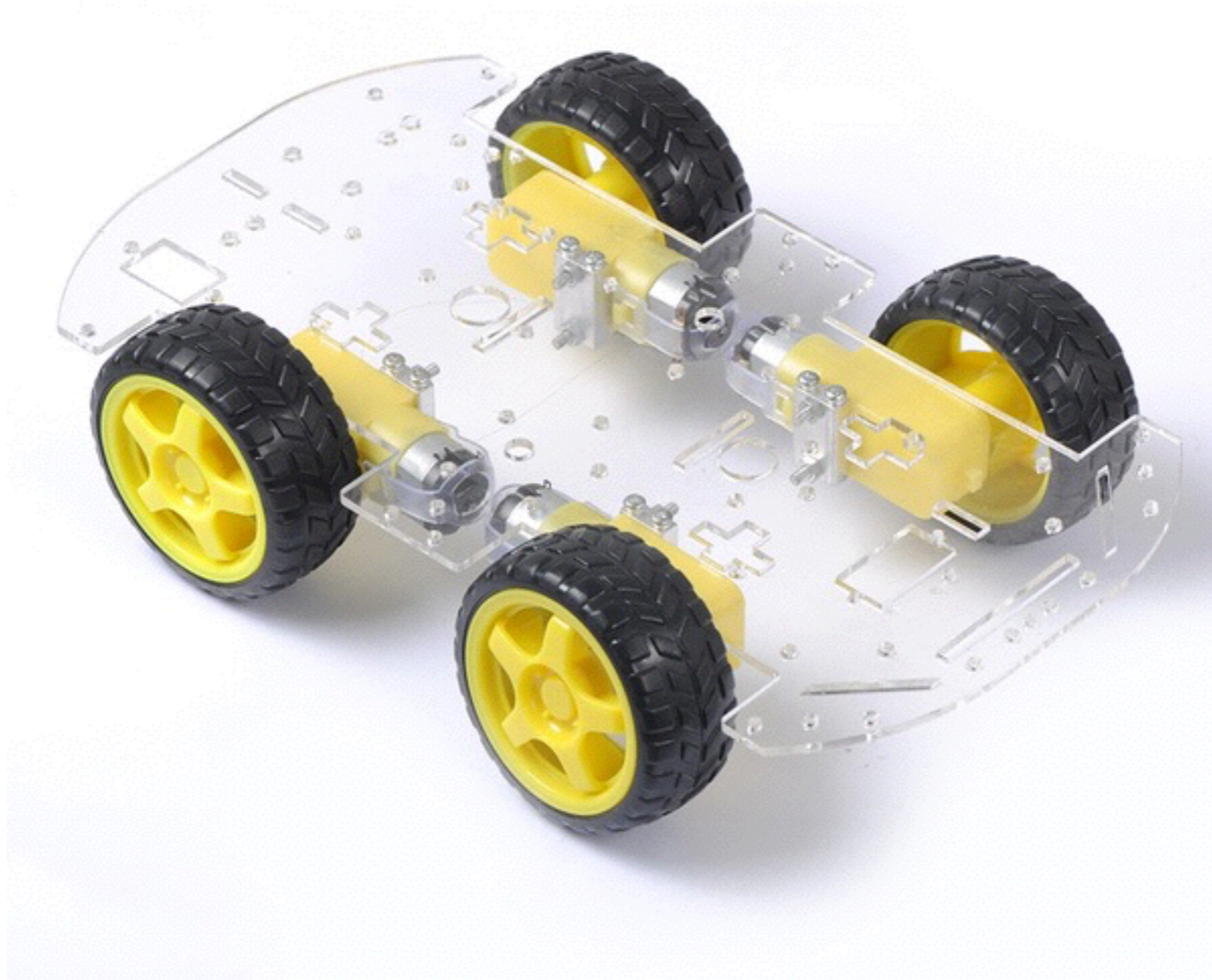


Robot Workshop

Josh Harle



What we'll do

1. robots in art
2. robot layout - talk about what everything does, things that could be added
3. instructions for building the robots
4. how we can change the path they draw
5. get started building
6. play with finished robots!

