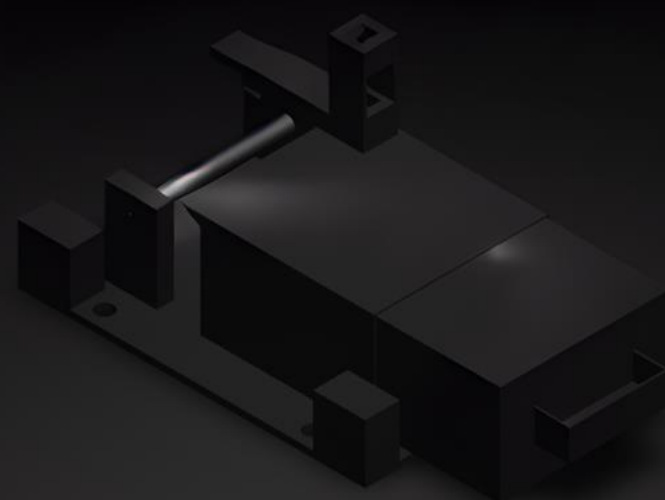


M87

SMART VACUUM CLEANER



មាតិកា

- តួនាទី
- មូលហេតុដែលបង្កើត/ទស្សនវិស័យ
- សម្ភារៈ និងការចំណាយ
- ដំណើរការនៃការបង្កើត
 - បង្កើនបរាង និងតួឡាន
 - បង្កើនខ្សែនិងគ្រឿងអេឡិចត្រូនិច
 - ការបញ្ចូលកូដដើម្បីគ្រប់គ្រងលើ Motor Driver
- បញ្ហាដែលយើងបានដោះស្រាយ និង បញ្ហាគួរតែប្រុងប្រយ័ត្ន
- សម្ភារៈ និងការចំណាយ
- ដំណើរការនៃការបង្កើត
- Timeline
- សរុបសេចក្តី

GROUP:

1

M87



ឃ្យៀប ឈុនសាត្ត

សយ ទិត

សារី គីមហុង

សុខ ជីននី



សាំង សេងថាយ

លឹម លក្ខណ៍សុផន

សាយ សក្កីសុវឌ្ឍនៈ



តួនាទី

សារី គីមហុង Build Up, Bugged , Warning

សុខ ជីននី Materials Management

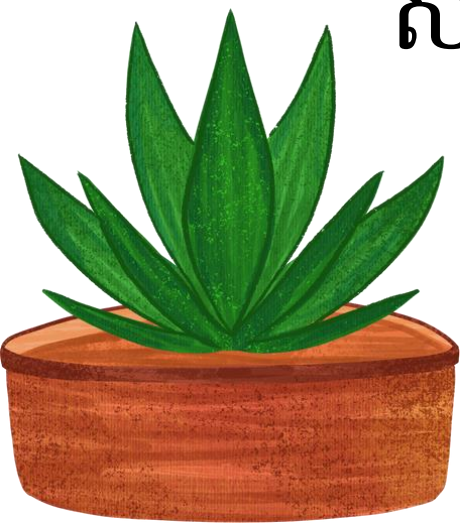
លីម លក្ខណ៍សុធន Assitant

សាំង សេងថាយ Build Up, 3D drawing

យ៉េប ឈុនសាន្ត Code Installation

សយ ទិត Assitant

សាយ សក្តិសុវឌ្ឍនៈ Presentation Slide | Design

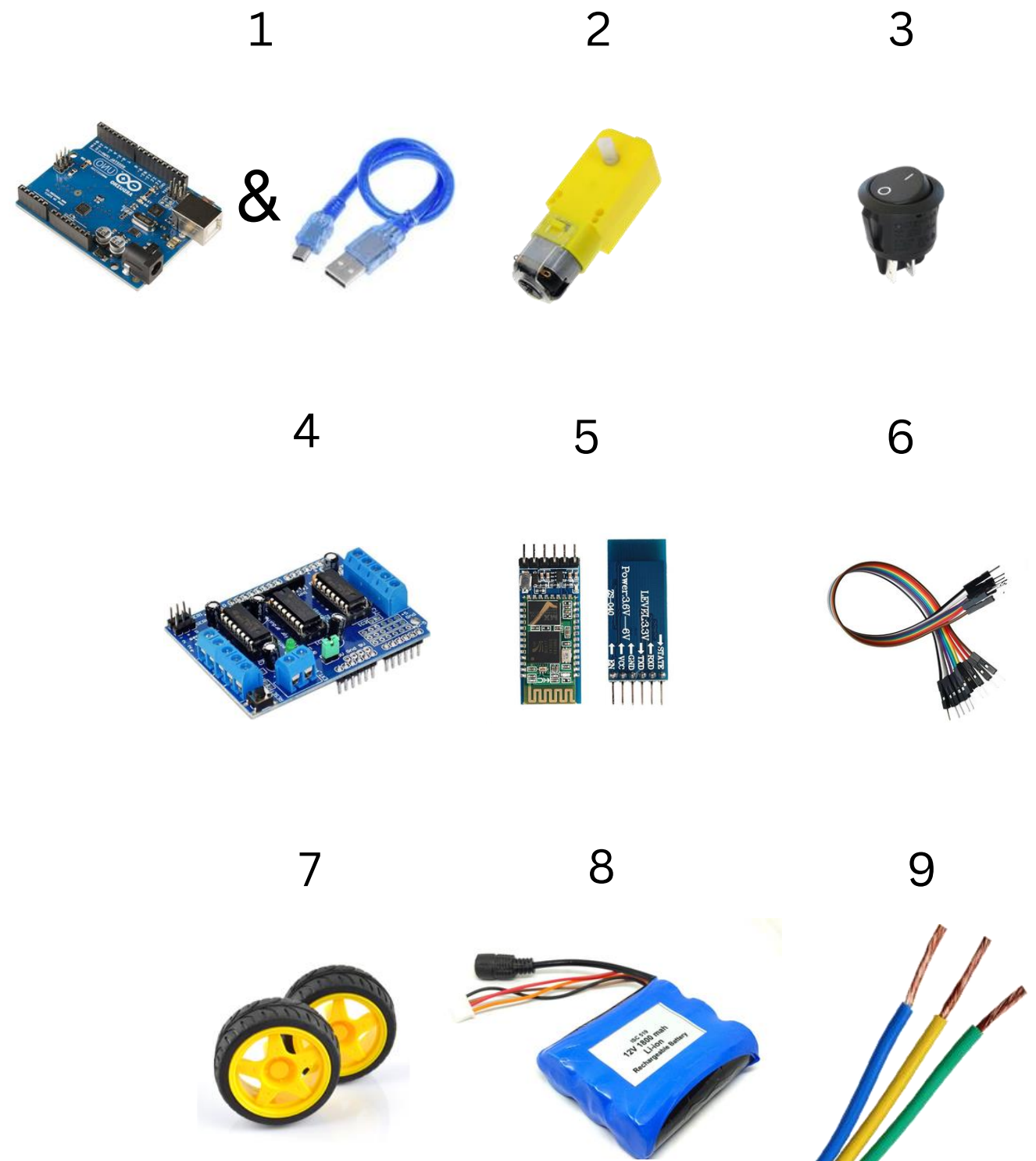


មូលហេតុដែលបង្កើត/ទស្សនវិស័យ

- (M87) គឺជា ឧបករណ៍ម្យ៉ាងដែលអាចអនុញ្ញាតអោយអ្នកប្រើប្រាស់អាចបញ្ជាវាបានតាមរយៈទូរស័ព្ទ ហើយវាមានសមត្ថភាពអាចបូមយកផ្ទាល់ និងកំទេចកំទីតូចៗបាន។
- ដោយការមើលឃើញថា សម្រាមមានគ្រប់ទីកន្លែង ប្រសិនបើយើងមិនសំអាតប៉ុន្តែពេលខ្លះខ្លួនមិនចង់សំអាតគិតថាហាត់ ហើយដៃច្រើនកាន់តែទូរស័ព្ទ ទើបពួកយើងបង្កើតប្រភេទនេះដែលអាចបញ្ជារតាមទូរស័ព្ទបាន

សម្ភារៈ និងការចំណាយ

Name	Amount	Price
Aduino Uno board+11.Aduino USB CABLE TYPE A/B 1M	1	\$ 8.50
Gear Motors	4	\$ 1.00
Switch	1	\$ 0.20
L293D Motor Driver Shield	1	\$ 2.00
Bluetooth Module	1	\$ 5.00
Jumper Wire	1 Set	\$ 0.70
Wheel	4	\$ 0.50
Lithim-ion battery cell	1 Set	\$ 2.00
electrical wire	1 Set	\$ 0.20
Grand Total		\$ 24.60



