Research Paper

E-ISSN NO : 2455-295X | VOLUME : 8 | ISSUE : 3 | MARCH 2022

DESIGN AND FABRICATION OF PROTOTYPE ROBOT FOR VARIOUS AGRICULTURAL TASKS

***DR. M. S. DHUTTARGAON [1] | PROF. S. M. GIDAVEER [2]**

[1] ASSOCIATE PROFESSOR, DEPT. OF MECHANICAL ENGINEERING, TKIET, WARANANAGAR, INDIA - 416113.

[2] ASSISTANT PROFESSOR, DEPT. OF MECHANICAL ENGINEERING, TKIET, WARANANAGAR, INDIA - 416113.

ABSTRACT:

Many countries in Asia including India are agrarian economies and most of their rural populations depend on agriculture to earn their livelihood. Aimed at increasing the productivity and reducing the labor involved, this robot is designed to execute the basic functions required to be carried out in farms. We aim to create a multitasking agriculture robot which will focus on basic work of plantation. This multipurpose system gives an advanced method to seed sowing, ploughing, watering the crops with minimum manpower and labor making it efficient vehicle. The machine will cultivate the farm by considering particular rows and specific columns at fixed distance depending on crop. Moreover, the vehicle can be controlled through Bluetooth medium using a Remote smart phone. The whole process calculation, processing, monitoring is designed with motors and interfaced with Microcontroller.

KEYWORDS:

ROBOT, AGRICULTURE, MULTIPURPOSE ROBOT.

1. INTRODUCTION

Agriculture plays a vital role in the economy of every nation. Since the dawn of history, agriculture has been the significant source of food for human beings. Today more and more lands are brought under cultivation to produce variety of crops. Agricultural processes involve handling of heavy materials. Most of the processes are monotonous, repetitive and requires strength and skills on part of labours. From early 19th century, researchers have explored the application of robots in agriculture filed. The potential benefits of automated agricultural vehicles include increased productivity, increased application accuracy, and enhanced operational safety. Additionally, the rapid advancements in electronics, computers, and computing technologies have inspired and renewed the interest in the development of vehicle guidance systems. Various guidance technologies, including mechanical guidance, optical guidance, radio navigation, and ultrasonic guidance, have been investigated.

A robot is a machine that can be programmed and reprogrammed to do certain tasks and usually consists of a manipulator such as a claw, hand, or tool attached to a mobile body or a stationary platform. Autonomous robots work completely under the control of a computer program. They often use sensors to gather data about their surroundings in order to navigate. Tele-controlled robots work under the control of humans and/or computer programs. Remote-controlled robots are controlled by humans with a controller such as a joystick or other hand-held device. The word "Robot" came from the Czech word Robot, which means forced labour or work. The number of agricultural robots, agribot, is increasing each year. The jobs they can do are also increasing with new technology in hardware and software. Robots are milking cows, shearing sheep, picking fruit, weeding, spraying, and cultivating, they use GPS and sensors for navigation. The robots having applications in agriculture are also known as agribot. The new robots are getting smaller and smarter. There are many advantages accompanied with deploying robots to perform agricultural tasks. Monotonous and repetitive tasks are performed effectively by robots every time, resulting in quality improvements. There is no limitation of time for robots to perform the tasks continuously.

India is agrarian economies and most of rural populations depend on agriculture to earn their livelihood. The farming methods at present are manual or semi-automatic with high involvement of labours. In the recent years, the number of labour availability is reducing continuously along with increase in their wages. The application of robots in Indian agriculture scenario can resolve this problem. Hence, the device which can help farmers to overcome the stated problem is needed. In this study, an attempt is made to develop a prototype of multipurpose agribot that will be able to plough the land and sow the seeds. The intended agribot should consist of attributes as follows:

i. It should dig the ground to the specified depth
ii. It should sow adequate number of seeds in the dug hole
iii. It should perform these operations automatically in a sequence without human intervention

iv. It should be safe and easy to operate

2. MATERIALS AND METHODS:

After identifying the problem, a plan of action is needed for successful completion of this study. As a first step, the literature pertaining to the problem was surveyed to collect the information about the existing agricultural robots. Different design alternatives were identified based on literature survey. After evaluating each design alternative critically, the best alternative was chosen.

