**Trade: Mobile Robotics**

**OBSTACLE AVOIDANCE ROBOT COMPETITION**

Introduction

The object of the contest is to build a small Audrino Uno microprocessor-controlled robot vehicle with the components you were given,program the vechicle to navigate its way, through an unknown mazeboard, to the final target in the shortest possible time. The final target is the marked point “Finish” with colour in the Maze Board.

The purpose of this competition is to provide a technically demanding yet enjoyable problem for the participants.

The challenge is to build a small robot vehicle capable of fast controlled motion,and provide it with sufficient intelligence to explore and negotiate around obstacle**Sharp GP2Y0D810Z0F Digital Distance Sensor with Carrier board**s in the shortest possible time to reach the target

**Final Outcome expected from Candidates**

In this contest the contestant should understand from the parts diagram and build small self-contained robots to negotiate a maze run in the shortest possible time

**Description of the project/Tasks**

1.Each Contestants will be given with EllipZo-Robotic Chassis with Servo Pan kit unassembled
Parts included:

1 x Top Chassis
1 x Bottom Chassis
4 x DC Motor Hold
1 x Array Hold
1 x Pan Head
1 x Servo Neck
2 x 4xAAA//9V Battery Holder
1 x Toggle Switch
1 x Caster Ball Wheel
2 x 100RPM BO1 plastic geared motor
1 x Off Road Wheels 65mmx30mm (Pair)
1 x FS90 Analog Plastic Geared Micro Servo Kit
Stand offs
4 x M3 6mm HEX
10 x M3 25mm HEX
Screws
4 x M3 35mm STAR HEAD
26 x M3 6mm STAR HEAD
3 x M3 12mm STAR HEAD
2 x M2 12mm STAR HEAD
2 x 7mm 3mm diameter Mushroom head
Nuts
5 x M3 2mm HEX
2 x M2 2mm HEX

**Parts included are:-**

Specifications:

DC motor (100RPM BO1 Plastic Gear Motor)

- Operating Voltage 3V-12V DC
- RPM 100rpm
- No load Current 40-80 mA
- Output Torque : 4 Kg-cm

Wheel(RW002 Off Road Wheels 65mm X 30mm(Pair))

- 65mm diameter
- 30mm wide
- Rubber wheel on plastic hub

Servo Motor (FS90 Micro 1.3kg Torque Analog Plastic geared Servo)

- Operating voltage 4.8-6V DC
- Operating Angle 120°
- Operating Speed:
  - 0.12sec/60° (4.8V)
  - 0.10sec/60° (6V)

Stall Torque:

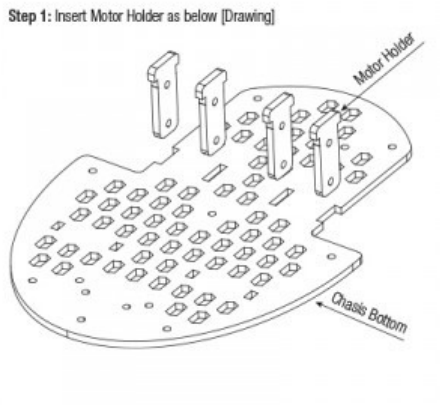
- 1.3kg.cm/18.09oz.in(4.8V)

Toggle Switch

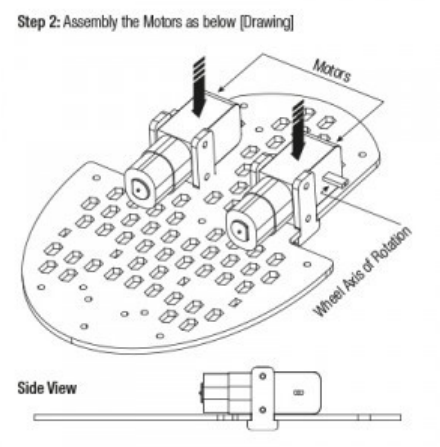
- Double Pole Double Through with
- Rated Current 3A , Rated Voltage 50V