some use of robots in art

Rebecca Horn



Stelarc



Katsu

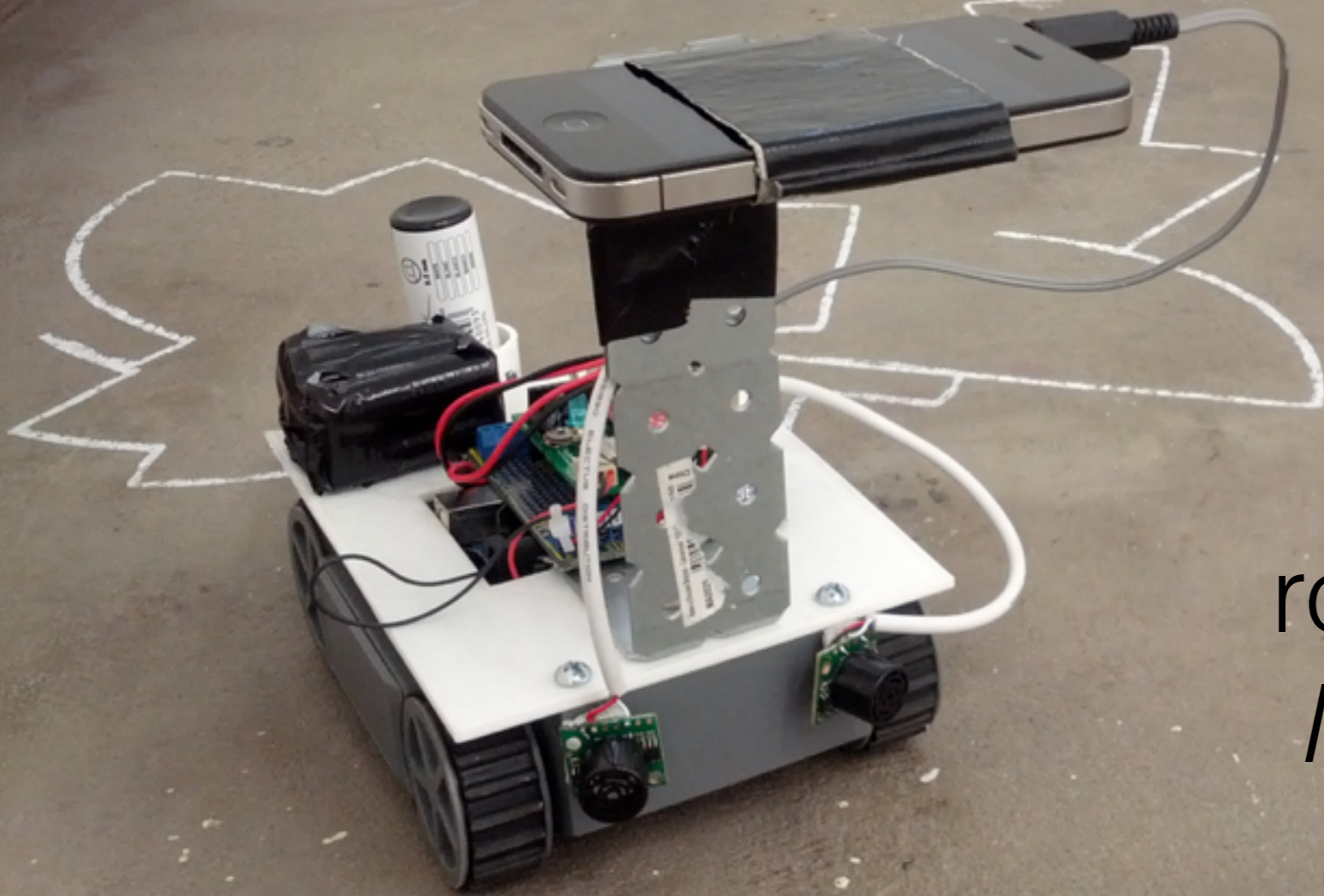


Calvin Klein

AT OPENING CE

#mycalv

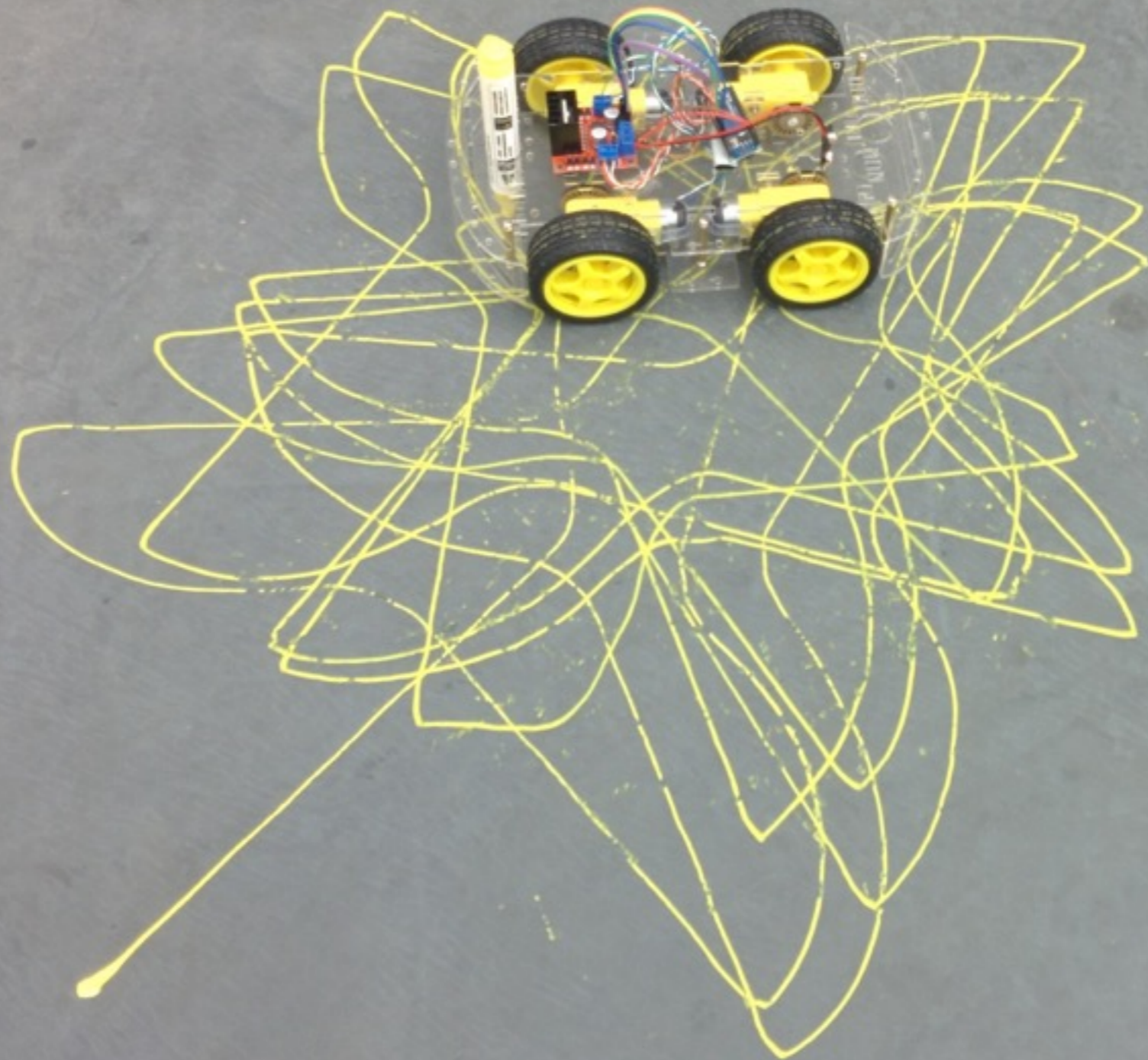




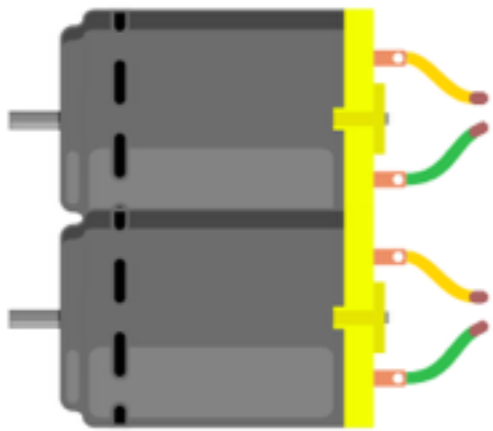
robot based on
Making Sense



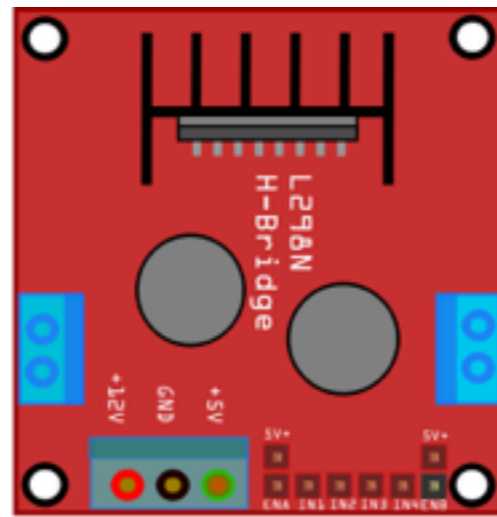
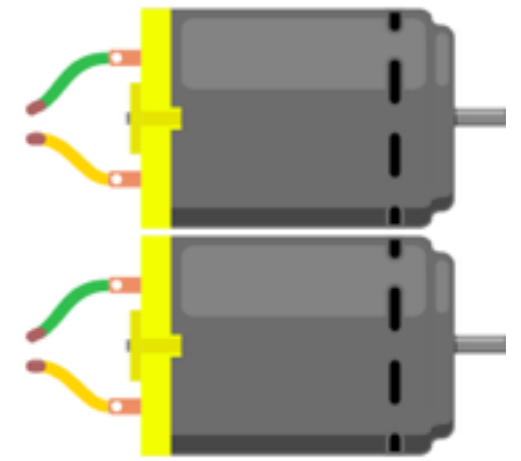
robot layout



