RESEARCH ARTICLE OPEN ACCESS

Development of Iot Controlled Robot

[1]Yashas M S, [2]Navaneeth Kashyap K V, [3]Anil Kumar M M
[1,2]Research Group, Department of Mechanical Engineering, MIT Mysore-571477
[3]Associate Professor, Department of Mechanical Engineering, MIT Mysore-571477
Corresponding Author: Yashas M S

ABSTRACT: *In many situations, autonomous robots can provide effective solutions to gruelling tasks. In this case, it is desirable to create an autonomous robot that can identify objects and relocate them if the object meets certain criteria. Dealing with a large number of objects is a very menial task, this is an excellent application for a robot of this type. In this case, to keep costs and design complexity low i.e an Arduino microcontroller is choosen instead of raspberry pi as raspberry pi requires another device to interface microcontroller with the motor drivers which are costly and also we are using joystick app instead of keyboard which increases the flexibility to use. By using Arduino c++ programming language is used.*

Key Words*: Arduino Uno, NODE MCU, RGB SENSOR, ROBOT ARM, PMDC MOTOR, MOTOR DRIVER,Arduino IDE, BLYNK*

Date of Submission: 01-01-2020 Date of Acceptance: 16-01-2020

I. INTRODUCTION

Robotics is the branch of mechanical engineering, electrical engineering and computer science that deals with the design, construction, operation, and application of robots, as well as computer systems for their control, sensory feedback, and information processing. The first industrial robot was designed by GEORGE DEVOL in 1954The internet of things (IoT) is the network of physical devices, vehicles, buildings and other items embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The IoT allows objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit. When IoT is augmented with sensors and actuators, the technology becomes an instance of the more general class of cyber-physical systems, which also encompasses technologies such as smart grids, smart homes, intelligent transportation and smart cities.

Figure 1: Typical Pick & Place Robot

II. LITERATURE SURVEY

AJINKYA DHEKNE proposed the robotic control using internet via webpage & keyboard. But by using the joystick or in built gyroscope, the controlling can be made more easy and flexible. RASPBERRY PI controller robot was proposed by VATSAL SHAH. However we are interfacing it with an ARDUINO microcontroller which reduces the cost MD. FOKHRAY HOSSAIN explained robot controlled by voice which uses Bluetooth. This had limited range but IOT controlled robot had enhance working range. A P SHINDE explained the knowledge of how different sensors responsible for robot control.

2.1 Problem Definition

The colour sensor used had higher light reception rate during day time and less reception rate at night.to overcome this problem we are using light dependent resistors which optimizes light receptivity.

Interfacing motor directly to Arduino was not possible, because motor requires 250mA of current to operate while Arduino works on 5mA.Direct interfacing of motor to Arduino destroys it.hence to overcome this problem we are using motor drivers.

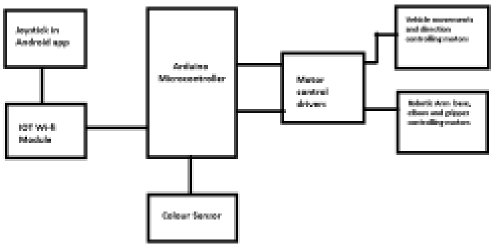
Instead of other motors only PMDC motors were used. Because other motors tends to reduce speed requirement as it is loaded. But pmdc motor has less variation to loading.

2.2 Objectives

This system is very beneficial for places where there is a need to pick an object move it and place it to some other place safely. If the object is being picked by a human, there is a risk of damage to the object which is avoided by this system.

- The main objective of Colour Identification is to sort the object according to their colour using a colour sensor and also to sort the object to the station accordingly.
- To control the displacement of the robotic arm so that the arm can be used pick and place the element from any source to destination
- The objective of the system to perform repetitive tasks over long period of time which is achieved ineffectively by human.

III. METHODOLOGY



Block diagram of the system

When the operator issues a command from the Joystick control to the robotic vehicle all necessary tasks will be carried out by sending signals to the microcontroller via IOT. The microcontroller then issues command to the respective channels that makes up the communication links. The electric motor response will depend on the type of command issued and the direction, speed and motion of the motor is regulated by the microcontrollerThe robotic arm has five degrees of freedom. It is made of various links forming an open chain. The arrangement of these links depends on the adopted design. The arm has a rotating base that is resting on the upper region of the vehicle. The arm terminates with a gripper or a specialized tool holder,it has five degrees of freedom.The first three links of the arm form the body and which helps to place the tool holder at the desired position at a location inside the workspace or environment. The remaining three links make up the wrist of the robotic arm and are used to define the orientation of the robotic arm end points. For the purpose of analysis, the robotic arm will made of joints, which will be named as gripper, wrist, shoulder, and base.