**EllipZo Assembly Steps:-**

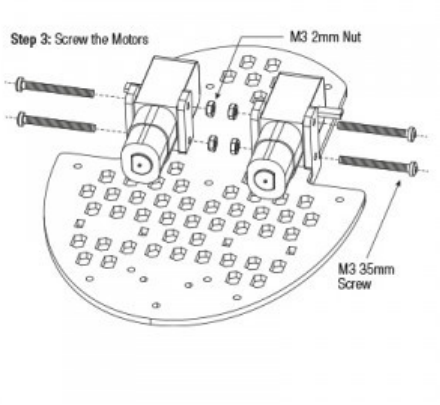
**Step 1:** Insert the motor holders in the slots as shown in the picture.



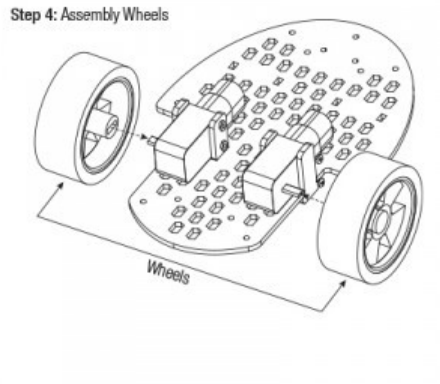

**Step 2:** Place both the DC motors in such a way that the motor shaft is at the middle of the  wheel slot.



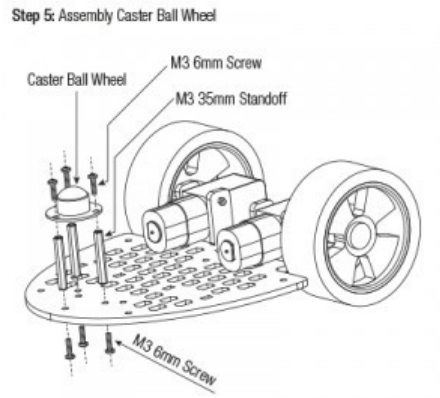

**Step 3:** Screw each motor  using two M3 25mm screw & ensure that it fits well.



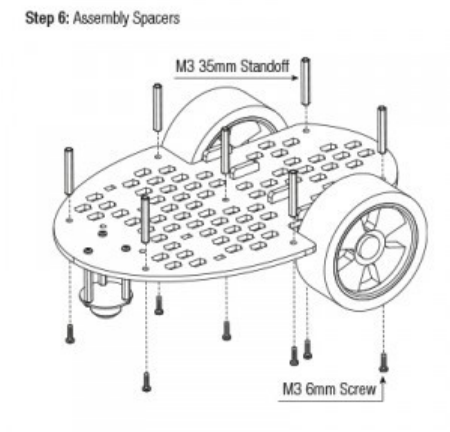

**Step 4:** Attach Wheels to the motor shaft as shown in the figure



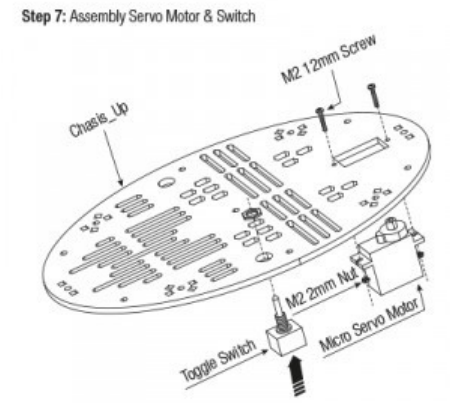

**Step 5:** Attach caster  wheel as shown in the figure below, use M3 12mm screw and M3 25mm stand offs.



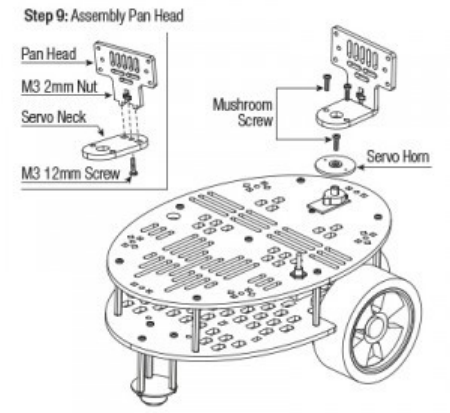
**Step 6:** Attach M3 25mm stand offs,use M3 6mm screw for fit it