motors



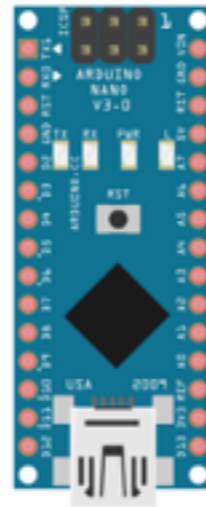
motors



motor controller



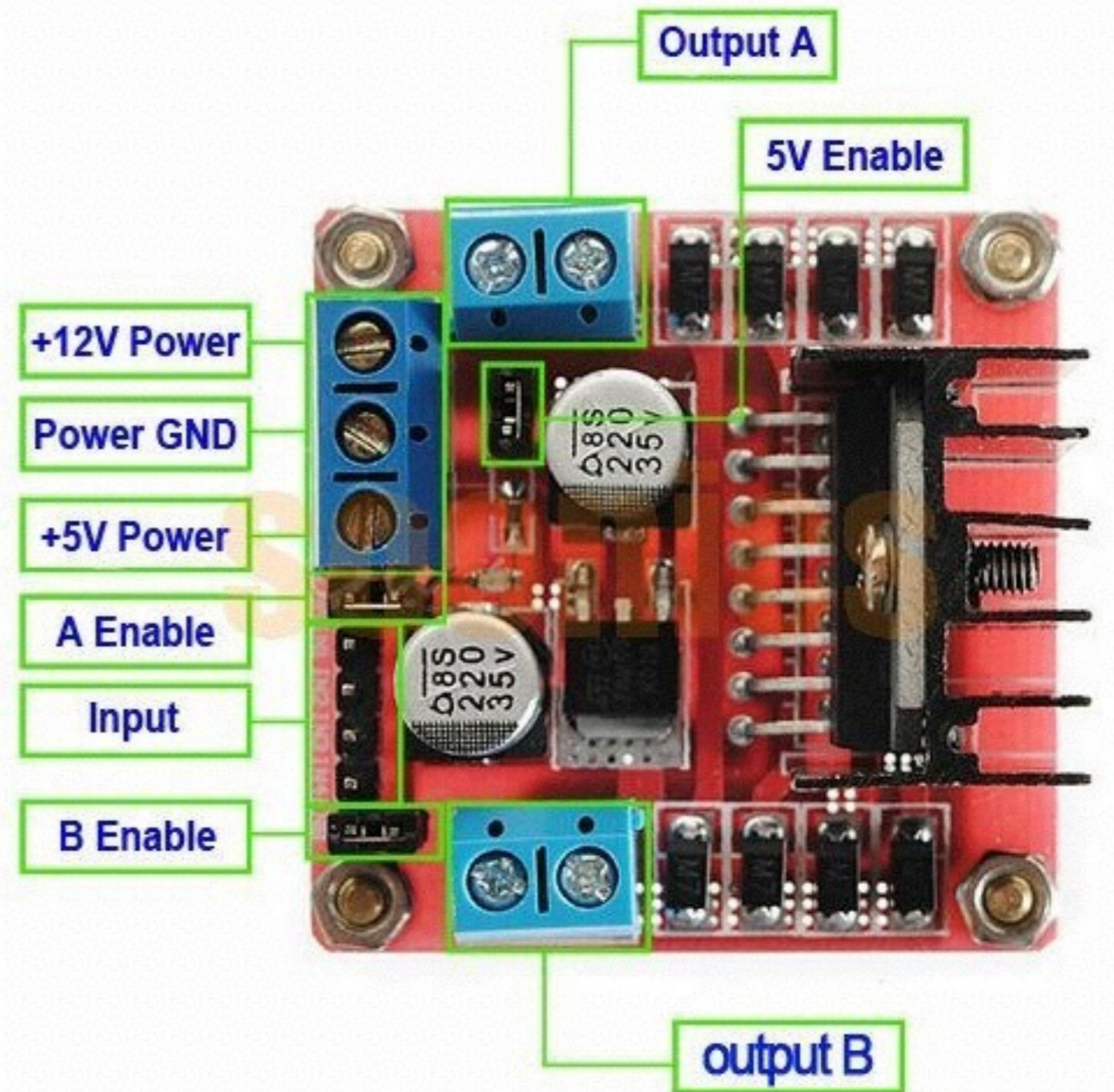
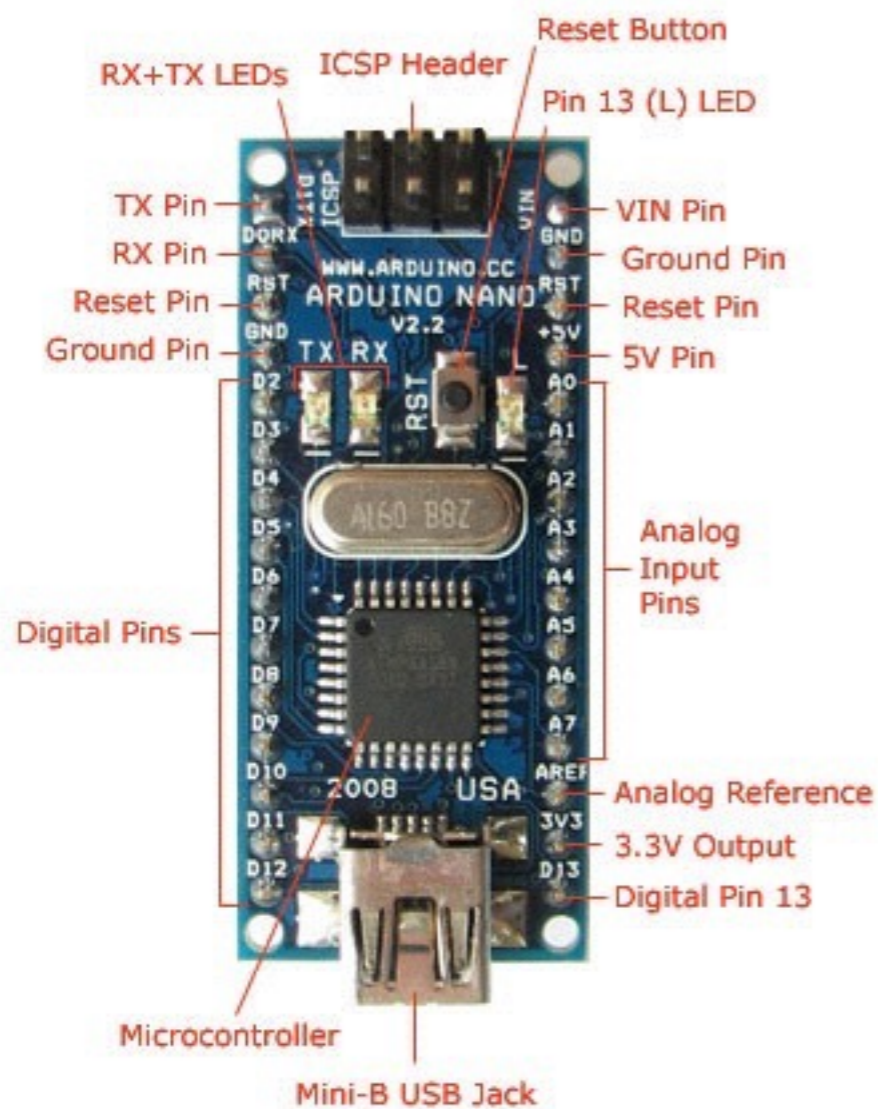
battery



