Description

The obstacle avoidance robotics is used for detecting obstacles and avoiding the collision. This is an autonomous robot. The design of obstacle avoidance robot requires the integration of many sensors according to their task.

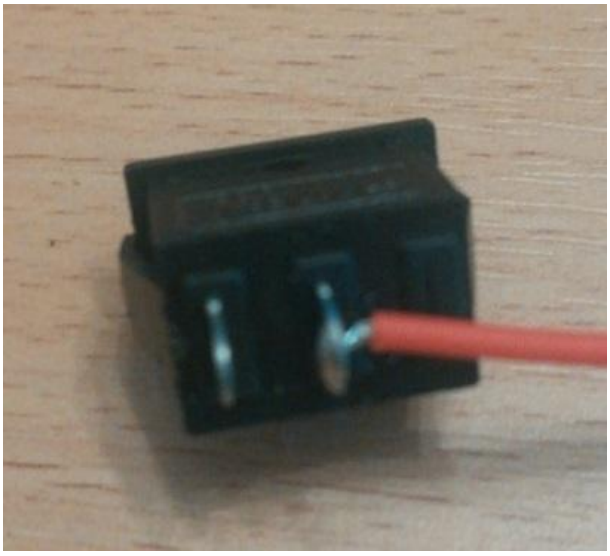
The obstacle detection is primary requirement of this autonomous robot. The robot gets the information from surrounding area through mounted sensors on the robot. Some sensing devices used for obstacle detection like bump sensor, infrared sensor, ultrasonic sensor etc.

Components and tools required

- Arduino UNO board
- Arduino sensor shield
- LN289 motor driver
- 2 geared motors
- Ultrasonic sensor with mount
- Chassy
- Wheels
- 4x AA battery holder
- Small rocker switch
- Jumper wires
- Various screws and spaces
- Soldering iron, screwdriver, hot glue gun, double sided tape

Step 1:

Solder wires on the geared motors and the small switch like shown in the pictures below



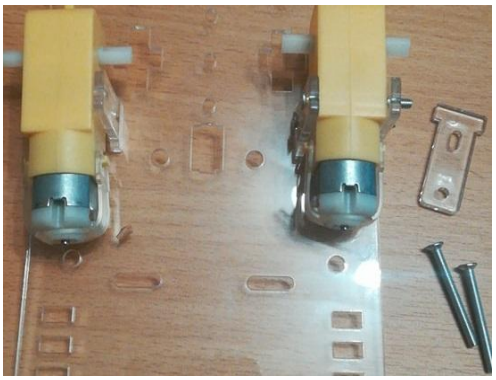
You only need to solder one wire to the switch for now, The other will be soldered in the final steps, after it is mounted on the chassy.

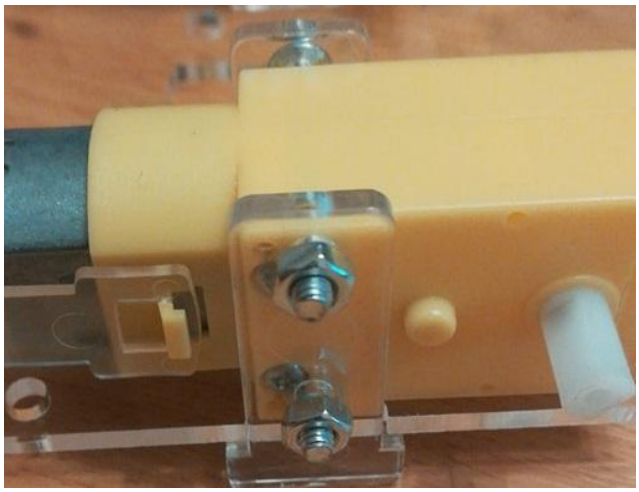
Step 2:

Slide in 2 motor mounts in the designated slots on the chassy as shown in the picture.

