Smart Phone Control Peck and Place Robotics Arm With Robot Car

SONARGAON UNIVERSITY (SU)

Supervised By

Md. Rakibul Alam

Lecturer
Department of EEE
Sonargaon University (SU)

<u>Submitted By</u>

MD. OMAR FARDOUS	**ID: EEE1802014053**
NAYON KUMAR	**ID: EEE1802014064**
RAZUAL KARIM RAJU	**ID: EEE1802014067**
JUBAYER HOSSAIN	**ID: EEE1802014069**
SAHIDUL ISLAM	**ID: EEE1802014070**
SAMIUR RAHMAN	**ID: EEE1502005171**
MD.SAMIUL ISLAM	**ID: EEE1602008234**

Department of Electrical & Electronic Engineering (EEE)
Sonargaon University (SU)
147/I, Panthapath,
Dhaka-1215, Bangladesh

Date of Submission: January 2022

Declaration

We hereby declare that this submission is our own work and to the best of our knowledge it contains no materials which are exactly same which were previously published anywhere in print or soft.

....

MD. OMAR FARDOUS

…..

NAYON KUMAR

....

MD.RAZUAL KARIM RAJU

…...

MD.JUBAYER HOSSAIN

....

MD.SAMIUL ISLAM

....

SAMIUR RAHMAN

....

SAHIDUL ISLAM

Under Supervision of

....

Md. Rakibul Alam

Lecturer

Department of EEE

Sonargaon University (SU)

Certification

B.Sc. Project entitled:

“Smartphone Control Peck and Place Robotics Arm With Robot Car”

Done under my supervision, meets acceptable presentation that is standard and can be submitted for evaluation to the department of Electrical and Electronic Engineering in partial fulfillment of the requirement for the degree of Bachelor of Science (B.Sc.) in Electrical and Electronic Engineering.

.

Supervisor

……………………………………………

Md. Rakibul Alam

Lecturer

Department of Electrical and Electronic Engineer (EEE)

Sonargaon University (SU)

ACKNOWLEDGEMENT

The report titled as on **"Smartphone Control Peck and Place Robotics Arm With Robot Car"** has been prepared to fulfill the requirement of our practicum program. In the process of doing and preparing our practicum report, we would like to pay our gratitude to some persons for their enormous help and vast co-operation.

At first, we would like to show our gratitude to the University authority to permit us to do our practicum. Specially, we would like to thank to our honorable teacher **Md. Rakibul Alam,** Lecturer, Department of Electrical & Electronics Engineering, SU– Sonargaon University, Dhaka, for his valuable and patient advice, sympathetic assistance, co-operation, contribution of new idea. Deep theoretical and hardware knowledge & keen interest of our supervisor in this field influenced us to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

Finally, we would like to thanks again to the respected **Vice- Chancellor of SU, Professor Dr. Md. Abul Bashar** also thanks to **Head of Department of SU , Electrical & Electronics Engineering, Professor Dr. M. Bashir Uddin** because they are designated such an environment for learning through which we got the opportunity to acquire knowledge under Bsc in EEE program, and that will be very helpful for our prospective career.

We are, indeed, grateful to all those from whom we got sincere cooperation and help for the preparation of this report

ABSTRACT

In recent years the industry and daily routine works are found to be more attracted and implemented through automation via Robots. The smart phone control pick and place robot is one of the technologies in manufacturing industries which is designed to perform pick and place operations. The system is so designed that it eliminates the human error and human intervention to get more precise work. There are many fields in which human intervention is difficult but the process under consideration has to be operated and controlled this leads to the area in which robots find their applications. Literature suggests that the pick and place robots are designed, implemented in various fields such as; any thing taken from one place to another, in bottle filling industry, packing industry, used in surveillance to detect and destroy the bombs etc. The project deals with implementing an pick and place robot using RoboArduino for any pick and place functions. The pick and place robot so implemented is controlled using bluetooth signal.

TABLE OF CONTENTS

Chapter One
1. INTRODUCTION

1.1 Introduction

Robot is associate integral half in automating the versatile producing system that's greatly in demand lately.Robots are currently quite a machine, as robots became the answer of the longer term as price labor wages and customers' demand.Robots is classified into completely different classes counting on their operate and therefore the market wants they're designed for. Here it’s determining 2 major categories of robots, industrial robots and repair robots. in line with the Robotic Industries Association, associate industrial golem is associate mechanically controlled, reprogrammable, useful manipulator programmable in 3 or a lot of axes which can be either mounted in site or mobile to be used in industrial automation applications. Research and development of future robots is movingly at a awfully speedy face owing to the perpetually up and upgrading of the standard standards of product. Golem and automation is used so as to exchange human to perform those tasks that are routine dangerous, uninteresting and in a very dangerous space.Now a day’s within the world of advanced technology, automation greatly will increase production capability; improve product quality and lower cost.Robots are indispensable in several producing interties.Robots are designed and programmed to be job specific.We used programming to implement our project.We have used arduino as the controller.

1.2 Objective of The Project

The main objective of this project is to“ Smart phone Control Peck and Place Robotics Arm with Robot Car’’ system Project for The main purpose is to create a smart system so that we can take many benefits from one project.

- Reduce costs from conventional systems.
- Convenience in control via smartphone.
- Make it usable for all.
- To reduce human effort in industries.
- To make a manual automation.

1.3 Background of the Project

The name of our project is Smartphone Control Peck and Place Robotics Arm With Robot Car .This is a modern project. The main goal of our project is to move a robot car and product from one place to another.In the industry we see this is done using CNC technology which is very costly.We have created a robot car and and robotics arm that can transfer any product under the command of a mobile phone.Here we have done the work with servo motor, Bluetooth module and microcontroller..

1.4 Robotic Arm Definition

A robotic arm is a robot manipulator, usually programmable, with similar functions to a human arm. The links of such a manipulator are connected by joints allowing either rotational motion (such as in an articulated robot) or translational (linear) displacement .The links of the manipulator can be considered to form a kinematic chain. The business end of the kinematic chain of the manipulator is called the end effectors and it is analogous to the human hand. The end effectors can be designed to perform any desired task such as welding, gripping, spinning etc., depending on the application. The robot arms can be autonomous or controlled manually and can be used to perform a variety of tasks with great accuracy. The robotic arm can be fixed or mobile (i.e. wheeled) and can be designed for industrial or home applications. The wireless mobile robots also have been developing in previous years..

1.5 Mechanical Design

Fig :1 illustrate the designing of the robot with (a) main structure arm robot (b) arm robot design . Robot arm wills have 5 outputs which consist of the base, shoulder, elbow, wrist and grippe .For arm robot, servo holder such as bracket and U joint is use to place the motor and it is made from aluminums because it is lightweight.

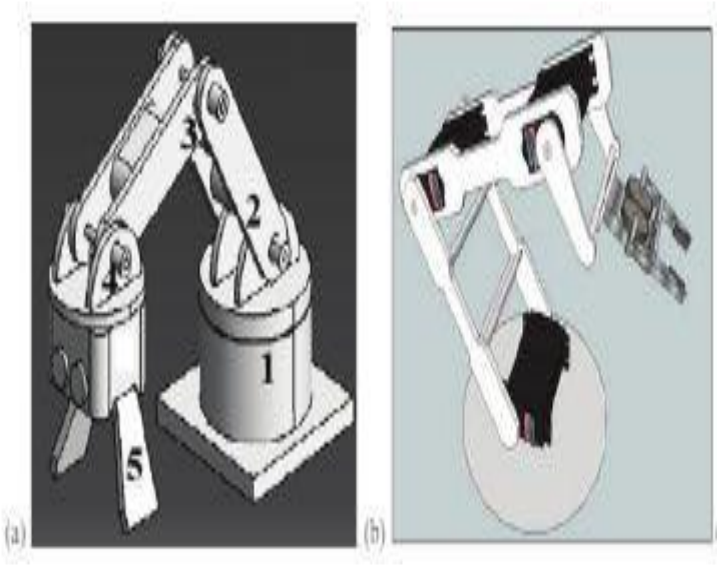


Fig- 1.1: Robotic arm design (a) Main structure arm robot (b) Robot arm design

The robot gripper is also made of aluminum because of the same reason as the main robot arm structure.The mobile robot, it has dimension (28 x 15.5 x 8.5) cm which is the length, width, and height respectively,as shown in Fig :1.Acrylic is used as the main material for mobile robotic arm because it is easy to be formed, cheap, strong and can bear the motor weight and movement.There are 3 servo motor and servo wheel attached to this system.