As the robot moves forward the digging of the ground is done by using the pointed structures forming continuous part. The seed container is spaced at equal and calculated distance from these structures. The seeds are sown in the hole dug by using the stepper motors. A container is used for holding the seeds. A hole is drilled in the bottom of the container and is covered with small metal sheet. This sheet provides way to the dropping of seeds at periodic intervals. The metal sheet is connected to a stepper motor, as the stepper motor rotates to a specific angle the metal sheet opens and a seed is dropped. Then, immediately it rotates the same angle in reverse direction thus, closing the metal sheet. The specific amount of seeds is dropped in the dug hole. At the end of the robot a metal sheet is provided for front filling of the soil. The operations are carried out simultaneously. It means, as the robot digs the ground and moves forward the seeds are sown in the dug hole meanwhile the digging is carried out in the next calculated distance space. These three functions are performed automatically in a sequence by using a control circuit. The source of the power is the battery which can be charged. Mechanical structure of a device consists of solid base in which other parts are assembled. Front wheels are driven by the rear wheels by v-belt and pulley arrangement which is driven by 3 phase induction motor. The axel of the front wheels is connected to the rod which is in contact with the worm gear arrangement. Quarter turn gear box helps to turn the wheels to particular angle. This can be controlled using the limit switches. The front wheels are used for turning and rear wheels for movement of the robot. The rear wheel is a spiked wheel of diameter 18inch and width 50mm. The wheel has 6 spokes of 12mm thickness. Ploughing tools are designed based on the specifications i.e. 3 inch depth. The three teeth like projections which form the path are attached on a shaft. The seed container is about 80cm in length, 40cm in height and 20cm width. The three holes are drilled in the box for the seed fall. There is a conical provision to which a pipe is attached through which the specific number of seeds fall. A plate is attached to the bottom of the seed container to control the opening and closing of seeds. The front filling of the soil is done by attaching the front filling provisions at the front of the robot. In this robot there are two mechanism one for turning the robot and other control of seed flow. Steering mechanism is of worm and sector arrangement. In the worm and sector steering gear the pitman arm shaft is connected to the motor which carries the sector gear that meshes with the worm gear on the steering gear shaft. Only a sector of gear is used because it turns through an arc of approximately 60 degrees either to the left or right. The steering wheel turns the worm on the lower end of the steering gear shaft, which rotates the sector and the pitman arm through the use of the shaft. The robot consists of a seed container in which seeds are stored. The container is provided with the holes of particular diameter depending on the size of the seeds. The holes are provided with a moving plate at the bottom which moves to-and-fro thus guiding the opening and closing of holes for the seed flow when required. It is done through stepper motor and the timer is placed to control the timing for seed flow.

A prototype was designed to test the working and functioning of the actual robot AS. The body is made up of the foam sheet, and the whole process is controlled by a microcontroller. This prototype is shown in Fig-1. Based on total weight of the robot torque required to rotate the wheels of the robot is calculated considering rolling resistance and grade resistance. The appropriate battery is selected considering torque required.

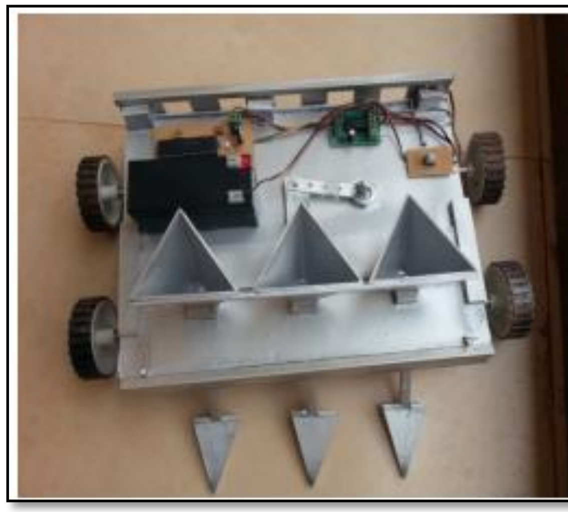

FIG 1: PROTOTYPE OF ROBOT

The prototype robot designed in this study is expected to be controlled by Smartphone through Bluetooth connectivity. System requirements specification (SRS) is a text written to specify in detail the system components, both hardware and software, which are needed for the system implementation, along with functional and non-functional and operational requirements, as anticipated from the system.

- **HARDWARE SPECIFICATIONS:**

AGRICULTURAL ROBOT REQUIRES THE FOLLOWING HARDWARE COMPONENTS:

1) A Remote smart phone having Bluetooth facility that acts as a controller of the robot.
2) A microcontroller 8051 development board which acts as the heart of the robot and controls the entire device.
3) The IC used is 89S52.
4) Motor driver circuit L293 to increase the current

rating.

