‘SPACE’else go to step 25.
23. If mx<340 then press key ‘A’ else go to step 2.
24. If mx>680 then press key ‘D’else go to step 2.
25. If my>500 then prss key ‘X else go to step 2.
26. To continue go to step 2.
27. STOP.

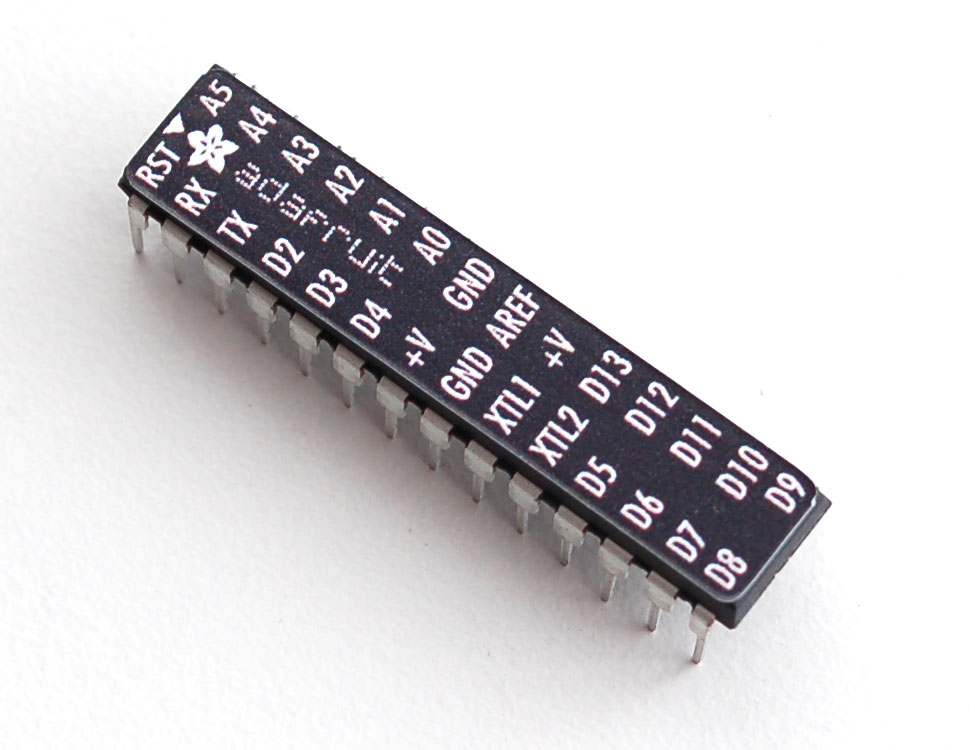
**CHAPTER 8.1**

**SYSTEM DESIGN**

**HARDWARE SELECTION**

In hand gesture controlled mouse following hardware components are used which plays an important role for giving desired output.

**Microcontroller:**

****

**Fig. 8.1.1**

Here, this project uses the P89V51RD2 because of following specifications:

Features:

• High Performance, Low Power AVR® 8-Bit Microcontroller

• Advanced RISC Architecture

– 131 Powerful Instructions – Most Single Clock Cycle Execution

– 32 x 8 General Purpose Working Registers

– 32K Bytes of In-System Self-Programmable Flash progam memory

– 6-channel 10-bit ADC in PDIP Package

– Easy USB connection with PC

– Programmable Serial USART

– 28-pin Pin DIP

• Operating Voltage:

– 1.8 - 5.5V for ATmega328P

• Speed Grade:

– 0 - 20 MHz @ 1.8 - 5.5V

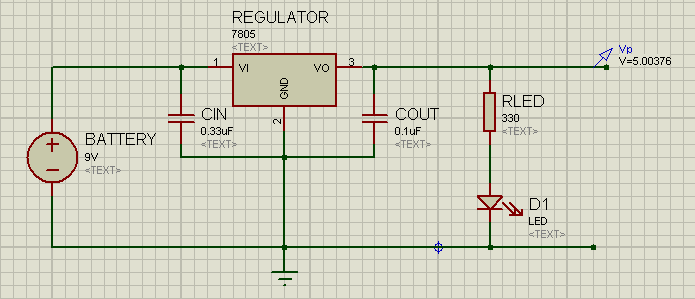
The main advantage of this microcontroller is that it has inbuilt ADC with 6 analog inputs which gives easy interfacing of sensors with AT328.