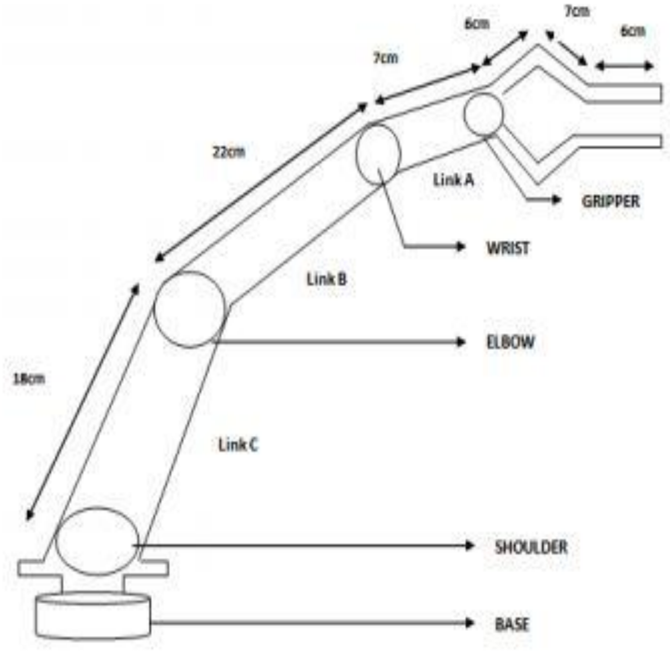


Fig- 1.2: Robotic arm design

1.6 Methodology

- Collection of data from books and internet.
- Collection of required components from local market and online shop.
- The concept of the design is based on using a microcontroller for processing.
- Designing a schematic for creating the circuit board.
- Programming for the microcontroller (ATMEGA328P) to controlled the energy saving system.
- Setting all the components in a PCB board, Car Chassis and soldering .
- Turn on the device experimental.
- And finally, run the system using microcontroller.

1.7 Motivation For Project

A poorly motivated team has been known to unravel even the best project plan. A good project manager needs to know how to harness the initial excitement that comes with starting a project and use it to maintain motivation – leading to success throughout the project's lifecycle. We now know that contemporary project managers need to be more than just schedulers and contract managers. They need excellent skills in managing those complex human elements that have the potential to bring any project down.

Chapter Two
2. THEORETICAL BACKGROUND

2.1 Introduction

The design analysis of an Android Controlled “Pick and Place” Robotic vehicle has been presented in this paper.This work unravels the fact that man would always want to adhere to safety precautions at workplace and even in its environment, to be able to handle some specific tasks, like sending the robotic vehicle to hazardous environment to obtain samples for chemical analysis. It is a microcontroller based control system which works in alliance with Android Application. It can be accessed by android application and the application can control the movement of vehicle as well as its robotic arms.This system comprises of a Bluetooth module which work as the receiver for vehicle. This sends commands to the microcontroller which execute according to the signals received by Bluetooth.In this work, the design of a robot is presented which will move around in four directions and is equipped with gripper for pick and place operation.These operations will be controlled by a user friendly interface present on operator’s mobile phone. Depending upon the button clicked on the application, proper motional commands are given to robot by microcontroller. This project is in aimed to design and develop a mobile robot which can move according to the button pressed on App.

This prototype can be employed in chemical industry for handling of chemical materials of hazardous nature, or for movement of heavy objects in any industry.Robots are indispensable in many manufacturing industries. The reason is that the cost per hour to operate a robot is a fraction of the cost of the human labour needed to perform the same function. More than this, once programmed, robots repeatedly perform functions with a high accuracy that surpasses that of the most experienced human operator. Human operators are, however, far more versatile. Humans can switch job tasks easily. Robots are built and programmed to be job specific. Today's most advanced industrial robots will soon become dinosaurs. Robots are in the infancy stage of their evolution. As robots evolve, they will become more versatile, emulating the human capacity and ability to switch job tasks easily.While the personal computer has made an indelible mark on society, the personal robot hasn't made an appearance. Obviously there's more to a personal robot than a personal computer. Robots require a combination of elements to be effective: sophistication of intelligence, movement, mobility, navigation, and purpose. In the beginning, personal robots will focus on a singular function (job task) or purpose.Building a useful personal robot is very difficult.But recent days Robots occupies a good place in community.The main functioning of robotics in recent days are.

2.2 Literature Survey

John Iovine , in this book various aspects of designing a Robot is described. It deals with different types of Arm design, controlling techniques, vehicle design etc… In the remaining references different types controlling technique are explained. The Idea about the Android software „Blue control‟ and its application in controlling Robotics is described in the Google play store website. From all these we find a different controlling technique using android device and software with a different approach for soft catching arm (electronic pressure sensor).

2.3 Android Application

Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language that has complete access to the Android APIs.Java may be combined with C/C++, together with a choice of non-default runtimes that allow better C++ support; the Go programming language is also supported since its version 1.4, which can also be used exclusively although with a restricted set of Android APIs. The SDK includes a comprehensive set of development tools, including a debugger, software libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available,including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled an framework based on Apache Cordova for porting Chrome HTML 5 web applications to Android, wrapped in a native application shell.

Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's APK (Android application package) file, or by downloading them using an application store program that allows users to install, update, and remove applications from their devices. Google Play Store is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services software.We in this project are developing the software in the Eclipse Software. The development of the app is divided into three parts:

(1) Android Manifest : This section of the code deals with the permissions that the application has to take from the android application device or the mobile phone. Since, the application is using the Bluetooth feature of the device, it takes permission for the usage.

(2) XML File : This section of the code deals with the layout design of the application. For example, the position of the buttons and their function is incorporated in this section of the application coding.

(3) JAVA File : This section of the code incorporates main function of the application. This section deals with the main functions being performed by the application i.e. sending of ASCII codes to the Bluetooth module connected in the circuit. The communication is carried out using the UART feature of the Bluetooth module. Hence, the mode of communication is serial communication.

2.4 Hazardous duty

Without risking human life or limb, robots can replace humans in some hazardous duty service . Take for example bomb disposal. Robots are used in many bomb squads across the nation. Typically these robots resemble small armoured tanks and are guided remotely by personnel using video cameras attached to the front of the robot.Robotic arms can grab a suspected bomb and place it in an explosion-proof safe box for detonation and/or disposal.Similar robots can help clean up toxic wastes. Robots can work in all types of polluted environments, chemical as well as nuclear. They can work in environments so hazardousthat an unprotected human would quickly die. The nuclear industry was the first to develop and use robotic arms for handling radioactive materials. Robotic arms allowed scientists to be located in clean, safe rooms operating controls for the robotic arms located in radioactive rooms

Chapter Three

3. DESIGN AND DEVELOPMENT

3.1 Introduction

The project is designed to develop a pick n place robotic arm. The project is aimed to design and develop a pick and place robotic arm with robot car. The project will include development of an android app and mobile robot which will be able to move objects from one place to another. Thus, controlling the arm remotely. Remote operation is achieved by any smart-phone/Tablet etc. This prototype will pave path for real life models using the same concept.

The vehicle will be controlled by the android application device at the transmitting end, which will send signal i.e. ASCII code, to Bluetooth module (HC-05) which is employed as an interface between Mobile and project (robot). The microcontroller is coded using IDE Studio for controlling of the motors, both to move the robotic arm the objects through the arm.

At the transmitting end using android application device, commands are sent to the receiver to control the movement of the robotic arm either to move forward, backward and left or right etc. At the receiving end four motors are interfaced to the microcontroller where two for them are used for arm movement of the robot while the other one are for the body movement. The android application device transmitter acts as a remote control that has the advantage of adequate range, while the receiver end Bluetooth device is fed to the microcontroller to drive DC motors via motor driver IC for necessary work. Remote operation is achieved by any smart-phone/Tablet etc., with Android OS; upon a GUI (Graphical User Interface) based touch screen operation. The main advantage of this robot is its soft catching arm that is designed to avoid extra pressure on the suspected object for safety reasons.

The main objective of this project is to design and implementation a ‘Smartphone Control Peck and Place Robotics Arm with Robot Car’ prototype by using Arduino MEGA, bluetooth module, and servo motor and to achieve the goal of this project.

3.2 Block Diagram of This Project

In this proposed system we have used the Arduino MEGA development board and Bluetooth module. In this project the system is been totally operated by power supply. The main aim of ‘Smart phone Control Peck and Place Robotics Arm with robot car’. In above block diagram there are Bluetooth module , Arduino Mega ,Servo motor , Gear motor and Motor Driver Module .Microcontroller continuously check the output of sensor and gives signal to the iot system circuit which drives the apps.