M 87

ជំរឿនកំរ

បង្កើត

PHNOM PENH, TOUL KORK, RUPP

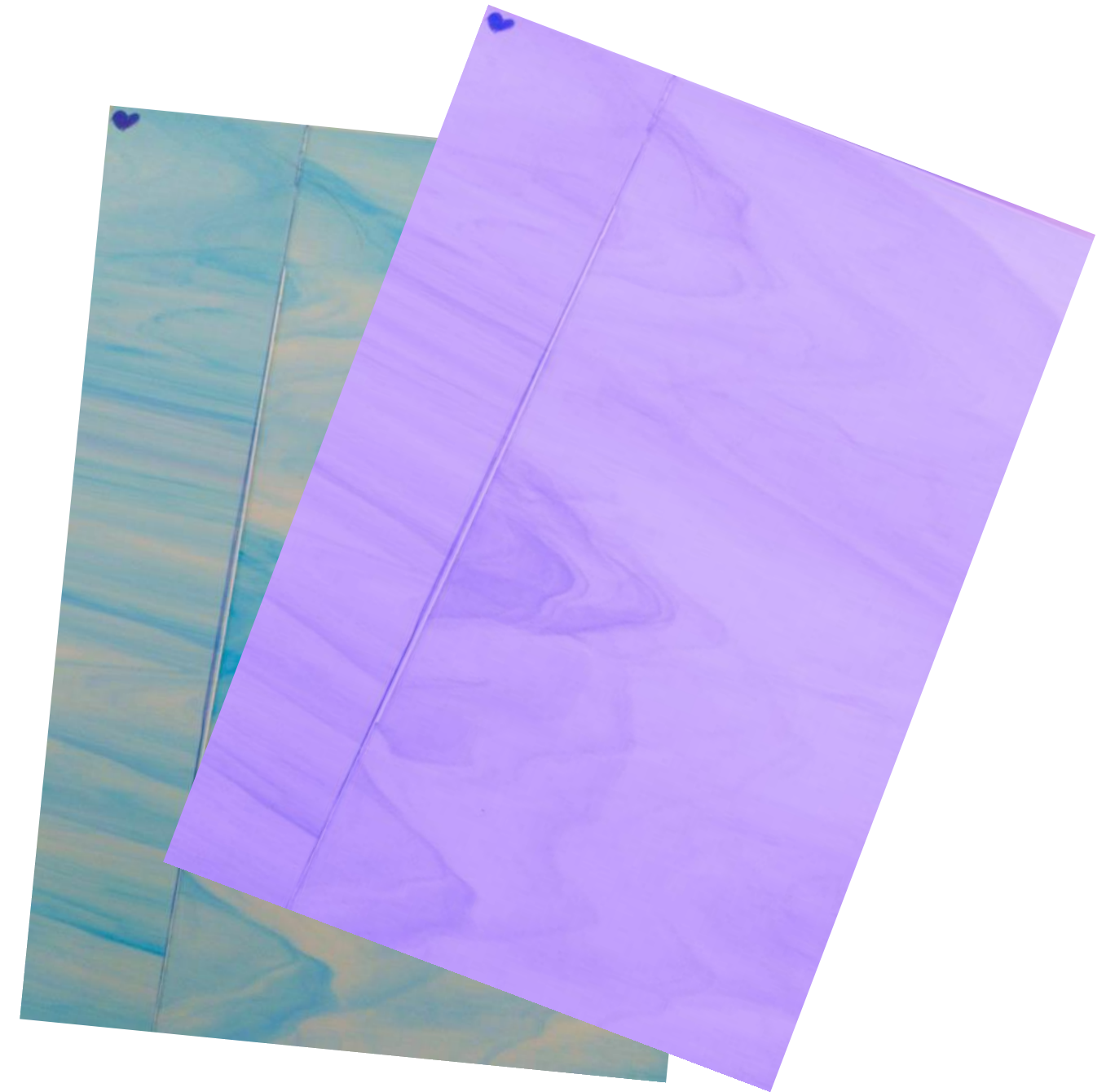
សម្ភារៈនៃបង្គំតួរឡាន

ជំរឿករីង

ពួកយើងបានទិញវាមកពីកន្លែងលក់គ្រឿងសំណង់នៅ

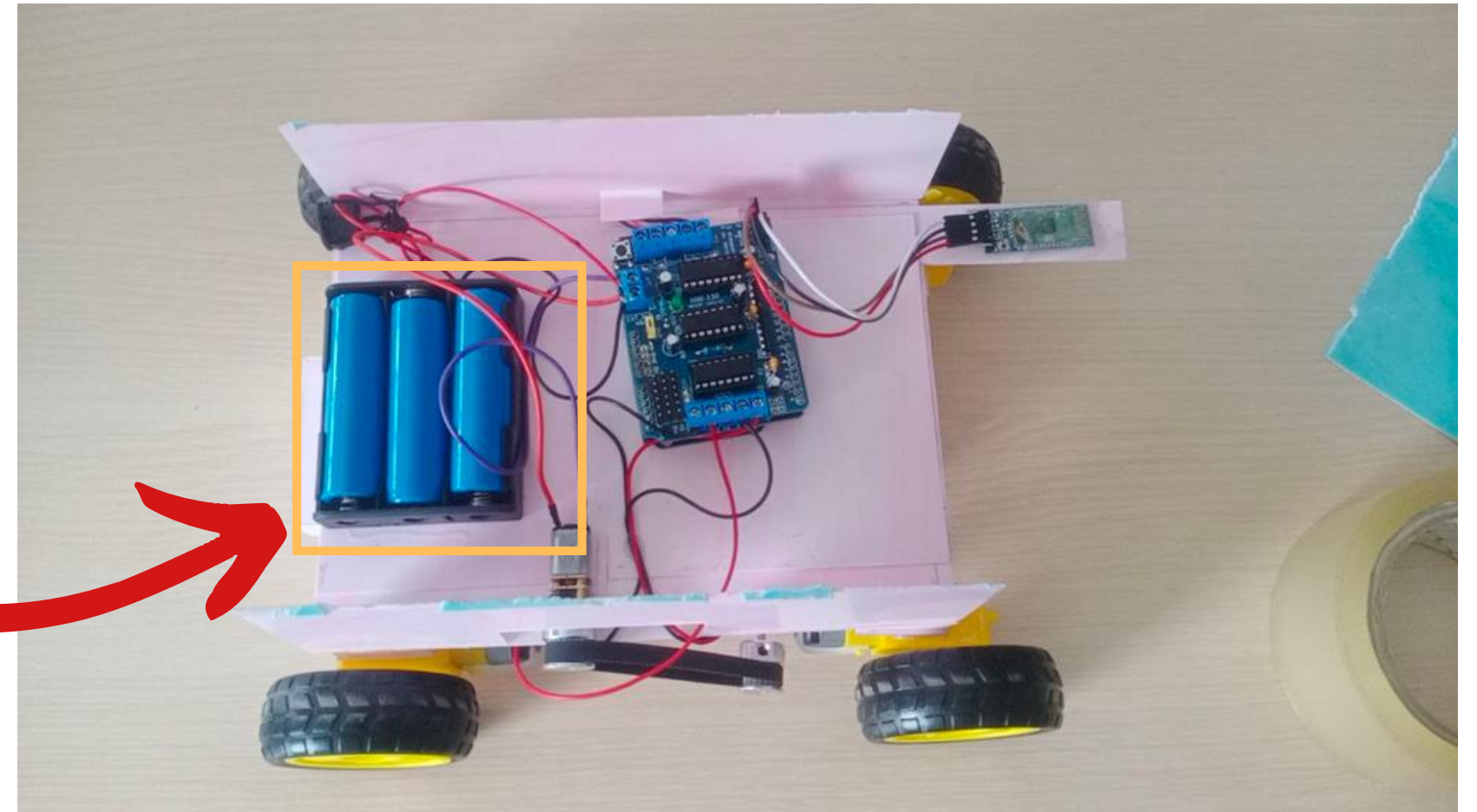
ម៉ូជិតផ្ទះ ក្បែរផ្សារ

- បន្ទះក្តារដែលបានប្រើប្រាស់៖ ៨ សន្លឹក
- ក្នុងបង្គំតួរនៃឡាន ត្រូវបានប្រើ ២ ពណ៌
 - Pink
 - Blue