**Step 7:** Attach micro servo & Toggle switch on to the top chassis as shown in the picture. Use two M2 12mm screws and M2 2mm nuts to fit micro servo.

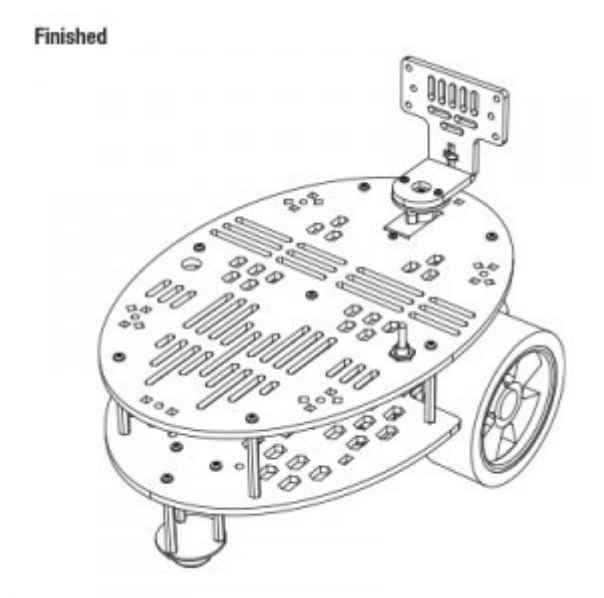



**Step 8:** Attach top chassis to the bottom chassis, Use M3 6mm screws to fit it. This step is crucial as the screw hole being symmetrically positioned, user can either place the top chassis with micro servo near caster ball wheel or near DC motor. This is important because it determines the forwardfacing side of the final assembly.**:** Attach servo neck and pan head using M3 12mm screw & M3 2mm nut. Attach this to micro servo horn and then to servo motor using mushroom head 7mm  screw.