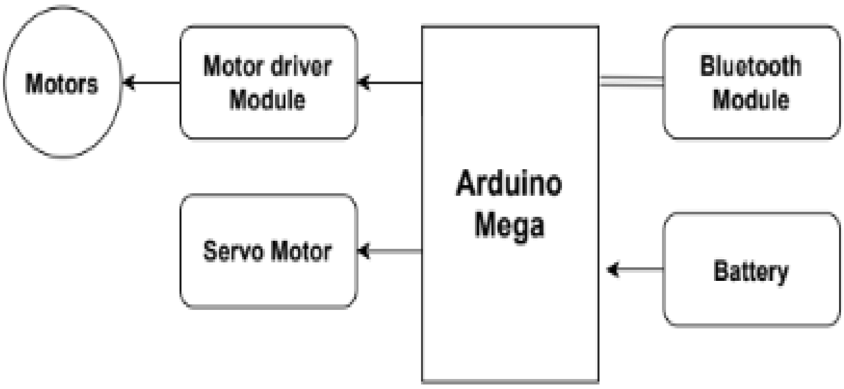


Fig: 3.1 Block Diagram

3.3 Circuit Diagram of This Project

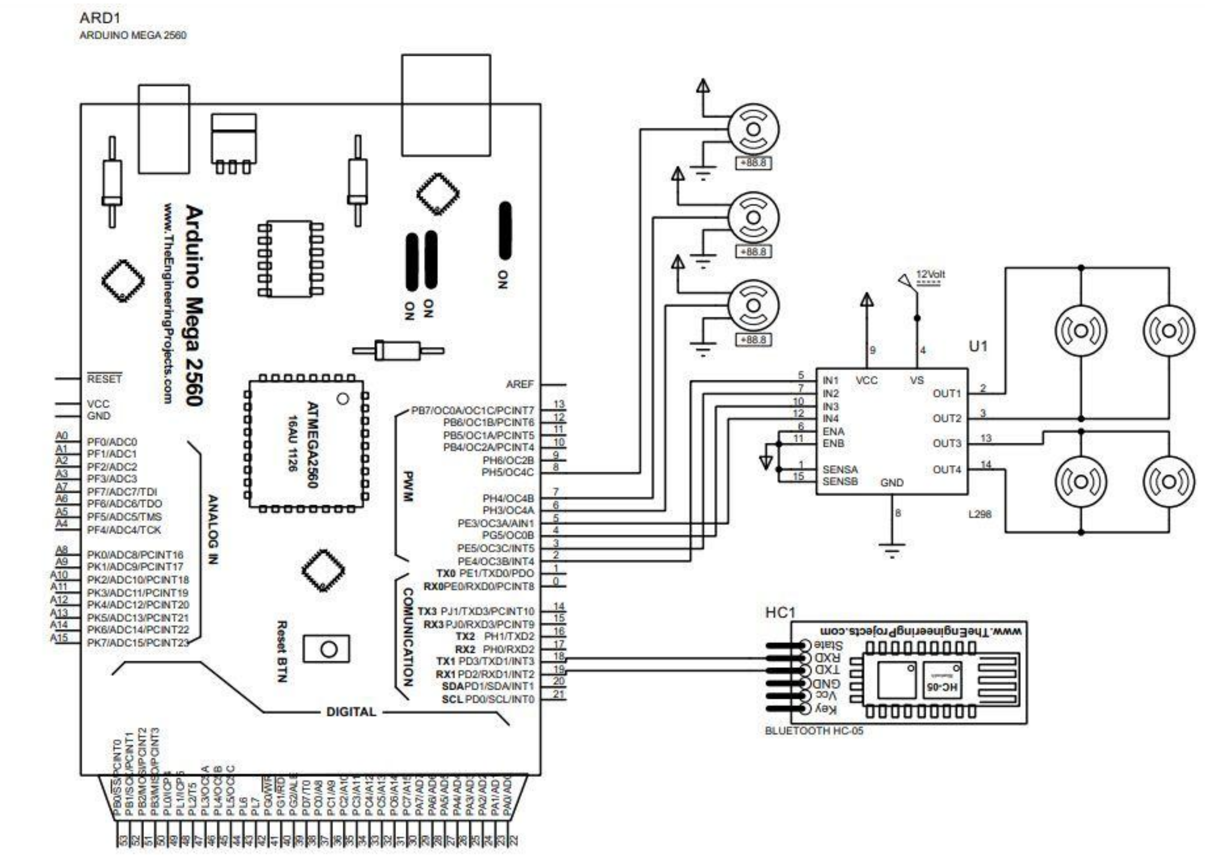


Fig: 3.2 Circuit Diagram

3.3.1 Working Principle of Circuit Diagram

First we will connect the mobile phone with the Bluetooth module in the project.We will give commands using a software on the mobile phone. The command sent from the mobile phone will receive the bluetooth module in the project. The Bluetooth module commands will be delivered to the project programming device arduino. The arduino controller will operate the motor driver module and servo motor to perform the command. Through the motor driver module we indicate the gear motor for operating the robot car and the servo motor is basically a robotics arm, which can move any product according to the command and .

3.4 The list of devices used in the project is given below

- Arduino MEGA.
- LM2596 Voltage Converter.
- Bluetooth Module.

- Servo Motor.
- Gear Motor.
- Battery.
- Motor Driver Module.

3.5 Arduino MEGA

The Arduino Mega is a microcontroller board based on the ATmega2560 . It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller.

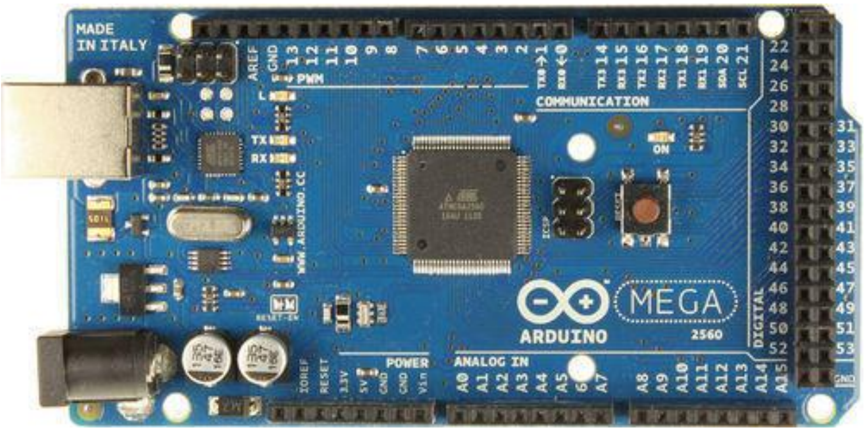


Fig-3.3: Arduino MEGA

3.5.1 Background

The Arduino project started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of $100, a considerable expense for many students. In 2003 Hernando Barragán created the development platform wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language.

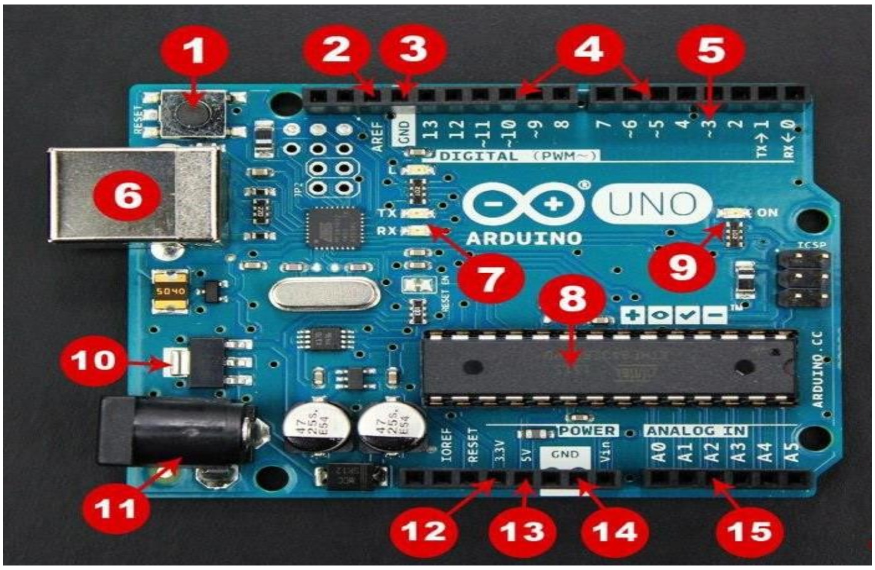


Fig-3.4: Arduino UNO

The project goal was to create simple, low-cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it Arduino. Early arduino boards used the FTDI USB-to-serial driver chip and an ATmega168.The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

3.5.2 History

The Arduino project was started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy.At that time, the students used a BASIC Stamp microcontroller at a cost of $50, a considerable expense for many students. In 2003 Hernando Barragán created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas. Casey Reas is known for co-creating, with Ben Fry, the Processing development platform. The project goal was to create simple, low cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it Arduino.

The initial Arduino core team consisted of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis, but Barragán was not invited to participate.

Following the completion of the Wiring platform, lighter and less expensive versions were distributed in the open-source community.

It was estimated in mid-2011 that over 300,000 official Arduinos had been commercially produced,and in 2013 that 700,000 official boards were in users' hands.

In October 2016, Federico Musto, Arduino's former CEO, secured a 50% ownership of the company. In April 2017, Wired reported that Musto had "fabricated his academic record.... On his company's website, personal LinkedIn accounts, and even on Italian business documents, Musto was until recently listed as holding a PhD from the Massachusetts Institute of Technology. In some cases, his biography also claimed an MBA from New York University." Wired reported that neither university had any record of Musto's attendance, and Musto later admitted in an interview with Wired that he had never earned those degrees.

