Paper ID: AM-10

Development of Autonomous Line-Following Robots for Solving a Maze Problem

Md. Rokunuzzaman, Asif Ahmed Shishir, Md. Saeed Sharman, Amit Roy*
Department of Mechanical Engineering
Rajshahi University of Engineering & Technology (RUET), Rajshahi-6204, Bangladesh
E-mail: royamit.me@gmail.com

Abstract

This paper describes about the development of two autonomous robots (Bots) which cooperate each other to solve a maze. In this work, we have prepared two autonomous line-following robots which can communicate with each other so as to simultaneously solve two mazes via line following. We have been provided with turn indicators on 1 maze. A maze is formed by concentric circles connected to each other at some specific locations. There are two similar mazes with one maze having dimensions in a ratio greater than 1 radially. The maze is made up with white and black strips. In the maze solving process, First Bot has followed white strip to reach the center of the maze such that, if a black strip is encountered adjacent to line on left side, the bot will take the next left turn and If a black strip is encountered adjacent to line on the right side, the bot will take the next right turn. End of the maze is indicated by innermost black circle. This bot has communicated to the second bot and guide it to reach the end of respective maze. The second bot has received signals from the first bot and follow the line to reach the end of the maze.

Keywords: *Autonomous robots, Maze solving, Line-following method*

1. Introduction

iARC: International Autonomous Robotics Competition is a great and entertaining way to enhance the learning process in the field of robotics. Students would gain first -hand experience in constructing and programming robots for competition, as well as allowing them to understand further the concept Mechatronics System Design [4].Realizing this, iARC is organized by IIT Kanpur, one of the premier technical institute of India and Techkriti the annual technical and entrepreneurial festival of IIT Kanpur in conjunction with the Mechatronics System. This project-based course requires the students to build maze solving robot via line following and then to compete with each other. Its performances in this competition is measured and recorded as part of the evaluation scheme. In this competition, students are divided into groups of one, two, or three people maximum four and they are given freedom to design their own robots in terms of hardware and software used. For our group of two people, we developed maze solving robots with different kinds of drive trains. The requirements include that many equipment and sensors have to be synchronized, needing powerful real-time capabilities, under limited onboard batteries power [1].

2. Methodology

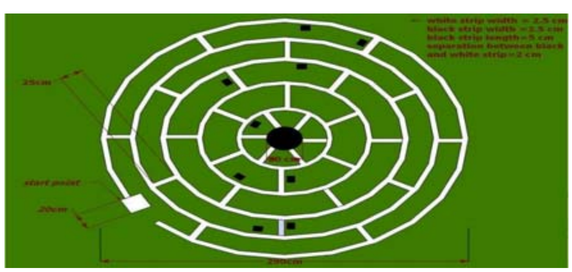

Fig.1 International Autonomous Robotics Competition (iARC) Arena

The rules of the International Autonomous Robotics Competition (iARC) are as below:

Arena Specifications:

a. White strip width = 2.5 cm
b. Black strip width =2.5 cm
c. Black strip length=5 cm
d. Separation between black and white strip=2 cm
e. Outermost diameter =290 cm

Robot size:

a. Maximum width: 200 mm
b. Maximum length: 200mm
. c. Maximum height: 200 mm

• The Robot must be stable and able to move on its own. A Bot not fulfilling these criteria will be disqualified.
• Both the bots should be able to follow the line according to event specifications. In addition the bots should be able to communicate with each other. Communication can be wired or wireless
• If communication is through wiring the team will be entirely responsible for problems due to entanglement of wires. Teams should bring sufficient length of wire for the same.
• The wire should remain slack at all times.
• Each team has to bring its own power supply for its robots.
• The voltage difference between any two points on the bot must not exceed 24 volts.
• Teams are advised to use an on-board power supply. In case they are using external power supply they will be responsible for any problem created by entanglement of wires.

3. Materials

3.1 Mechanical construction

The development of our robot (see Fig. 2) utilizes plastic as its main body together with two 2 inch wheels with each wheel attached to one Bipolar stepper motor. One Ball castor (metal) is mounted further up front to provide stability for the robot. The differential drive method is implemented to enable the robot to rotate clockwise and counter clockwise. This means that one motor would rotate while another stops to let the robot turns towards the desired angle of rotation.

