РОБОТИЗИРОВАННАЯ ПЛАТФОРМА

На колесах Бенгта Илона

Выполнил Павловский Иван,
ученик 10 класса МАОУ “Лицей №38”
Руководители:
Еделев А.Ю.
Еделев Ю.А.

Актуальность

- Ограниченные пространства для маневров на складских помещениях замедляют транспортировку
- Высокоманевренные платформы стоят значительно дороже стандартных устройств
- Ускорение транспортировки делает её значительно дороже

Цель

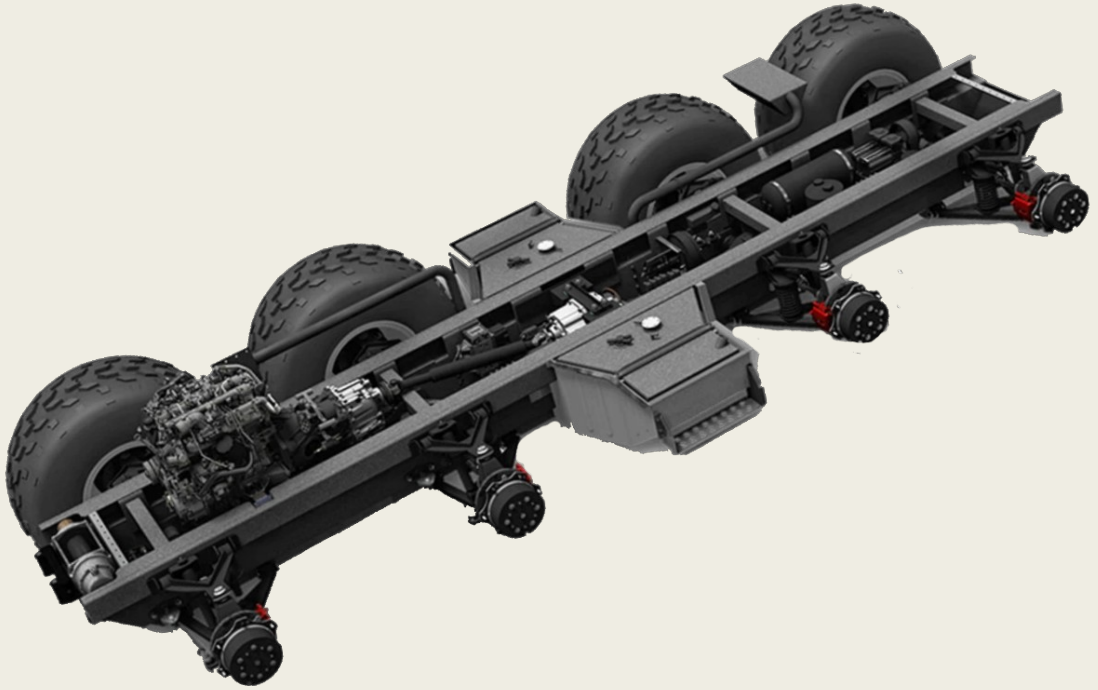
Создать высокоманевренное роботизированное устройство для транспортировки грузов.

Задачи

- Исследовать аналоги стандартным складским транспортировочным устройствам
- Спроектировать и создать собственное устройство
- Провести опыты и тестирования