5) Bluetooth HC05 module to establish connection with the Remote and receive commands.

6) 5 DC motors for performing the activities.

- **SOFTWARE SPECIFICATIONS:**

AGRICULTURAL ROBOT REQUIRES THE FOLLOWING SOFTWARE COMPONENTS:

1) Operating System: This application works well with Remote 4.0 and all versions above this.
2) Backend: Java will be used for developing backend of the robot application. Embedded C is used to program the agricultural robot.
3) Front End: It is developed using ADT Bundle toolkit and the language used is XML.
4) The cross compiler used is Keil Micro Vision 3.
5) Flash Magic software is used for dumping code into microcontroller.

SYSTEM DESIGN

When the user opens the application, a message requesting Bluetooth to be enabled appears. Once the Bluetooth is switched on and the robot is paired a list of activities appears. The user selects the desired activity and the id corresponding to the activity is sent to the robot via Bluetooth. The following Fig-2 shows the activity diagram at user end.

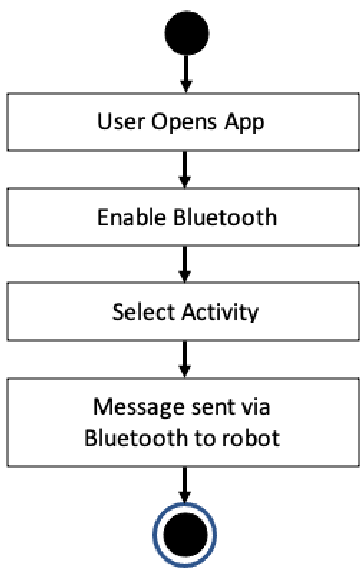

FIG-2. ACTIVITY DIAGRAM AT USER END

Bluetooth receiver at the robot receives the Identifier passes by the user and passes the message on to the Microcontroller. Appropriate DC Motor that must be activated is deduced by the Microcontroller. The Motor Driver activates the corresponding motor and performs the selected operation. The robot now waits for the stop signal to be received to stop functioning. This is indicated in following Fig-3.

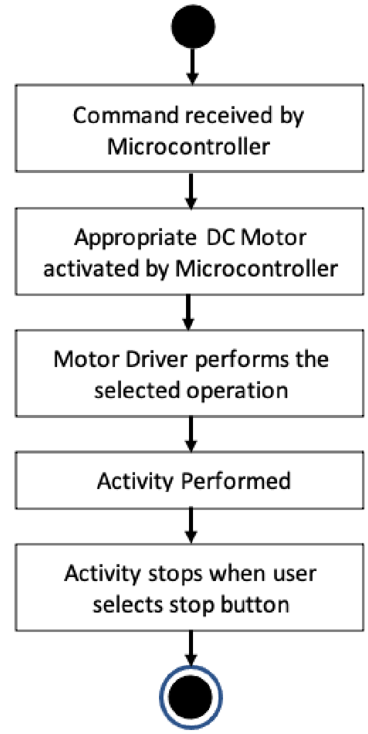

FIG-3. ACTIVITY DIAGRAM AT ROBOT END

SYSTEM IMPLEMENTATION

This section explains how WSN based agricultural robot operated using smart phone can be implemented.

1) MODULE 1: SENDER SIDE - REMOTE PHONE ALONG WITH ITS USER

Step 1: Initialize the intent request codes.

private static final int REQUEST_CONNECT_DEVICE = 1;

private static final int REQUEST_ENABLE_BT = 2;

Step 2: Note the message types sent from the Bluetooth read service

Handler. public static final int

MESSAGE_STATE_CHANGE = 1; public static final int

MESSAGE_READ = 2;

public static final int MESSAGE_WRITE = 3;

public static final int MESSAGE_DEVICE_NAME = 4;

public static final int MESSAGE_TOAST = 5;

Step 3: Call an instance when the activity is first created.

super. on Create(saved Instance State);

Step 4: Set up custom title and select the led device using its id.

cmdBtn = (Button) findViewById(R.id.Load1ON);

cmdBtn.setOnClickListener(this);

cmdBtn = (Button) findViewById(R.id.Load1OFF);

cmdBtn.setOnClickListener(this);

Step 5: Save the connected device's name.

mConnectedDeviceName =

msg.getData().getString(DEVICE_NAME);

Step 6: Create a dialog box to indicate successful or unsuccessful message.

