IC201P: Design Practicum (Feb 2017)

Sports Field Marking Robot

Team Number 17

Product Design Report

Team Members :

Ankit Amrit Raj	(B15107)	9479088619
Abhishek Sonal	(B15204)	9005759878
Anand Mohan	(B15307)	9736259326
Revu Sri Harsha	(B15328)	7382217573
Dheeraj Meena	(B15314)	7827807695
Vikas	(B15424)	9416835645

Team Mentor : Dr Dhiraj V. Patil

Contents

Problem Definition

What is the problem?

We all enjoy playing games with friends and family, but without the field lines on the ground, there would be no game at all! Line marking process of sports ground is time consuming, manual and resource intensive task. At present this is done either manually by hands or by using a lawn mowing technique in order to mark the lines. This require a lot of expertise and it also leads to inconsistency in the final result.

Who has the problem?

This project will be beneficial for many sports organizations as well other schools and colleges who frequently organize sports events. This robot will be a one time investment and thus in long term it would be worthwhile to buy. At present we do not have any such kind of robot in any sports organization, colleges or schools which can mark the lines on playing area precisely and without consuming much time.

What are the goals?

Using microcontroller, magnetometer and a spraying mechanism this sports ground marking robot will draw the field line on the ground based on the input about the dimensions and shape of the ground given by the operator. This will save both time and human efforts and would have a great precision.

Literature review

Current State

Currently there are two field marking robots which have been already made. One of them has been made by a group of students in India. This model uses "Theo jansen principle" and has four legs. It is operated manually by hand . Other one is "Intelligent marking ApS", a company founded in denmark. Its field marking robots which uses GLONASS satellite for finding the coordinates of the path on which it has to move.

Difference in our model

To avoid inaccuracy in our design we have used a magnetometer in the place of a GPS sensor. Small amount of deviation from a path can change the entire design of the field. Magnetometer will be connected to our microcontroller and based upon the readings we can make our robot to move in a straight line or a circle.

Gap/Problem in existing literature

Two bots which can mark the field lines are based on different techniques. One of the field marking robot is based on " Theo Jansen principle". When the bot moves the lines get marked from the back side of the robot and the movement of the robot is done manually, it is not autonomous.

The second one uses GPS and mark the field lines with +/- 2 cm accuracy. This bot comes with a interchangeable lithium ion battery so that it can move for a long time.

Reference : https://www.youtube.com/watch?v=l_ya_UzEO8s
Reference:http://www.intelligentmarking.com/

Solution Methodology

Picture of final product

For the problem defined above we have decided to build an autonomous robot that can mark the sports field lines accurately on the ground upon giving instruction by the operator through an android device application over a bluetooth network. Operator has to enter which sports ground he/she wants the robot to mark give its dimensions (length and width of ground) and the rest will be done by the robot.

For this project the major components that we will be using are microcontrollers (Raspberry Pi/Arduino), bluetooth module for giving input to microcontroller from an android application, magnetometer to make sure that we are moving in the right direction, stepper motors and a spray mechanism which spray paint only for the time when we need it. After getting input from the user over bluetooth network, the microcontroller will set the readings coming from magnetometer as the reference angle and all the later calculations and instructions are done with respect to this. After setting the reference position and direction, the microcontroller will define a path for the robot on which it have to move, like for how much distance it has to move straight and at what time it has to take left/right turns. During the time for which the robot is on its correct path the spray mechanism sprays the paint on the ground and for the rest of the time it will not.

In our design we have decided to form two compartments in our robot a top and a bottom. In the bottom compartment we will be placing all our circuits. This compartment will have proper air in/out facility because mostly this product will be used during day-time hence overheating of circuit will be a issue as result microcontrollers and sensors will not process data correctly which led to inaccuracy in marking. The upper chamber will be used as a container to hold the paint, thus in this way the paint which we are using being cold also act as a coolant for the circuit in lower chamber and avoid overheating of the circuits.

For marking a perfect 90^0 turn instead of making our robot longer in length we would be making it longer in width so that rear and front wheels can be placed closer to each other thus point rotation(rotating by standing on the same point) will possible, in this way we can avoid marking curved corners and make an almost perfect 90^0 corners.

Now let's see how the robot will mark a volleyball field ground of dimensions (length = 18m and width = 9m). This dimension will be given by the operator to the robot via bluetooth network.