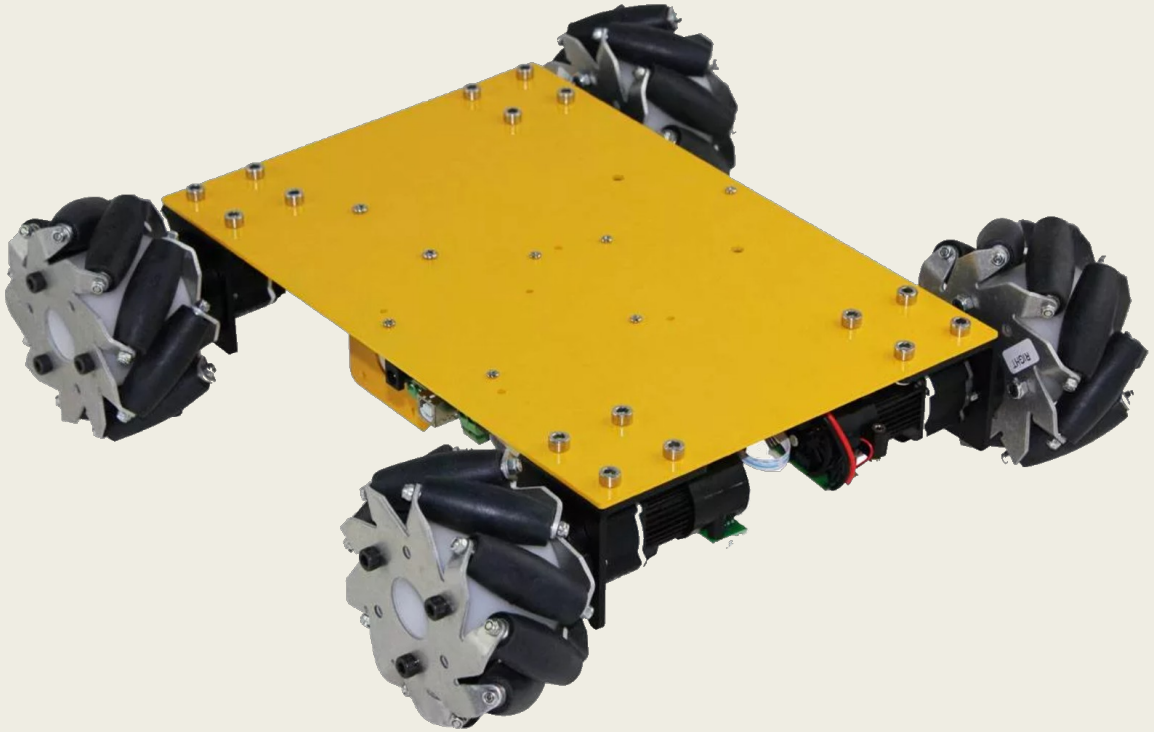
АНАЛОГИ СКЛАДСКИМ ПОГРУЗЧИКАМ

Гусеничная тележка



Тележка с четырьмя независимыми приводами

Тележка с двумя осями вращения колеса