Around that same time, Massimo Banzi announced that the Arduino Foundation would be "a new beginning for Arduino."But a year later, the Foundation still hasn't been established, and the state of the project remains unclear.

The controversy surrounding Musto continued when, in July 2017, he reportedly pulled many Open source licenses, schematics, and code from the Arduino website, prompting scrutiny and outcry.

In October 2017, Arduino announced its partnership with ARM Holdings (ARM). The announcement said, in part, "ARM recognized independence as a core value of Arduino ... without any lock-in with the ARM architecture." Arduino intends to continue to work with all technology vendors and architectures.

3.5.3 Hardware

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available.

Although the hardware and software designs are freely available under copyleft licenses, the developers have requested the name Arduino to be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product.Several Arduino-compatible products commercially released have avoided the project name by using various names ending in -duino.

An early Arduino board with an RS-232 serial interface (upper left) and an Atmel ATmega8 microcontroller chip (black, lower right); the 14 digital I/O pins are at the top, the 6 analog input pins at the lower right, and the power connector at the lower left.

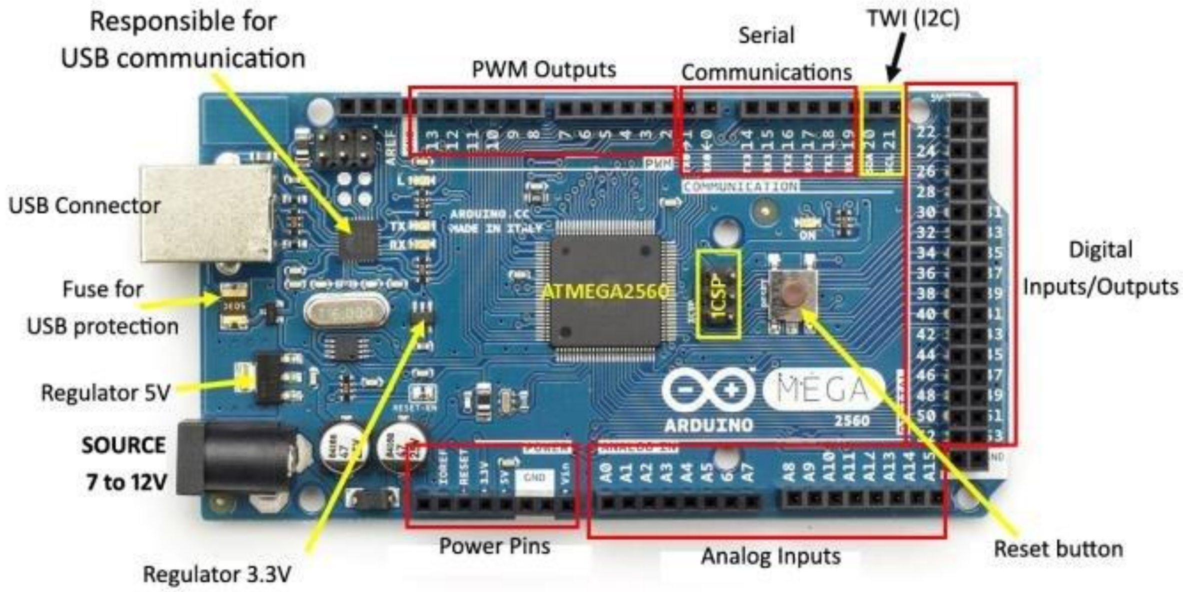


Fig-3.5: Pinout of Arduino MEGA

Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012.The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an I²C serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the LilyPad, run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions.

Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default bootloader of the Arduino UNO is the optiboot bootloader.Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between RS-232 logic levels and transistor–transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE, standard AVR in-system programming (ISP) programming is used.

An official Arduino Uno R2 with descriptions of the I/O locations

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila,Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

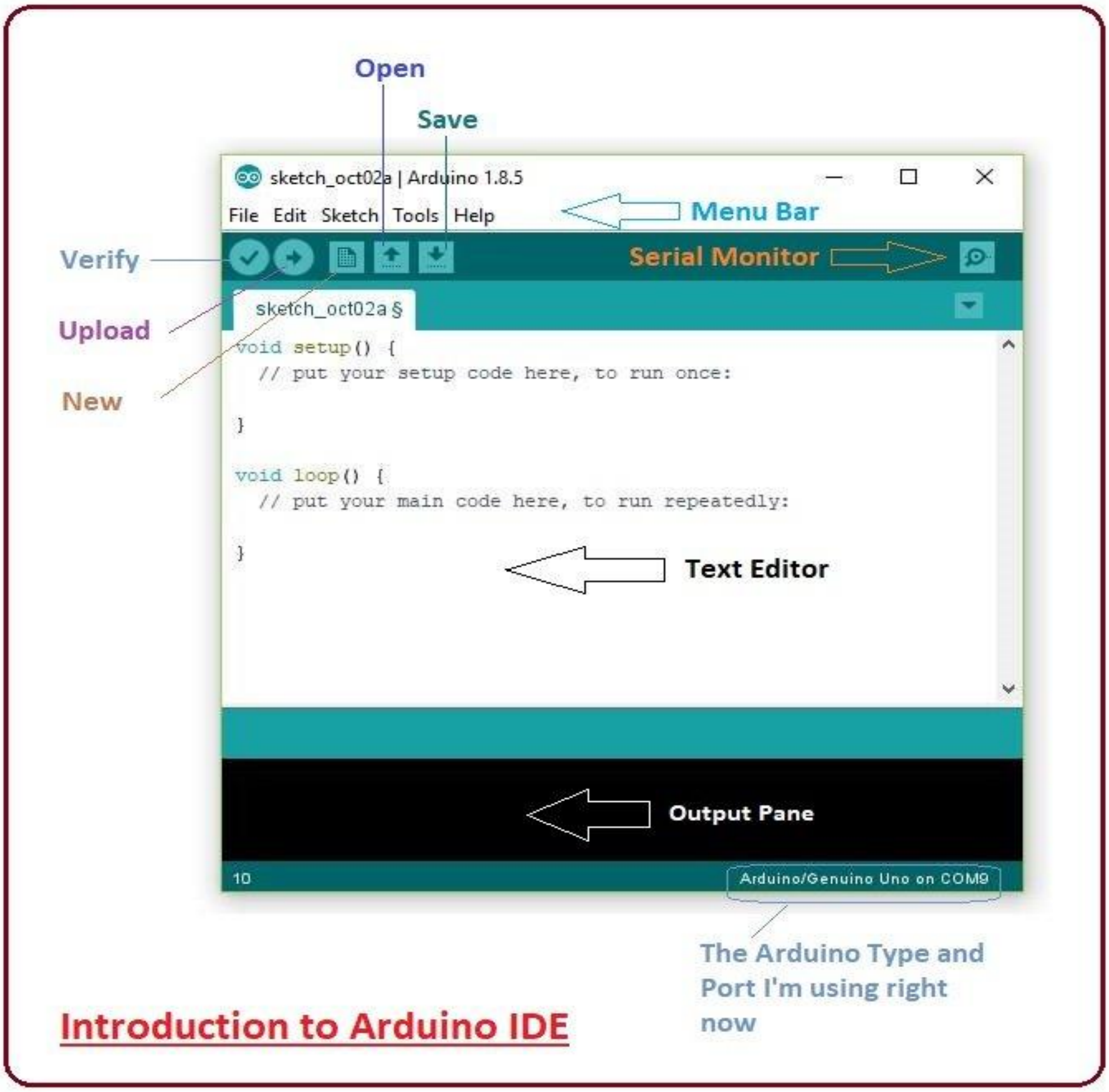
Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.

3.5.4 Software

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer).

1.1.1 3.5.5 Integrated Development Environment

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program AVR dude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.



Open
Save
sketch_oct02a | Arduino 1.8.5
File Edit Sketch Tools Help
Menu Bar
Verify
Upload
New
Serial Monitor
sketch_oct02a §
Text Editor
Output Pane
10
Arduino/Genuino Uno on COM9
The Arduino Type and Port I'm using right now
Introduction to Arduino IDE

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Fig-3.6: IDE software

1.1.2 3.5.6 Sketch

A sketch is a program written with the Arduino IDE.Sketches are saved on the development computer as text files with the file extension .ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde.

A minimal Arduino C/C++ program consists of only two functions:

setup(): This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. It is analogous to the function main().

loop(): After setup() function exits (ends), the loop() function is executed repeatedly in the main program. It controls the board until the board is powered off or is reset. It is analogous to the function while(1).

3.5.7 Blink example

Most Arduino boards contain a light-emitting diode (LED) and a current limiting resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions.A typical program used by beginners, akin to Hello, World!, is "blink", which repeatedly blinks the on-board LED integrated into the Arduino board. This program uses the functions pinMode(), digitalWrite(), and delay(), which are provided by the internal libraries included in the IDE environment. This program is usually loaded into a new Arduino board by the manufacturer.