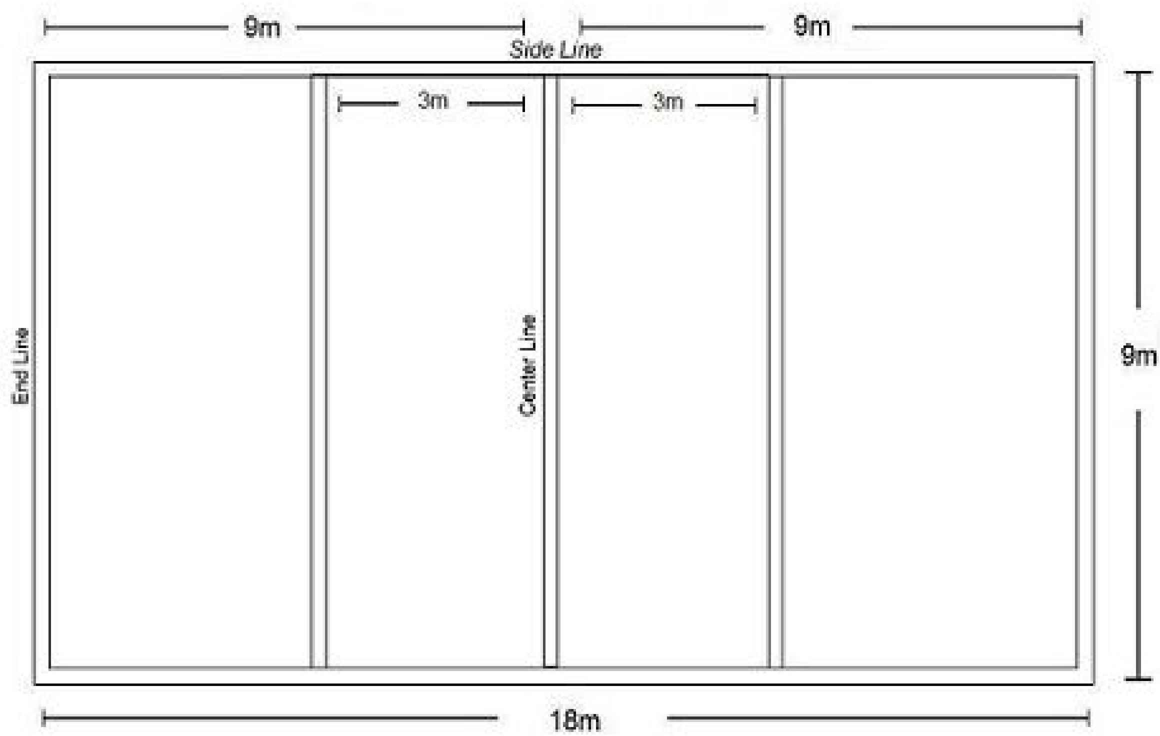
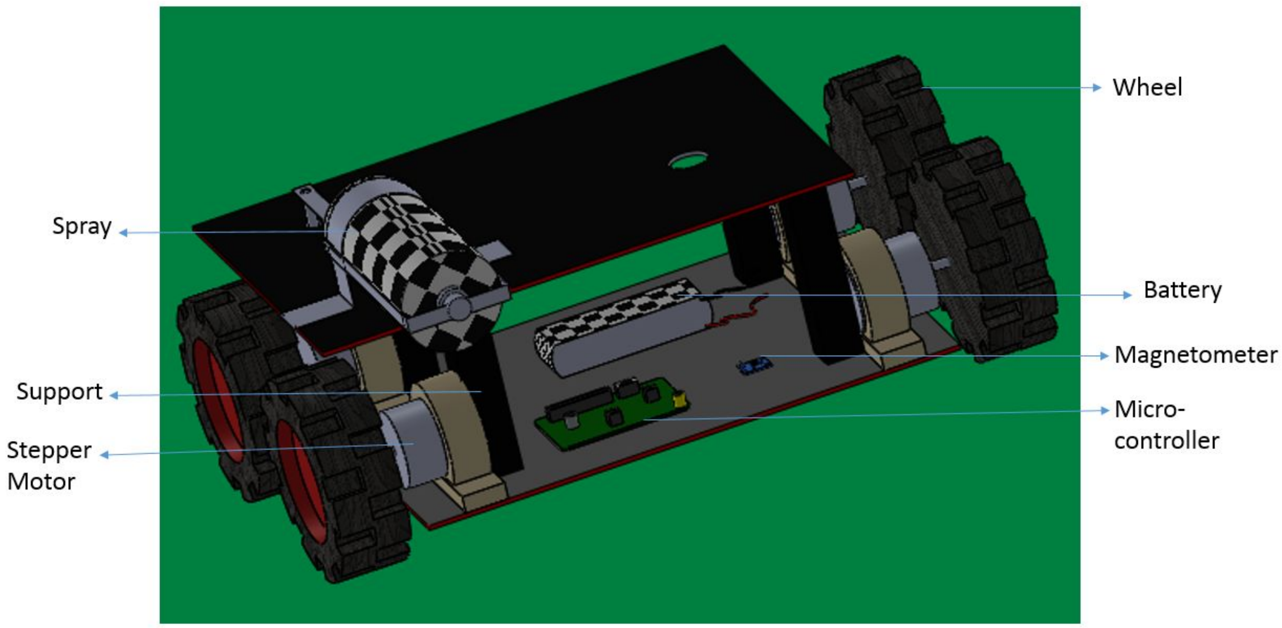


Figure 1

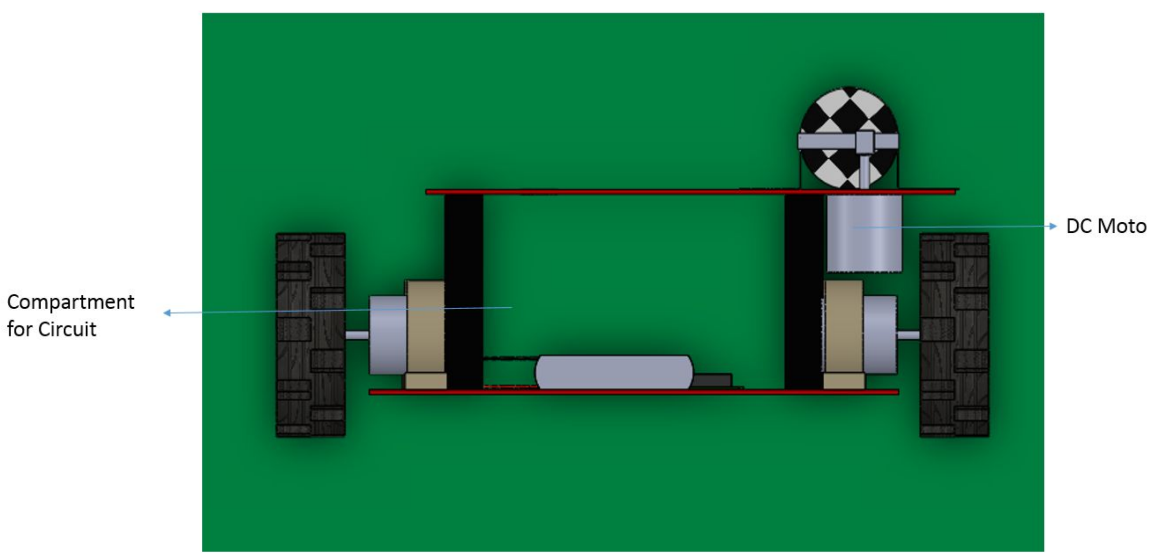
Now, after getting length and width, the microcontroller will automatically calculate the length of all sections of the ground(scaled sections as length and breadth can be different) like length of centerline, length of attack-line, distance of attack line from the centerline. Let's say initially the operator has placed the robot at the lower left corner of the ground. Then, on pressing the start button the robot will start moving straight for 18m with the help of reading coming from magnetometer. After covering 18m it will take point rotation of 90^0 left turn. And again in it moves straight for 9m. In this way it will mark the outer lines of the ground with the help of spray mechanism. After this the robot will again reach its starting point i.e. lower left corner. Now it will move 9m in straight line without spraying the paint and reach the point from where it should start marking the center line by taking a 90^0 point left rotation and move 9m straight to mark the center line. In the similar way it will traverse the field and at the same time marking the field lines.

