assume that you have read problem statement of Micromouse thoroughly it is available on www.techfest.org . I will suggest you to read every page from http://micromouse.cannock.ac.uk/ A terrific site that gives lots of information on building a Micromouse. It covers all aspects of the design from hardware to software and it's a great place to start with for a beginner.

I will discuss very few points here which we found crucial.

Before starting any work first of all make some decisions and **never change them during development** otherwise you will move one step forward and two step back (This I can tell you from our first micromouse building experience). Points to decide are

1) **Chassis**: Chassis design, motors. (Important)
2) **Sensors**: Side looking or Top down, Position of sensors (Important)
3) Batteries: NiMh, Lithium ion, NiCd
4) Electronics: Controller, Schematics

Chassis :

Is bigger better? Probably not.

Make your mouse as thin as possible. Thinner and lighter the micromouse greater the maximum speed. Try to fit your mice in square of 8cm x 8cm. Micromouse which came first at Techfest was only 7.5cm wide and our mice was 10cm x 10cm. Watch this video http://youtube.com/watch?v=peEpkRIKDEs a micromouse from Darek Hall.

Try to concentrate maximum mass near centre to reduce moment of inertia. It counts when turning at high speed. Make your PCB a part of chassis.

Thumb rules to chassis design are.

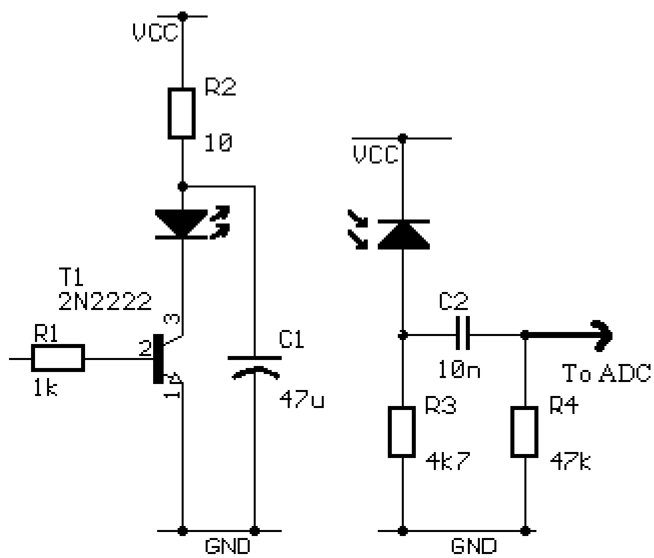
- keep the Center of mass low
- keep size as small as possible
- put the skids or casters as far out from the wheels as you can
- keep the total mass small
- keep the mass concentrated in the middle of your mouse

Sensors :

There are two choices available for a micromouse builder. You can either have side looking sensors or top down sensors.

Side looking sensor:

A good side-looking wall sensor will return an analogue value representing distance to the wall. The simple presence or absence of a wall is inadequate for steering. The greater the level of accuracy, the better. In principle a sideways looking, reflective sensor will give better results than a top-down sensor. However, variations in wall reflectivity, from cell to cell and maze to maze can make this difficult to use. You could try some kind of auto calibration while the mouse is running although it is likely that you will need manual calibration at the start of a run.



Circuit is pretty much simple. It is a differentiator applied to Photodiode output. It differentiates the constant output voltage due to ambient light.

Turn ON the transmitter when you want to sense. ADC conversion should be started at least 7us after turning ON the IR LED. Turn OFF transmitter after sensing.

Top down sensor:

A popular choice is sensors that live on wings and look down at the walls from above. These can be simple reflective switches connected to an eight-bit port on the micro. A perfectly adequate mouse can be constructed using eight of these on each side. Say, seven for the walls and one each shared to detect the forward wall.

The wing *must* be placed well forward of the wheels. You will need to see where the walls are before you get to them or the control problem is impossible.

As for other IR sensors these should really be AC coupled and synchronously read. You might like to just use ready-made reflective switches designed to work in adverse conditions.

The main problem with these arrangements is to do with the rotational inertia they present to the mouse. However light you make them, they have to live a relatively long way from the center of mass and so contribute a relatively large component of the total inertia. As long as you are prepared for that and don't expect breathtaking turn performance, they can have a lot going for them. Build them light and there will still be plenty of other things to worry about before rotational inertia becomes a limiting factor in performance.

For top down sensors circuit is same as the circuit for line follower reflection sensor.

Side looking	Compact	Simple circuitry, complex program
Top down	Bulky	Simple but more number of sensors required. Simple to program

Controller:

Choose controller according to your requirements of program memory(flash), RAM, ADC, EEPROM. 16kB of flash is sufficient to implement floodfill algorithm.