Step 3:

- Put the 2 motors near the mounting supports.
- Add the second pair of mounting supports on the other side of the motors
- Fasten everthing with 4 long screws that are provided in the kit



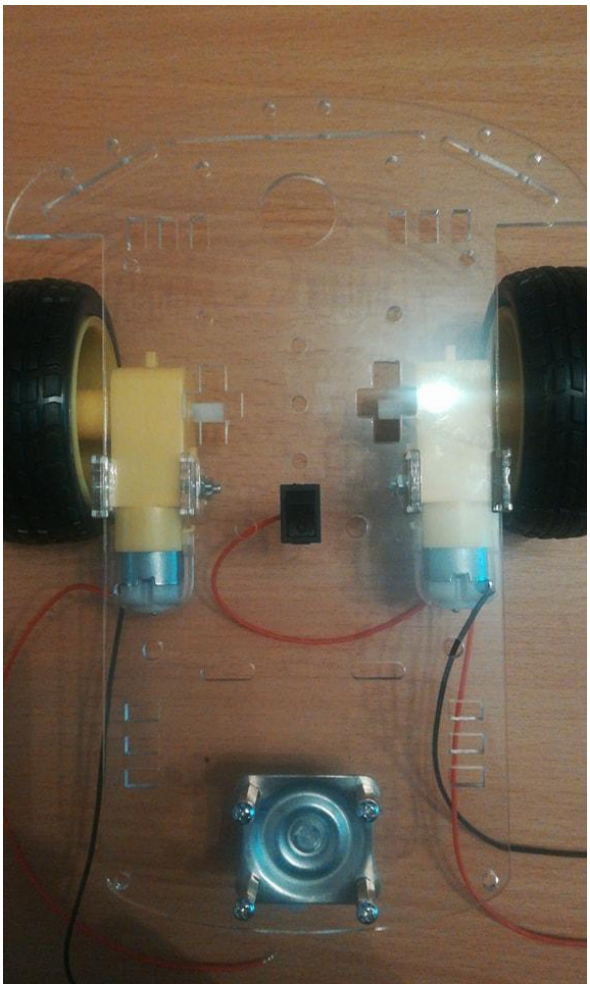


Step 4:

- Fasten the spacers for the caster wheel in the 4 holes at the back of the chassy.
- Mount the caster wheel on top of the spacers and secure it with screws.

Step 5:

Mount the wheels in the motor shafts, flip the chassy over and insert the rocker switch into the centre hole.