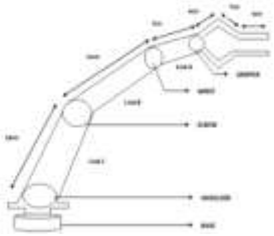


FIG 2:ROBOT PARTS

3.1SELECTION OF HARDWARE

ArduinoUno: The Arduino Uno is a microcontroller board based on the ATmega328. Arduino is an open-source, prototyping platform and its simplicity makes it ideal for hobbyists to use as well as professionals. The Arduino Uno has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button

FIG 3: ARDUINO MICROCONTROLLER

Power (USB / Barrel Jack): Every Arduino board needs a way to be connected to a power source. The Arduino UNO can be powered from a USB cable coming from computer or a wall power supply that is terminated in a barrel jack. The USB connection is also how you will load code onto your Arduino board.The recommended voltage supply should be in the range of 6-12V.**Pins 5V,**

3.3V, GND, Analog, Digital, PWM are used.

NODE MCU

FIG 4 : NODE MCU

Node-MCU is an open source development board and firmware based in the widely used ESP8266 -12E Wi-Fi module. It allows you to program the ESP8266 Wi-Fi module with the simple and powerful LUA programming language or Arduino IDE.

- Finally, programable Wi-Fi module.
- Arduino-like (software defined) hardware IO.

- Can be programmed with the simple and powerful Lua programming language or Arduino IDE.
- USB-TTL included, plug & play.
- 10 GPIOs D0-D10, PWM functionality, IIC and SPI communication, 1-Wire and ADC A0 etc. all in one board.
- Event-driven API for network applications.
- PCB antenna.

RGB SENSOR

This section is used for detecting the colour of the object to be sorted. There are many colour sensing ICs available today. In this project TCS3200 is selected.

FIG 5: RGB SENSOR

The TCS3200 and TCS3210 programmable colour light-to-frequency converters that combine configurable silicon photodiodes and a current-to-frequency converter on a single monolithic CMOS integrated circuit. The output is a square wave with frequency directly proportional to light intensity. The light sensor works by shining a white light at an object and then recording the reflected colour. It can also record the intensity of the reflection (brightness). Through red, green and blue colour filters the photodiode converts the amount of light to current. The converter then converts the current to voltage which our arduino can read. The sensor also consists of four LED lights.The sensor has 10 pins S0, S1 are for setting the frequency and S2, S3 are for reading the color values. The **out** pin is supposed to give the output to a Arduino in the form of a square wave. The other pins are for powering the sensor.

ROBOT ARM:

A robotic arm is a programmable mechanical arm, analogous with functionality of a human arm. The links of a robotic arm are interconnected by joints allowing either translational or rotational motion. Robotic arm has three segments namely base, elbow and gripper. Each segment has one degree of freedom which means it will have six movements. Motor will be used to move the robot arm and will be located at every joint of the robot arm.

MOTORS

Permanent-magnet direct-current (PMDC) motor is used. DC Motor is inexpensive, small and powerful motors. These motors are flexible because both their direction and speed can be controlled; direction by polarity and speed by voltage. DC Motors are extensively used in robotics because of their lesser size and more energy output, they are excellent for driving the mechanical assemblies.

FIG 6:PMDC MOTOR

300RPM 12V DC motors with Gearbox
3000RPM base motor
6mm shaft diameter with internal hole
125gm weight
Same size motor available in various rpm
0.35kgcm torque
No-load current = 60 mA(Max), Load current = 300 mA(Max)

MOTOR DRIVER

A push-pull high current, high voltage four channel driver, L298D is used to control the robotic arm The L298N is an integrated monolithic circuit in a 15- lead Multiwatt and PowerSO20 packages.

It is a high voltage , high current dual full-bridge driver de-signed to accept standard TTL logic level sand drive inductive loads such as relays, solenoids, DC and stepping motors.