**Supply section:**

**7805** is a **voltage regulator** integrated circuit. It is a member of 78xx series of fixed linear voltage regulator ICs. The voltage source in a circuit may have fluctuations and would not give the fixed voltage output. The **voltage regulator IC** maintains the output voltage at a constant value. The xx in 78xx indicates the fixed output voltage it is designed to provide. 7805 provides +5V regulated power supply. Capacitors of suitable values can be connected at input and output pins depending upon the respective voltage levels.

Ci=330nF is required if regulator is located an appreciable distance from power supply ﬁlter.

Co=100nF which improves stability and transient response.



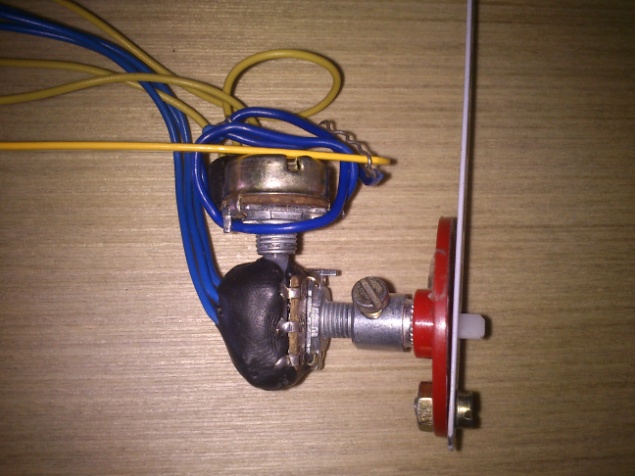
**Fig. 8.1.2**

**Potentiometer Sensors:**

A potentiometer informally a pot, is a three-[terminal](http://en.wikipedia.org/wiki/Terminal_(electronics)) [resistor](http://en.wikipedia.org/wiki/Resistor) with a sliding contact that forms an adjustable [voltage divider](http://en.wikipedia.org/wiki/Voltage_divider). If only two terminals are used, one end and the wiper, it acts as avariable resistor or rheostat.

Potentiometers are commonly used to control electrical devices such as volume controls on audio equipment. Potentiometers operated by a mechanism can be used as position [transducers](http://en.wikipedia.org/wiki/Transducer), for example, in a [joystick](http://en.wikipedia.org/wiki/Joystick).

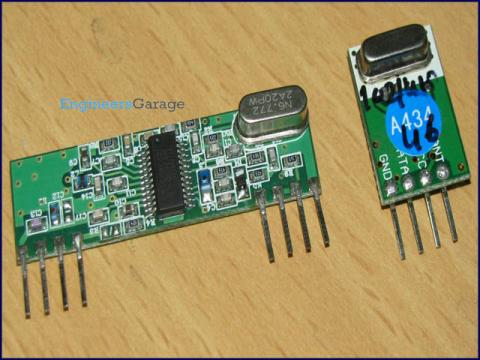
* These are used for both X and Y position control.
* It is also low cost sensor.
* There is no need of signal conditioning as required in case of many other sensors
* To get free movement in 2-D plane i.e. control of X and Y coordinate potentiometer are attached perpendicular to each other as shown in photograph.



**Fig. 8.1.3**

**RF Module:**

An RF Module (Radio Frequency Module) is a small electronic circuit used to transmit and/or receive radio signals on one of a number of carrier frequencies.



**Fig. 8.1.4**

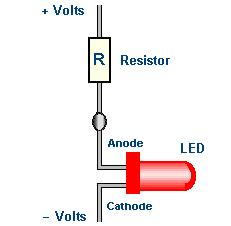
Several carrier frequencies are commonly used in commercially-available RF modules, including 433.92MHz, 315MHz, 868MHz and 915MHz. These frequencies are used because of national and international regulations governing the used of radio for communication.