1.1.3 3.5.8 Technical Specifications

I. Microcontroller : ATmega2560
II. Operating Voltage : 5V
III. Input Voltage (recommended) : 7-12V
IV. Input Voltage (limit) : 6-20V
V. Digital I/O Pins : 54 (of which 15 provide PWM output)
VI. Analog Input Pins : 16
VII. DC Current per I/O Pin : 20 mA
VIII. DC Current for 3.3V Pin : 50 mA
IX. Flash Memory : 256 KB of which 8 KB used by bootloader
X. SRAM : 8 KB
XI. EEPROM : 4 KB
XII. Clock Speed : 16 MHz
XIII. Length : 101.52 mm
XIV. Width : 53.3 mm
XV. Weight : 37 g

3.5.9 Applications

I. Arduboy, a handheld game console based on Arduino.
II. Arduinome, a MIDI controller device that mimics the Monome.
III. Ardupilot, drone software and hardware.
IV. ArduSat, a cubesat based on Arduino.
V. C-STEM Studio, a platform for hands-on integrated learning of computing, science, technology, engineering, and mathematics (C-STEM) with robotics.
VI. Data loggers for scientific research.
VII. OBDuino, a trip computer that uses the on-board diagnostics interface found in most modern cars.
VIII. OpenEVSE an open-source electric vehicle charger.
IX. XOD, a visual programming language for Arduino.

3.6 Buck Module

The LM2596 series of regulators are monolithic integrated circuits that provide all the active functions for a step-down (buck) switching regulator, capable of driving a 3-A load with

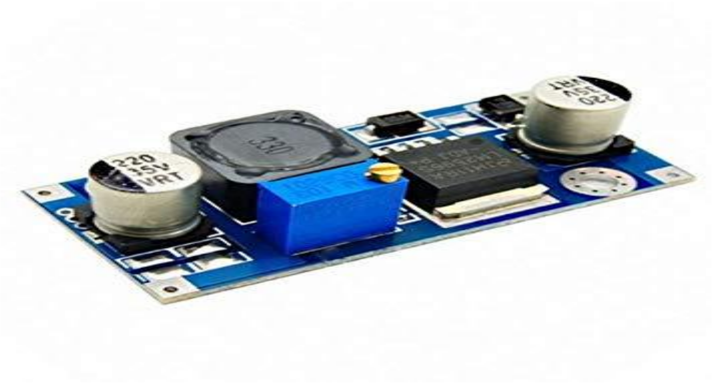


Fig: 3.7 LM2596 Voltage Converter

excellent line and load regulation.These devices are available in fixed output voltages of 3.3 V, 5 V, 12 V, and an adjustable output version. Requiring a minimum number of external components, these regulators are simple to use and include internal frequency compensation, and a fixed frequency oscillator. The LM2596 series operates at a switching frequency of 150 kHz, thus allowing smaller sized filter components than what would be required with lower frequency switching regulators. Available in a standard 7-pin TO-220 package with several different lead bend options, and a 7-pin TO-263 surface mount package.

3.7 Bluetooth Module

HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module,designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller or PC.HC-05 Bluetooth module provides switching mode between master and slave mode which means it able to use neither receiving nor transmitting data.

3.7.1 Introduction

This Bluetooth module can easily achieve serial wireless data transmission. Its operating frequency is among the most popular 2.4GHz ISM frequency band (i.e. Industrial, scientific and medical).It adopts Bluetooth 2.0+EDR standard. In Bluetooth 2.0, signal transmit time of different devices stands at a 0.5

seconds interval so that the workload of bluetooth chip can be reduced substantially and more sleeping time can be saved for bluetooth. This module is set with serial interface, which is easy to use and simplifies the overall design/development cycle.

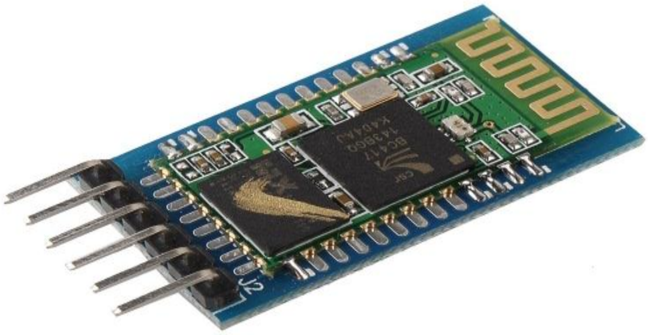


Fig: 3. 8: Bluetooth Module HC-05

3.7.2 Specification

I. Bluetooth protocol: Bluetooth 2.0+ EDR standard
II. USB protocol: USB v1.1/2.0
III. Operating frequency: 2.4GHz ISM frequency band
IV. Modulation mode: Gauss frequency Shift Keying
V. Transmit power: ≤ 4dBm, second stage
VI. Sensitivity: ≤-84dBm at 0.1% Bit Error Rate
VII. Transmission speed: 2.1Mbps(Max)/160 kbps(Asynchronous)
VIII. 1Mbps/1Mbps(Synchronous)
IX. Safety feature: Authentication and encryption
X. Supported configuration: Bluetooth serial port (major and minor)
XI. Supply Voltage: +3.3 VDC 50mA
XII. Operating temperature: -20 to 55℃
XIII. Size: 36.5*16mm
XIV. Weight: 4g

3.7.3 Pin Description

'VCC: positive pole of the power source.

GND: Ground.

TXD: serial interface, transmitting terminal.

RXD: serial interface, receiving terminal.

3.7.4 Test method

HC-05	Sun Founder Uno Board
VCC	5V
GND	GND
TXD	RX0 (pin0)
RXD	TX0 (pin1)

Table 01:- Pinout 0f bluetooth module HC- 05

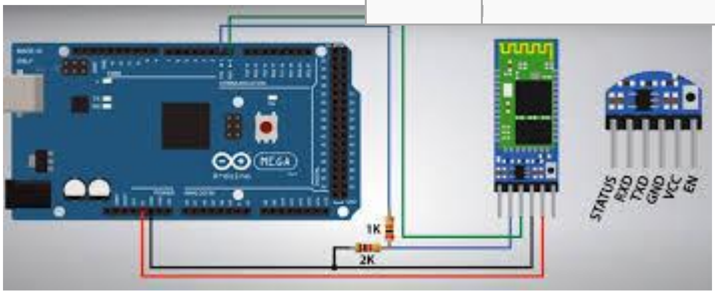


Fig: 3.9 Bluetooth Module With Arduino MEGA

You need to download a bluetooth debugging assistant for mobile phone to connect your phone with the Bluetooth HC-05 module. We've provided the tool in Android (SPP). Just download the package on your smart phone and decompress. Then you can see a file BlueSpp. Enable Bluetooth on your phone and install the BlueSpp. Open the installed.

3.8 Servo Motor

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration.It consists of a suitable motor coupled to a sensor

Fig: 3.10 Servo Motor

for position feedback.It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors.Servomotors are not a specific class of motor, although the term servomotor is often used to refer to a motor suitable for use in a closed-loop control system.MG996 Tower Pro Copper Servo Motor.

3.8.1 Specification

I. Power Supply: Through External Adapter.
II. Weight : 55g
III. Operating Speed : 0.17sec / 60 degrees (4.8V no load)
IV. Operating Speed : 0.13sec / 60 degrees (6.0V no load)
V. Stall Torque : 13 kg-cm (180.5 oz-in) at 4.8V
VI. Stall Torque : 15 kg-cm (208.3 oz-in) at 6V
VII. Operation Voltage : 4.8 - 7.2Volts
VIII. Gear Type: All Metal Gears

3.9 Gear Motor

A gear motor is a specific type of electrical motor that is designed to produce high torque while maintaining a low horsepower, or low speed, motor output. Gear motors can be found in many different applications, and are probably used in many devices in your home. Gear motors are commonly used in devices such as can openers, garage door openers, washing machine time control knobs and even electric alarm clocks.

Fig-3.11 Gear Motor

Common commercial applications of a gear motor include hospital beds, commercial jacks, cranes and many other applications that are too many to list.

3.9.1 Gear Motor Work

A gear motor can be either an AC (alternating current) or a DC (direct current) electric motor. Most gear motors have an output of between about 1,200 to 3,600 revolutions per minute (RPMs). These types of motors also have two different speed specifications: normal speed and the stall-speed torque specifications.

Gear motors are primarily used to reduce speed in a series of gears, which in turn creates more torque. This is accomplished by an integrated series of gears or a gear box being attached to the main motor rotor and shaft via a second reduction shaft. The second shaft is then connected to the series of gears or gearbox to create what is known as a series of reduction gears. Generally speaking, the longer the train of reduction gears, the lower the output of the end, or final, gear will be.