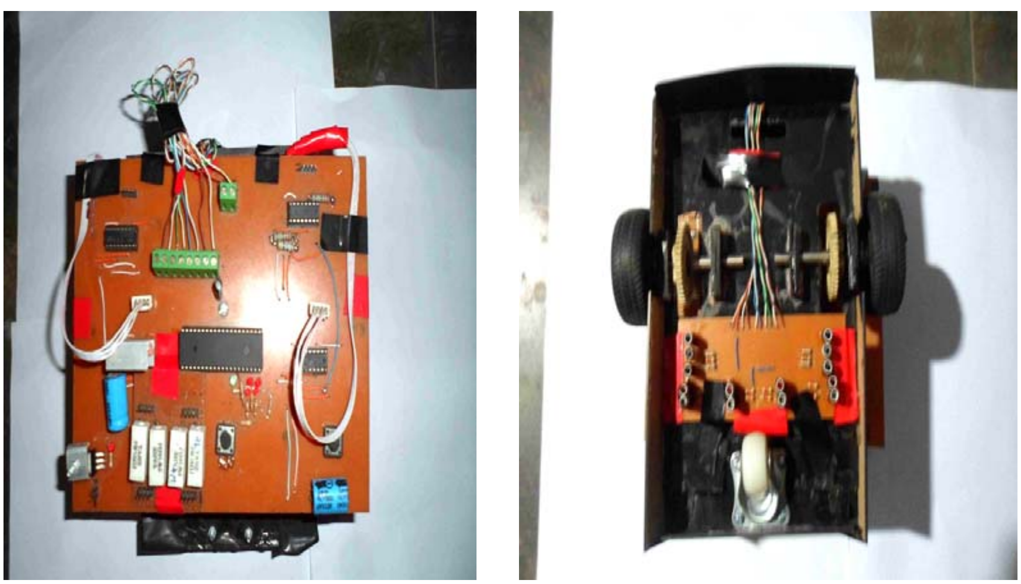

Fig. 2: Autonomous Robot with Bipolar Stepper motors as drive train

3.2 Circuitry interfacing

For the on-board microcontroller of the robot, the avr microcontroller and embedded system is implemented as the board set provides easy interfacing with sensors and actuators [5].Furthermore, the ATmega32 microcontroller can be easily programmed using the C language .As for the line tracing sensors, three light dependent resistors (LDRs) are arranged in straight line and mounted below the motors and in parallel to their shafts. Three light emitting diodes are also mounted next to each of the LDR to provide uniform light source for the sensors.

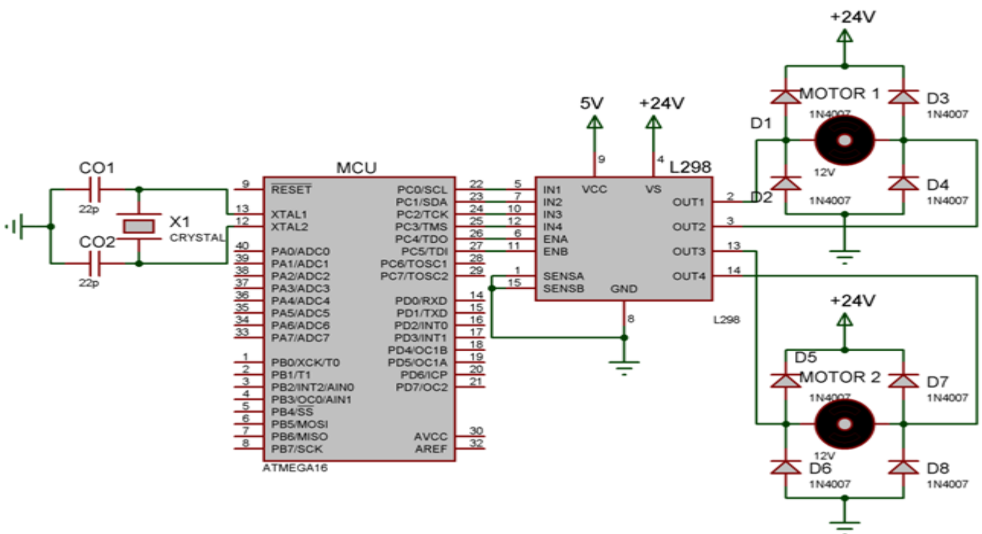

Fig.3: Circuit Diagram for motor control using L-29

3.2 Programming algorithm

The issue of control manifests itself in at least two aspects of the Robot Design project: in the development of the robots higher-level strategy from a conceptual standpoint, and in the actual programming of that strategy into the machine through the process of writing computer code. The analog signal received of each of the phototransistors is proportional to its coverage of the line, and lies between 0 and 2V. The signals of the sensors are weighted by 1, 2 and 3 for left, middle and right sensors respectively. The average of the weighted readings gives a value for deviation of the robot from the line. This average has the range between 1 and 3 when the line is at far left or far right of the robot respectively. This average is used to control the speed of the rear left and right wheels of the robot.Timer-0 and timer-2 in phase correct PWM mode are used for the left and right wheels. It is such that if the average of the weighted readings is 2, the robot is on the line and the OCR0, and OCR2 values are half the top, and correspondingly by an average value of 1 the right speed is highest and the left is 0, and by an average value of 3 the right speed is 0 and the left is highest. In between the speed (i.e. the values of OCR0 and OCR2) change linearly. It is assumed that the microcontroller frequency is to be 1 MHz, and the A/D clock to be lowest.