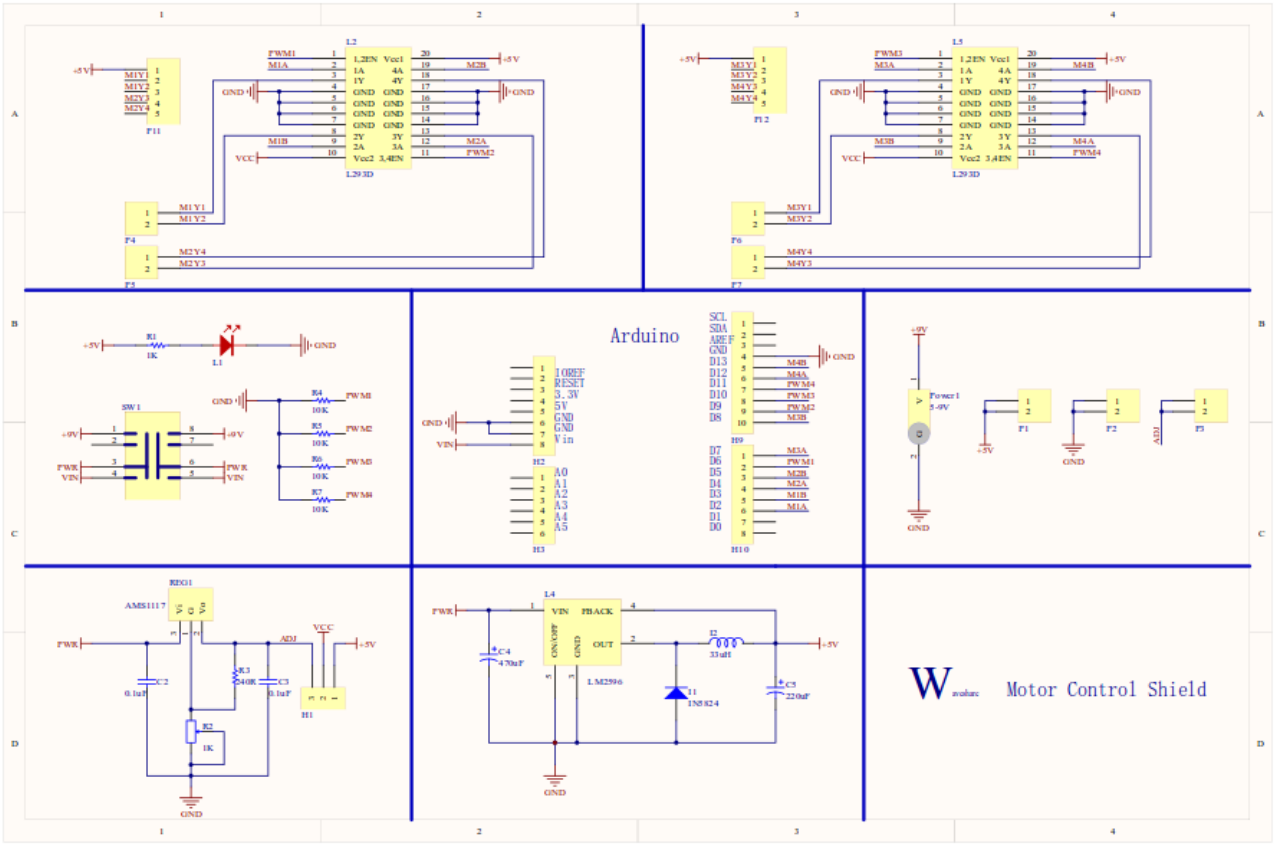

Now **EllipZo** is ready to use.

Finished



## Motor Control Shield

The Motor Control Shield  is based on the L293D motor driver IC, which is a dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. With dual L293D ICs, the shield lets you drive 4 DC motors or 2 stepper motors at the same time with your Arduino board along with the control over speed and direction of each one independently. The shield has an adjustable power supply which can have output voltages ranging from 1.25V to 6.45V DC at a time



**Sharp GP2Y0D810Z0F Digital Distance Sensor with Carrier board**

Description: This small digital distance sensor detects objects between 2 cm and 10 cm (0.8″ and 4″) away. With its quick response time, small size, and low current draw, this sensor is a good choice for non-contact object detection, and our compact carrier PCB makes it easy to integrate into your project.

General Specifications:

Maximum range: 10 cm

Minimum range: 2 cm

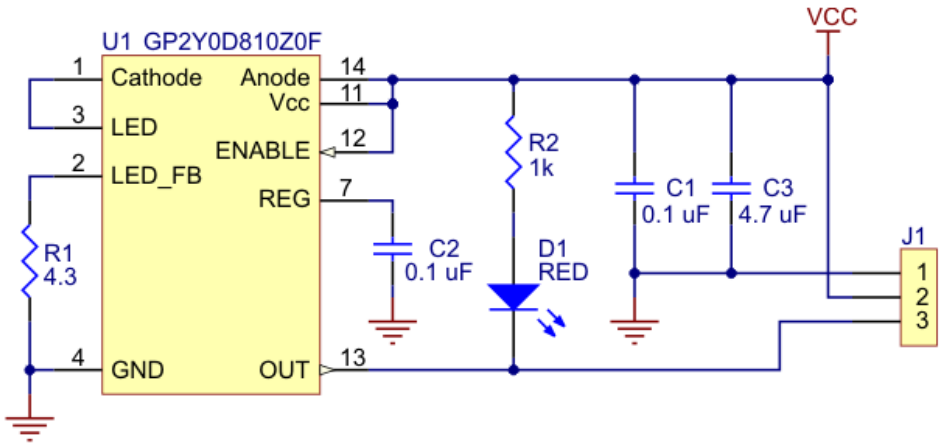
Sampling rate: 390 Hz2

Minimum operating voltage: 2.7 V

Maximum operating voltage: 6.2 V

Supply current: 5 mA3

**Output type: digital4**




**Pololu carrier for Sharp GP2Y0D805Z0F, GP2Y0D810Z0F, and GP2Y0D815Z0F sensors schematic diagram.**

**Arduino UNO - R3**

Package includes:

1 x Arduino UNO - R3 (Arduino-Italy)

1 x USB A to B cable

Description: The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable

Specifications:

Microcontroller ATmega328 (PDIP)

Operating Voltage 5V

Input Voltage (recommended) 7-12V

Input Voltage (limits) 6-20V

Digital I/O Pins 14 (of which 6 provide PWM output)

Analog Input Pins 6

DC Current per I/O Pin 40 mA

DC Current for 3.3V Pin 50 mA

Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader

SRAM 2 KB (ATmega328)

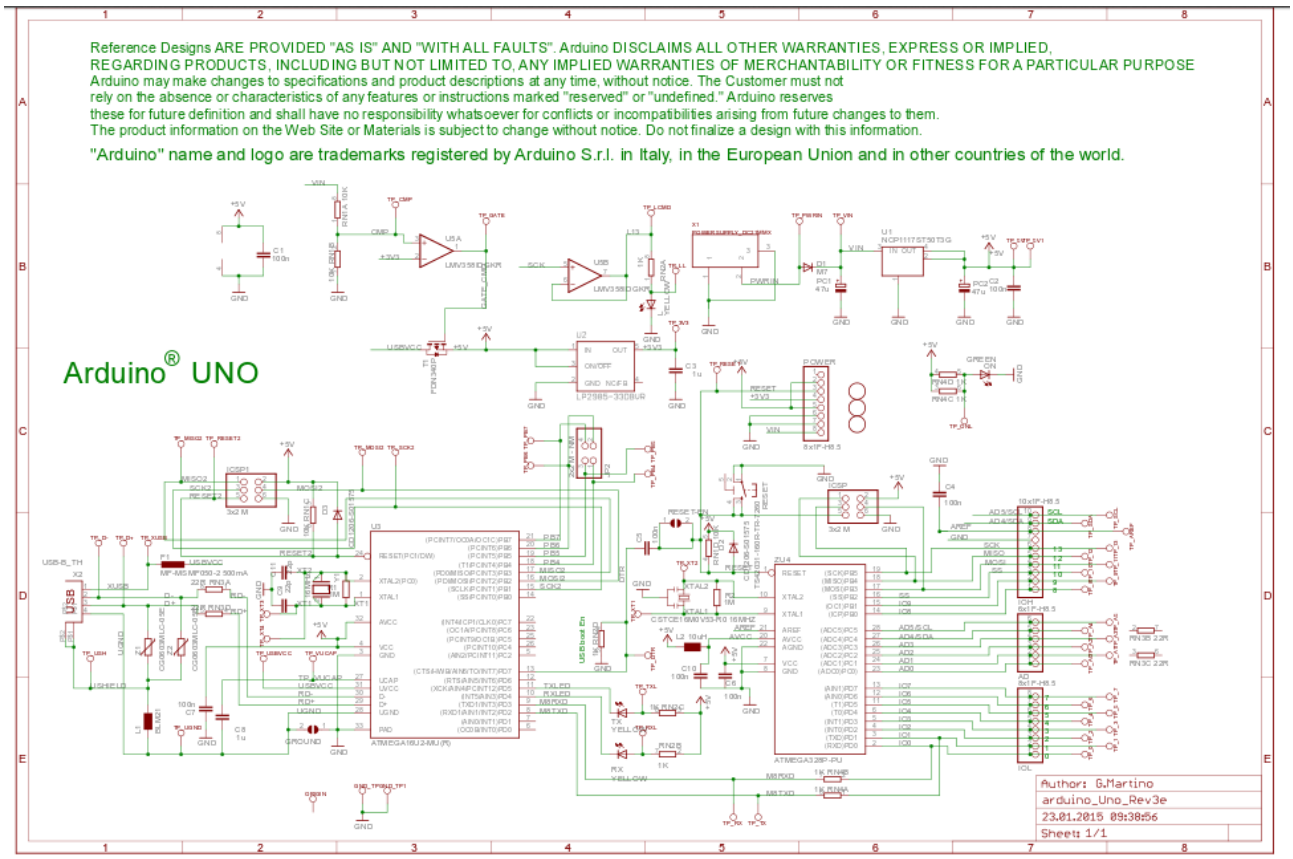
EEPROM 1 KB (ATmega328)