An excellent example of this principle would be an electric time clock (the type that uses hour, minute and second hands). The synchronous AC motor that is used to power the time clock will usually spin the rotor at around 1500 revolutions per minute. However, a series of reduction gears is used to slow the movement of the hands on the clock.

For example, while the rotor spins at about 1500 revolutions per minute, the reduction gears allow the final secondhand gear to spin at only one revolution per minute. This is what allows the secondhand to make one complete revolution per minute on the face of the clock.

3.9.2 Gear Motors and Increased Force

Gear motors are commonly used in commercial applications where a piece of equipment needs to be able to exert a high amount of force in order to move a very heavy object. Examples of these types of equipment would include a crane or lift Jack.

If you've ever seen a crane in action, you've seen a great example of how a gear motor works. As you have probably noticed, a crane can be used to lift and move very heavy objects. The electric motor used in most cranes is a type of gear motor that uses the basic principles of speed reduction to increase torque or force.

Gear motors used in cranes are usually specialty types that use a very low rotational output speed to create incredible amounts of torque. However, the principles of the gear motor used in a crane are exactly the same as those used in the example electric time clock. The output speed of the rotor is reduced through a series of large gears until the rotating, RPM speed, of the final gear is very low. The low RPM speed helps to create a high amount of force which can be used to lift and move the heavy objects.

3.10 Lipo Battery

A lithium polymer battery, or more correctly lithium-ion polymer battery (abbreviated as LiPo, LIP, Li-poly, lithium-poly and others), is a rechargeable battery of lithium-ion technology using a polymer electrolyte instead of a liquid electrolyte. High conductivity semisolid (gel) polymers form this electrolyte. Power Lithium Polymer Battery 11.1V 2200mAh 3S 30C for Drones, Racing Cars, Helicopters.

Fig-3.12: Lipo Battery

3.10.1 Features

1) High energy density

2) High working voltage for single battery cells

3) Pollution-free

4) Long cycle life >500times

5) No memory effect

6) Capacity, resistance, Voltage, platform time consistency is good

7) With short-circuit production function, safe and reliable

8) Factory price& High quality

9) Good consistency, low self-discharge

10) Light weight, small and customized size

11) Wide application for electronics and industry

12) Environmentally compatible.

3.10.2 Specifications

Item: Li-po 3S

Capacity: 2200 mAh

Voltage: 11.1V

Continuous Discharge C-rate: 30

Max charging rate: 3C

3.10.3 Application

1). mini 4-axis aircraft, Large 4-axis 6-axis and multi-axis aircraft

2). Align famous 3D helicopter RC Helicopter, Trex 450/500/600/700

3). Indoor Helis, Indoor UFO 3D Airplane Aerobatics

4). Micro Slow Flyers, Park Flyers

5). Running electronic Car, HSP car

6). Wireless Communication System

7). Electric Instrument

8). UPS Aero Model, Military Use

9). RC electronic boat

3.11 Motor Driver module

The L293D is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors.

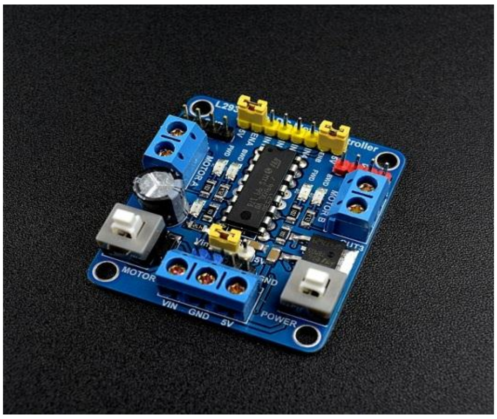


Fig: 3.16: Motor Driver Module

Two enable inputs are provided to enable or disable the device independently of the input signals.The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.Check here for L293D datasheet.

3.11.1 Specifications

- Driver: L293D
- Driver power supply: +5V~+46V
- Driver peak current: 2A
- Logic power output Vss: +5~+7V (internal supply +5V)
- Logic current: 0~36mA
- Controlling level: Low -0.3V~1.5V, high: 2.3V~Vss
- Enable signal level: Low -0.3V~1.5V, high: 2.3V~Vss
- Max drive power: 25W (Temperature 75 ℃)
- Working temperature: -25℃~+130℃

3.11.2 Pin Description

- Out 1: Motor A lead out(+ve)
- Out 2: Motor A lead out(-ve)
- Out 3: Motor B lead out(+ve)
- Out 4: Motor B lead out(-ve)
- 12V:12V input from DC power source(12V jumper – remove this if using a supply voltage greater than 12V DC. This enables power to the on board 5V regulator)
- GND: Ground
- 5V: 5V input (unnecessary if your power source is 7v-35v, if the power source is 7v-35v then it can act as a 5v out)
- EnA: Enables PWM signal for Motor A
- In1: Enable Motor A
- In2: Enable Motor A
- In3: Enable Motor B
- In4: Enable Motor B
- EnB: Enables PWM signal for Motor B

3.11.3 Hardware & Software Requiewd

- L293D H Bridge Drive
- Arduino Mega

- Arduino IDE

1.1.4 3.11.4 Hardware Connections

Here a single DC motor is used and controlled by the H-Bridge driver.It should be connected to Uno as follows:

- OUT1 - DC Motor(+ve)
- OUT2 - DC Motor(-ve)
- Gnd - Gnd of Arduino Uno
- 5V - 5V of Arduino Uno
- ENA- digital pin 10.Leave this in place when using a stepper motor. Connect to PWM output for DC motor speed control.
- IN1 - digital pin 9
- IN2 - digital pin 8

3.11.5 Features

1)High operating voltage, which can be up to 40 volts;

2)Large output current, the instantaneous peak current can be up to 3A;

3)With 25W rated power;

4)Two built in H-bridge, high voltage, large current, full bridge driver, which can be used to drive DC motors, stepper motors, relay coils and other inductive loads.

5)Using standard logic level signal to control.

6)Able to drive a two-phase stepper motor or four-phase stepper motor, and two-phase DC motors.

7)Adopt a high-capacity filter capacitor and a freewheeling diode that protects devices in the circuit from being damaged by the reverse current of an inductive load, enhancing reliability

8)The module can utilize the built-in stabilivolt tube 78M05 to obtain 5v from the power supply. But to protect the chip of the 78M05 from damage, when the drive voltage is greater than 12v, an external 5v logic supply should be used.

9)Drive voltage: 5-35V; logic voltage: 5V

10)PCB size: 4.2 x 4.2 cm

3.12 List of Components With Price

SL NO	Components Name	Quantity	Unite Price	Total Price (BDT)

1	Arduino MEGA Made In Italy	1	2500	2500
2	Buck Module	1	500	500
3	Bluetooth Module	1	600	600
4	Gear Motor and Wheel with Car Chassis	4	600	2400
5	Lipo Battery	1	2000	2000
6	Motor Driver Module	1	800	800
7	Servo Motor	3	600	1800
6	Robotic Arm Frame 6D		4000	4000
11	Others		1000	1000
	Total Cost			15600tk

Table02 -List of Components With Price

Chapter Four

4. SOFTWARE AND SIMULATION

4.1 Introduction

Software part is one of the main part of the MCU based Single phase induction motor driving, controlling & protection system. The Algorithm is based on different conditions & measuring parameters. Total function of the system is controlled by the software. The code is written on C.

4.2 Software Tools

The software that is used to program the microcontroller is open-source-software and can be downloaded for free on www.arduino.cc. With this “Arduino software” we can write little programs with the microcontroller. These programs are called “Sketch”.

In the end the sketches are transferred to the microcontroller by USB cable. More on that later on the subject “programming”.

Installation

Now one after another the Arduino software and the USB driver for the board have to be installed.

Installation and setup of the Arduino software

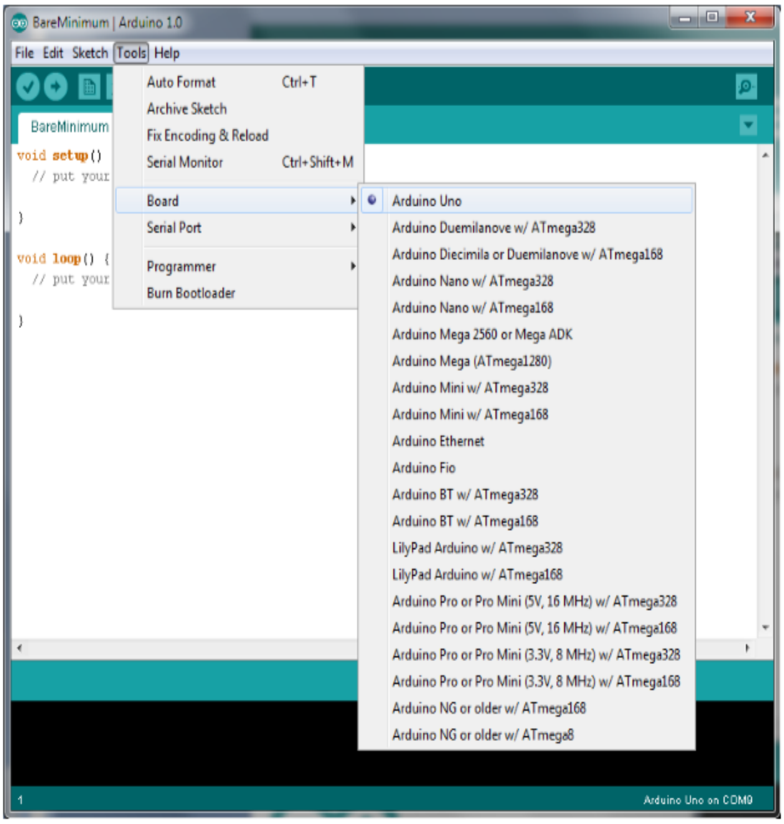
1. We have downloaded the Arduino software from www.arduino.cc and installed it on the computer (This was NOT connected to the PC). After that we opened the software file and installed the program named arduino.exe.