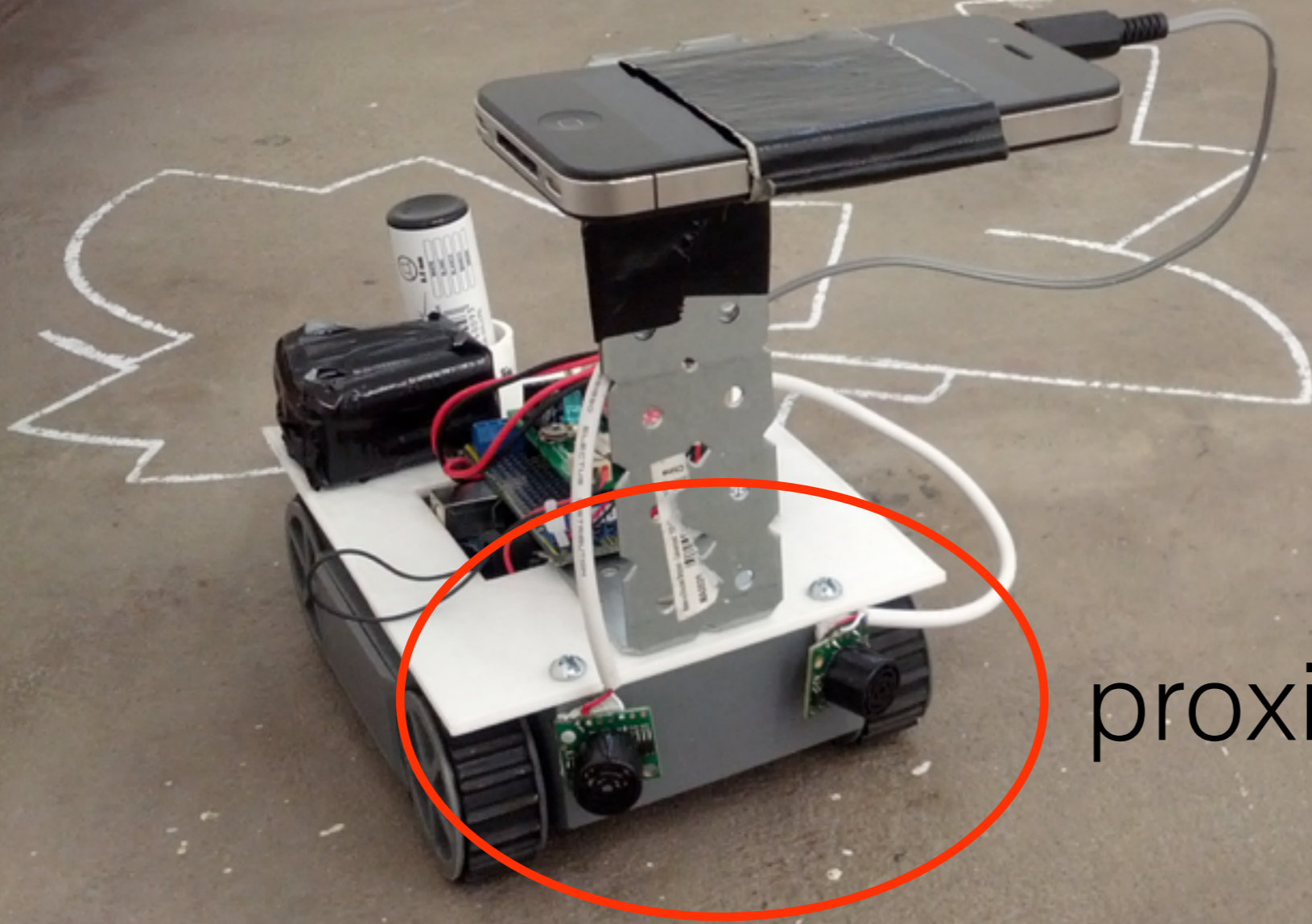
arduino nano

fritzing

Arduino (brains) and motor controller



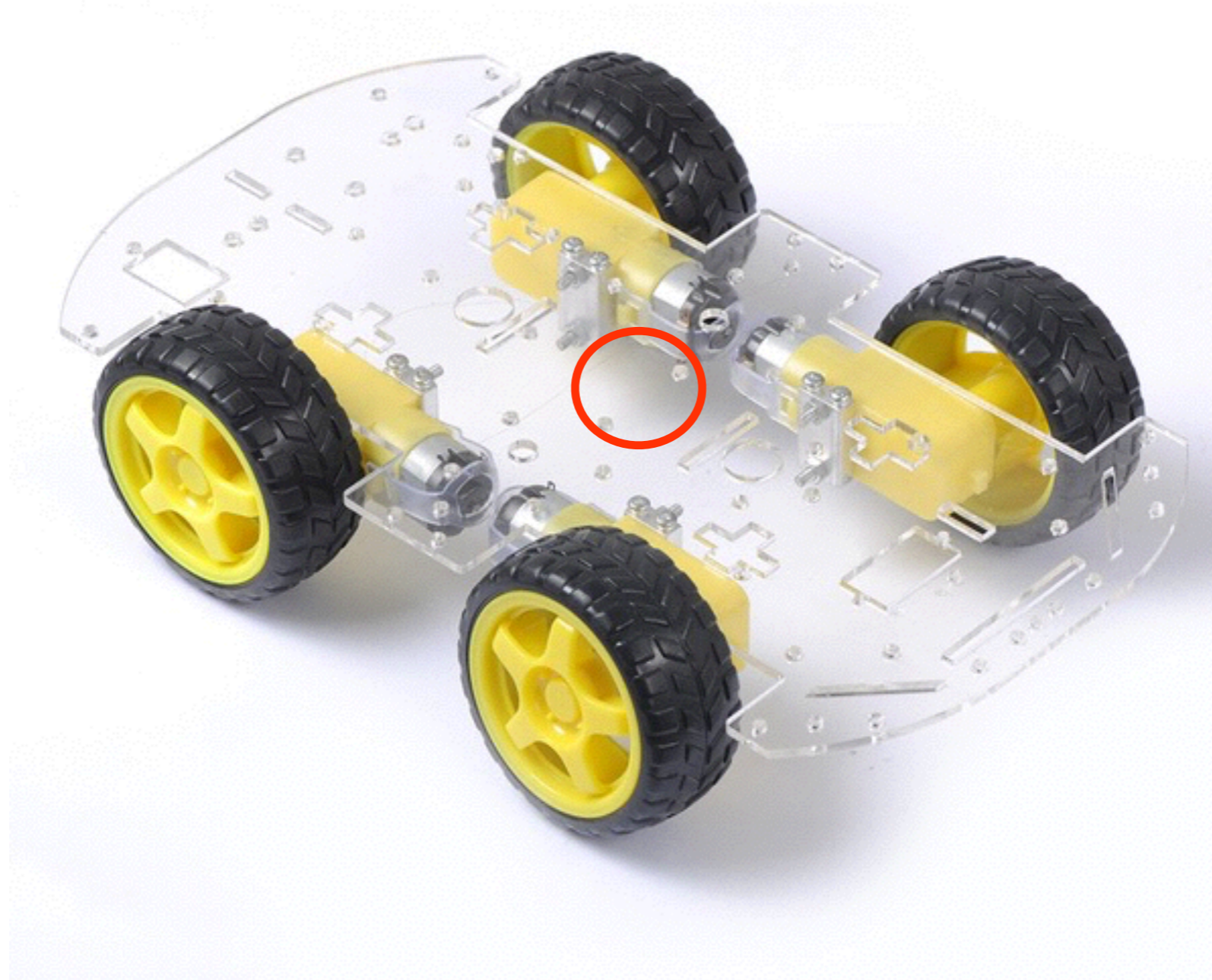
optional parts

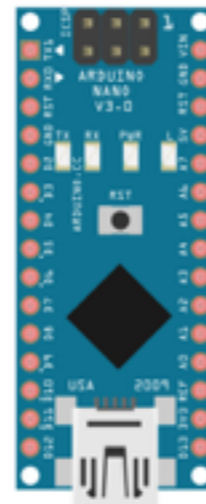
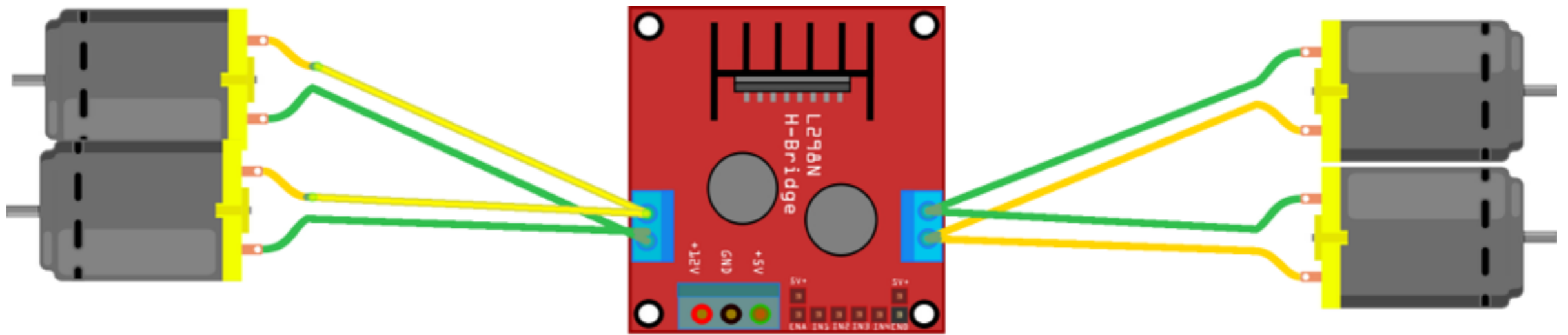