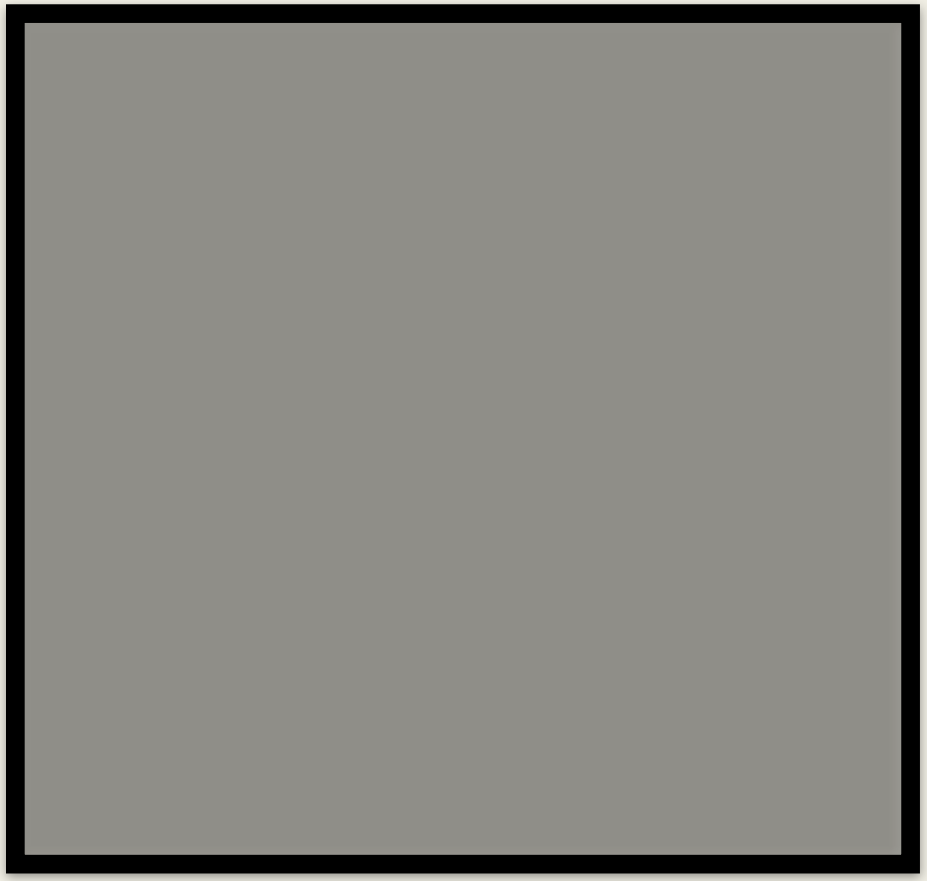


Тележка на колесах Илона

КОЛЕСО ИЛОНА

Колесо Илона

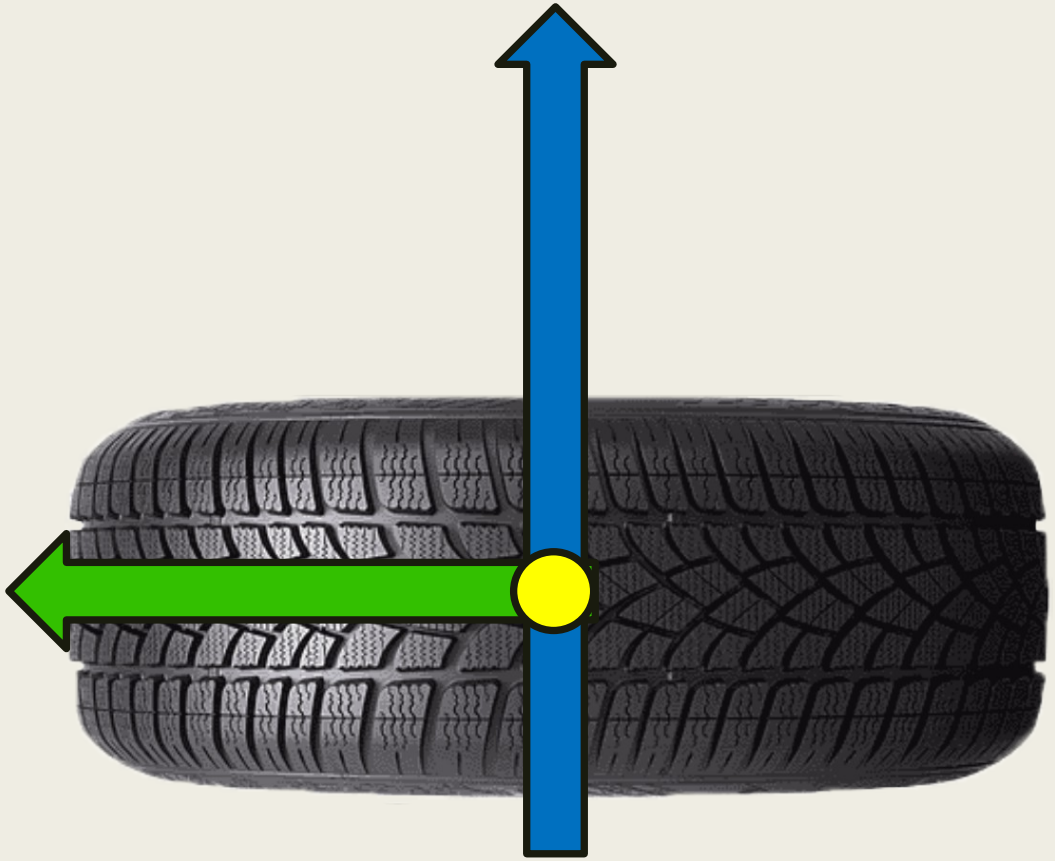
Колесо

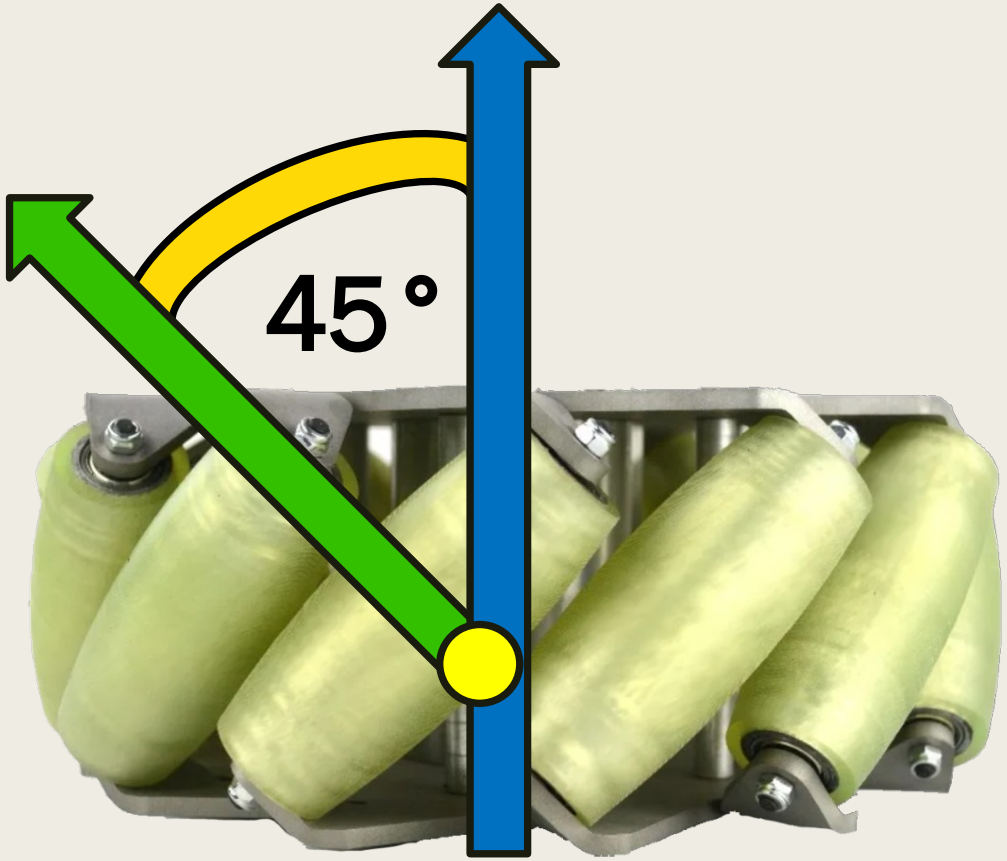