Two set ups on the program are important and should be considered.

a) The board that we want to connect has to be selected on the arduino software. The “Arduino Uno” is here known as “Arduino / Genuino Uno”.

b) We have to choose the right “Serial-Port”, to let the Computer know to which port the board has been connected. That is only possible if the USB driver has been installed correctly. It can be checked this way:

At the moment the Arduino wasn’t connected to the PC. If we now choose “Port”, under the field “Tool”, we will already see one or more ports here (COM1/ COM2/ COM3…). The quantity of the shown ports doesn't depend on the quantity of the USB ports on the computer. When the board gets connected to the computer, we will find one more port.


BareMinimum | Arduino 1.0
File Edit Sketch Tools Help
BareMinimum
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Board
Serial Port
Programmer
Burn Bootloader
Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Mini w/ ATmega328
Arduino Mini w/ ATmega168
Arduino Ethernet
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
Arduino NG or older w/ ATmega8
1
Arduino Uno on COM9

```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Fig. 4.1: Program installation process

4.3 Programming

The development cycle is divided into 4 phases:

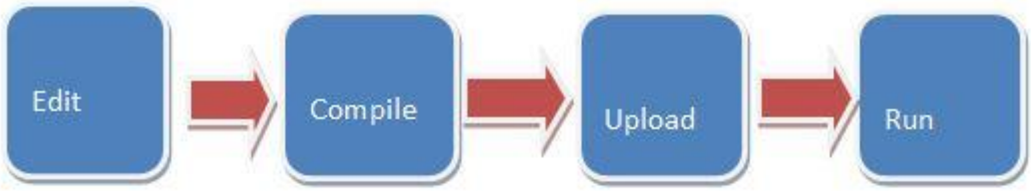


Fig.4.4 Flowchart of the compiling process

Compile: Compile means to translate the sketch into machine language, also known as object.

Code Run: Arduino sketch is executed as soon as terminates the step of uploading on the board.

4.4 Arduino Program Development

- Based on C++ without 80% of the instructions.
- A handful of new commands.
- Programs are called 'sketches'.
- Sketches need two functions:
- void setup ( )
- Void loop ( )
- Setup ( ) runs first and once.
- loop ( ) runs over and over, until power is lost or a new sketch is loaded.

4.5 Proteus8.1

Proteus 8 is best simulation software for various designs with microcontroller. It is mainly popular because of availability of almost all microcontrollers in it. So it is a handy tool to test programs and embedded designs for electronics hobbyist. You can simulate your programming of microcontroller in Proteus 8 Simulation Software.After simulating your circuit in Proteus 8 Software you can directly make PCB design with it so it could be a all in one package for students and hobbyists. So I think now you have a little bit idea about what is proteus software. Proteus 7.0 is a Virtual System Modeling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical prototype in real time. This program allows users to interact with the design using on-screen indicators and/or LED and LCD displays and, if attached to the PC, switches and buttons. One of the main components of Proteus 7.0 is the Circuit Simulation -- a product that uses a SPICE3f5 analogue simulator kernel combined with an event-driven digital simulator that allow users to utilize any SPICE model by any manufacturer.Proteus VSM comes with extensive debugging features, including breakpoints, single stepping and variable display for a neat design prior to hardware prototyping. In summary, Proteus 7.0 is the program to use when you want to simulate the interaction between software running on a microcontroller and any analog or digital electronic device connected to it.

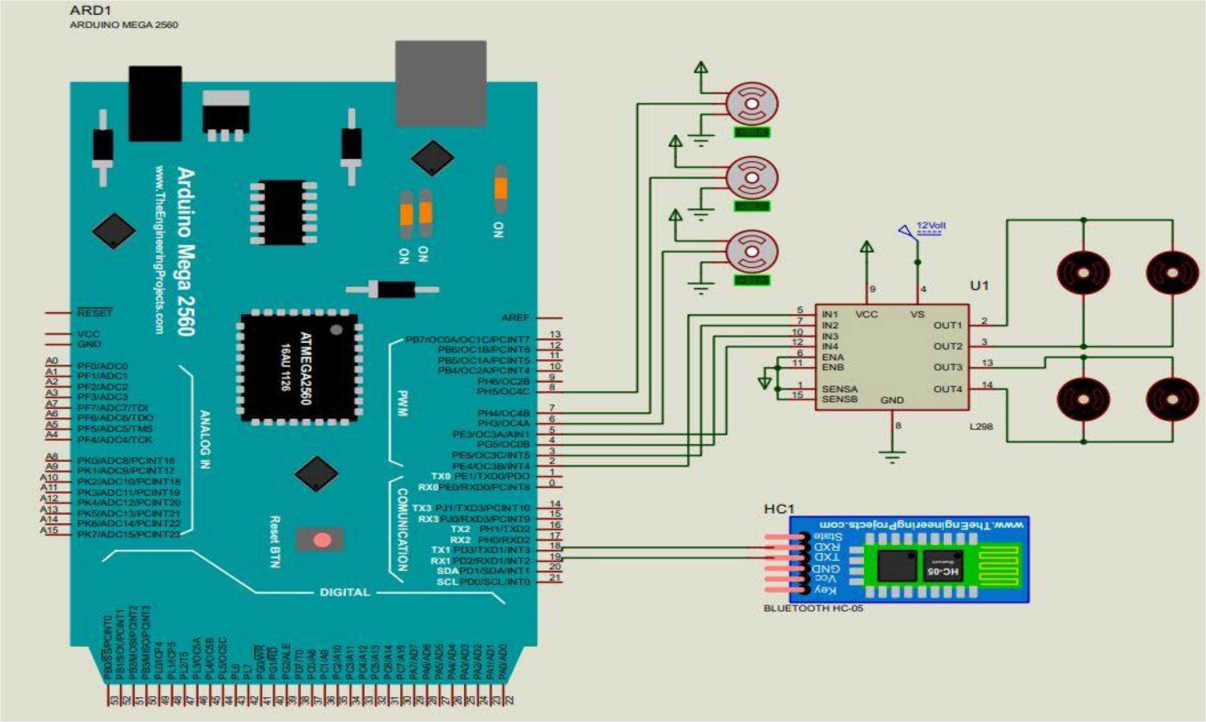


Fig 4.5: User interface of proteus 8.1

Chapter Five

5. RESULT AND OUTPUT OBSERVATIONS

5.1 Introduction

The name of our project is Smartphone Control Peck and Place Robotics Arm .This is a modern project. The main goal of our project is to move a product from one place to another.In the industry we see this is done using CNC technology which is very costly.We have created a robotics arm that can transfer any product under the command of a mobile phone.Here we have done the work with servo motor, Bluetooth module and microcontroller.

5.2 **Working Principle of this Project**

First we will connect the mobile phone with the Bluetooth module in the project.We will give commands using a software on the mobile phone.The command sent from the mobile phone will receive the Bluetooth module in the project.The Bluetooth module commands will be delivered to the project programming device arduino.The arduino controller will operate the servo motor and motor driver module to perform the command.The servo motor is basically a robotics arm, which can move any product according to the command.The below image is showing the total system is in Run mode.

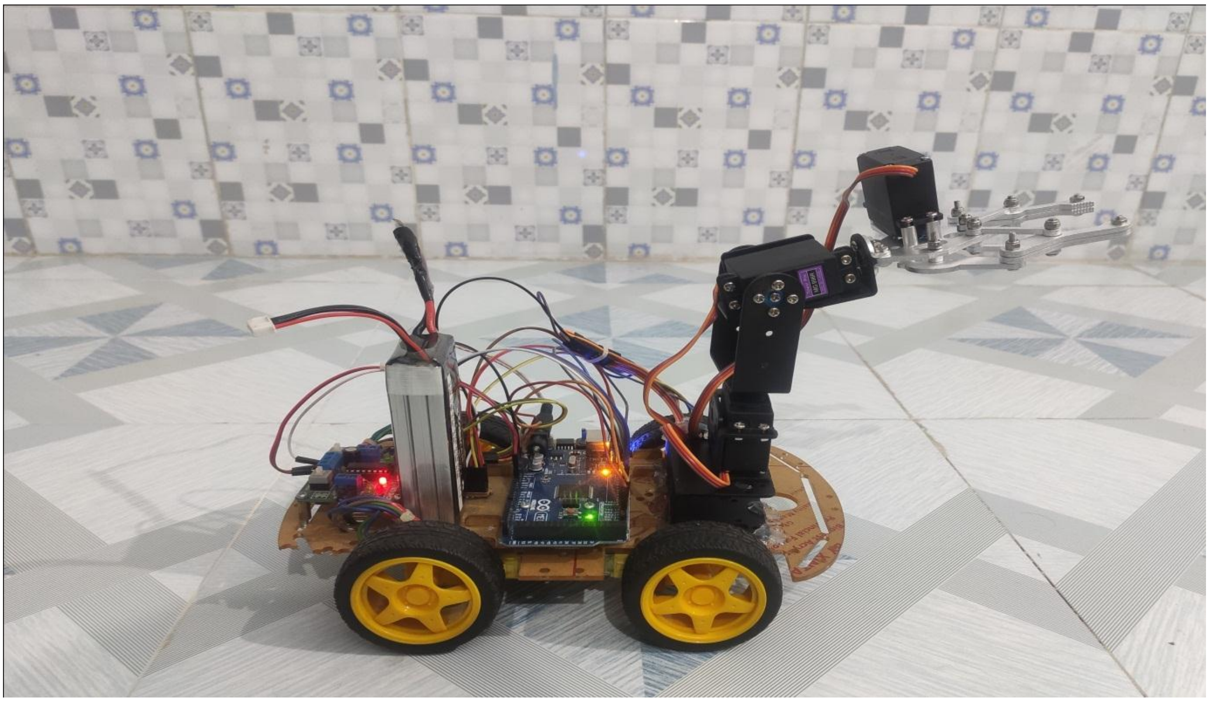


Figure 5.1: Project picture

5.3 Result

After switching on, the project will start.The mobile module must be connected to the Bluetooth module in the project.We have robot made by smart phone apps.It can be easily taken to the place we want only at the command of the smartphone command. Robotics arm with robot car that can transfer any product under the command of a mobile phone.

Chapter Six

6. CONCLUSION AND DISCUSSION

6.1 **Conclusions**

An autonomous robot with adjustable gripper that perform pick and place operation has been successfully designed and developed.The robot is also able to perform lifting upward and downward smoothly. By using microcontroller, the robot has performed the task perfectly according to the program that being made.This system can be used in various applications like processing, spraying, stamping and welding for work piece.Due to this advantage, the robot can pick.

6.2 **Advantages**

I. Advantages of Robotics – High performance, high accuracy
II. low labour cost.
III. Cost is less as Arduino is open source.
IV. Digital Assistance.
V. Anyone can manage.
VI. Products can be easily transferred.
VII. Commands are being handled wirelessly.

6.3 Applications

I. Industrial Automation.
II. Pharmaceutical Factory.
III. Food Factory.
IV. Load unloading zone.
V. Port area.

6.4 Limitation

I. Fixed Working place.
II. Unable to carry excess weight.

6.5 Future works

In future this method will be implemented in most of the organization and industry. It can be implemented even in the small scale industries because energy conservation and paying penalty cause a major issue in all scale industry and this method rectifies all the problem and so this will be a compulsion to minimize their expenses. This will be the best solution in future because man power will be eradicated completely and will be automated. This will play a major role in it.It can be implement IOT Based monitoring and Controlling system , this will allow the use of advanced sensors like color detector sensors and to include moveable system.

7. REFERENCES

[1] Ratheesh Rajan "Foundation Studies for an Alternate Approach to Motion Planning of Dynamic Systems" M.S.E., the University of Texas at Austin, 2001.

[2] Mohamed Naufal bin Omar,"Pick and place robotic arm controlled by Computer", UniversititeknikalMalaysia, Melaka April 2007.

[3] Yanjianghuang, ryosukechiba, tamioarai, tsuyoshiueyama and junota. ,"Integrateddesign of multi-robot system for pick-and-place tasks", Proceeding of theIEEE International conference on robotics and biomimetic (ROBIO) Shenzhen, china, December 2013.

[4] Sungwookmoon ,youngjinkim, ho junmyeong , changsookim, namjucha,and dong hwankim . ,"Implementation of smart phone environment remote controland monitoring System for android operating system-based robot platform" ,The 8thinternational conference on ubiquitous robots and ambient intelligence (URAI 2011) Tc1-1 Nov. 23-26, 2011 in songdoconventia, incheon, korea 978-1-4577-0723-0 / 11 /$26.00 2011 IEEE

[5] ATMEL ATmega48A/PA/88A/PA/168A/PA/328/P [DATASHEET] 2 Atmel-8271I-AVR- ATmega-Datasheet10/2014.

[6] SGS Thomson Microelectronics L293D - L293DD [DATASHEET] PUSH-PULLFOUR CHANNEL DRIVER WITH DIODE .June 1996.

[7] Fairchild semiconductor KA78XX/KA78XXA 3-Terminal 1A Positive Voltage Regulator Data sheet.2001 Fairchild Semiconductor Corporation.

[8] http://en.wikipedia.org/wiki/Android_(operating_system). April 2015.

[9] https://play.google.com/store/apps/details id=com.gundel.bluecontrol&hl=en .April 2015.

[10] http://en.wikipedia.org/wiki/Arduino Programmer. April 2015.

[11] John Iovine.,“Robots,Androids, andAnimations 12 Incredible Projects You Can Build”, Second Edition,McGraw-Hill.2002

[12] Richard E. Pattis. Karel the Robot: A Gentle Introduction to the Art of Programming. John Wiley &Sons, 1981.

[13] RK Mittal and IJ Nagarath “Robotics and Control” BITS Pilani, 2003.

[14] https://www.arduino.cc/en/software/

[15] https://en.wikipedia.org/wiki/Proteus_Design_Suite

8. APPENDIX

Program