AlertDialog.Builder builder = new AlertDialog.Builder(this);

Step 7: Prevent the phone from sleeping

2) <u>MODULE 2: BLUETOOTH- CONNECT THE AGRIBOT AND THE REMOTE</u>

Step 1: Connect to Bluetooth module.

Step 2: Detect the list of devices.

Step 3: Check if visible devices are greater than zero.

Step 4: Check if the Bluetooth adopter is equal to NULL. Terminate the activity if it is

equal to NULL.

```
if (mBluetoothAdapter == null)
{
finishDialogNoBluetooth();
return;
}
```

Step 5: Continue the process if the Bluetooth adopter is not equal to NULL.

```
mConnectThread = new ConnectThread(device);
mConnectThread.start();
setState(STATE_CONNECTING);
```

Step 6: Start pairing of devices

3) <u>MODULE 3: AGRIBOT- ESTABLISH CONNECTION AND PERFORM ACTIVITY</u>

Step 1: Check whether DeviceListActivity returns with a device to connect.

Step 2: Get the device MAC address.

```
String address = data.getExtras()
.getString(DeviceListActivity.EXTRA_DEVICE_ADDRESS);
```

Step 3: Get the BLuetoothDevice object.

```
BluetoothDevice device = mBluetoothAdapter.getRemoteDevice(address);
```

Step 4: Attempt to connect to the device if the request has been received.

```
mSerialService.connect(device);
```

Step 5: Once device connected perform the activity based on the id received.

1. *READ THE SELECTED ACTIVITY AND SEND THE ACTIVITY ID TO AGRIBOT*

THE FOLLOWING PROCEDURE IS FOLLOWED:

Step 1: The Farmer selects an option from the list of activities.

Step 2: OnClickListener method is activated and implemented.

Step 3: The ID associated with that particular activity is fetched and the passed via RF

signals onto the Agribot.

The following code snippet shows the implementation of above-mentioned steps.

```
importRemote.app.Activity;
importRemote.os. Bundle;
importRemote.os.Handler;
importRemote.os.Message;
importRemote.preference.PreferenceManager;
importRemote.view.KeyEvent;
importRemote.view. Menu;
importRemote.view.MenuInflater;
importRemote.view.MenuItem;
importRemote.view.View;
importRemote.view.View.OnClickListener;
importRemote.view.Window;
importRemote.view.inputmethod.InputMethodManager;
importRemote.widget.Button;
importRemote.widget.TextView;
importRemote.bluetooth.BluetoothAdapter;
importRemote.bluetooth.BluetoothDevice;
importRemote.content.Context;
importRemote.content.DialogInterface;
importRemote.content.Intent;
importRemote.content.SharedPreferences;
importRemote.content.res.Configuration;
importRemote.util.Log;
importRemote.widget.Toast;
public class BlueTerm extends Activity implements OnClickListener {
// Intent request codes
private static final int REQUEST_CONNECT_DEVICE = 1;
private static final int REQUEST_ENABLE_BT = 2;
private static TextViewmTitle;
// Name of the connected device
private String mConnectedDeviceName = null;
public static final boolean LOG_CHARACTERS_FLAG = DEBUG && false;
public String TAG="BlueTerm";
public static final boolean LOG_UNKNOWN_ESCAPE_SEQUENCES = DEBUG &&
false;
public static final String LOG_TAG = "BlueTerm";
public static final int MESSAGE_STATE_CHANGE = 1;
```

```
public static final int MESSAGE_READ = 2;
public static final int MESSAGE_WRITE = 3;
public static final int MESSAGE_DEVICE_NAME = 4;
public static final int MESSAGE_TOAST = 5;
public static final String DEVICE_NAME = "device_name";
public static final String TOAST = "toast";
privateBluetoothAdaptermBluetoothAdapter = null;
private static BluetoothSerialServicemSerialService = null;
private static InputMethodManagermInputManager;
privatebooleanmEnablingBT;
privatebooleanmLocalEcho = false;
privateintmControlKeyId = 0;
privateint i;
public static final int WHITE = 0xffffffff;
public static final int BLACK = 0xff000000;
public static final int BLUE = 0xff344ebd;
private static String[] CONTROL_KEY_NAME;
privateMenuItemmMenuItemConnect;
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
if (DEBUG)
getResources().getStringArray(R.array.entries_controlkey_
preference);
requestWindowFeature(Window.FEATURE_CUSTOM_TITL
E);
setContentView(R.layout.main);
getWindow().setFeatureInt(Window.FEATURE_CUSTOM_T
ITLE,
R.layout.custom_title);
if(v.getId() == R.id.Load2ON)
{
cmd =74;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
if(v.getId() == R.id.Load2OFF)
{
cmd =68;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
if(v.getId() == R.id.Load3ON)
{
cmd =69;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
if(v.getId() == R.id.Load3OFF)
{
cmd =70;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
if(v.getId() == R.id.Load4ON)
{
cmd =71;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
if(v.getId() == R.id.Load4OFF)
{
cmd =72;
buffer[0] = (byte)cmd;
mSerialService.write(buffer);
}
```