Product Design

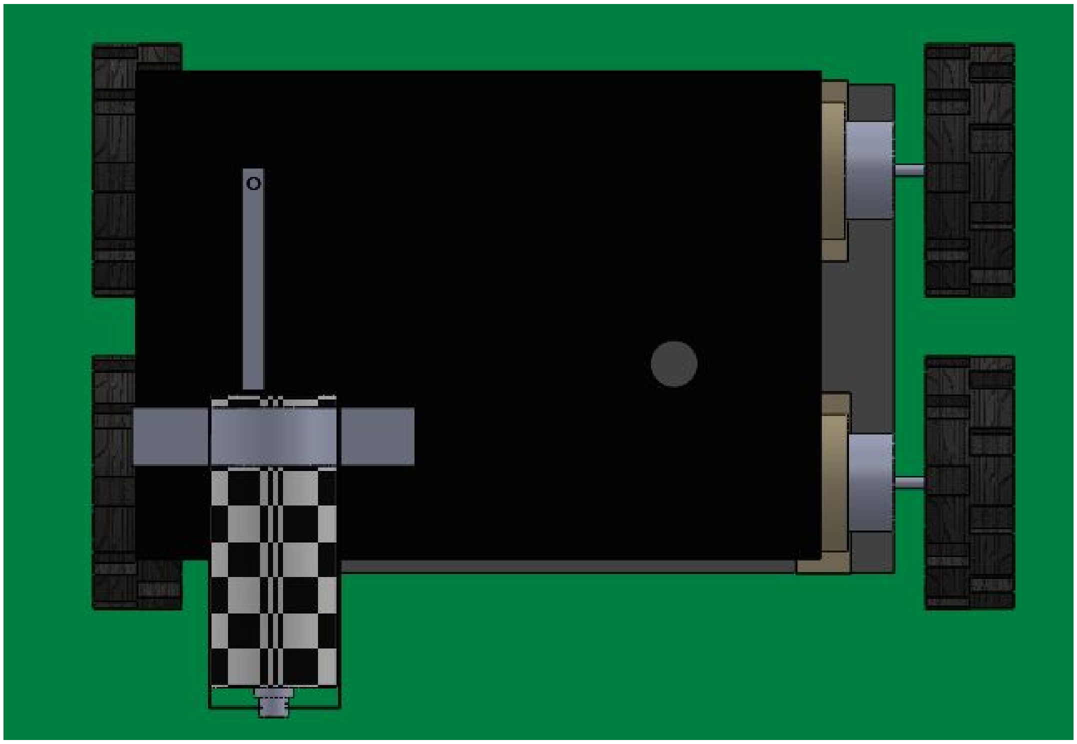
- **CAD Model**



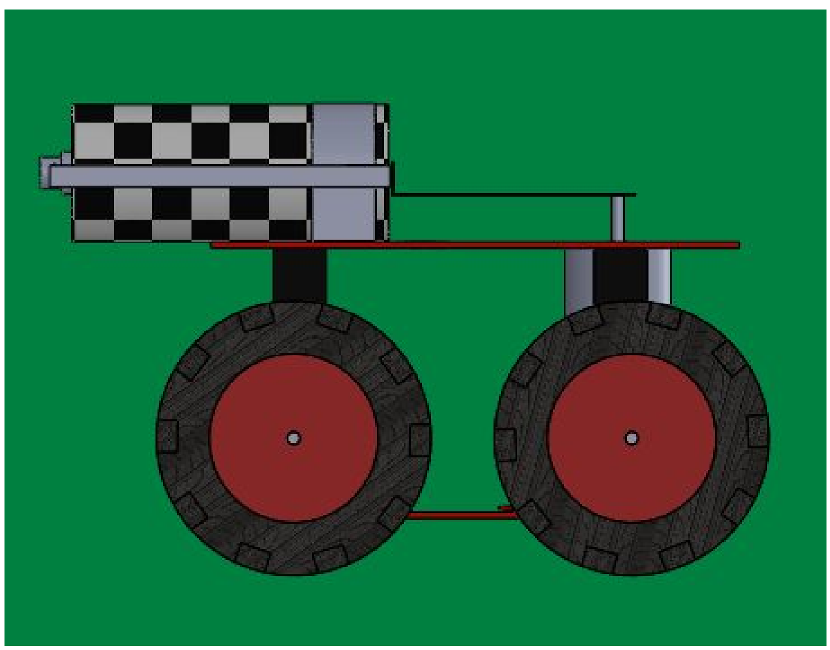
Isometric view



Front View



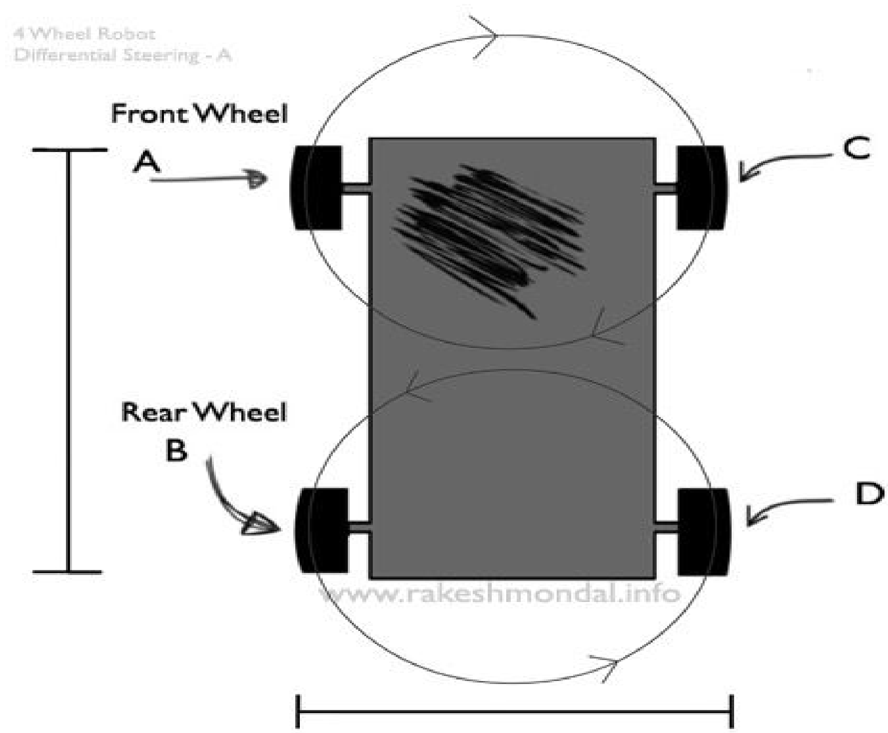
Top view



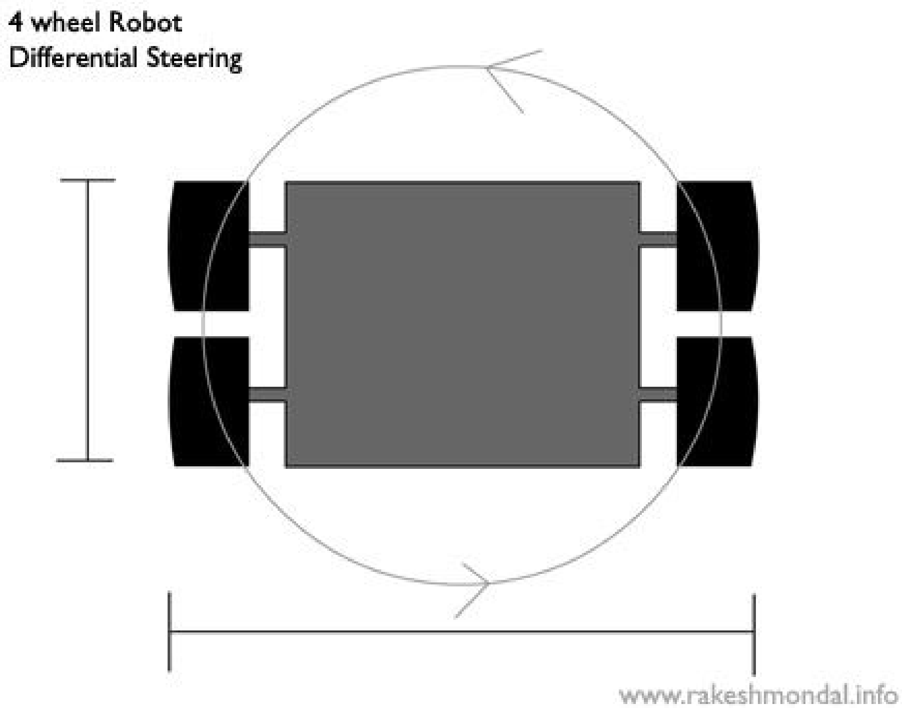
Side View

- **Stability and turning of robot**

Since our robot has to move in sports field so after completing one straight line it will have to take 90^0 turn so our rotation should be smooth and there should curved corner. For turning purpose we will have to use differential steering. Differential steering is nothing but creating difference in the speed of the adjacent left or right wheels to take turns. For a complete 360^0 spin at its pivot position the two wheels on each adjacent side (left and right) must rotate on opposite spin at same speed. This approach might appear easy, but it is not always efficient with all kinds of 4 wheel chassis. For the rotation of 90^0 rear wheels B and D spins in opposite direction which are connected to the Motor, (the wheel B rotates backwards and D forward) here by trying to make a left circular motion. But since the distance of the forward wheels A and C is too high from the rear B and D , hence it cuts out the circular pull generated by the back wheel which has a very less diameter (distance between B and D is the diameter). Hence front set of tyre will skid on ground causing loss of energy and wear and tear in tyres.



So for effective turning of robot we will use drawing below . In this design robot takes a zero-radius 360 degree turn (pivot turning), the front wheel gives almost negligible resistance to the overall circular motion generated by the rear wheels. Advantage of this design are :- No loss of torque, fast turning, no wheel slips while turning, better traction on wheels, cost saving by use of regular wheels.