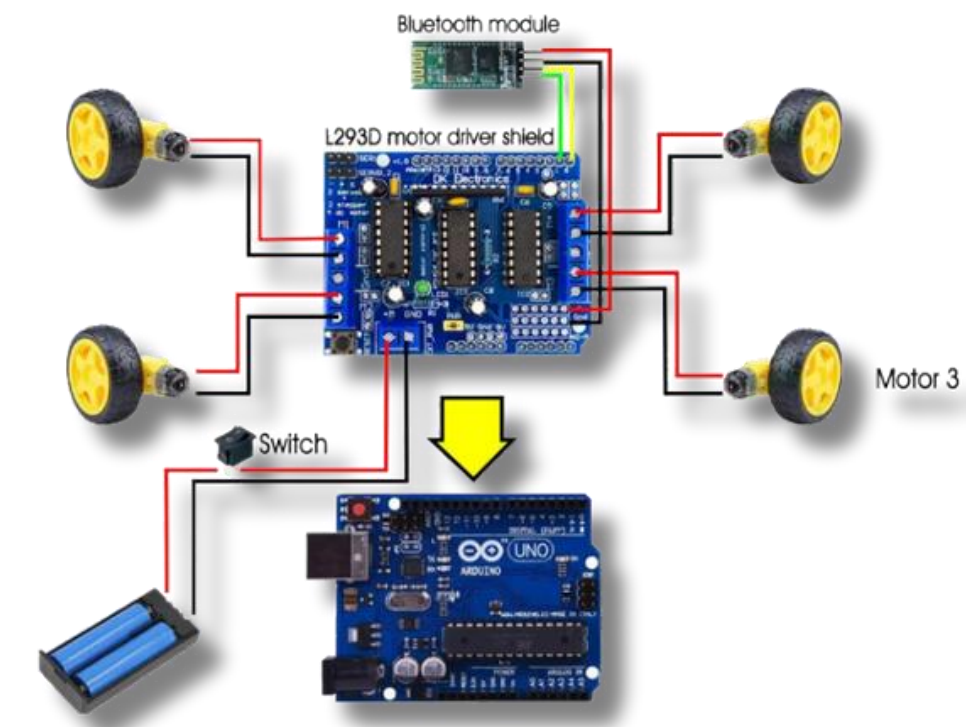
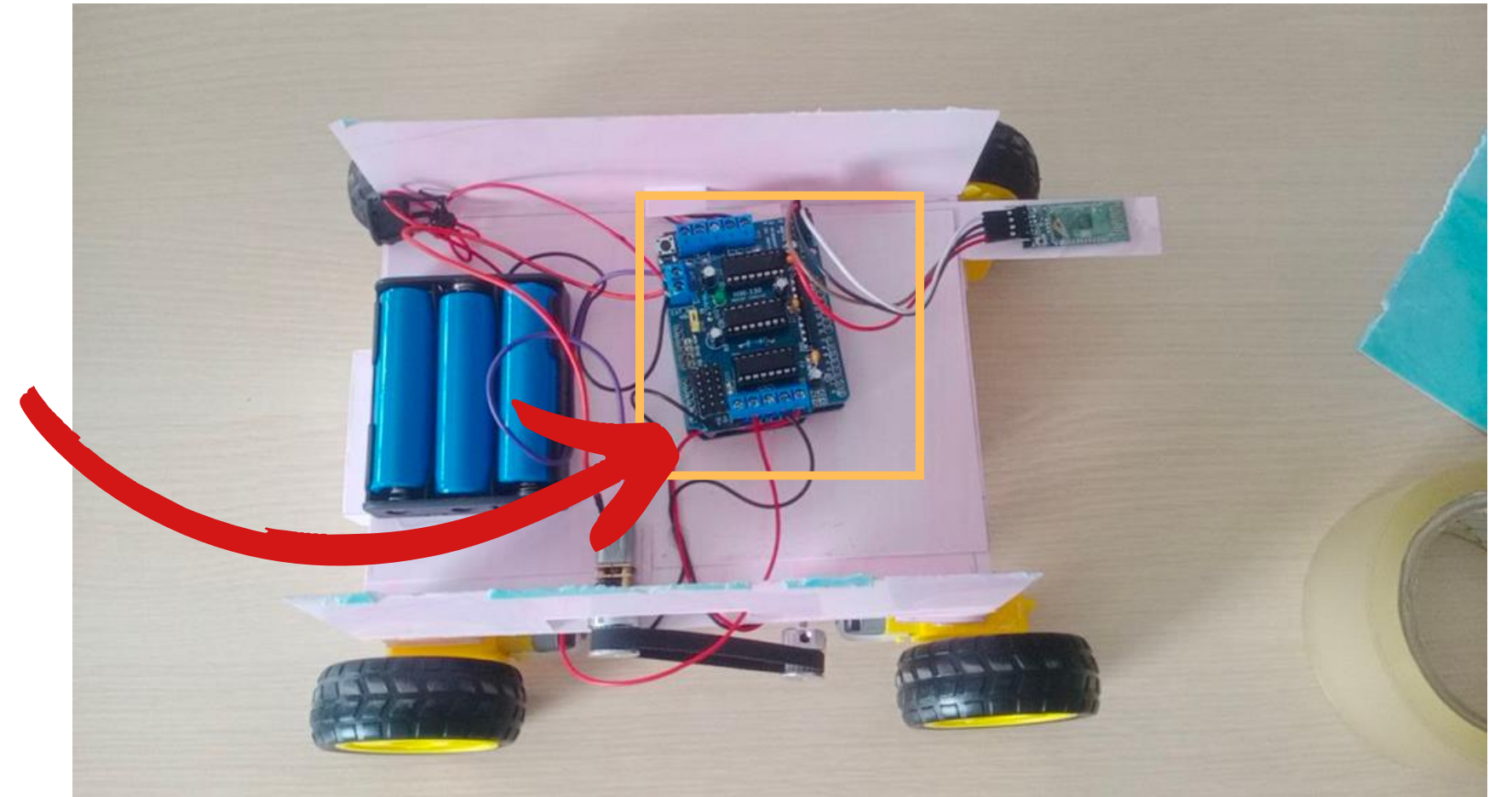
BATTERY

ពួកយើងបានប្រើប្រាស់ថ្នូរ
ដែល១គ្រប់ មានថាមពល 3.7
V ហើយយើងបានប្រើចំនួន៣
គ្រប់ (បង្កើតជាខ្ទង់)