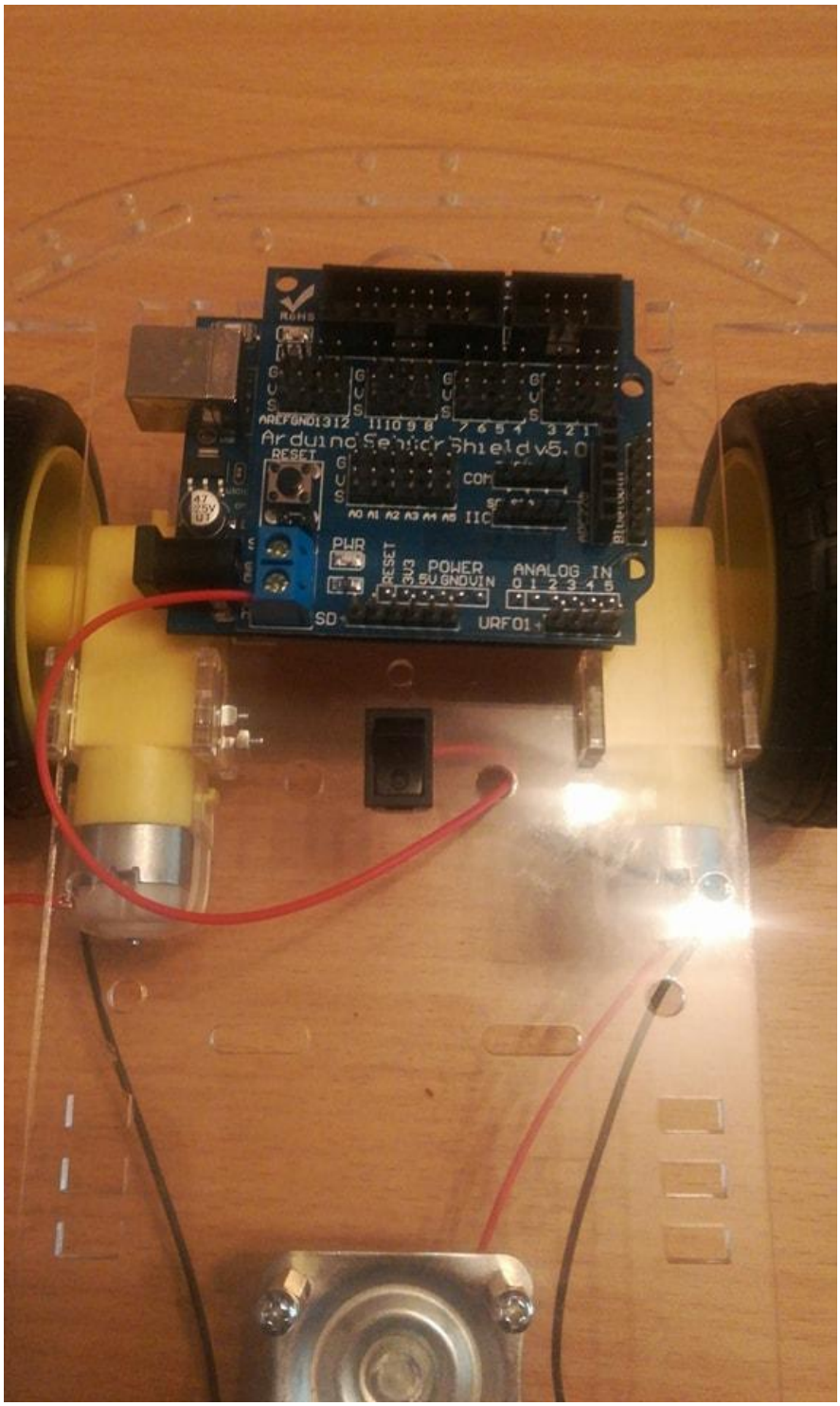


Step 6:

Mount the Arduino sensor shield on top of the Arduino

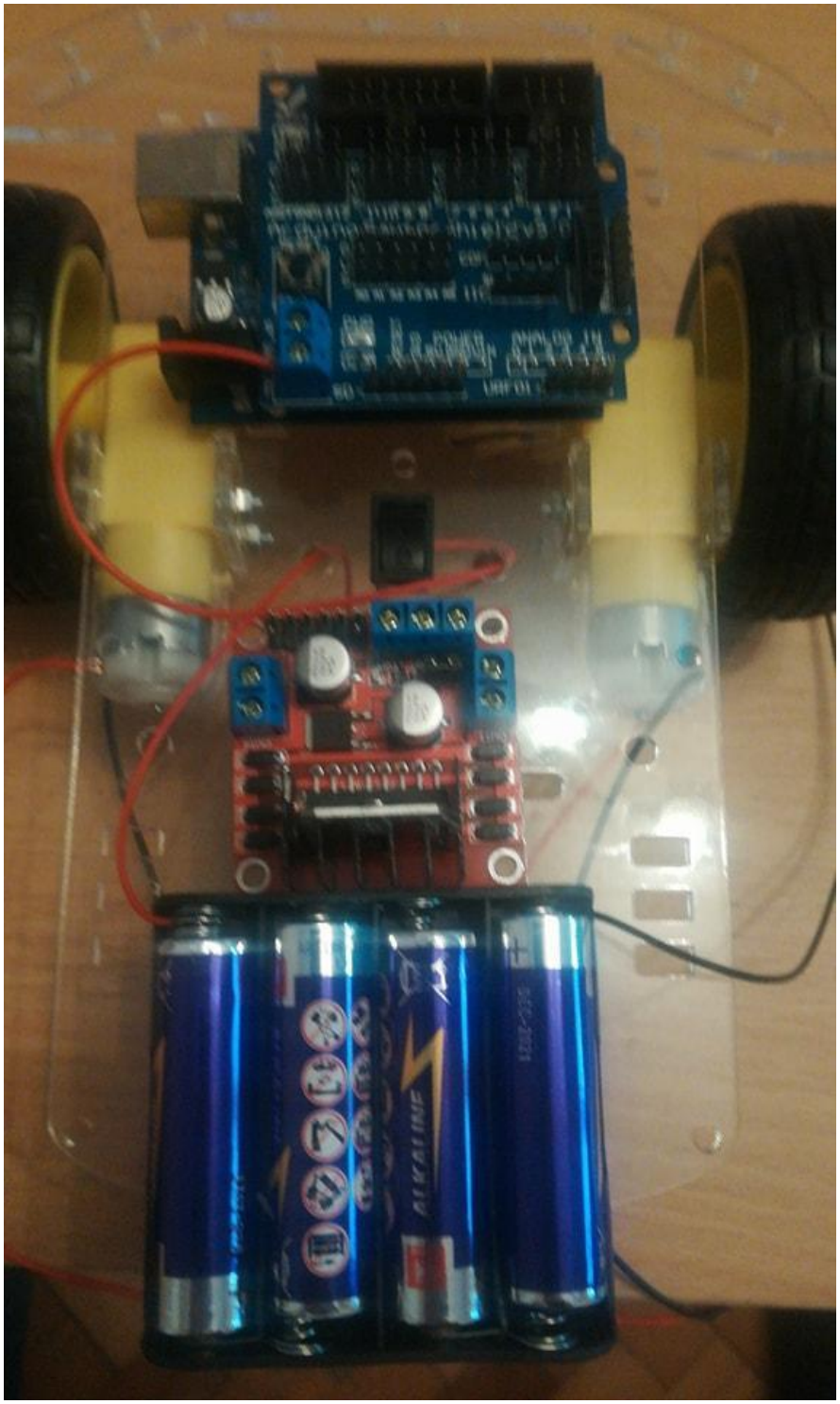
Step 7:

Mount the Arduino and sensor shield on the chassy, and connect the wire from the switch to VCC screw terminal on the sensor shield.



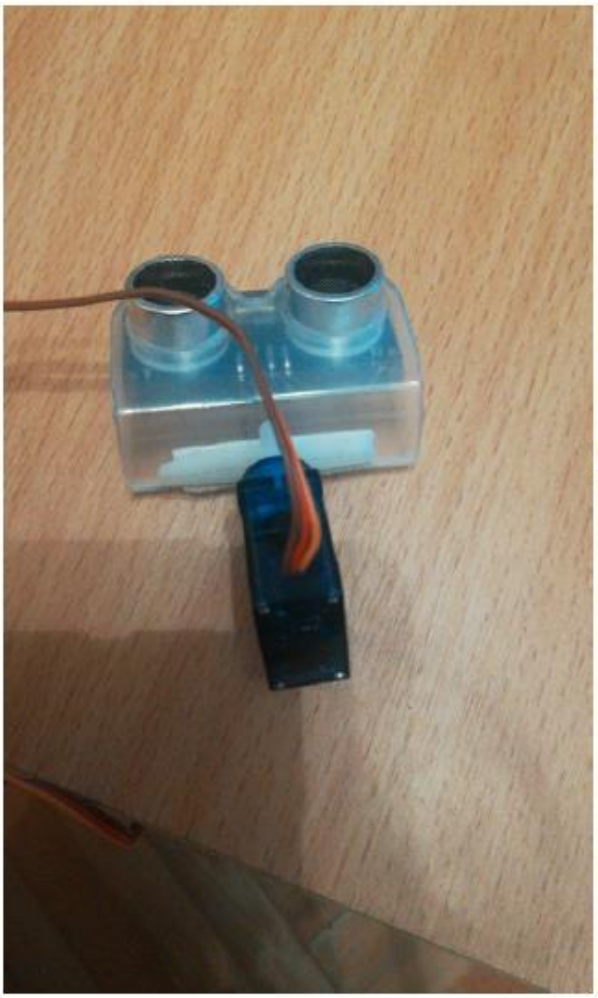
Step 8:

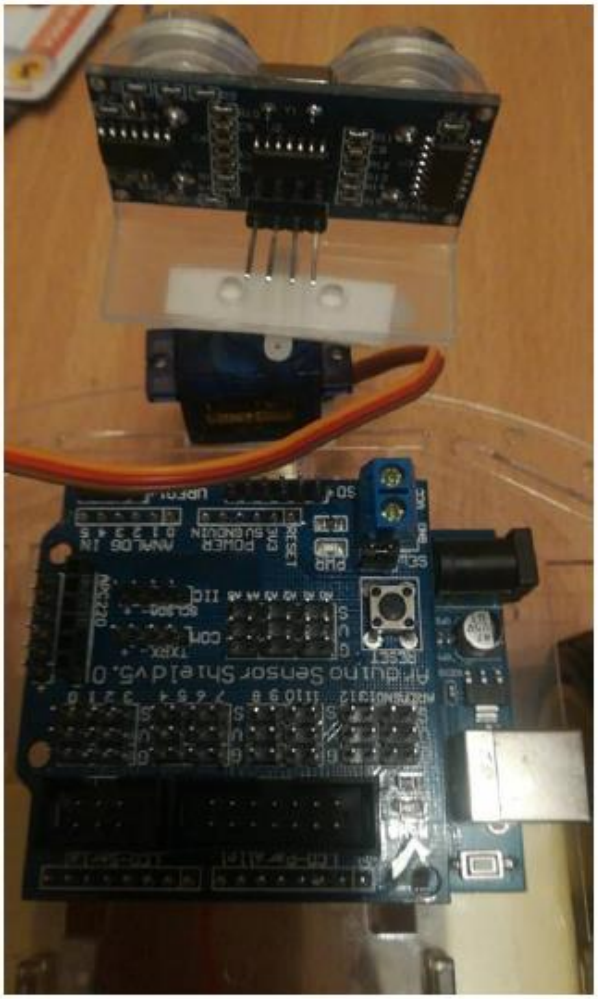
Mount the motor driver and battery holder using hot glue or double sided tape.



Step 9:

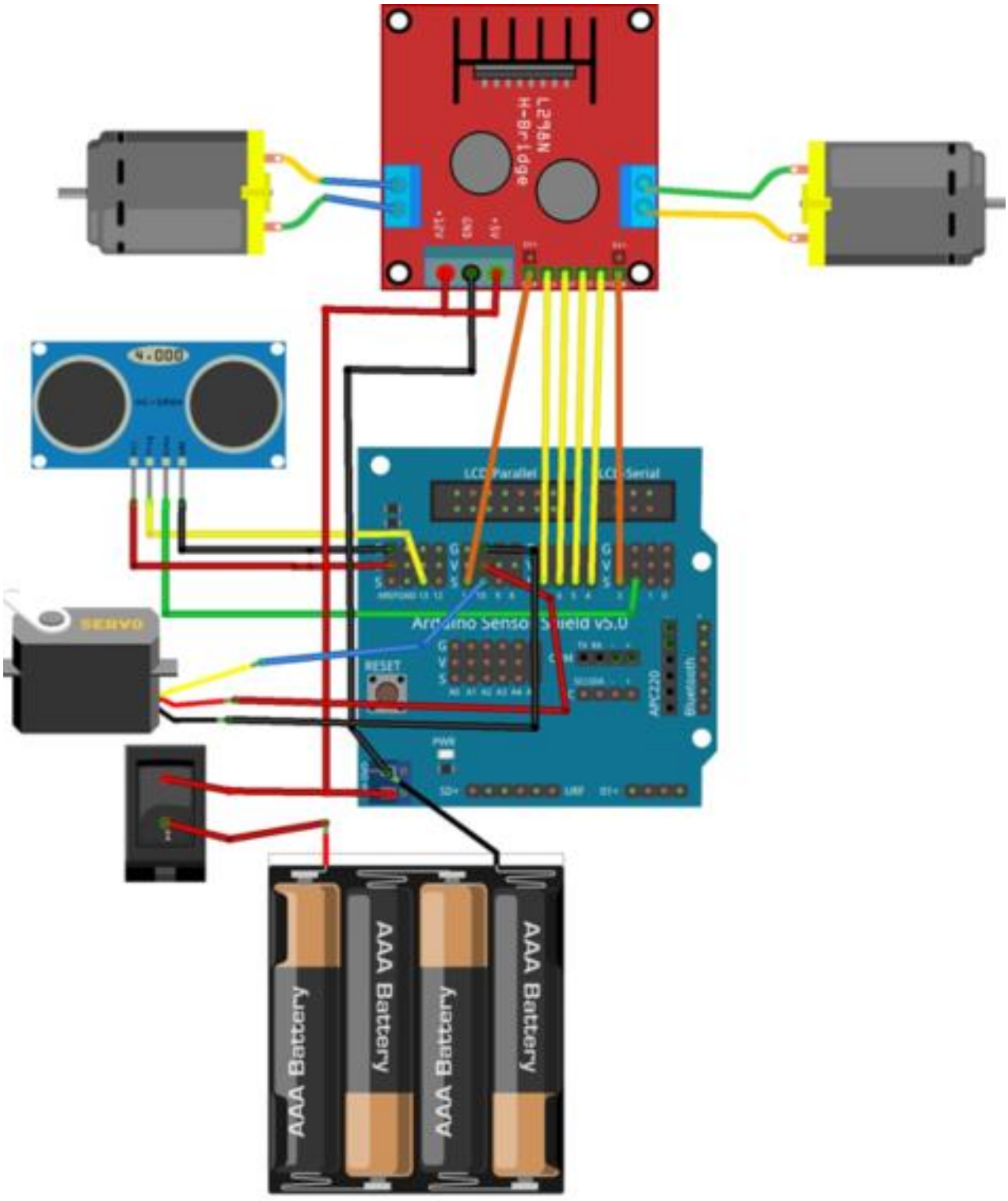
Mount the ultrasonic sensor in the holder and mount it with double sided tape on the servo, after that mount the servo - sensor assembly to the chasis.





Step 10:

Connect the electrical wires according to the schema below, and upload the provided code to the arduino.



Step 7: Programming the Robot

After you have finished connecting the motors to the motor shield you have now finished the assembly of robot. We now need to program it.

All you have to do is plug the UNO Board into your computer, load the Arduino software and write your code. Alternatively, you can use the sample code below:

```
/*
 *  Short code for the robot kit
 * When the ultrasonic sensor detects an obstacle the robot makes a left
turn
 * and continues the forward movement
 *
 * It's possible that the motor speed may vary. in order to correct this
you modify the variables vitezaMotor1 or VitezaMotor2
 * with values between 0 and 255
 *
 *
 */


#include<Servo.h>
Servo SM;


// define the pins used by the ultrasonic sensor
#define echoPin2 2
#define trigPin2 13

//define motor speed
//modify the values if the motors are not spinning at the same speed
//motor 1
#define vitezaMotor1 140
//motor 2
#define vitezaMotor2 140

/*
 define pins utilized for the motor control
 pins enA and enB are used to control the motor speed using the PWM signal
//definiti pinii utilizați pentru controlul motoarelor
//pinii enA si enB sunt folositi pentru a controla viteza motoarelor in
functie
//de semnalul PWM

//pins in1, in2, in3 si in4 are used to control the dirrection of the
motors (forward or backward)
*/

//pins used by the first motor
#define enA 11
#define in1 7 //LA
#define in2 6 //LB

//pins used by the second motor
#define enB 3
#define in3 5 //RA
#define in4 4 //RB

// forward = HIGH , backward = LOW
#define inainte HIGH
#define inapoi LOW

//declare the variables to determine the distance of the obstacles in front
of the robot
long duration1,duration2;
int distance1,distance2;

void directie (int dir1, int dir2)
```

Disclaimer: Libraries, example code, diagrams and resources are provided as an additional free service and are not sold as part of this product. Phipps Electronics Pty Ltd does not provide any guarantees or warranties as to their accuracy or fitness for purpose.