FIG 7: MOTOR DRIVER

Two enable inputs are provided to enable or disable the device independently of the in-put signals .The emitters of the lower transistors of each bridge are connected together rand the corresponding external terminal can be used for the connection of an external sensing resistor.

3.2SOFTWARE REQUIREMENTS
Arduino IDE

FIG 8:Arduino IDE

- Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, arduino micro and many more.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- This environment supports both C and C++ languages.
- The IDE environment is mainly distributed into three sections.

BLYNK

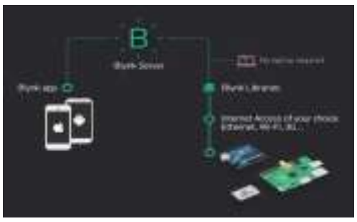


FIG 9: BLYNK

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, vizualize it and do many other cool things.

- Similar API & UI for all supported hardware & devices
- Connection to the cloud using:
- WiFi
- Bluetooth and BLE
- Ethernet
- GSM
- Set of easy-to-use Widgets
- Direct pin manipulation with no code writing
- Easy to integrate and add new functionality using virtual pins
- History data monitoring via Super Chart widget
- Device-to-Device communication using Bridge Widget

IV. PROGRAM

```
#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
char
auth[]
"f2817a8811824a428a29f238b3e44d42";
char ssid[] = "vinay";
char pass[] = "9844131711";
const int ml1 = 15;
const int ml2 = 2;
const int mr1 = 4;
const int mr2 = 5;
const int rot1 = 18;
const int rot2 = 19;
const int elb1 = 21;
const int elb2 = 22;
const int grip1 = 23;
const int grip2 = 13;
WidgetLCD lcd(V0);//virtual port number on app lcd data to be connect

//rotation//rotation//rotation
BLYNK_WRITE(V1)
{
int a = param[0].asInt();
int b = param[1].asInt();
Serial.print("X = ");
Serial.print(a);
Serial.print("; Y = ");
Serial.println(b);
if(a<500 && b==1023)
{
digitalWrite(rot1, HIGH);
digitalWrite(rot2,LOW);
}
else if(a>500 && b==0)
{
digitalWrite(rot1, LOW);
digitalWrite(rot2,HIGH);
}
else if(a == 512 && b == 512)
{
digitalWrite(rot1, HIGH);
digitalWrite(rot2, HIGH);
}
else
{
```

```
}
}

//elbow//elbow//elbow//elbow
BLYNK_WRITE(V2)
{
int c = param[0].asInt();
int d = param[1].asInt();
Serial.print("X = ");
Serial.print(c);
Serial.print("; Y = ");
Serial.println(d);
if(c<500 && d==1023)
{
digitalWrite(elb1, HIGH);
digitalWrite(elb2,LOW);
}
else if(c>500 && d==0)
{
digitalWrite(elb1, LOW);
digitalWrite(elb2,HIGH);
}
else if(c == 512 && d == 512)
{
digitalWrite(elb1, HIGH);
digitalWrite(elb2, HIGH);
}
else
{
digitalWrite(elb1, HIGH);
digitalWrite(elb2, HIGH);
}
}

//gripper//gripper//gripper//gripper
BLYNK_WRITE(V3)
{
int e = param[0].asInt();
int f = param[1].asInt();
Serial.print("X = ");
Serial.print(e);
Serial.print("; Y = ");
Serial.println(f);
if(e<500 && f==1023)
{
digitalWrite(grip1, HIGH);
digitalWrite(grip2,LOW);
}
else if(e>500 && f==0)
{
digitalWrite(grip1, LOW);
digitalWrite(grip2,HIGH);
}
else if(e == 512 && f == 512)
{
digitalWrite(grip1, HIGH);
digitalWrite(grip2, HIGH);
}
else
{
 digitalWrite(grip1, HIGH);
digitalWrite(grip2, HIGH);
}
}

// movements// movements// movements

BLYNK_WRITE(V4)
{
int x = param[0].asInt();
int y = param[1].asInt();
Serial.print("X = ");
Serial.print(x);
Serial.print("; Y = ");
Serial.println(y);
if(x<500 && y==1023)
{
lcd.clear();
lcd.print(1, 0, "Forward"); // use: (position X: 0-15, position Y: 0-1, "Message you want to print")
digitalWrite(ml1, HIGH);//forward
digitalWrite(ml2,LOW);
digitalWrite(mr1,HIGH);
digitalWrite(mr2,LOW);
}
else if(x>500 && y==0)
{
lcd.clear();
lcd.print(1, 0, "Reverse"); // use: (position X: 0-15, position Y: 0-1, "Message you want to print")
digitalWrite(ml1, LOW);//reverse
digitalWrite(ml2,HIGH);
digitalWrite(mr1,LOW);
digitalWrite(mr2,HIGH);
}
else if(x<=0 && y<500)
{
lcd.clear();
lcd.print(1, 0, "Left"); // use: (position X: 0-15, position Y: 0-1, "Message you want to print")
digitalWrite(ml1, HIGH);//left
digitalWrite(ml2,LOW);
digitalWrite(mr1,LOW);
digitalWrite(mr2,LOW);
}
else if(x==1023 && y>500)//right
{
lcd.clear();
lcd.print(1, 0, "Right");
digitalWrite(ml1, LOW);
digitalWrite(ml2,LOW);
digitalWrite(mr1,HIGH);
digitalWrite(mr2,LOW);
}
else if(x == 512 && y == 512)//stop
{
```

```
lcd.clear();
lcd.print(1, 0, "Stop");
digitalWrite(ml1, HIGH);
digitalWrite(ml2, HIGH);
digitalWrite(mr1, HIGH);
digitalWrite(mr2, HIGH);
}
else
{
lcd.clear();
lcd.print(1, 0, "Stop");
digitalWrite(ml1, HIGH);
digitalWrite(ml2, HIGH);
digitalWrite(mr1, HIGH);
digitalWrite(mr2, HIGH);
}
}
void setup()
{
pinMode(ml1, OUTPUT);
pinMode(ml2,OUTPUT);
pinMode(mr1, OUTPUT);
pinMode(mr2,OUTPUT);
pinMode(rot1,OUTPUT);
pinMode(rot2,OUTPUT);
pinMode(elb1,OUTPUT);
pinMode(elb2,OUTPUT);
pinMode(grip1,OUTPUT);
pinMode(grip2,OUTPUT);
Serial.begin(9600);
Blynk.begin(auth, ssid, pass);
}
void loop()
{
Blynk.run();
```