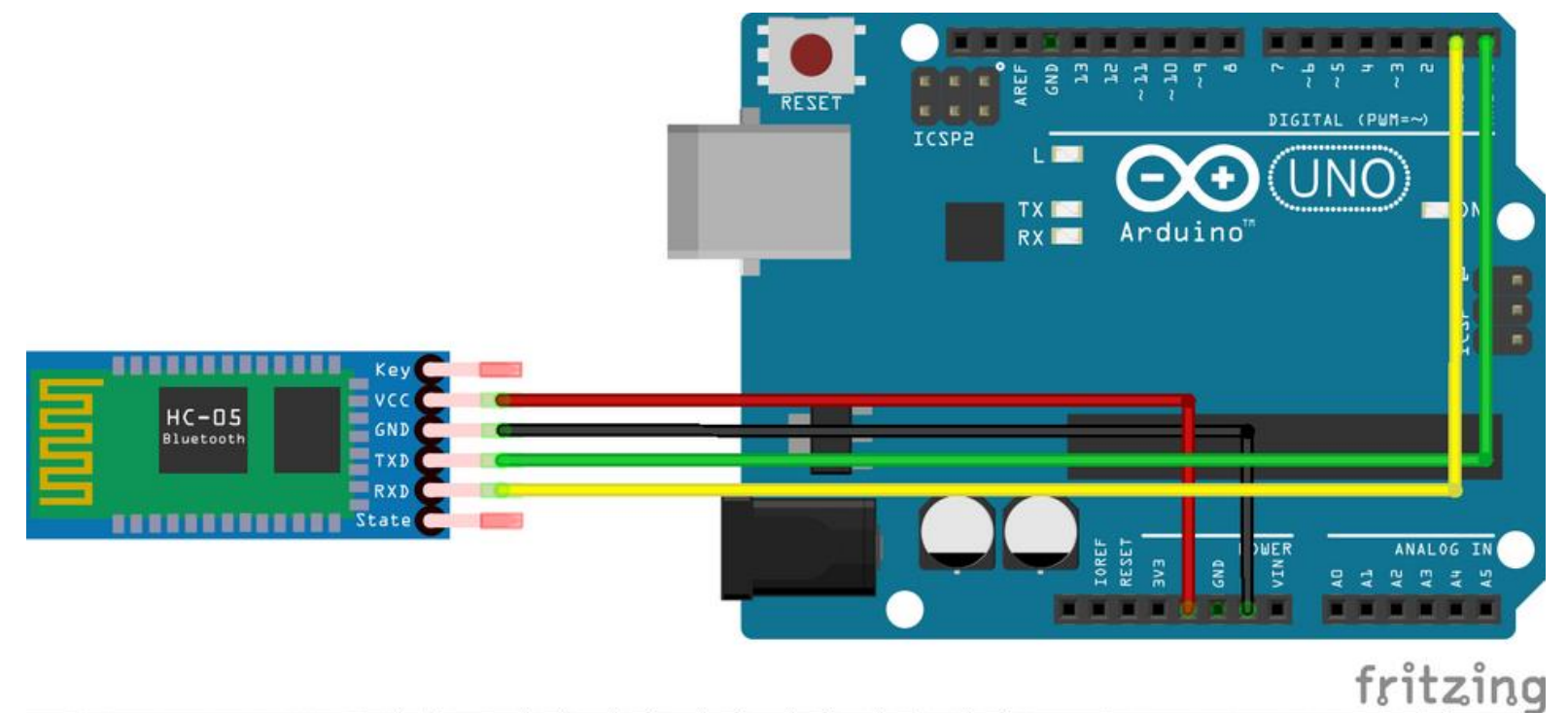
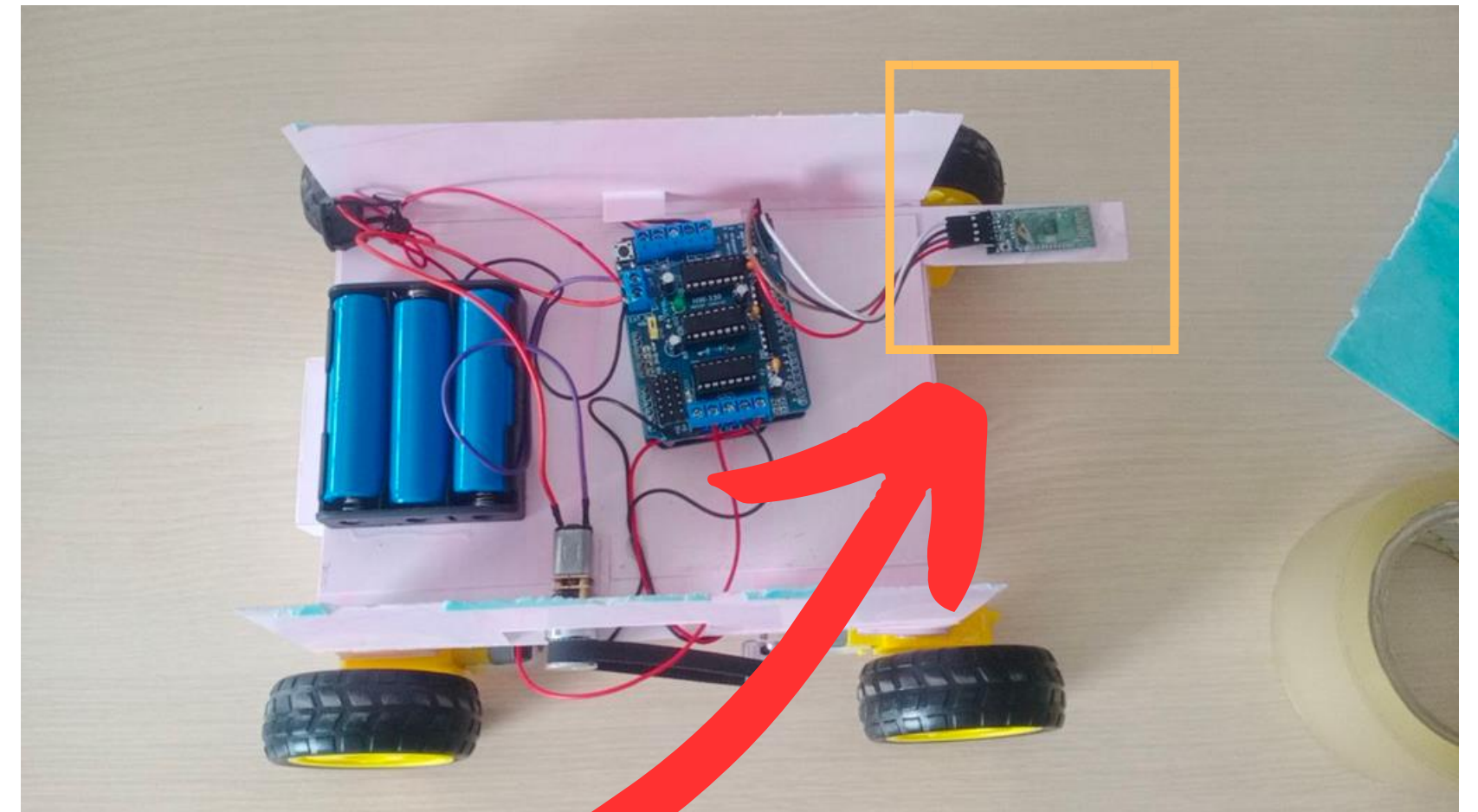
MOTOR DRIVER

ពួកយើងបានប្រើប្រាស់ Motor driver Shield ដើម្បីគ្រប់គ្រងលើការបញ្ជូន Motor ទាំង៤ ជៀសជាង ការគ្រប់គ្រងដោយផ្ទាល់ទៅលើ Arduino តែម្តង !



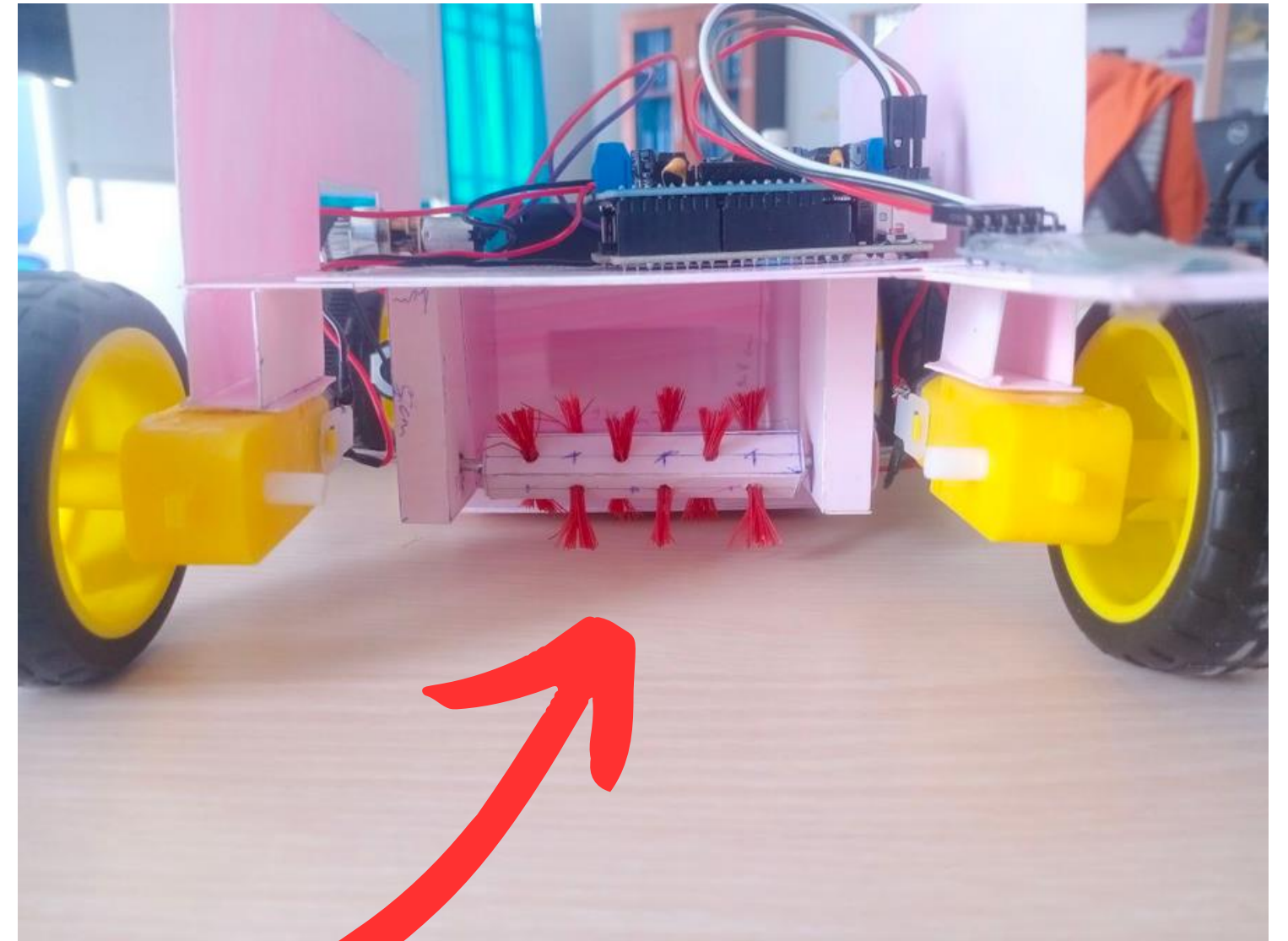
BLUETOOTH MODULE

ក្នុងដំណាក់កាលនេះ យើងបាន
ភ្ជាប់ Bluetooth Module
ទៅនឹង Motor driver shield
ដើម្បីគ្រប់គ្រងពីចម្ងាយតាម
រយៈ Bluetooth



