Micromouse : Maze solving algorithm

This is my maze solving robot project which worked out pretty well. I have put up my whole project report that i submitted to my college but i have chucked out the exact code. **if i get a good response and demans then i will surely give you all the exact working code of my project.**
If you are interested only to learn about the algorithm and not worried about the design pl skip to the section 5 of this text.

1 INTRODUCTION
Atonomous robots have wide reaching applications.From Bomb sniffing to finding humans in wreckage to home automation.Major problems facing designers are power and reliable sensing mechanism and unfamiliar terrain obotic competitions have inspired engineers for many years.Competitions are held all around the world based on autonomous robots. One of the competions with the richest history is micromouse . The micromouse competitions have existed for almost 30 years in the United States and it has changed little since its inception. The goal of the contest is simple. The robot must navigate from a corner to the center as quickly as possible.The actual final score of the robot is primarily a function of the total time in the maze and the time of the fastest run.The specificatons for the micromouse event is specified in appendix A.
The Design incorporates various techniques to simplify the approach and make an effecient automated robot.

2 MICROMOUSE DESIGN AND HARDWARE
The Major criteria of micromouse design remained the size of the robot which will allow smooth 90-degree turns and U-turns possible. After detailed analysis regarding the maximum dimensions of the robot the initial dimensions to start with were finalised as 10cm x 10cm.

The Micromouse hardware required two stages.
1. Choosing the type of motor
2. Building the chasis

The micromouse was made initially with a DC motor,since the strategy revolved around using very accurate sensors which can be easily used to regulate the non-linearity of the DC motor.DC motor has its own advantages of higher torque even at low cost motors.The initial design planned incorporated four 6F22 9v general batteries,which posed considerable weight considerations. This was tackled successfully by the use of a good gear system. The weight of the robot was planned to be lesser than 500gm
which would facilitate free motion of the robot even on rough surfaces.The number of wheels was a major factor of thought,A four wheeled robot would find it difficult to negotiate turns while giving a steady straight motion. the three wheeled robot was on cards that can negotiate turns with ease, Major disadvantage being ,it capable

of maintaing steady straight motion on straight runs. Sensing devices have been traditionally classified as “Over-thewall” or “Under-the-wall” . The original micromice used the red paintted wall top to determine the orientation,like a long wing like sensor arrays extending over the walls.Recent designs avoid the large moment of inertia due to huge wing arrays of the sensors and have opted for low riding mice that measure the distance from inside the wall.The latter design was markedly superior, and permitted extremely compact designs.Sensor design will be discussed in section 3. In hardware consideration of the design it was decided to use optical sensors rather than the ground-contact(rolling) sensors. The mechanical design ot the micromouse was completed on paper, drawn with relative scale.

3 SENSOR

In order to execute the algoritm accurately and and prevent the robot from crashing into obstacles the robot has to see the environment it is moving in.There are major considerations on the design of the robot since varied approaches can be introduced in the way the robot sees its environment. One elaborate but accurate technique is to measure the intensity of the optical wave and finding the distances of the robot from the obstacles at short distances. A very simple rather not so accurate technique is the move at accurate distances per move and keep counting the cells and keep the robot aware of its current location in the maze.Major problem posing this approach was the fact that when a motion is set up after a halt the wheels would slip before they actually start covering their ground,what automotive engineers call “grip-slip” for a typical rubber tyre.The wheels selected for the design were plastic hard wheels for easier design approach that offrerd more slip over smooth surfaces. It is obvious that we need some amount of wheel slip is necessary to exert the acceleration force.Worse,the actual grip slip is dependent on the surface type and all that is known is that the ground is black in color and it absorbs light.Thus to capitalise all the drawbacks on the accurate movement of the robot, repeated testing was required to find average yet accurate motion. primarily it was decided to design short range sensors that can just detect the presence of obstacles and not calculating the distance of the robot from the obstacle.A simple hardware approach essentially required more tedious programming technique. it was a trade off between hardware or software approach. It was decided to tackle problems on software grounds than hardware.

3.1 Short Range IR Sensor

The short range IR sensors needed to be designed with a dynamic range of 1-8cm. the Ir sensor designed was having one IR Led and one Photodiode whose configuration is ass shown in the figure 1. It can be noted that the angle of acceptance of the photodiode is small compared to the beam angle of the IR Led.