If you are thinking of implementing floodfill algorithm then 256 bytes are required to store Map of the maze and 512 bytes are required for processing so you need atleast 1kB RAM to implement floodfill. ATmega16,32 and PIC18f452 are some controllers popularly used in micromouse.

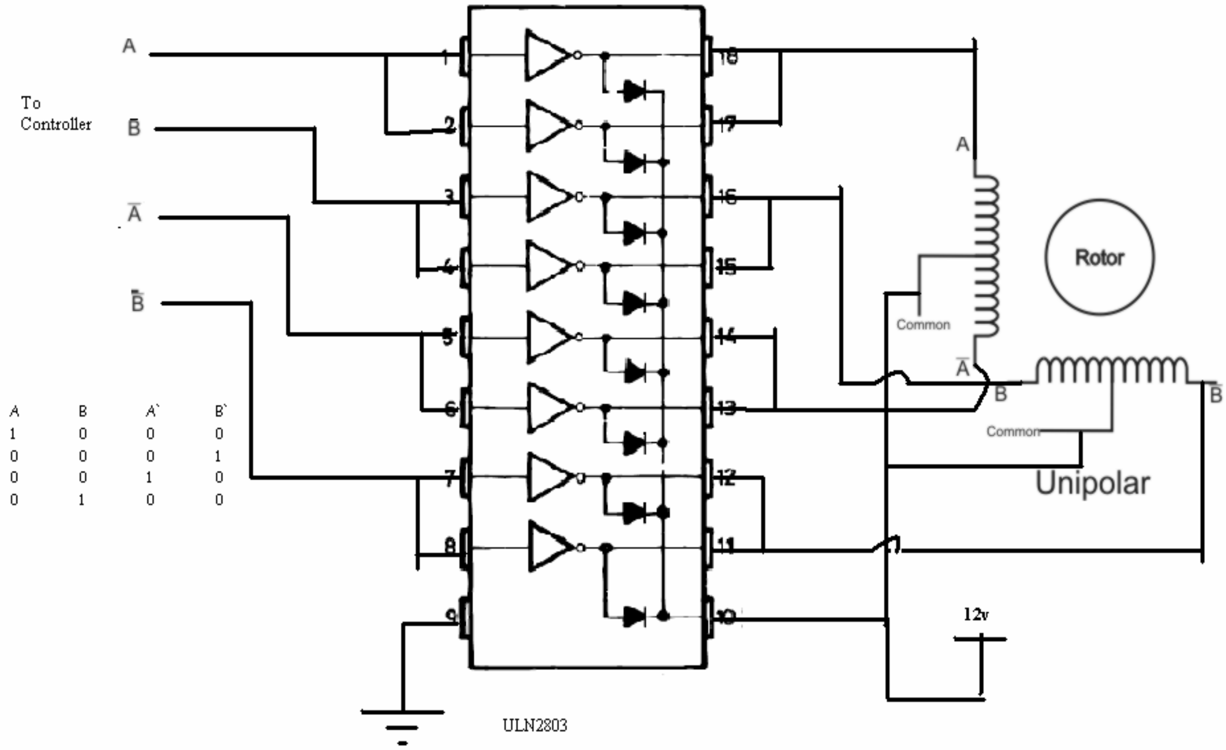
If you are thinking of building just a wall follower then simple 8051 based controller like AT89s52 can also be used.

Stepper driving:

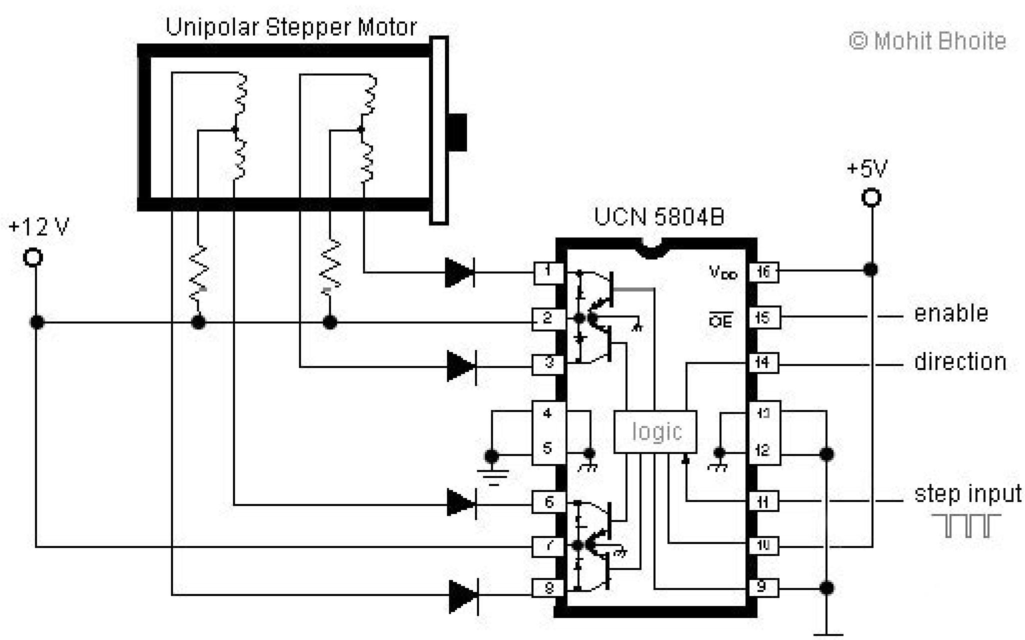
Please read http://www.triindia.co.in/resources/?p=40 a step by step stepper motor guide first.