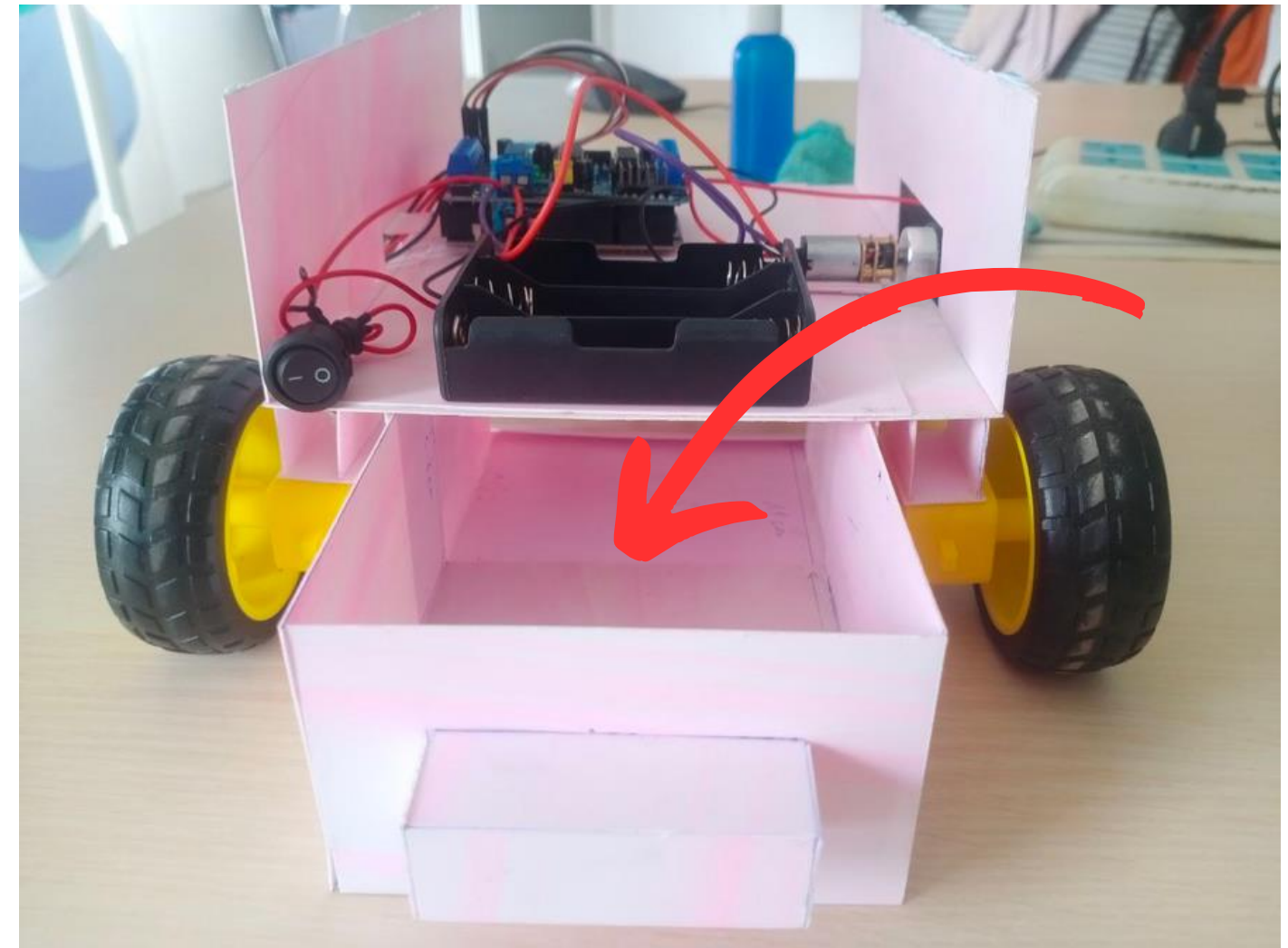
CLEAN SPINNER

ពួកយើងបានបង្កើតវាពីសម្ភារៈ
ដែលមាន ជ័រ កម្រូចអំបោស
ឆ្លឹងអាវធ្វើជាស្ពាន



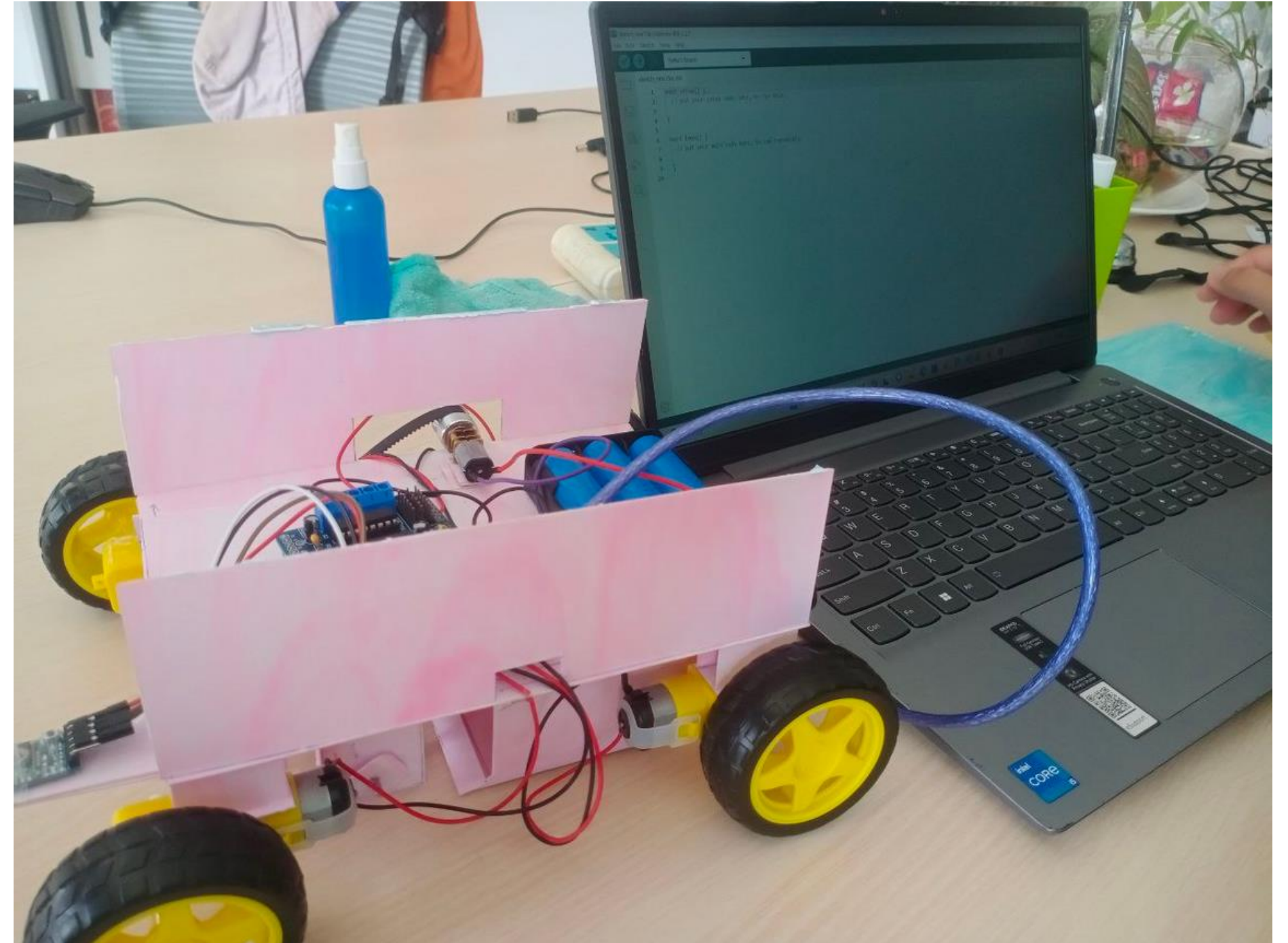
TRASH CONTAINER

ពួកយើងបានបង្កើតវាដោយ
ជ័រ បន្ទុះក្តារ និងប្រើការធ្វើជា
ការបិទភ្ជាប់



CODE UPLOADING

ពួកយើងបានប្រើប្រាស់
AFMotor Library ដើម្បី
ធ្វើការបញ្ជាទៅលើ
Arduino ដឹងManual
Control តាមរយៈ
Bluetooth