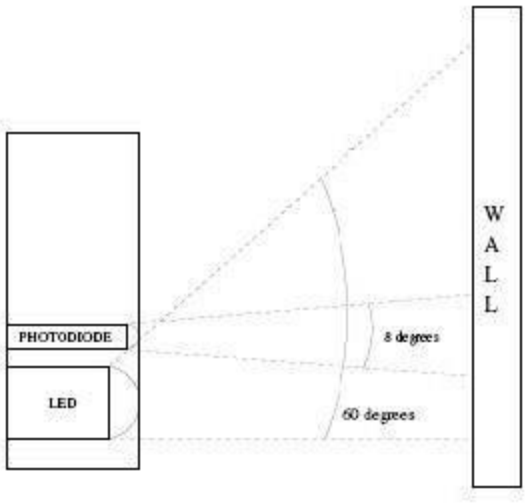
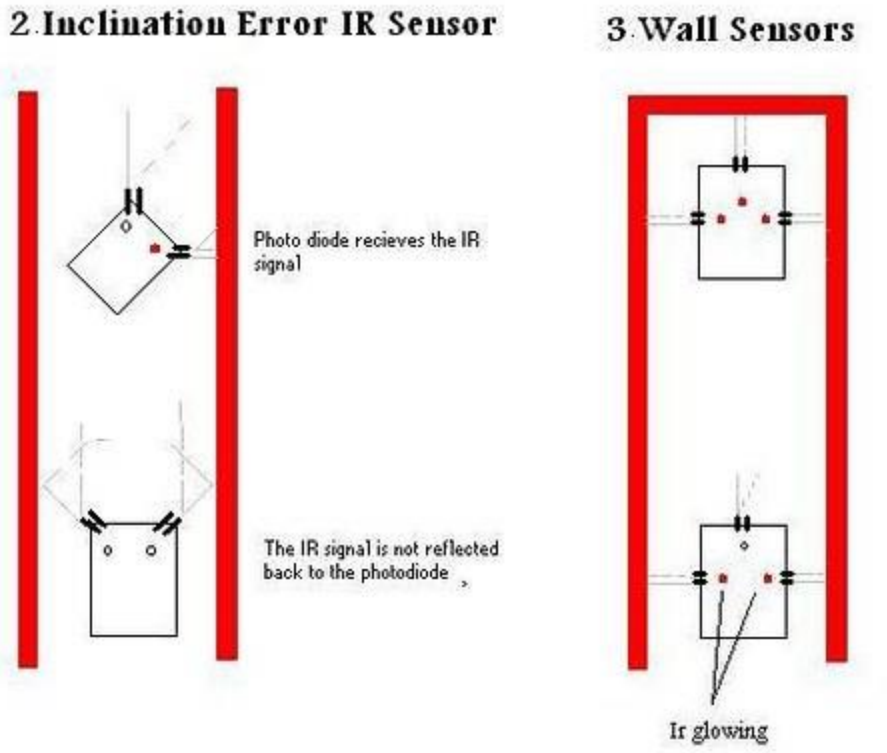


Figure1: Design for short range sensors

Since the technique adopted does not involve measuring of the ambient light and measuring the difference, appropriate care has to be taken to prevent ambient light to disterb the ir sensor and inaccurately detect the presence of an obstacle while there was none due to infeference of ambient light , thus the ir sensors were placed far lower in the robot architecture such that the maze walls are solely enough to restrict most of the extenal light disturbances that possibly can disturb our detecting system.the IR were places low far from the circuit site fixing it to the robot body.

The reqired number of IRs were five. Three sensors to detect the presense of walls on three sides namely front,left and right. and two sensors one on each sides to detect the inclination of side walls to the robot's line of motion. The inclination sensors were based on the fact that IRs respond only within a particular range of inclination with side walls. in the designed inclination sensor the robot would recept an inclination error signal when the robot in at an angle less than 72.5 degrees with rhe walls. Thus as long as the sensors i,e robot was 90-degrees with the side walls there would be no eroor signal. If the robot was to deviate from its path and move at an angle towards the wall the inclination error sensors would be set high which can be detected and processed.



---

4 HARDWARE PLATFORM

The electronic design centres around a Microchip processor. the PIC16F877 has 5 ports that make our interface with external hardwares easier.PIC could be interfaced with external EEPROM memory to facilitate extensive programming. To keep the hardware small and compact, the inbuilt EEPROM code memory of 8k was used for programming and the data memory of 256 bytes were used for runtime memory map storage. Other data storage requirements are implemented on the 256 byte RAM.

The processor is the only onboard programmable chip,other peripherals included a shmitt trigger IC 74HC14N.the voltage levels from the sensors were a change from 1.45v to 0.25 volts when an obstacle was detected. The inverting schmitt trigger was interface to bring the detecting signal to TTL logic.

The motor selection decided the type of motor driving hardware.