proximity sensors

build instructions

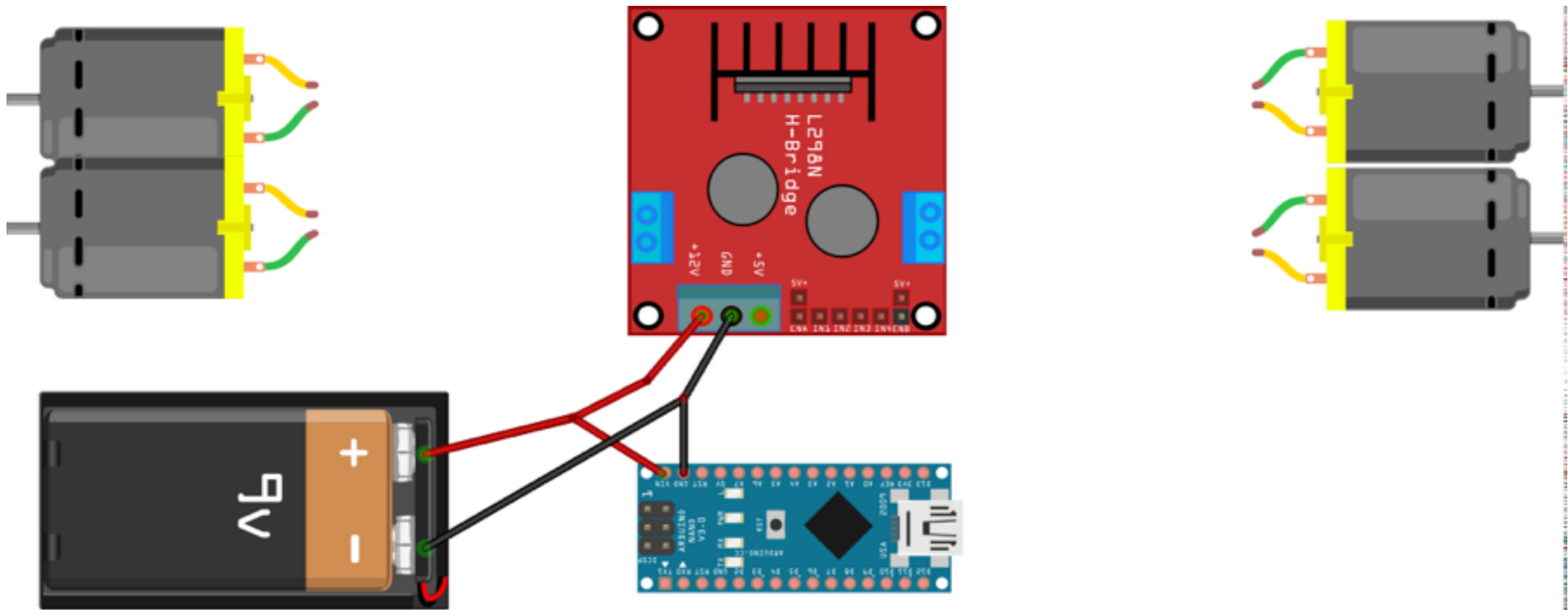
- follow instructions for base construction (connected motors are on the same side left + left, right + right)
- Pull motor connectors through the central hole in the top plate





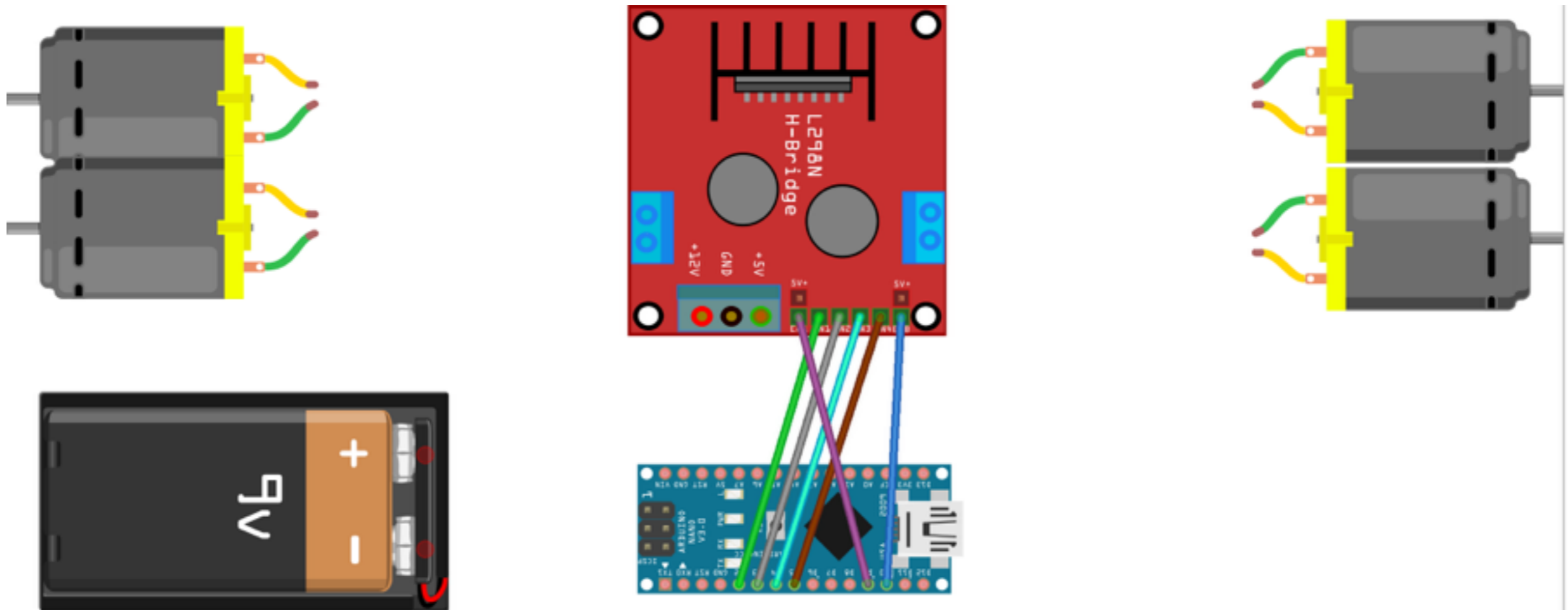
fritzing

Screw in the motor connectors



fritzing

Screw in the battery connectors, and attach
+ to “vin” and
- to “grn” on nano



fritzing

- Remove the 'jumpers' from the two end pins
- connect D2, D3, D4, D5 to the middle 4 pins on the motor controller
- connect D9 and D10 to A Enable and B Enable on the motor controller