< Source Code >

7

< CODE />



```
1  #include <AFMotor.h>
2  #include <SoftwareSerial.h>
3
4  SoftwareSerial bluetoothSerial(9, 10); // RX, TX
5
6  //initial motors pin
7  AF_DCMotor motor1(1, MOTOR12_1KHZ);
8  AF_DCMotor motor2(2, MOTOR12_1KHZ);
9  AF_DCMotor motor3(3, MOTOR34_1KHZ);
10 AF_DCMotor motor4(4, MOTOR34_1KHZ);
11
12 char command;
13
14 void setup()
15 {
16     bluetoothSerial.begin(9600); //Set the baud rate to your Bluetooth module.
17 }
18
19 void loop() {
20     if (bluetoothSerial.available() > 0) {
21         command = bluetoothSerial.read();
22
23         Stop(); //initialize with motors stoped
24
25         switch (command) {
26             case 'F':
27                 forward();
28                 break;
29             case 'B':
30                 back();
31                 break;
32             case 'L':
33                 left();
34                 break;
35             case 'R':
36                 right();
37                 break;
38         }
39     }
40 }
```


7

< CODE />



```
1 void forward()
2 {
3     motor1.setSpeed(255); //Define maximum velocity
4     motor1.run(FORWARD); //rotate the motor clockwise
5     motor2.setSpeed(255); //Define maximum velocity
6     motor2.run(FORWARD); //rotate the motor clockwise
7     motor3.setSpeed(255); //Define maximum velocity
8     motor3.run(FORWARD); //rotate the motor clockwise
9     motor4.setSpeed(255); //Define maximum velocity
10    motor4.run(FORWARD); //rotate the motor clockwise
11 }
12
13 void back()
14 {
15     motor1.setSpeed(255); //Define maximum velocity
16     motor1.run(BACKWARD); //rotate the motor anti-clockwise
17     motor2.setSpeed(255); //Define maximum velocity
18     motor2.run(BACKWARD); //rotate the motor anti-clockwise
19     motor3.setSpeed(255); //Define maximum velocity
20     motor3.run(BACKWARD); //rotate the motor anti-clockwise
21     motor4.setSpeed(255); //Define maximum velocity
22     motor4.run(BACKWARD); //rotate the motor anti-clockwise
23 }
24
25 void left()
26 {
27     motor1.setSpeed(255); //Define maximum velocity
28     motor1.run(BACKWARD); //rotate the motor anti-clockwise
29     motor2.setSpeed(255); //Define maximum velocity
30     motor2.run(BACKWARD); //rotate the motor anti-clockwise
31     motor3.setSpeed(255); //Define maximum velocity
32     motor3.run(FORWARD); //rotate the motor clockwise
33     motor4.setSpeed(255); //Define maximum velocity
34     motor4.run(FORWARD); //rotate the motor clockwise
35 }
```



7

< CODE />



```
1
2 void right()
3 {
4     motor1.setSpeed(255); //Define maximum velocity
5     motor1.run(FORWARD); //rotate the motor clockwise
6     motor2.setSpeed(255); //Define maximum velocity
7     motor2.run(FORWARD); //rotate the motor clockwise
8     motor3.setSpeed(255); //Define maximum velocity
9     motor3.run(BACKWARD); //rotate the motor anti-clockwise
10    motor4.setSpeed(255); //Define maximum velocity
11    motor4.run(BACKWARD); //rotate the motor anti-clockwise
12 }
13
14 void Stop()
15 {
16     motor1.setSpeed(0); //Define minimum velocity
17     motor1.run(RELEASE); //stop the motor when release the button
18     motor2.setSpeed(0); //Define minimum velocity
19     motor2.run(RELEASE); //rotate the motor clockwise
20     motor3.setSpeed(0); //Define minimum velocity
21     motor3.run(RELEASE); //stop the motor when release the button
22     motor4.setSpeed(0); //Define minimum velocity
23     motor4.run(RELEASE); //stop the motor when release the button
24 }
```





bluetooth controller arduino



Arduino Bluetooth Control

broxcode • Tools

✓ Installed



Arduino bluetooth controller

Giumig Apps • Tools

4.2 ★ 7.6 MB ↓ 500K+



Arduino Bluetooth Controller

Ioannis Tzanellis • Education

3.8 ★ 1.4 MB ↓ 100K+



Arduino Bluetooth RC Car

Andi.Co • Education

4.0 ★ 3.9 MB ↓ 500K+



Arduino Bluetooth Controller – All in One

MyValley® • Tools

4.2 ★ 5.0 MB ↓ 50K+



បញ្ហាដែលយើងបានដោះស្រាយ និង បញ្ហាគួរតែប្រុងប្រយ័ត្ន

CODE PROCESS

- Forgot Bluetooth Code
- First Experience with the code language
- use other library and coding the wheel for the wrong turn
- separate board before uploading the code
- Always research before doing on our own



BUILD UP PROCESS

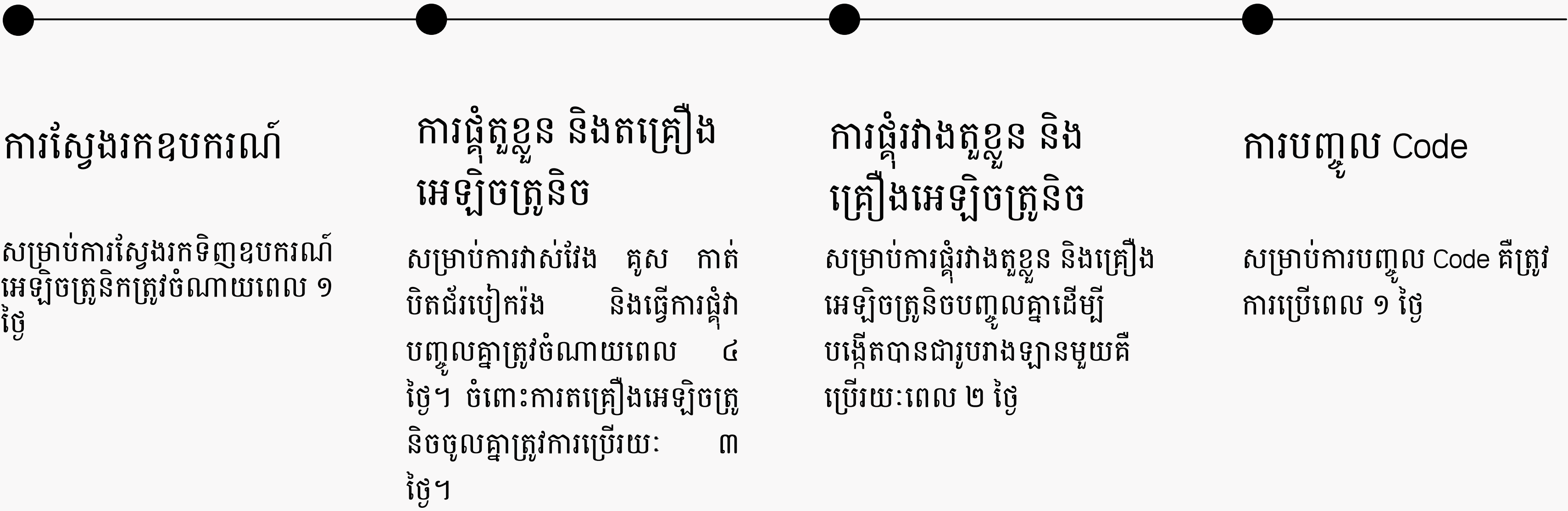
- Waste Time For Plastic Printer
- Less Tools and Materials
- New Experience with Board
- Do before research lead us to almost broke a board
- Right Measurement of the Car Board (Plastic)