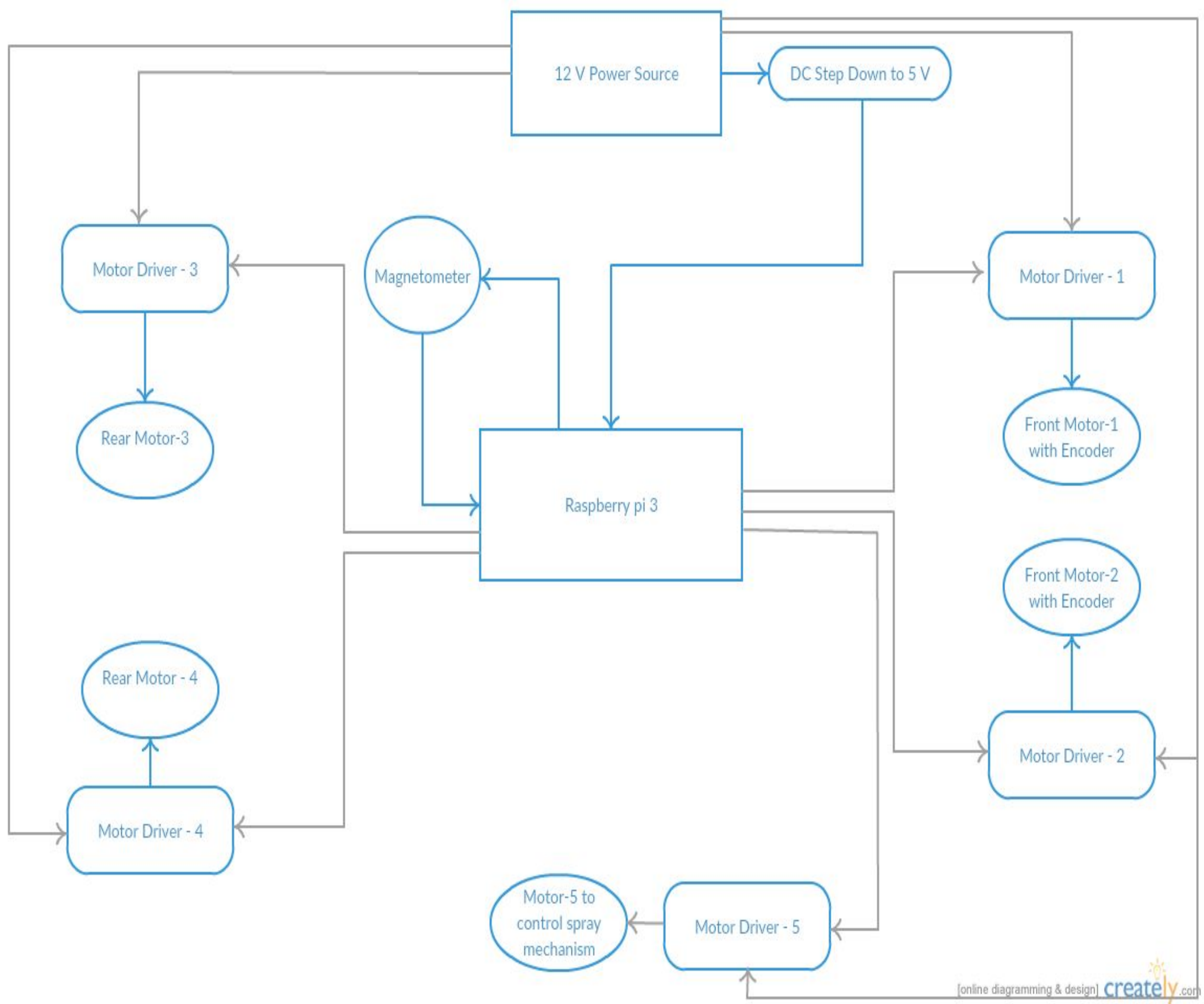


So Decreasing the distance between the front and rear set of wheels, results in better zero-radius turning in any direction with 4 wheel drive robot. We keep these points in mind while designing our robot.