There are two types of motors a) unipolar b) bipolar.

Unipolar stepper motors can be used as bipolar steppers so prefer unipolar while purchasing stepper. You can use simple ULN2803 IC the circuit can be found in 8051 microcontroller by Mazidi. ULN2803 has eight darlington pair in a single package. It has current carrying capacity of 500mA. so it will get heated up frequently will get damaged. So two ULN2803 can be connected in parallel to fix the problem but this is a *chalta hai* way of driving stepper.



Another simple and better way of driving stepper is to use UCN5804B a BiMOS II Unipolar stepper-motor translator/driver. I don't know availability of this ic in Aurangabad but definitely available in Mumbai.



Add a proper current limiting resistance according to your stepper current rating. ie if your motor is 5v 500mA(Ri=10 ohm) and you are driving motor at 12v then you will have to maintain current below 500mA therefore value of current limiting resistor is 12v/0.5A-10 ohm=14 ohm(2 Watt). Lower this value if you don't get sufficient torque.

You have to provide two signals to the IC that are 1) direction 2) step input. Enable can be connected permanently to +5v.

1) Direction input decides direction of rotation of stepper.

2) Step input provide clock input here for one clock pulse motor will rotate by one step ie 1.8 degree in full stepping mode. by varying time period of the clock you can change the speed of the stepper.

Only one disadvantage with this IC is it doesn't provide current chopping. So current is wasted in heating up the motor coils and current limiting resistors.

Chopper drive:

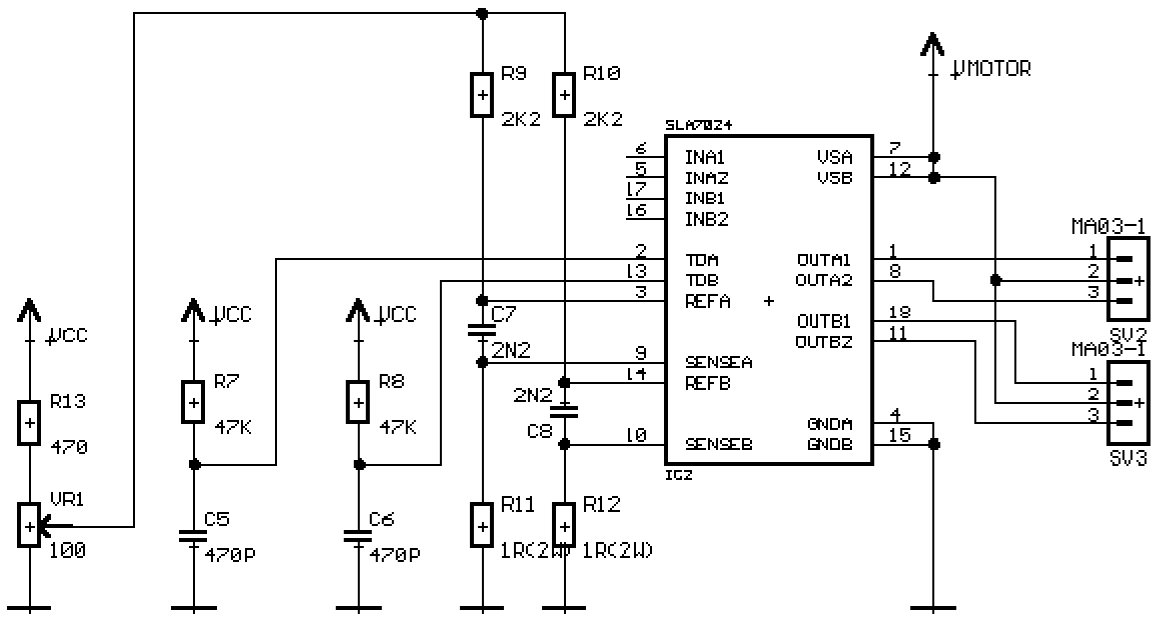
Download this pdf it gives detailed information on how to drive the stepper motor.