2. *PERFORM THE ACTIVITY BASED ON RECEIVED ID*

Step 1: Read the ID written in the Buffer.

Step 2: Compare the ID with the list of IDs present. When match is found switch on the

corresponding motor associated with that particular ID.

Step 3: Perform the activity until stop signal is received.

The following code snippet shows the implementation of the above mentioned steps.

```
sbit IN2N=P1^3;
sbit IN3=P1^4;
sbit IN4=P1^5;
sbit IN5=P0^0;
unsigned char ch,bh;
{
unsignedintj,k;
for(j=0;j<i;j++)
for(k=0;k<1275;k++);
}
voidserial_init()
{
SCON=0x50;
TMOD=0x20;
TH1=0xfd;
```

```
TL1=0xfd;
TR1=1;
}
void send(unsigned char ch)
{
SBUF=ch;
while(!TI);
TI=0;
}
unsigned char receive()
{
while(!RI);
bh=SBUF;
RI=0;
returnbh;
}
```

3. RESULTS AND DISCUSSION:

It is important to test the performance of agricultural robot to check if hardware and software operate in perfect coordination without any error. Testing is a process of exercising software with the objective of ensuring that the software system meets the anticipated requirements, user expectations and does not fail in an unacceptable manner.

The following are the design of the test cases that validate the functioning of the internal program logic and that the inputs produce valid outputs.

TEST CASE: ACTIVATING BLUETOOTH

Test Case ID:	Activate Bluetooth
Description:	The application requires Bluetooth to be enabled for working
Input:	Pair the Bluetooth of the Remote with the Bluetooth of Robot
Expected Output:	Connection established. Devices paired
Actual Output:	Connection established. Devices paired
Result:	Success

TEST CASE: SELECT AN ACTIVITY (BASIC OPERATION)

Test Case ID:	Select any activity from a list of options
Description:	To select an activity from a list of activities. The basic operations are forward, reverse, left, right, stop.
Input:	Choosing of an activity
Expected Output:	Movement of robot
Actual Output:	Visible Movement of the robot based on activity selected.
Result:	Success

TEST CASE: SELECT AN ACTIVITY

Test Case ID:	Select any activity
Description:	To select an activity from a list of activities. The main operations are seeding, harvesting, water pumping
Input:	Choosing of an activity
Expected Output:	Movement of robot
Actual Output:	Visible Movement of the robot based on activity selected.
Result:	Operation performed successfully

In general testing of overall agricultural robot was completed successfully with some technical glitches. Wherever the technical problems or deviation from expected operations were observed, the immediate troubleshooting was conducted, and accordingly corrective measures were implemented. At the end the all test cases resulted in successful operation.

4. CONCLUSIONS:

This section should comprise a brief statement of the major findings and implications of the study. It is not the function of this section to summaries the study; this is the purpose of the abstract. New information must not be included in the conclusions.

REFERENCES

1. J. Katupitiya, R. Eaton, A. Cole, C. Meyer and G. Rodnay, "Automation of an Agricultural Tractor for Fruit Picking," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 3201-3206, doi: 10.1109/ROBOT.2005.1570603.

2. Herbert Schildt, (2007). *JAVA-The Complete Reference Java & J2EE, Seventh Edition*. McGraw-Hill

3. Hajjaj, S. S. H., & Sahari, K. S. M. (2014). Review of research in the area of agriculture mobile robots. *In The 8th International Conference on Robotic, Vision, Signal Processing & Power Applications* (pp. 107-117). Springer, Singapore.

4. Bogue, R. (2016). Robots poised to revolutionise agriculture. *Industrial Robot: An International Journal*.

5. Kawamura, N., & Namikawa, K. (1988). Robots in agriculture. *Advanced robotics*, *3*(4), 311-320.