changing the path

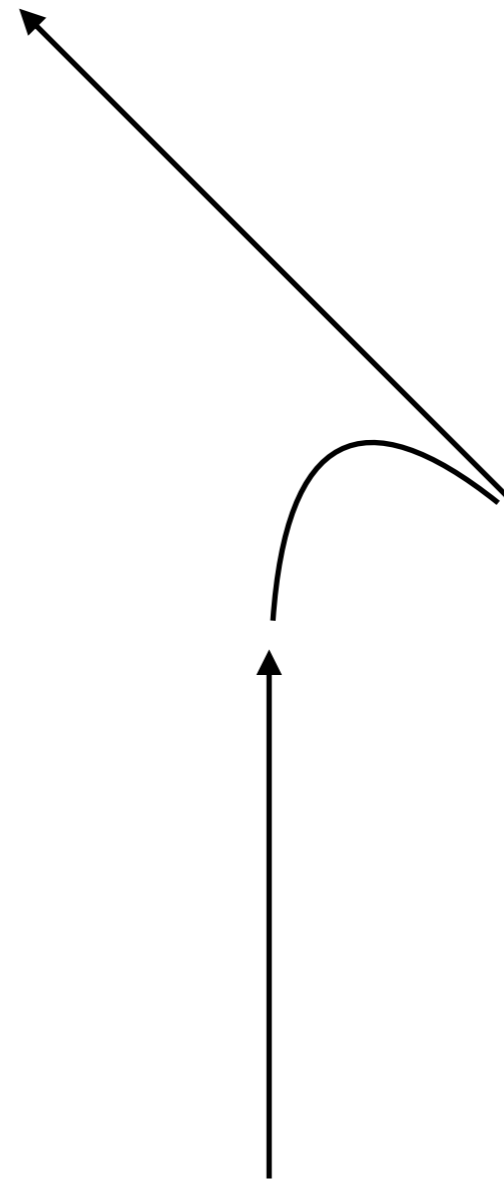
Programming the Arduino

```
void loop() {  
  int driveSpeed = 100;  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
}
```

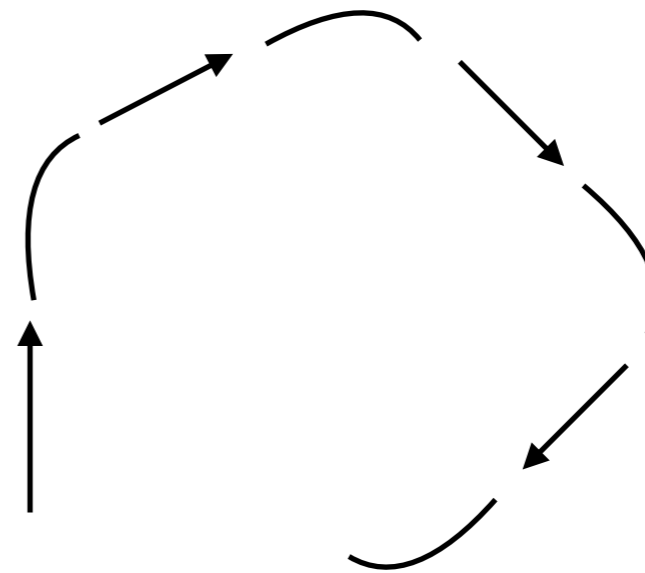
Use special instructions to control the behaviour of the robot. I have made the following movement commands:

- forward
- left
- right
- back
- curve
- randomChoice

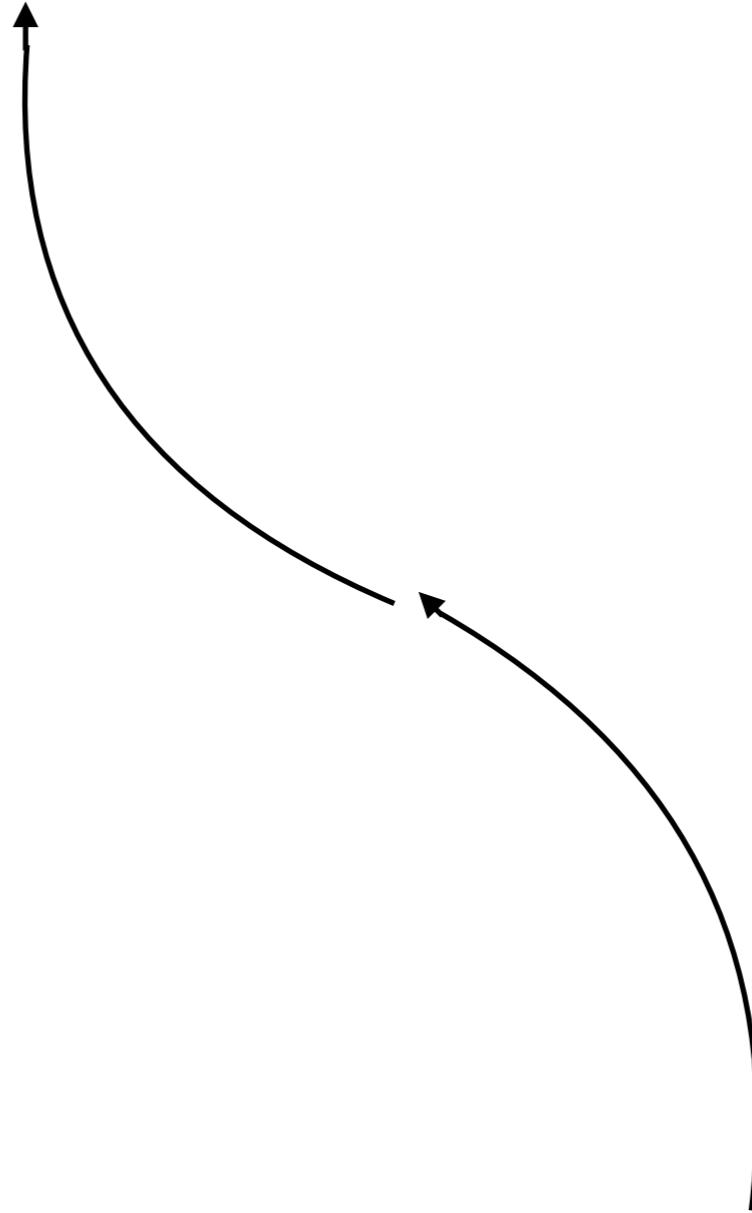
```
void loop() {  
  int driveSpeed = 100;  
  forward(driveSpeed, 4000);  
  delay(2000);  
  right(255, 2000);  
  delay(2000);  
  back(driveSpeed, 4000);  
  delay(2000);  
}
```



```
void loop() {  
  int driveSpeed = 100;  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
  forward(driveSpeed, 2000);  
  delay(2000);  
  right(255, 400);  
  delay(2000);  
}
```

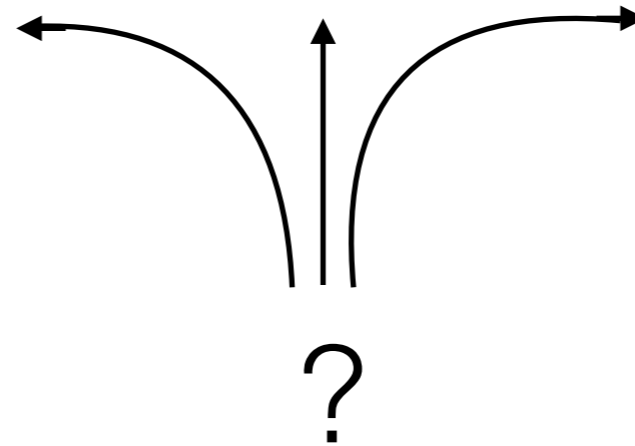



```
void loop() {  
  curve(120,255, 1000);  
  delay(2000);  
  curve(255, 120, 1000);  
  delay(2000);  
}
```



- randomChoice(leftChoice, rightChoice, forwardChoice)
- randomChoice(1,1,4);

```
void loop() {  
  randomChoice();  
  delay(2000);  
}
```



let's get building!