Clock Speed 16 MHz

**Programming**

For programming Arduino Arduino Software (IDE) installed PC / Laptop with USB port will be provided

Each contestant can make a folder in their name and can start their programming and porting




### Programming the Arduino UNO

The Arduino UNO can be programmed with the Arduino software . Select "Arduino UNO" from the "Tools > Board" menu.

ATmega328 on the Arduino UNO comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar.

### Installing Drivers for the UNO

### Windows

If you previously installed Arduino IDE, you already have drivers installed.

If you haven't yet installed Arduino IDE, download the Windows version of the Arduino software is available for download
When the download finishes, launch the file to install the software and the drivers.

If you downloaded the .zip version of the IDE, unzip the downloaded file. Make sure to preserve the folder structure.
Now your drivers will be downloaded and installed from Internet, directly from Windows.
If you have issues, you can find the "drivers" folder inside the unzipped file, for manual installation. Linux

There is no need to install drivers for Ubuntu 10.0.4

In some computers, you need to setup user permissions and some udev rules.

You can find detailed informations on how to achieve this at this page.

**Uploading Code to the UNO**

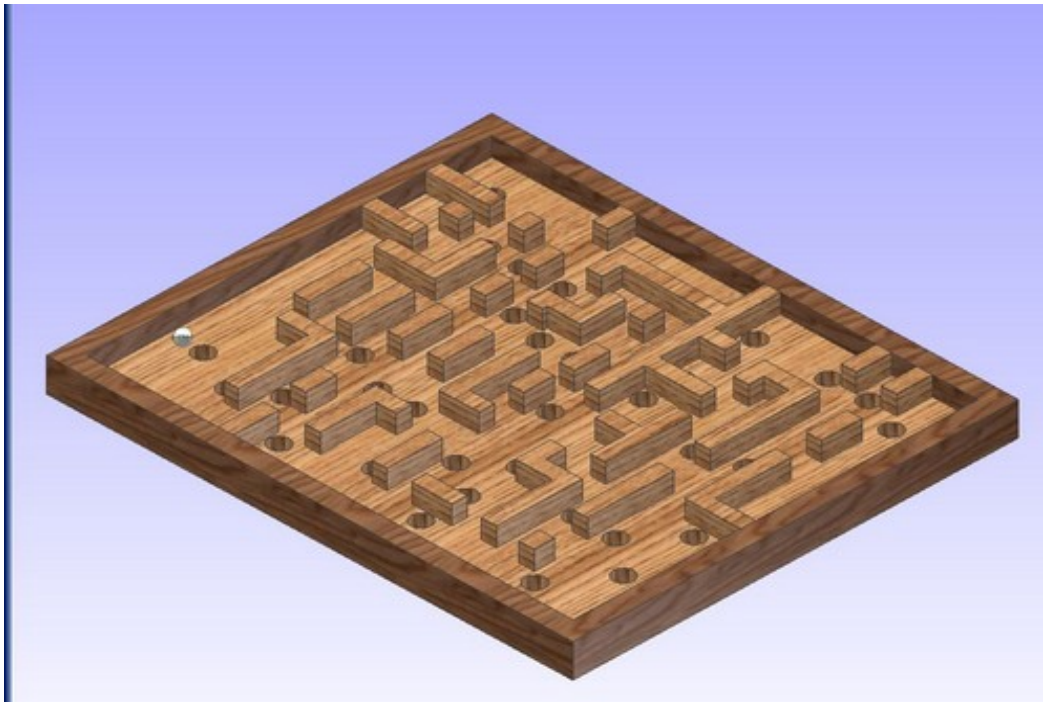
Click the upload button in the Arduino IDE and your sketch will be automatically uploaded onto the board and then started. The Arduino software initiates a reset of the board, launching the bootloader - which is responsible for receiving, storing, and starting the new sketch.