```
//#include <SoftwareSerial.h>
//SoftwareSerial android(7, 8);
#include <Servo.h>
char command ;
int motorR1 = 2;
int motorR2 = 3;
int motorL1 = 4;
int motorL2 = 5;

//servo setup
Servo myservo;
Servo myservo1;
Servo myservo2;

int servodata = 70;
int servodata1 = 50;
int servodata2 = 30;
char data;

void setup() {
  pinMode(motorR1, OUTPUT);
  pinMode(motorR2, OUTPUT);
  pinMode(motorL1, OUTPUT);
  pinMode(motorL2, OUTPUT);

  myservo.attach(6);  // attaches the servo on pin 3 to the servo object
  myservo1.attach(7);  // attaches the servo on pin 5 to the servo object
  myservo2.attach(8);  // attaches the servo on pin 6 to the servo object

  myservo.write(servodata);
  myservo1.write(servodata1);
  myservo2.write(servodata2);

  Serial.begin(9600);
  Serial1.begin(9600);// Starts the serial communication
}

void loop() {

  if (Serial1.available())
  {


  }



  // for bluetooth


  if (command == '1') //forward
  {
    analogWrite(motorR1, 150);
    analogWrite(motorR2, 0);
    analogWrite(motorL1, 150);
    analogWrite(motorL2, 0);

  }
  else if (command == '2') //Reverse
  {
    analogWrite(motorR1, 0);
    analogWrite(motorR2, 150);
    analogWrite(motorL1, 0);
    analogWrite(motorL2, 150);
    //digitalWrite(spker, LOW);
  }

  else if (command == '3') //Left
  {
    analogWrite(motorR1, 20);
    analogWrite(motorR2, 0);
    analogWrite(motorL1, 200);
    analogWrite(motorL2, 0);

  }
  else if (command == '4') //Right
  {
```

```
  analogWrite(motorR1, 200);
  analogWrite(motorR2, 0);
  analogWrite(motorL1, 20);
  analogWrite(motorL2, 0);

}
else if (command == 'c') //stop
{
  analogWrite(motorR1, 0);
  analogWrite(motorR2, 0);
  analogWrite(motorL1, 0);
  analogWrite(motorL2, 0);
}

if (data == 'E')
{
  Serial.println("left");
  myservo.write(servodata);
  delay(20);
}

if (data == 'e')
{
  Serial.println("right");
  myservo.write(servodata);
  delay(20);

}

if (data == 'a')
{
  Serial.println("up");
  myservo1.write(servodata1);
  delay(20);

}

if (data == 'b')
{
  Serial.println("down");
  myservo1.write(servodata1);
  delay(20);
}


if (data == 'c')
{
  Serial.println("close");

  myservo2.write(servodata2);
```

```
    delay(100);
  }

  if (data == 'd')
  {
    Serial.println("open");
    myservo2.write(servodata2);
    delay(100);
  }

  //command = 's';
}
```