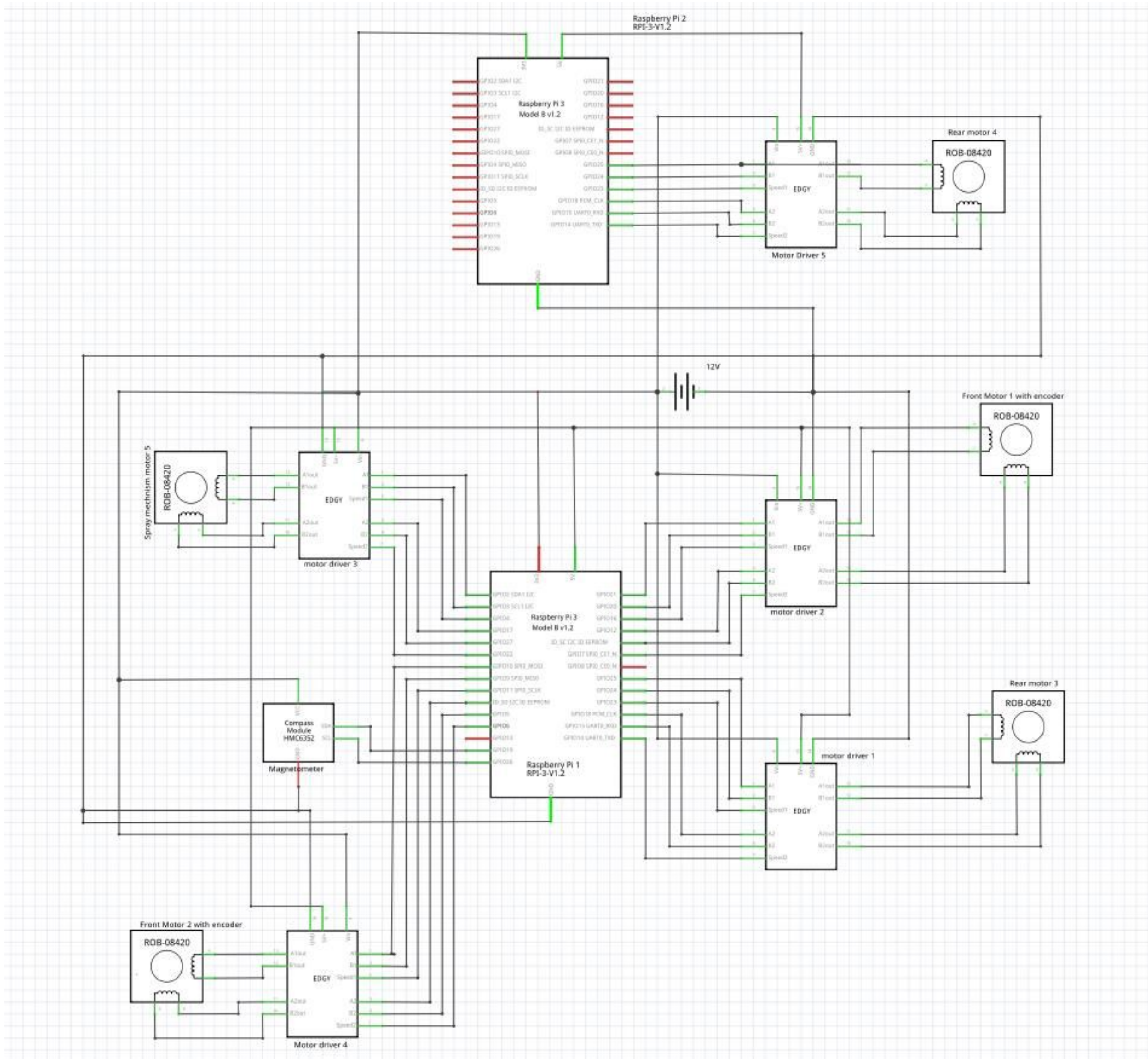
For stability of robot we will try to keep paint and battery such that their moment would be balanced about about horizontal axis passing through mid wheels A and C. Also we will try to keep paint near to this axis to minimize its moment.

Block Diagram of Circuit

Following is the simple block diagram of the circuit which we are going to make. The 12 V power supply coming from battery is lowered to 5 V before supplying it to raspberry pi, however 12 V supply is directly given to all five motor drivers. All the five stepper motor are controlled by raspberry pi via motor drivers. Magnetometer is connected to raspberry which also send back data of current direction of bot w.r.t. North back to raspberry pi for further calculations.



Circuit Diagram



Above figure shows the actual connections of different components stepper motors, motor drivers, magnetometer, power source and raspberry pi.

Algorithm

The following algorithm we will be going to use to align the robot in the right direction while spraying paint from outlet onto the ground.