4.1 DC MOTOR DESIGN

In this type of design, two individual motors were used to drive the wheels on either side. Appropriate reduction gears were used to optimise speed. The motors needed to be driven in both forward and reverse direction thus requiring circuitry to enable drive on either side with appropriate control signals.

A normal relay was used to implement this, 2 unipolar 16v relays were used to select appropriate motors and 2 bipolar 5v relays

were used to determine the direction of the motors.

4.2 STEPPER MOTOR DESIGN

Stepper motors require special driving mechanisms unlike DC motor that are two terminal driving devices.Our robot was implemented on a NEMA14 stepper motor and was driven with a serial pulse of 16v , 500 mA supply. IC ULN2003 was ised as drivers. the microcontroller port B was assigned for driving the motors and IC ULN2003 was interfaced with the microcontroller port.

————————————————————

THE MAIN FUNDA

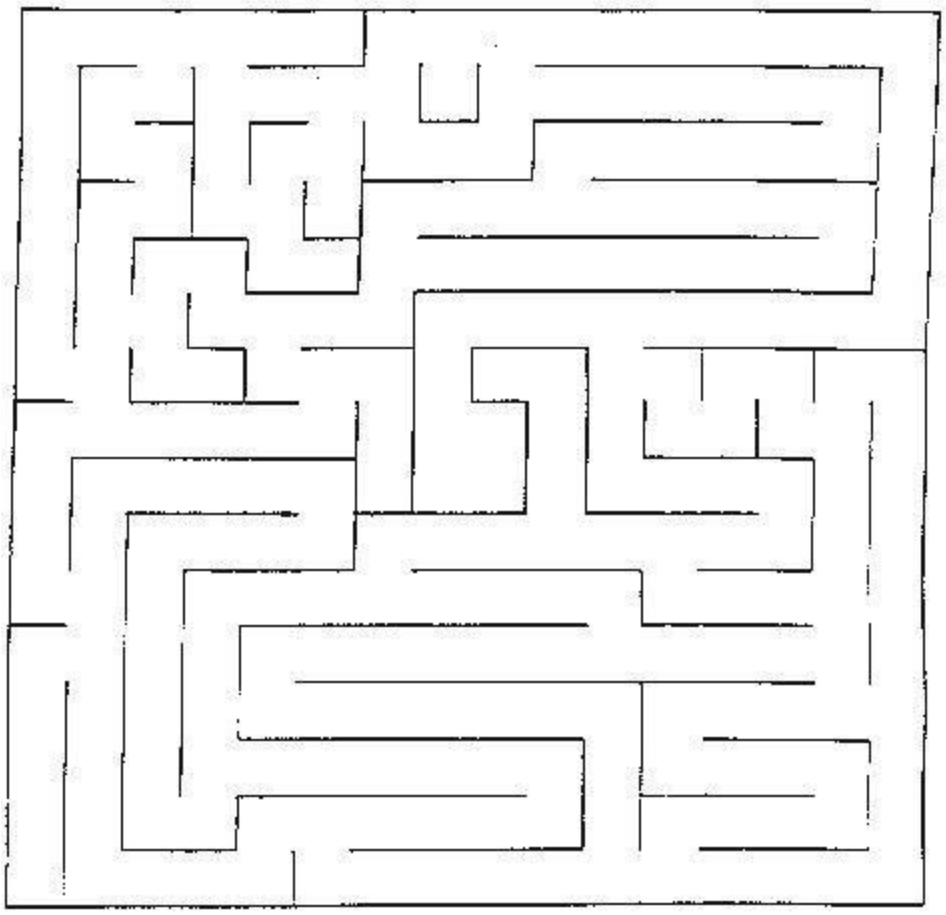
5 ALGORITHM

The maze solving algorithm implemented in the robot was self developed with improvements from the basic form of bellman flooding algorithm.The algorithm requires around 256 X 3 bytes of menory. The selected microcontroller for implementation had only 256 kbytes of memory, Thus a major memory crisis was to be tackled on the software basis. A very apt solution was to switch over to PIC 18FXXX series which have higher RAM and ROM memories. After appropriate analysis the problem statement was simplified to three rules which if followed would direct the robots to the centre of
the maze.