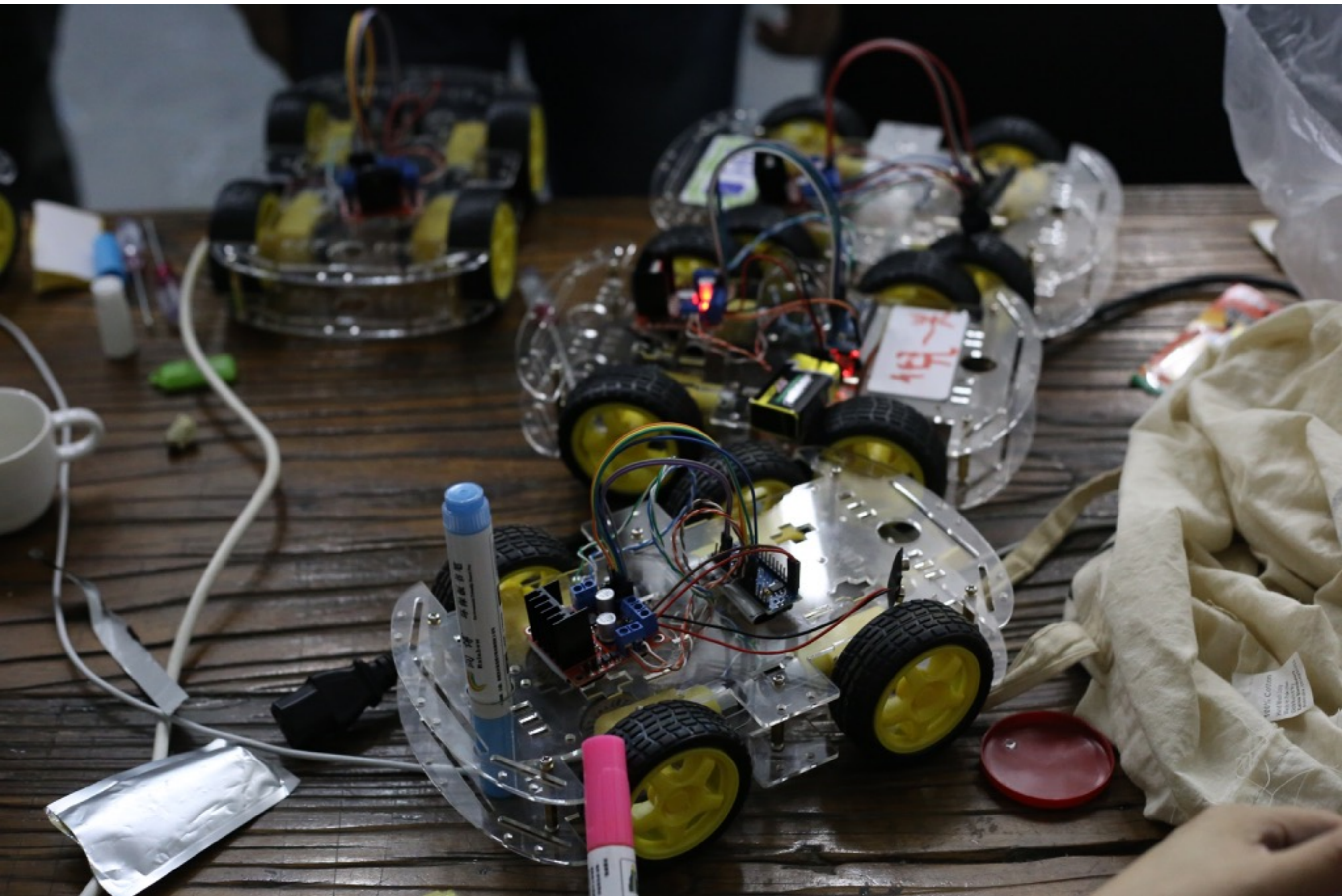




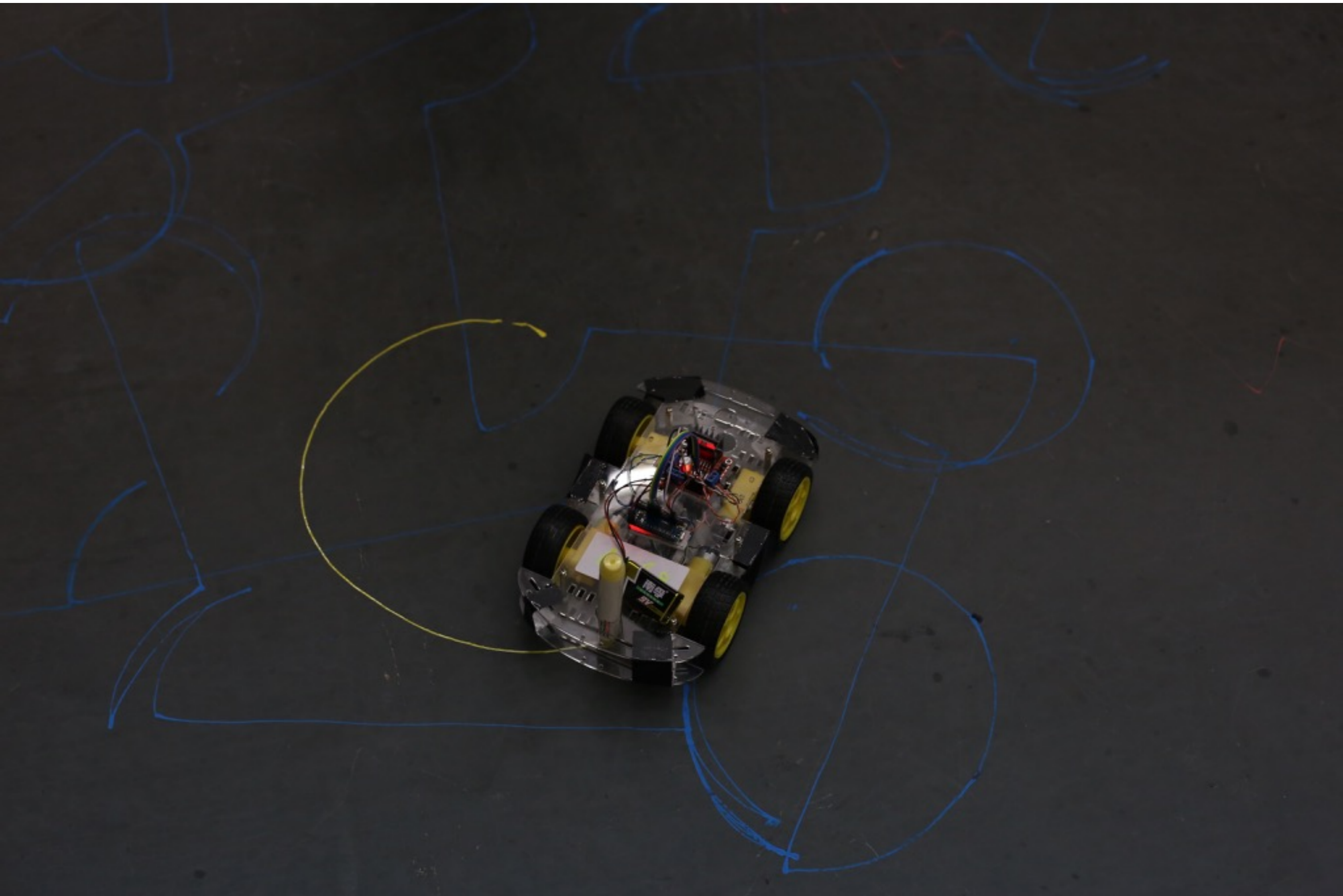


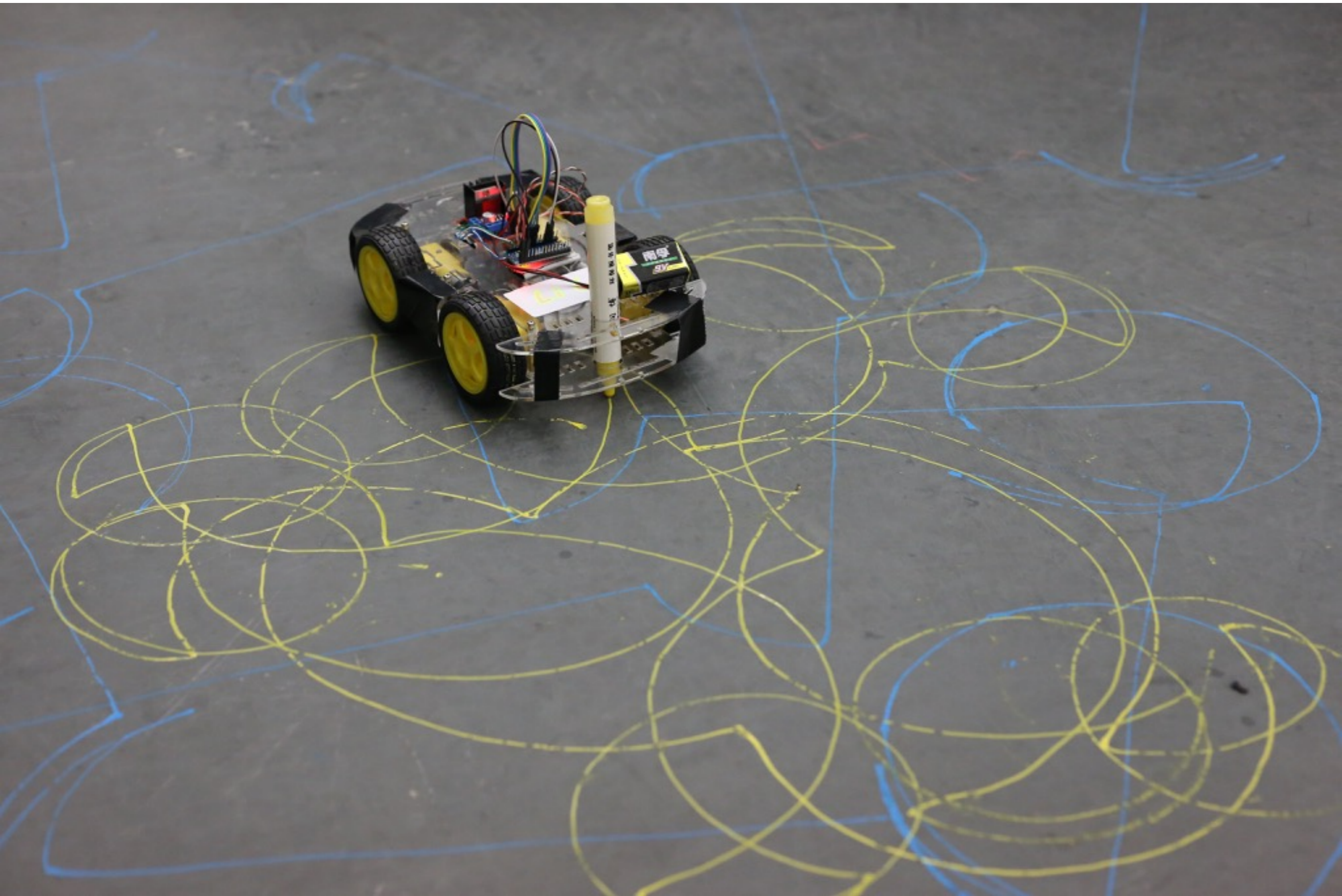


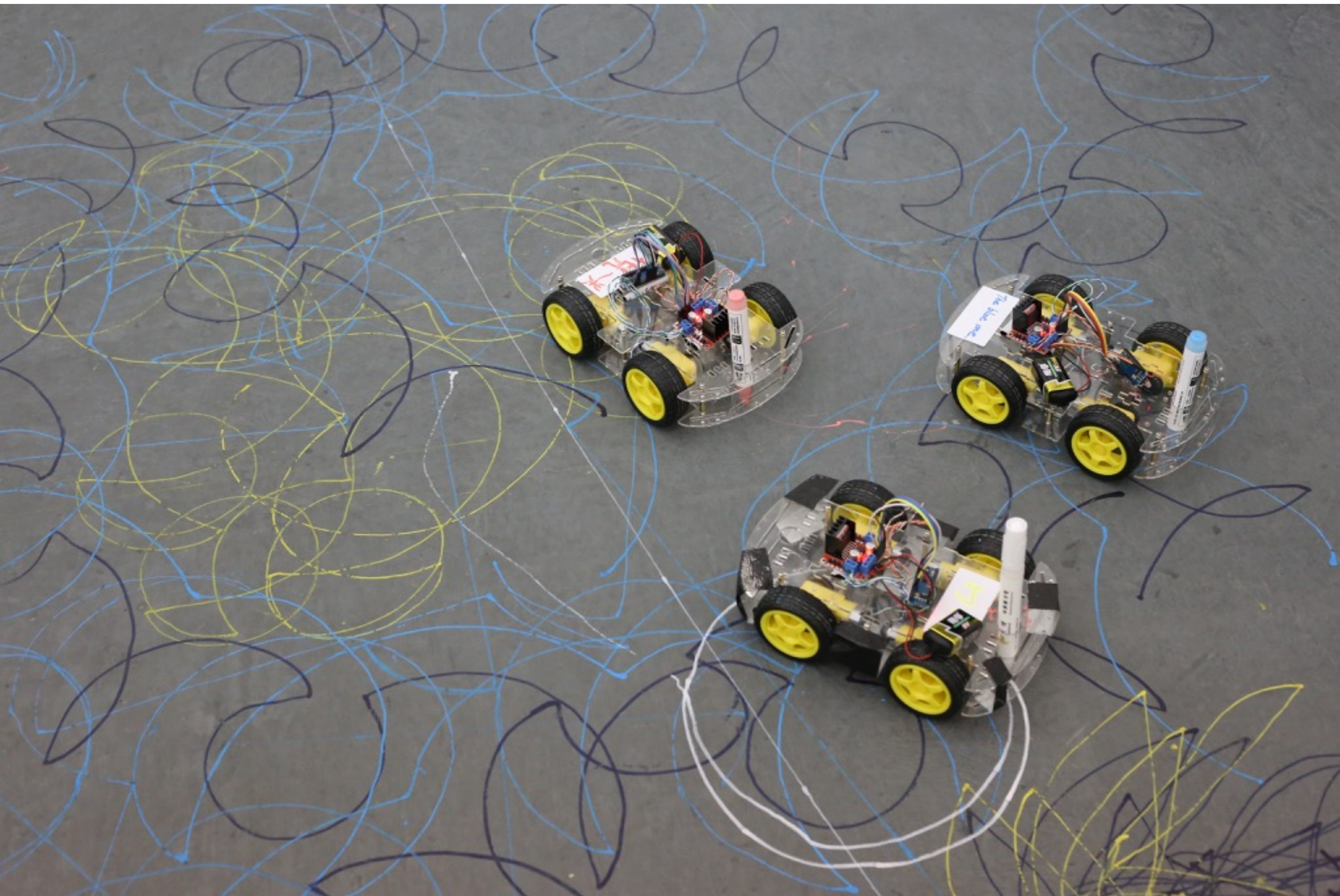














Useful Links

- The control program for the robot: http://tacticalspace.org/files/pattern_drawing.ino
- Arduino program for Mac and Windows, with useful learning resources: <http://arduino.org/>
- Instructions for installing driver for connecting to nano clone that comes with your robot: <http://kiguino.moos.io/2014/12/31/how-to-use-arduino-nano-mini-pro-with-CH340G-on-mac-osx-yosemite.html>
- Download for Windows driver for nano clone: <http://www.5v.ru/zip/ch341ser.zip>
- Download for Mac driver for nano clone: http://kiguino.moos.io/downloads/CH341SER_MAC.ZIP