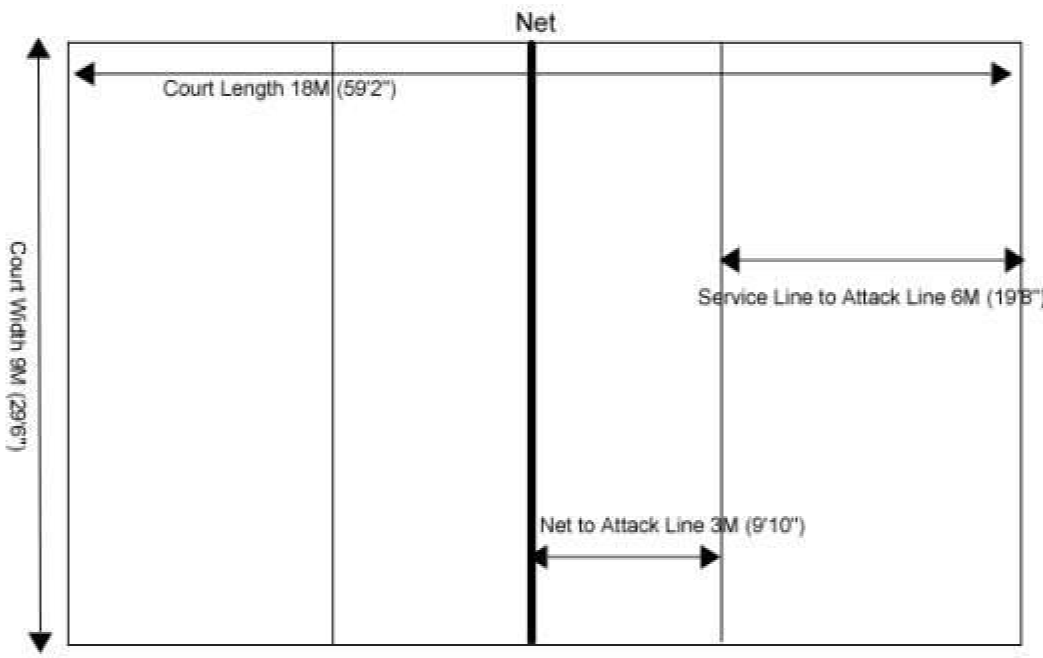


Figure 2

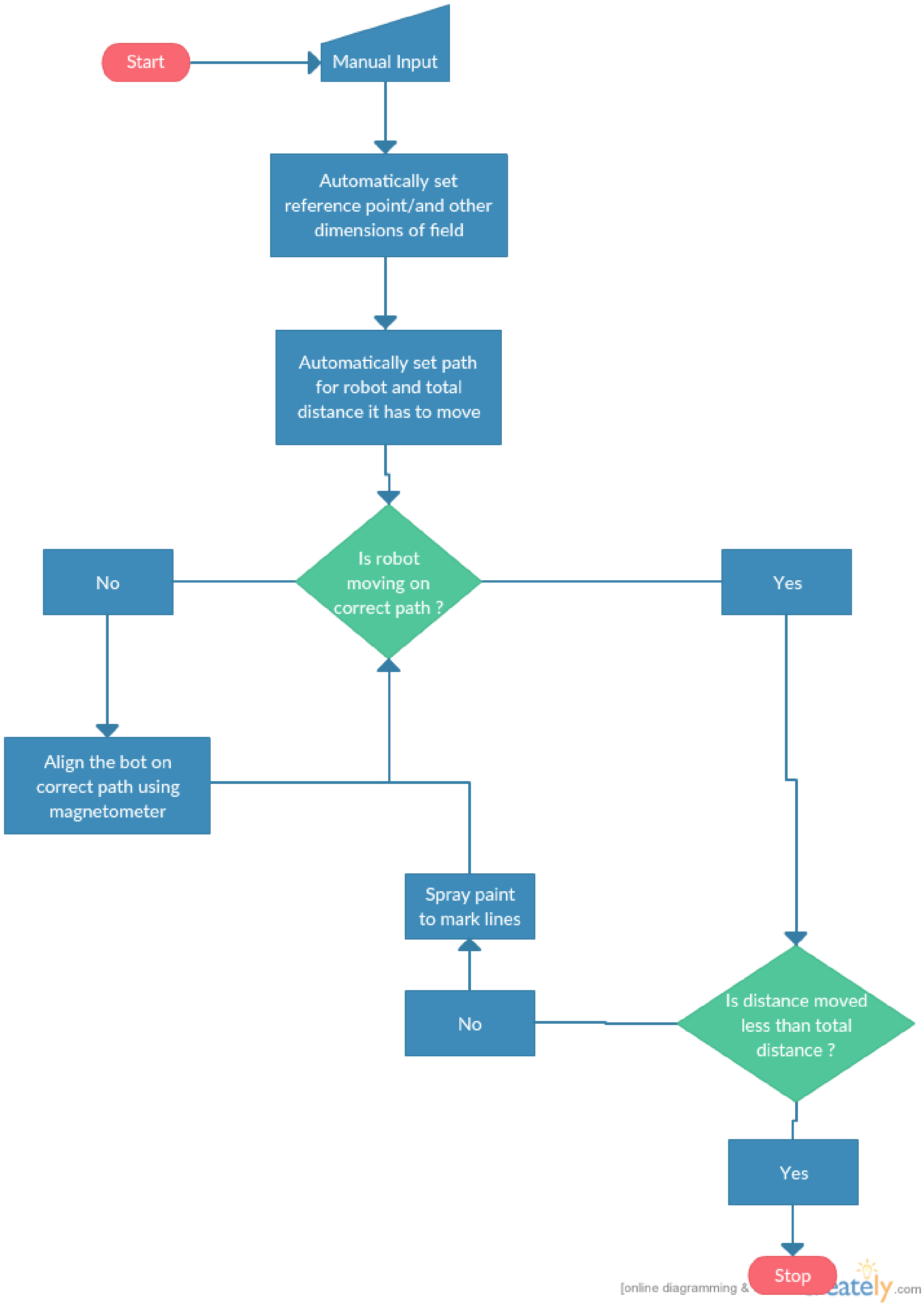
Algorithm : For marking a volleyball field ground. Place the robot at lower left corner of the ground i.e. point **A**(start point) refer Fig2 .

```
/*--------------------------------Function Declaration---------------------------------*/
getInput() ;                    // to get input from user
move(int d) ;                   // move 'd' distance in a straight line
turn_left(int a) ;              // turn 'a' degree left
turn_right(int a);              // turn 'a' degree right
/*-------------------------------------------------------------------------------------------*/

/*---------------------------------- Global Variables -----------------------------------*/
Length = getInput();            // user have to input length of field
Width   = getInput();           // user have to input width of field
LENGTH_ES = Length / 7.5 ;
```

```
LENGTH_FQ = Length / 20 ;
LENGTH_ST = Width  / 1.75 ;
LENGTH_QR = Width / 3.8 ;
RADIUS_CIRCLE = Length / 13.33 ;

TOTAL_DISTANCE = ( Length * 2 ) + (Width * 3) + (LENGTH_ES * 4) + 2 * π *
RADIUS_CIRCLE +  ( LENGTH_FQ * 4 ) + ( LENGTH_QR * 2 ) + ( LENGTH_ST *
2 ) +
DISTANCE_TRAVEL = 0;
/*---------------------------------------------------------------------------------------------------*/

While ( DISTANCE_TRAVEL <= TOTAL DISTANCE )

        /*---------------------------------Draw outline of field------------------------------*/

        move( Length ) ;                 // move Length distance straight
        turn_left( 90 ) ;                // take a left turn of 90^0
        move ( Width ) ;
        turn_left ( 90 ) ;
        move ( Length ) ;
        turn_left( 90 ) ;
        move  ( Width ) ;                // At this point we will be at point A
        DISTANCE_TRAVEL += 2 * ( Length + Width) ;
        /*-----------------------------------------------------------------------------------------*/

        /*------------------------------Drawing Left Goal Area-----------------------------*/

        move( -(Width - LENGTH_ST) / 2 ) ;              // move 15 m backwards H
        turn_left( 90 ) ;                               // align along HT
        move( LENGTH_ES ) ;
        turn_left( 90 ) ;                               // align along TS
        move( LENGTH_ST ) ;
        turn_left( 90 ) ;                               // align along SE
        move( LENGTH_ES ) ;
        DISTANCE_TRAVEL += 2 * LENGTH_ES + LENGTH_ST  ;
        turn_left( 90 ) ;
        move ( (LENGTH_ST - LENGTH_QR) / 2 )     // Reached point F
        turn_left ( 90 );                               // Align along FQ
        move ( LENGTH_FQ );
```

```
        turn_right( 90 ) ;                              // Align along QR
        move ( LENGTH_QR );
        turn_right ( 90 );                              // Align along RG
        move ( LENGTH_FQ ) ;
        DISTANCE_TRAVEL += 2 * LENGTH_FQ + LENGTH_QR ;

/*------------------------------------------------------------------------------------------*/

/*-------------------------------------Marking center line----------------------------*/
        // currently the robot is at point G
        move(LENGTH_GA);
        turn_left ( 90 ) ;
        move ( Length/2) ;                              // Reached V
        turn_left ( 90 ) ;
        move ( Width ) ;                                // Reached U
        move ( -(Width/2 - CIRCLE_RADIUS) )             // Reached intersection of circle
                                                                  and halfway line
        DISTANCE_TRAVEL += Width;

/*------------------------------------------------------------------------------------------*/

/*-------------------------------Marking Centre Circle-------------------------------*/
        int counter = 0;
        while( counter <= 360 )
                turn_left (2);
                move( 2 * π * RADIUS_CIRCLE / 180 )
                counter = counter + 2;
        end
/*-------------------------------------------------------------------------------------------*/
// Similarly we mark the right goal area.
/*-------------------------------------------------------------------------------------------*/
end
```