TEAM PHOTOGRAPH

TESTING

Timeline - Midterm

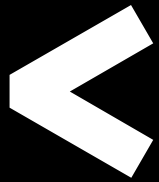


FINAL PROJECT

UPDATE

7

< CODE />



```
#include <AFMotor.h>
#include <SoftwareSerial.h>

SoftwareSerial bluetoothSerial(9, 10); // RX, TX

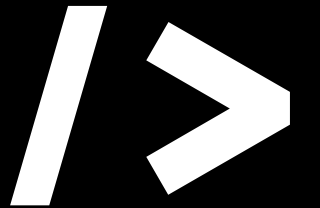
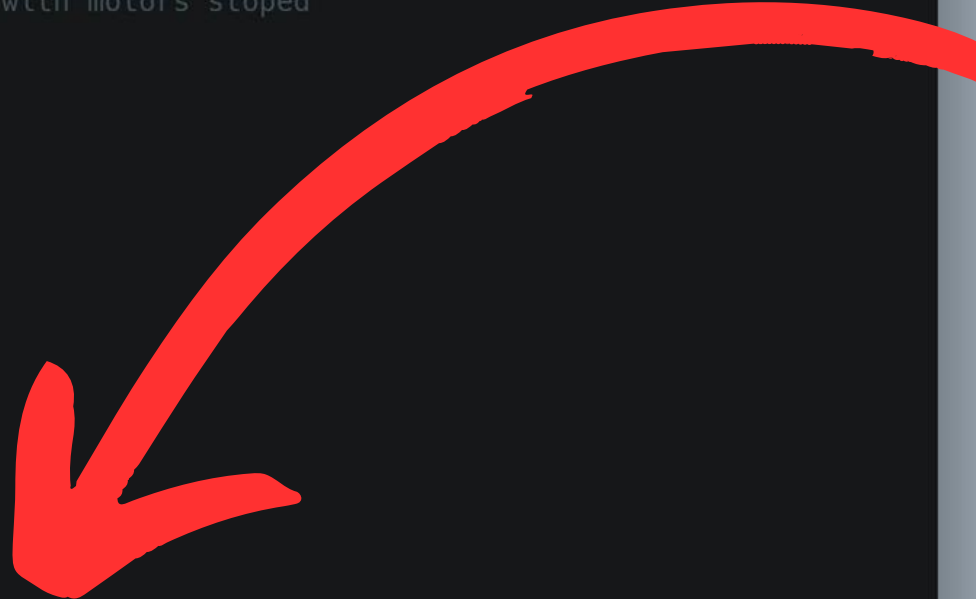
//initial motors pin
AF_DCMotor motor1(1, MOTOR12_1KHZ);
AF_DCMotor motor2(2, MOTOR12_1KHZ);
AF_DCMotor motor3(3, MOTOR34_1KHZ);
AF_DCMotor motor4(4, MOTOR34_1KHZ);

char command;
const int relayPin = A0;
void setup()
{
  bluetoothSerial.begin(9600); //Set the baud rate to your Bluetooth module.
}

void loop() {
  if (bluetoothSerial.available() > 0) {
    command = bluetoothSerial.read();

    Stop(); //initialize with motors stoped

    switch (command) {
      case 'F':
        forward();
        break;
      case 'B':
        back();
        break;
      case 'L':
        left();
        break;
      case 'R':
        right();
        break;
    }
    if (command == 'w'){
      digitalWrite(relayPin, HIGH);
    }else if(command == 'W'){
      pinMode(relayPin, OUTPUT);
      digitalWrite(relayPin, LOW);
    }
  }
}
```

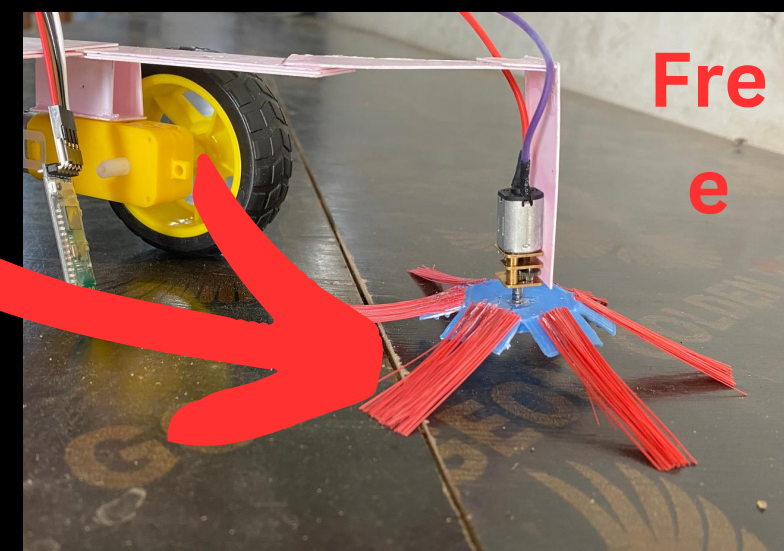




Water Battle Cap

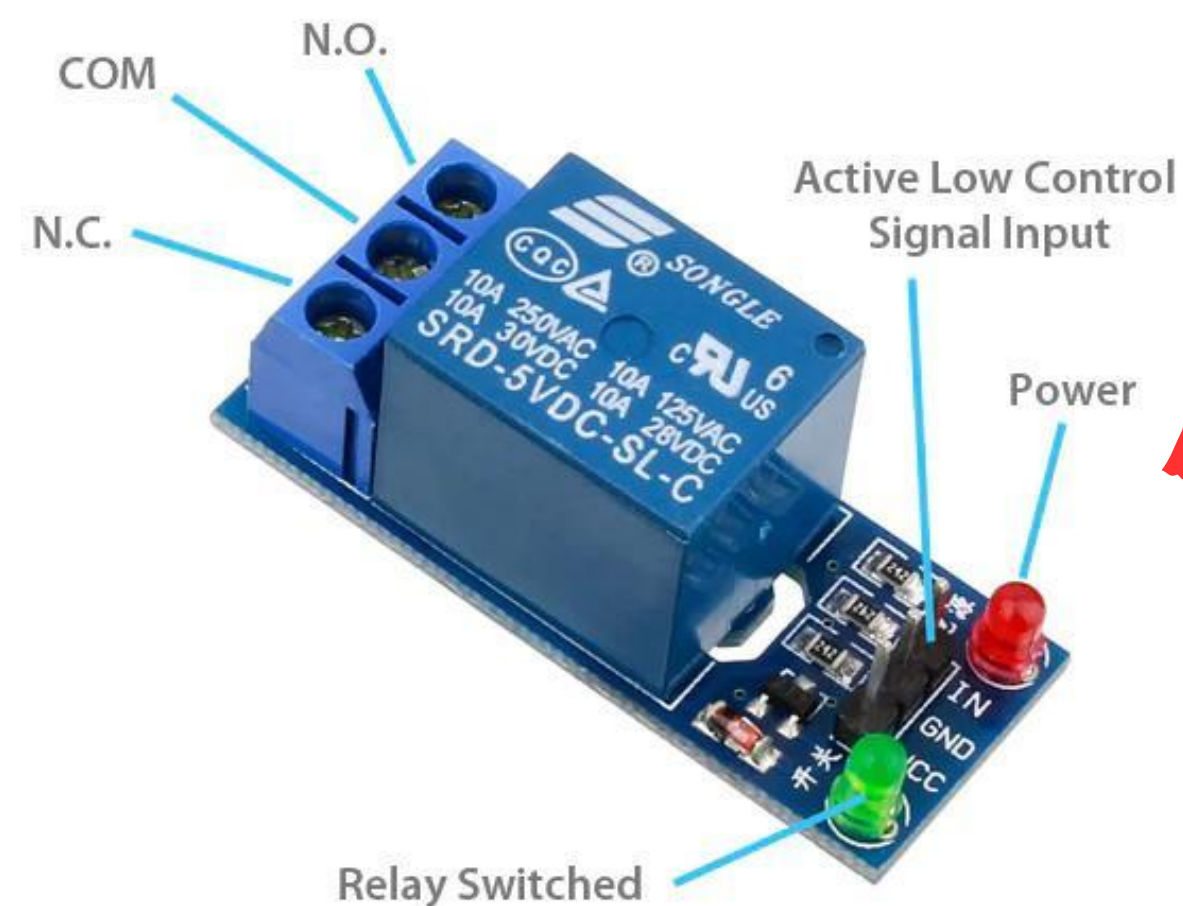
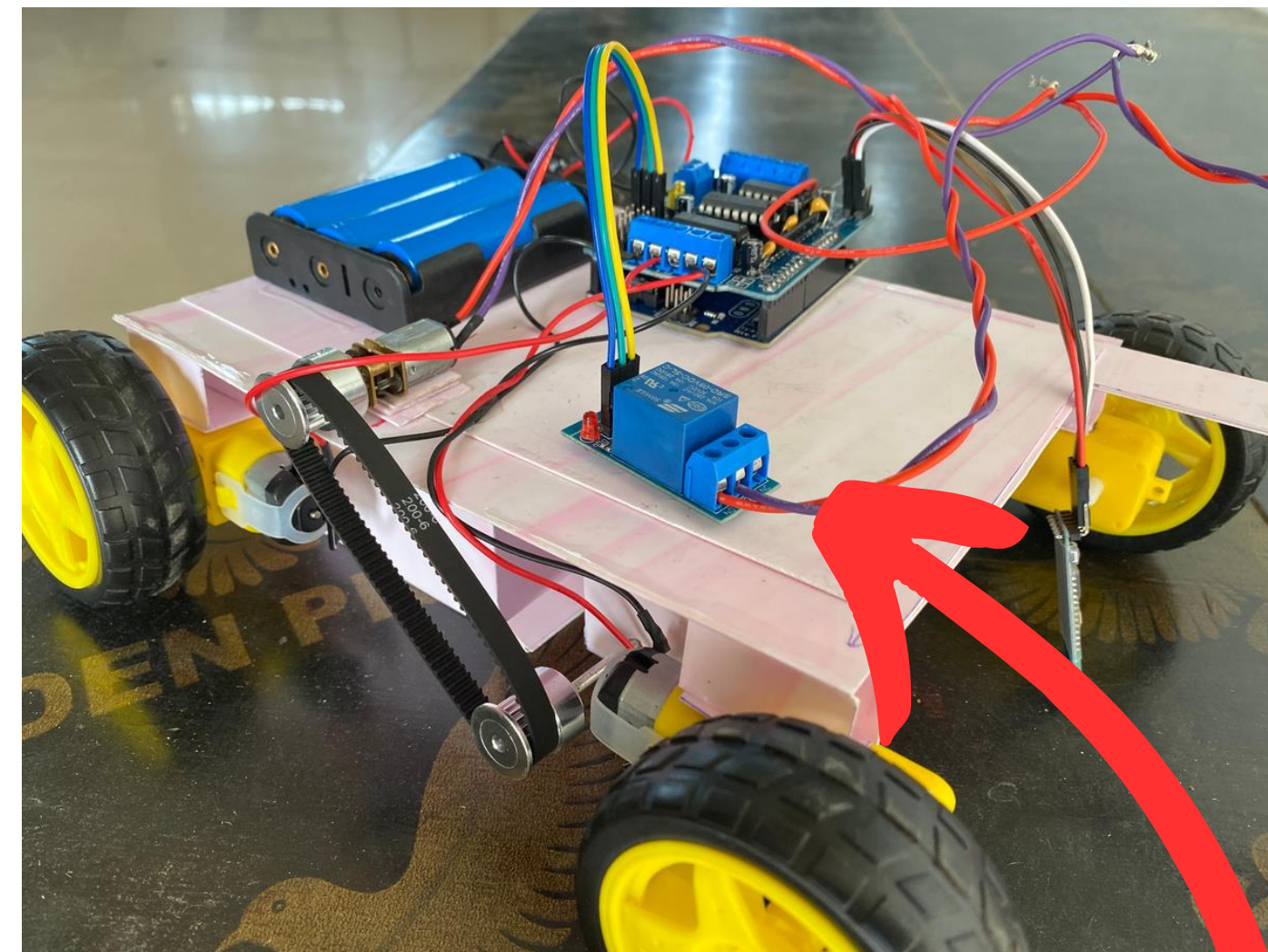