As we are controlling mouse wirelessly we used 433.92MHz RF module.

Transmission through RF is better than IR (infrared) because IR mostly operates in line-of-sight mode, RF signals can travel even when there is an obstruction between transmitter & receiver.

**LED Indications:**

For display of ON/OFF condition of supply to circuit LED can be used. The calculations for resistor according to supply voltage are given:



**Fig. 8.1.5**

Applying KVL,

+Volts-IR-Vled-(-Volts)=0

Here we are using 5 volts supply.

Considering voltage drop across LED=1.8V

Minimum forward current required=10mA

R=(5-1.8)/(10mA)

=320ohm~330 ohm (as available practically in market)

**CHAPTER 8.2**

**SOFTWARE SELECTION**

**COMPILER & DEBUGGER:**

As the code is written in arduino language so arduino software is used.It has following specifications

[Developer(s)](http://en.wikipedia.org/wiki/Software_developer): Arduino Software

Written in [Java](http://en.wikipedia.org/wiki/Java_%28programming_language%29), [C](http://en.wikipedia.org/wiki/C_%28programming_language%29) and [C++](http://en.wikipedia.org/wiki/C%2B%2B)[Operating system](http://en.wikipedia.org/wiki/Operating_system)

[License](http://en.wikipedia.org/wiki/Software_license): [LGPL](http://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) or [GPL](http://en.wikipedia.org/wiki/GNU_General_Public_License) license

Website [arduino.cc](http://arduino.cc/en/Main/Software)

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software.The Arduino [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) is a [cross-platform](http://en.wikipedia.org/wiki/Cross-platform) application written in [Java](http://en.wikipedia.org/wiki/Java_%28programming_language%29), and is derived from the IDE for the [Processing programming language](http://en.wikipedia.org/wiki/Processing_%28programming_language%29).Users only need define two functions to make a runnable [cyclic executive](http://en.wikipedia.org/wiki/Cyclic_executive)program:

* setup(): a function run once at the start of a program that can initialize settings
* loop(): a function called repeatedly until the board powers off.

**SIMULATOR:**

**Proteus VSM co-simulation software** has been used to simulate our project circuit and it has following specifications:

ISIS lies at the heart of the Proteus system, and is far more than just another schematics package. It combines a powerful design environment with the ability to define most aspects of the drawing appearance. Whether your requirement is the rapid entry of complex designs for simulation and PCB layout, or the creation of attractive schematics for publication, ISIS is the tool for the job.

**PROCESSING:**

Processing software is used for controlling mouse pointer and to do click action on PC. More about Processing software as follows:

Processing is an open source programming language and environment for people who want to create images, animations, and interactions. Initially developed to serve as a software sketchbook and to teach fundamentals of computer programming within a visual context, Processing also has evolved into a tool for generating finished professional work. Today, there are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning, prototyping, and production. Features are as follows:

* Free to download and open source
* Interactive programs using 2D, 3D or PDF output
* For Linux, Mac OS X, and Windows
* Over 100 libraries extend the software into sound, video, computer vision, and more...
* Well [documented](http://processing.org/reference/), with many [books](http://processing.org/learning/books/) available

**LAYOUT DESIGNING:**

For layout designing we used the EAGLE (**E**asily **A**pplicable **G**raphical **L**ayout **E**ditor)software and it has following specifications:

Version 6.3.0 for Windows Light Edition

**PCB Layout** — PCB designs with an easy-to-use manual routing tools, shape-based auto-router and auto-placer.

**Schematic**:-Schematic Capture with multi-level hierarchy and export to PCB Layout,

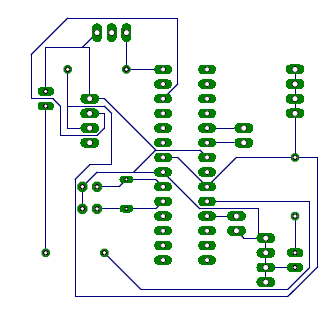
**Standard Libraries:** include almost all components with facility to create new component part in library.