I will try to describe the above detailed information in short. To attain higher speed with sufficient torque rated current Ip must flow through the coils. Coils being inductor oppose the current flowing through it and current doesn't reach to Ip resulting in lower torque.

To overcome this situation voltage across the coils is increased above rated voltage (stepper is overdriven). But increased voltage causes larger current than Ip to flow through the coils. This excess current heats up the motor. It doesn't harm the motor but the batteries drain very fast.

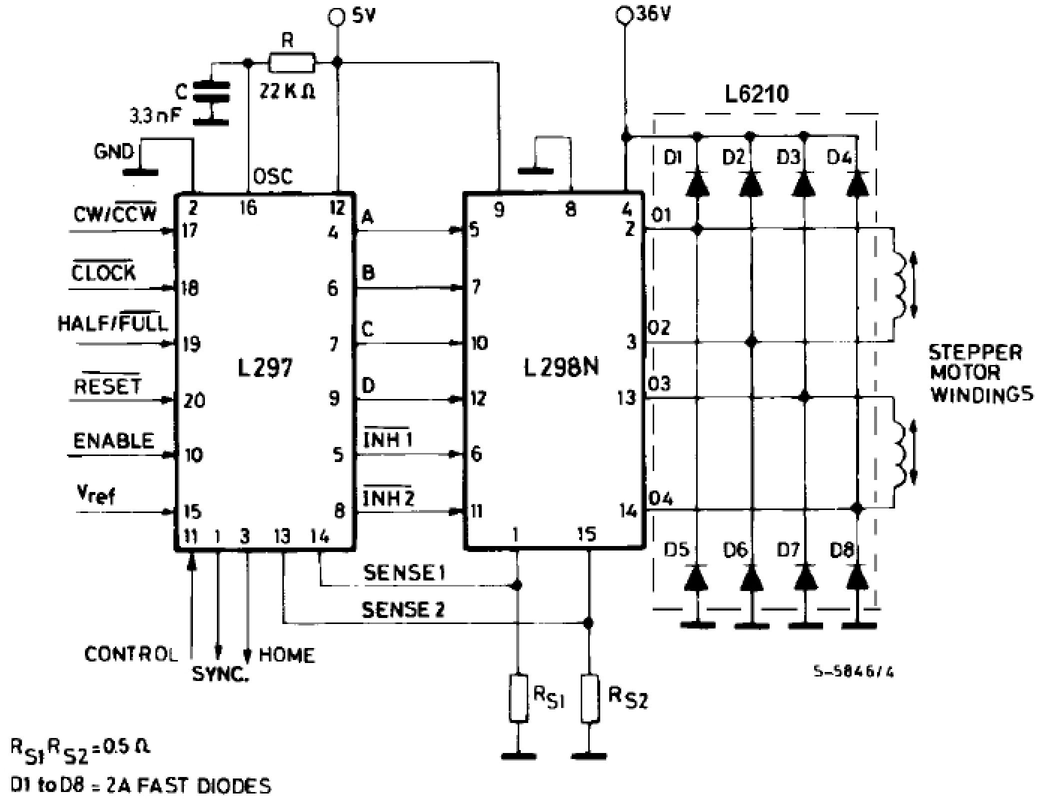
Now chopper drive cuts off the current to coil when current through the coil reaches Ip and mentains the current through the coil at Ip.

SLA7024M is a chopper drive IC for unipolar stepper motor.



You will have to provide stepper sequence at INA1-INB2 pins unlike UCN5804B where only clock pulses are required.

Another bipolar chopper drive is L297 with L298. L298 is dual H-bridge ic. and L297 stepper motor controller.



ULN 2803	Simple circuitry (In terms of size)	Complex programming	No chopping is provided IC gets heated.
UCN5804B	Simple circuitry	Simple	No chopping
SLA7024M	Complex circuitry	Complex	Chopper drive
L297 with L298	Complex circuitry	Simple	Chopper drive

Software:

Your software for micromouse should do following things.

1) Read sensor values continuously
2) Track the distance traveled by the mouse.
3) Keep the mice at the center of passage way.
4) Prepare the map.
5) Run shortest distance algorithm
6) Take decision at each center of cell about which direction to head now.
7) Repeat this until at the center.