Planetary Gear Brushed Motor
N60



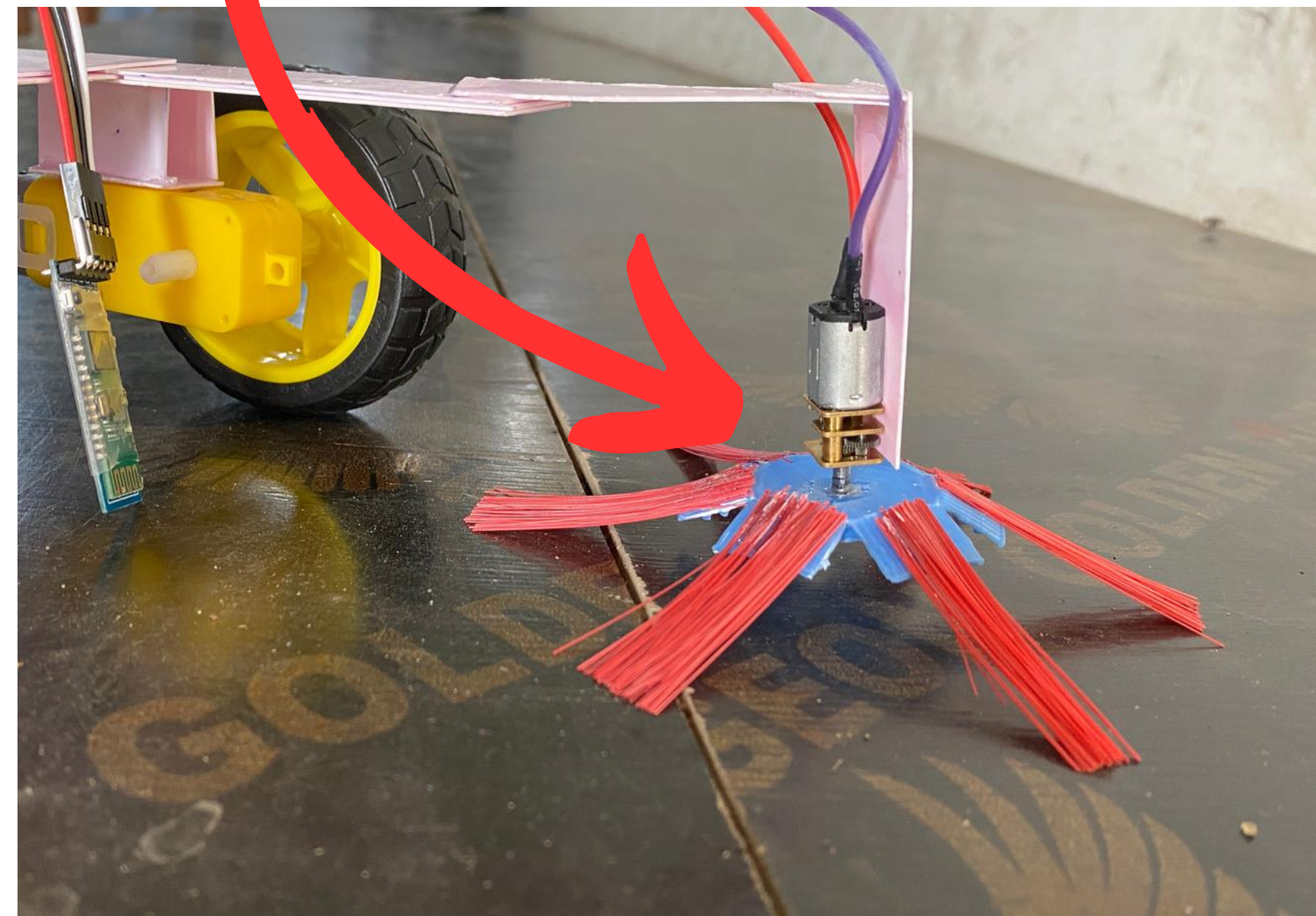
RELAY MODULE

យើងប្រើប្រាស់ដើម្បីគ្រប់គ្រងអគ្គិសនី
ដែលបានបញ្ជូនទៅកាន់ឧបករណ៍
សម្ងាត់សំឡេងដូចជាសំឡេងកាប្រើប្រាស់
ឡើងអគ្គិសនីច្រើន



FRONT CLEAN SPINNER

យើងប្រើប្រាស់ដើម្បីគាត់សំរាមដែល
នៅតាមកន្ទុកកន្ទាត់មកចំកណ្តាល
ដែលផ្តល់ភាពងាយស្រួលដល់ការ
សម្អាតចោទកាន់តែស្អាតជាងមុន



Timeline - Final



ការផ្ដុំតួខ្លួន និងត
គ្រឿងអេឡិចត្រូនិច

សម្រាប់ការវាស់វែង គូស កាត់
បិទជំរបៀករង និងធ្វើការផ្គុំ
បញ្ចូលគ្នាជាមួយគ្រឿងអេឡិចត្រូ
និចដែលបានត្រូវការ
ប្រើរយៈ ១ ថ្ងៃ។



ការបញ្ចូល Code

សម្រាប់ការបញ្ចូល Code គឺ
ត្រូវការប្រើពេល ១ ថ្ងៃ

សរុបសេចក្តី

ការងាររបស់យើងមានសកម្មភាពច្រើន នៅក្នុងនោះយើងបានចំណាយពេលធ្វើទៅលើ ការស្វែងយល់អំពី Electronic Components ច្រើនជាងគេ។ បើមើលទៅលើដំណើរការវិញ វាមិនមានបញ្ហាអ្វីគួរឲ្យចោតសួរនោះទេ តែចំពោះ Performance វិញ វានៅមានកម្រិត។ ម្យ៉ាងទៀត អ្វីដែលគ្រប់គ្នាអាចជៀសវាងបានអំឡុងពេលធ្វើ គឺអ្នកទាំងអស់គ្នាគួរតែជៀស វាងការប្រើរបស់ដែលមិនមានការវាស់វែងច្បាស់លាស់ ត្បិតតែវាអស់ការចំណាយថវិការតិច តែ Performance ក៏វាមិនល្អទៅតាមនោះដែរ។

យ៉ាងណាមិញ ការងារមួយនេះបានបង្រៀនយើងជាច្រើន ដូចជាការស្រាវជ្រាវ ការ សហការគ្នា និងជាពិសេសគឺចេះពិភាក្សាគ្នា សាមគ្គីគ្នាដើម្បីសម្រេចនូវគោលដៅរួមមួយ។

< Thank You! >