**Electrical rule check (**[**ERC**](#49)**):** ERC to look for open pins, etc and use the messages generated to correct any errors.

**Design rule check (**[**DRC**](#49)**):** The DRC is performed in a board window, and checks the design for overlaps, distance violations etc.

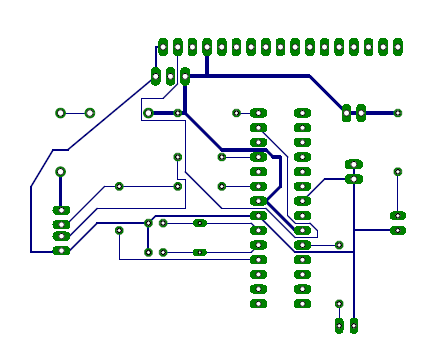
**CHAPTER 8.3**

**PCB layout for Transmitter:**

****

**Fig. 8.3.1**

**PCB layout for Receiver:**

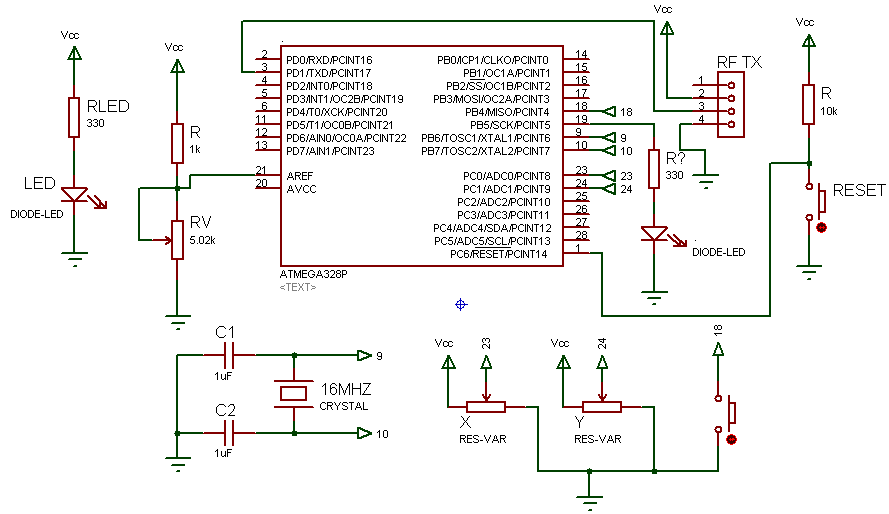
****

**Fig. 8.3.2**

**CHAPTER 8.4**

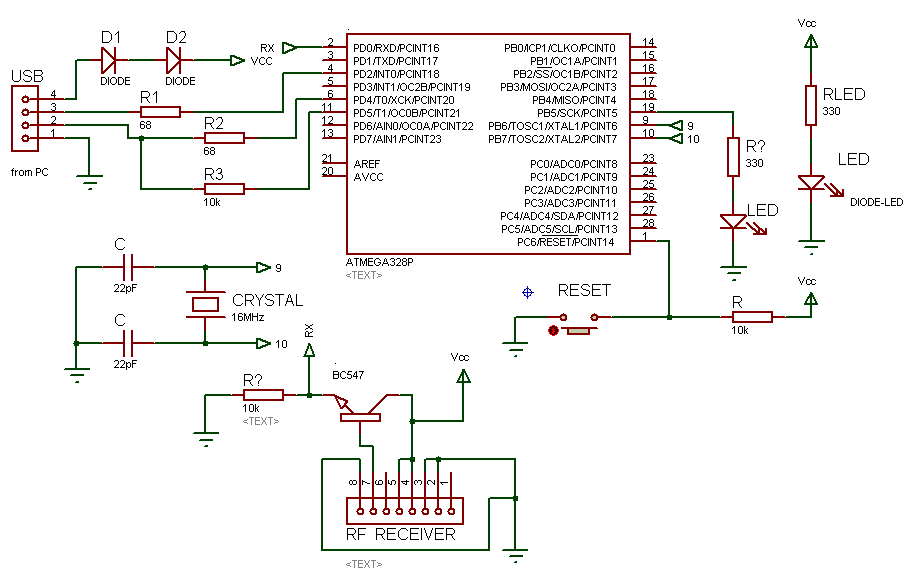
**DETAILED SYTEM DESIGN**

**Transmitter:**

****

**Fig. 8.4.1**

**Receiver:**

****

**Fig. 8.4.2**

**CHAPTER 9**

**COSTING**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. no. | Name of component | Qty. | Price in Rs. |
| 1 | Controller IC ATmega328P | 2 | 540 |
| 2 | RF module pair | 1 | 400 |
| 3 | Crystal Oscillator 16MHz | 2 | 40 |
| 4 | Transistor BC547 | 1 | 10 |
| 5 | Potentiometers | 3 | 60 |
| 6 | 7805 regulator IC | 1 | 15 |
| 7 | Battery 9v | 1 | 40 |
| 8 | Pin Header Strip | 2 | 20 |
| 9 | LEDs | 5 | 10 |
| 10 | Resistors | 10 | 30 |
| 11 | Capacitors | 6 | 20 |
| 12 | IC base | 2 | 20 |
| 13 | Metal strips | 5 | 80 |
| 14 | Diodes 1N4007 | 2 | 5 |
| 15 | Push Buttons | 4 | 20 |
| 17 | PCB Layout printing | 2 | 200 |
| 18 | Connecting wires | - | 60 |
|  | **TOTAL** |  | **1570/-** |

**Table. 9.1**

**CHAPTER 10**

**FUTURE SCOPE**

1. In future the technique can be used to control actions of keyboard.
2. Using the same principle as used in this project gaming experience through hand gesture could be more interesting for children.
3. To minimize the hardware image processing can be used.
4. The techniques can be used to create 3-D design using hand movements.
5. In future system can be upgraded for controlling mechanical movements (in robotics) for remote control operation.

**CHAPTER 11**

**MERITS, DEMERITS & APPLICATIONS**

**MERITS:**

1. Product is flexible due to wireless transmission.