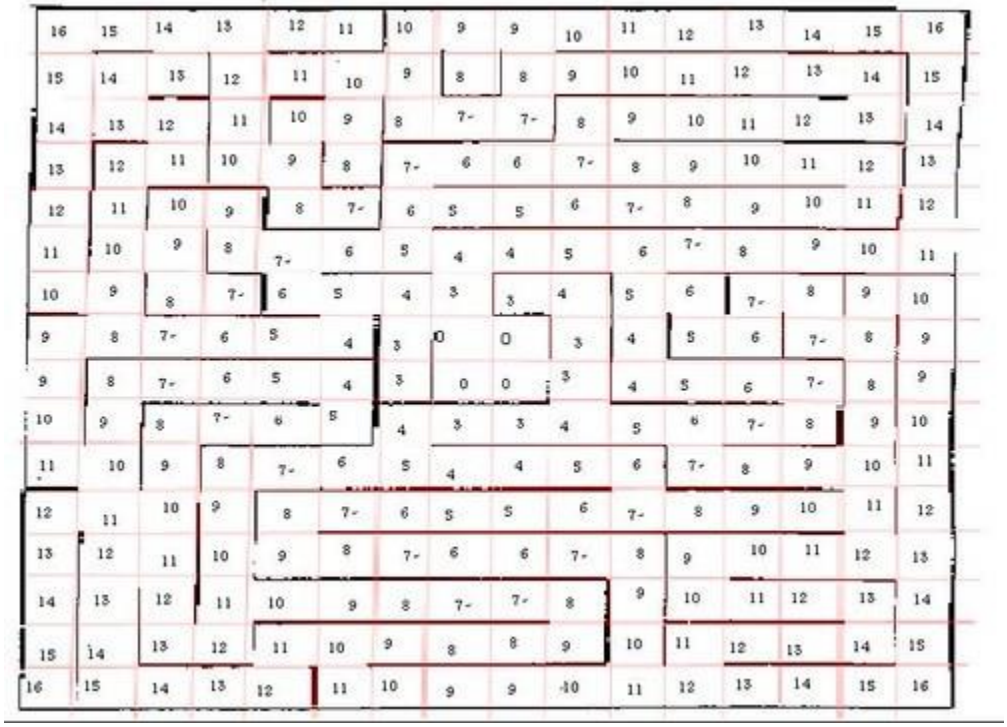
5.1 MEMORY MAPPING

The contest area has a matrix of 16 X 16 cells. the whole game area is mapped into the memory of the robot assigning the values as shown in the figure

SAMPLE MAZE



MAPPED SYSTEM IN MEMORY



As the cells are mapped with the numbers as sshown in the figure, at each cell the robot is expected to take three decisions.
1) move to cell which it has gone to least
2)move to the cell that has minimum cell value
3)if possible the robot must try to go straight.
It is evident that these three conditions if followed at each cell position the robot will reach the centre of the maze designated as ‘’0’
10
the mapping of the cell values in the memory requires huge memory , thus an alternative method was adopted to generate the cell values at runtime.

ALGORITHM TO GENERATE CELL VALUES AT RUNTIME

```
unsigned short gen(unsigned short row1,unsigned short
col1)
{
if(row1>0x08)
{
nr = row1 – 0x09;
row1 = 0x08 – nr;
}
if(col1>0x08)
{
nc = col1 – 0x09;
col1 = 0x08 – nc;
```

}
return(0x0f – (col1-0x01) – (row1 – 0x01));
}
Eg, consider the cell location where row = 0x08 , col = 0x08
Evaluating in the formula we get the return value as '0' which is the cell value.

5.3 NAVIGATION ROUTINE

The robot was designed to move each cell by exact distance and then the sensor reading is read by the processor. based on the values and applying the three criteria discissed earlier the robot decides its next action
At every junction if only one side is sensed open then the robot has to move only into that cell., decision comes into play only when there are two or three sides that are open to navigate.The robot records each location value as it proceeds towards the center. To come back to the starting point it just traces the path back from the memory map.

CORRECTION

Since the robot cannot strickly hold to its straight direction,neither striclt maintain a 90-degree turn the bot required software correction techniques. as discussed earlier in the hardware techniques about the correction IR sensor, the robot required to move in the other direction to the signal until the signal was off. thus involving a few lines of coding.

5 SIMULATION

Simulation was done using PIC simulator IDE . The siumulator shows the port pin logics and the EEPROM memory. as the code was run the appropriate ligics were checked and the memory value was recorded.

6 CONCLUSION

MicroMouse is a prime example of engineering challenge that most theoretically deviced solving techniques fail. the robot was designed to tackle most practical probles encountered in real situations.The cross-disciplinary nature of the project enabled us to learn elemnts of mechanical,control,signal and computer engineering.

I guess you guys learnt a bit out of it. well this post comes from one of the readers (Mr.Subhobroto Sinha) of this blog who requested to post my projects. i have just copy pasted it from my scrap box. If possible will make individual points clear in future posts .Do comment about what you need to know

Click below for the processor schematic diagram