FIG 8: FINAL ASSEMBLED MODEL

V. CONCLUSION

Robotics is a technology with a future, and it is a technology for the future.. But in our project we used colour sensor to sort the object based on colour and eventually the robot picks and places that object accordingly. By doing so the speed and the accuracy of the colour sorting process is increased. The cost for the colour sorting process is considerably reduced. And most importantly there is overall optimization in the productivity if an industry. This robot arm can perform an action which is much similar to human. Although there is significant progress in robotics, still its usage is limited due to less availability of resources and high cost of production.If we able to overcome these restrictions, more benefits can be achieved from robotics.

REFERENCES

[1]. Vindhya Devalla, Dr. Rajesh Singh, Amit Kumar Mondal and Vivek Kaundal, “Design and Development of Object Recognition and Sorting Robot for Material Handling in Packaging and Logistic Industries” International Journal of Science and Advanced Technology , Volume 2 No 9 September 2012.

[2]. L.A. Reeves and M. Farooq “A Multiprocessor Pick-and-Place Operation for TWO Robot Arms”“ Dept. of Electrical and Computer Engineering, Royal Military College of Canada, Kingston, Ontario,0-7803-1443-3 ,Proceedings of 1993 IEEE.

[3]. http://www.dfrobot.com/wiki/index.php/TCS3200\ Color Sensor_(SKU:SEN0 101)

[4]. Mohd Ashiq Kamaril Yusoffa, Reza Ezuan Saminb, Babul Salam Kader Ibrahimc, Wireless Mobile Robotic Arm, International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), 1072 – 1078.

[5]. Nikesh. R. Vange, Atul .V. Nar, Dhananjay .B. Surve, Anita. P. Trimukhe, Manisha .M. Patil, Rajesh .A. Patil, Object Sorting Robotic Arm Based on Color Sensing Mechanism , International Journal of Emerging Technology and Advanced Engineering , (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 5, Issue 5, May 2015)

[6]. http://www.robotics.org

Yashas M S "Development of Iot Controlled Robot". *International Journal of Engineering Research and Applications (IJERA*), vol.10(01), 2020, pp 01-06