2. No line of sight communication necessary.
3. Since AVR controller is used less hardware and software required. Therefore price is reduced.

**DEMERITS:**

1. Possibility of interference due to other RF devices in range.
2. Less security since no RFID feature as provided by other RF devices like CC2250.

**APPLICATIONS:**

1. It can be used as wireless keypad.
2. Simultaneously using two screens by adding new gesture.
3. Wireless gaming console control.
4. It can be used as wireless Kinect connection (XBOX 360).

**CHAPTER 12**

**CONCLUSION**

Thus we have successfully implemented RF module based Hand Gesture Controlled Game. While synthesizing this technology in project, we learnt lot about wireless data transmission protocol, coding techniques for safe data transmission. Again with this project we learnt module interfacing with AVR, signal conditioning, Communication with PC using USB (Universal Serial Bus). In today’s world wireless technology is on verge of upgradation to its extent. So safe wireless data transmission plays important role in upcoming technologies. Thus we have used same technology in our project and also we learnt lot while overcoming the obstruction that we faced during synthesizing technology

**REFERENCES**

**Books referred:**

* **From Pearson education,“Mechatronics” – by W. Bolton, 3rdedition.**
* **From Pearson education, “AVR microcontroller and Embedded System”—by M. Mazidi.**
* **From Oxford University Press, “Modern Digital and Analog Communication” by B.P.Lathi.**

**Websites referred:**

[**http://arduino.cc**](http://arduino.cc)

**www.ti.com**

**http://www.processing.org**

**http://**[**www.youtube.com**](http://www.youtube.com)

[**http://en.wikipedia.org**](http://en.wikipedia.org)

[**www.atmel.com**](http://www.atmel.com)

**TED Innovations**

**http://digitalcommunication.blogspot.**[**com**](http://www.youtube.com)