However, because the serial port is virtual, it disappears when the board resets, the Arduino software uses a different strategy for timing the upload than with the Uno and other boards. In particular, after initiating the auto-reset of the UNO (using the serial port selected in the Tools > Serial Port menu), the Arduino software waits for a new virtual (CDC) serial / COM port to appear - one that it assumes represents the bootloader. The it performs the upload on this newly-appeared port.

These differences affect the way you use the physical reset button to perform an upload if the auto-reset isn't working. Press and hold the reset button on the UNO, then hit the upload button in the Arduino software. Only release the reset button after you see the message "Uploading..." appear in the software's status bar. When you do so, the bootloader will start, creating a new virtual (CDC) serial port on the computer. The software will see that port appear and perform the upload using it. Again, this is only necessary if the normal upload process (i.e. just pressing the uploading button) doesn't work. (Note that the auto-reset is initiated when the computer opens the serial port at 1200 baud and then closes it; this won't work if something interferes with the board's USB communication - e.g. disabling interrupts.)

**INSTRUCTIONS TO THE CANDIDATES**

Checker plane with obstructions placed with markings start and end Example of MazeBoard given below



The obstacles will be placed, approximately with necessary clearence minimum passage width of at least for the robot which is fabricated is guaranteed.

**RULES FOR THE CONTEST**

The objective of the competition would be for the robot to reach the target in the shortest time.

All robots must travel on the surface of the domain. Clearing certain points will be awarding points according to predetermined check points

The time of start and finish for each one robot will be noted and bonus points for crossing certain points

Before the start of the competition, entrants are allowed to inspect the layout of the maze to satisfy themselves that all blocks are spaced in such a way that robot can pass through

Under all circumstances, participants are not allowed to touch the block obstacles.Should any positional adjustment be made to the block obstacles, these will be performed by the event timers/helpers/judges.

The start of the maze is located at one of the four corners. The start square is bounded on three sides by walls. The start line is located between the first and its second squares. That is, as the robot exits the corner square, the time starts. The destination goal is the four cells at the center of the maze. The destination square has only one entrance.

Small square zones (posts), at the four corners of each unit square are called *lattice points*. The maze is so constituted that there is at least one wall at each lattice point.

Multiple paths to the destination square are allowed and are to be expected.

## Each particiapnts should be provided with following

1.A PC with ARDUINO programming with USB
2.A Table for assembly of the kits provided to them
3.Soldering Iron ,Paste ,Solder Lead,Desoldering Pump
Different coloured wires ,
Cutter/Stripper,Heat Shrinkable sheild of necessary size according to the wires
One Multimeter
Glue Stick /Gun for Securing Wires
Insant Adhesive
4.Necessary Screwdriver Set, Plier,Tweaser
5.Maze Board fabricated as per the Question
6.Timer for each participants
7.Necessary batteries for the robot
8.UPS Points for the PC and Soldering Iron and SMPS Adaptor for UNO Board
9.CD Marker Pen for Each Contesant
10.Cello Tapes ,Insulation Tapes
11.A4 White Paper
12.Instruction Sheets