Flowchart



Start
Manual Input
Automatically set reference point/and other dimensions of field
Automatically set path for robot and total distance it has to move
Is robot moving on correct path ?
No
Yes
Align the bot on correct path using magnetometer
Spray paint to mark lines
No
Is distance moved less than total distance ?
Yes
Stop
[online diagramming &
ately.com

Components Required

- Wheels (4)
- Arduino Mega(1)
- Motor Driver(5)
- Stepper Motor (6)
- Spray (1)
- Jumper Wires (approx 50)
- Magnetometer (1)
- Li-Po Rechargeable Battery(1)
- PCB (1)
- Soldering Iron (1)

Cost of the Product

Item	**Cost/piece**	**Total Cost**
Wheels	45	180
Motor Driver	200	1000
Stepper Motor	400	1600
Spray	220	220
Jumper Wires(50)	-	110
PCB	20	20
Li-Po battery	2000	2000
Magnetometer	320	320
DC geared Motor	400	400
Arduino Mega	1000	1000

Total Cost : Rs. 6850

Proposed Manufacturing/Assembly process

- Firstly we have to make bottom compartment of the chassis of dimensions shown in CAD models using metal sheet. Then we have to fix two stepper motors in the front and two at the rear side of the chassis using nuts and bolts along with some adhesives. For fixing motors we also have to drill four circular holes in the chassis from which shaft of the motors comes out, onto which wheels will be fixed.

- Slits will be made on the backside of the bottom compartment which acts as a ventilation system for the circuit, preventing the circuit from overheating.

- Now we will safely place all the circuits in the bottom chamber .

- After this we have to plastic coat the bottom compartment to prevent any paint from top chamber entering the lower chamber.

- Now we make the upper partition of the robot using metal sheet. A hole is drilled into this section for the circular pipe of spray. This pipe is fixed into the drilled hole with adhesives.

- Paint is poured into this upper section after which this compartment is also covered from top using metal sheet.

- Now for a good outer design of the product it will be painted from outside too.

Limitations:

- Our current model can not move on curve lines(like circles) , it can mark only straight lines . So it can be used only for fields like badminton , volleyball etc.
- Since we are using stepper motor to calculate the distance moved by the robot hence the distance calculation may be inaccurate if we try to move it on uneven surfaces.
- After every 200m of painting , the paint spray has to be replaced.
- As we are using aerosol spray , so when the spray bottle is getting empty , the spray sprays air with the paint . Hence lines not marked properly.

Problems Faced:

- Operating the spraying mechanism , so that when the robot is taking a 90 degree turn , spray is off and when the turn is completed the spray turns on .
- Robot was cutting some of the marked lines.
- Placing the spray mechanism at the right place so that the error is minimum while taking 90 deg turn.
- Skidding of the wheels while turning.
- To control the amount of paint released by the spray so that all lines are of same width.
- To make sure that our robot moves straight and take perfect 90 degree turn.

Future Improvement:

- One of the major improvement can be the use of white powder instead of aerosol spray paint.
- The power gives better finishing at the edges than the spray paint, also the height of the spraying mechanism does not matter like in the the case of the spray paint.
- The spray can be installed on the side of the field marking robot so that the bot does not cuts the marked line.
- At turns there was skidding of the wheels so this problem can be resolved if we use motors having high torque.
- On the major improvements can be use of big wheels along with high torque motors and optical encoders so that our robot does not face any problem while moving on a rough surface.
- A cooling mechanism can also be installed so as to cool motor drivers.

Impact on Society

Our product is mainly beneficials for schools, institutes and sports organisations who conduct sports related event on a regular interval. Currently a huge amount of time and labour is spent every week by such organisation in order to mark sports field lines properly and accurately. On an average , to mark any ground at-least 3hrs are needed along with 3-4 persons who will be measuring distances and doing other calculations , this is a very hectic and time-consuming task. And every time organisation has to pay the labour cost. With the help of our robot which costs only Rs 7000, and can be controlled by only one person this hectic task of field marking can be made easier. By connecting the smartphone with the robot via bluetooth one can tell the robot which ground he/she want to mark and what are the dimensions of the ground and the rest is taken care by the robot. Moreover, its application is not only limited to marking on sports field, it can be use to mark center white of roads, we can also make various other designs on the ground according to our interests.

References

- http://www.intelligentmarking.com/
- https://creately.com
- https://www.pololu.com/picture/view/0J5645
- http://www.rakeshmondal.info/4-Wheel-Drive-Robot-Design