Coding : Proteus

```
#include<avr/io.h>
#include<avr/interrupt.h>
#include<util/delay.h>
#define left 1
#define right 2
#define s 3
#define hleft 4
#define hright 5
#define hs 6
#define stop 7
int c[6],a[6],r1,r2,r3,r4,w1=0,w2=0,k;
int speed1=233,speed2=236;
int go(int n)
    {
        switch(n)
            {
                case 1:
                    PORTB&=~((1<<PB0)|(1<<PB2));
                    PORTB|=(1<<PB0);                //PB0 connect to the right motor on to turn left
                    _delay_ms(10);
                    break;
                case 2:
                    PORTB&=~((1<<PB0)|(1<<PB2));
                    PORTB|=(1<<PB2);                // PB1 connect to the left motor on to turn right
                    _delay_ms(10);
                    break;
                case 3:
                    PORTB|=(1<<PB0)|(1<<PB2); //PB0 and PB1 on for go straight
                    _delay_ms(10);
                    break;
            }
    }
                //////////////////////////////// Function for get adc values /////////////////////////
int adcr(int adcselect)
{
//ADMUX &= 0b01000000; because AVCC as ref
    ADMUX &= 0b01000000;
    if(adcselect==0) ADMUX &= 0b01000000;
    if(adcselect==1) ADMUX |= (1<<MUX0);                //adc2
    if(adcselect==2) ADMUX |= (1<<MUX1);                //adc2
    if(adcselect==3) ADMUX |= (1<<MUX1)|(1<<MUX0);      //adc3
    if(adcselect==4) ADMUX |= (1<<MUX2);                //adc4
    if(adcselect==5) ADMUX |= (1<<MUX2)|(1<<MUX0); //adc5
    if(adcselect==6)  ADMUX |= (1<<MUX1) | (1<<MUX2);  //adc6
    if(adcselect==7)  ADMUX |= (1<<MUX1) | (1<<MUX2)|(1<<MUX0);//adc7
    ADCSRA|=(1<<ADSC);
    while(ADCSRA & (1<<ADSC));
    return ADC;
```

4. Results

Trial Video shots:

Fig 4.1

Fig 4.2

Fig 4.3

Fig 4.4

The results of our iARC robot competition are as follows:

Round	Result	Total teams
1st round	9th place	Out of 35 teams
Final Round	5th place	Out of 9 teams

5. Discussion

During the preliminary round, it has performed nicely in following the lines, but the speed is rather too slow to reach the goal. Total 9 teams were selected out of 35 teams. And we have managed to reach second round. The use of LDRs provides a great low-cost line-tracing sensor. However, it is prone to the ambient light and is also not fast enough in scanning the white lines above on green color. Quite many times the robot has diverted from the white lines and junctions since the readings from the LDRs are not fast enough. For the final rounds, a small changed arena was supplied. So we had to change our code instantly in final round. That's why we were unable to manage to reach the full arena. The use of two Stepper motors had enabled the robot to move at a higher speed with 6 volt battery supply .But we lost 4 opponent teams which had great autonomous robots with great strategies. All is not lost actually, since our robot has actually brought a good fight for the opponent robot teams. And we took 5th place in this International Autonomous Robotics Competition.

6. Conclusion

From this autonomous robot competition, there are so much knowledge and experience obtained by just building a small-sized autonomous robot. Robotics competition is indeed a great platform for students to enhance their robotics skills in terms of mechanical construction, circuit interfacing and programming. Besides that, the implementation of good strategies is undeniably important in determining the success of the game played.

7. Recommendation

Further improvements can be made by using a better Micro- processor which has better processing power, bigger memory and built-in pulse-width-modulation (PWM) function. It should be made by using DC Gear motors or Parallax servos with sufficient speed and high torque plus so their drivers would enhance the mobility of the robot. It is also recommended that the Sharp image processing sensor is to be used with an analog-to-digital converter (ADC) chip in order to supply the micro-processor with a digital input. Different configurations for the line tracing sensors could also provide better results in terms of effective scanning of the lines. Instead of three sensors, perhaps the configuration of five sensors being aligned in a semicircle shape might provide better line sensing capability for the robot. For such a speed-critical competition, perhaps it is better to implement the infrared sensors image processing sensor.

References

[1] Jun'ichi Shibata, Gen'ichi Yasuda and Hiroyuki Takai, "*A Microcontroller-Based Architecture for Locally Intelligent Robot Agents*", Proceedings of the Sm World Congress on Intelligent Control and Automation, June 15-19, 2004, Hangzhou, P.R. China

[2] Martin, F. "*Ideal and Real Systems: A Study of Notions of Control in Undergraduates Who Design Robots*", In Y. Kafaiand M. Resnick (Eds.), Constructionism in Practice: Rethinking the Roles of Technology in Learning, MIT Press, MA, 1994.

[3] McComb, G. (2000) "*The Robot Builder's Bonanza*", 2nd Ed. McGraw-Hill, New York, U.S.

[4] Miglino, O., Lund, H. H., Cardaci, M. "*Robotics as an Educational Tool*", Journal of Interactive Learning Research, 1998.

[5] http://www.parallax.com