Движение платформы на колесах Илона

Ось Вращения

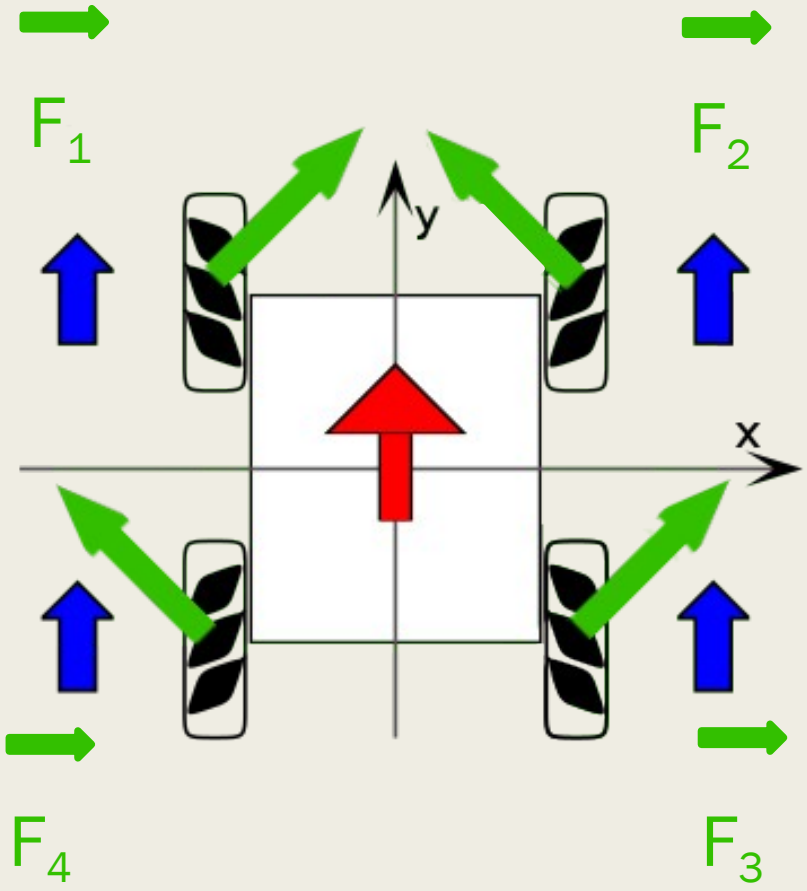
Сила Трения Качения

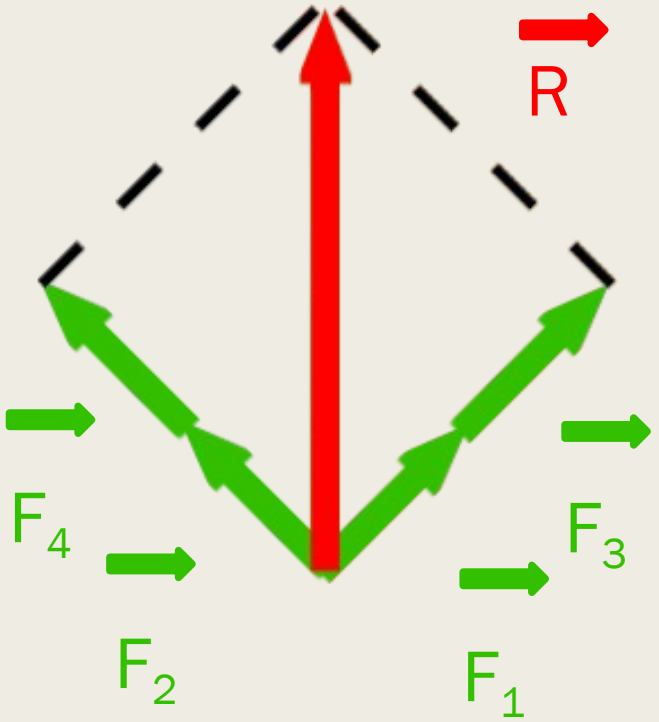




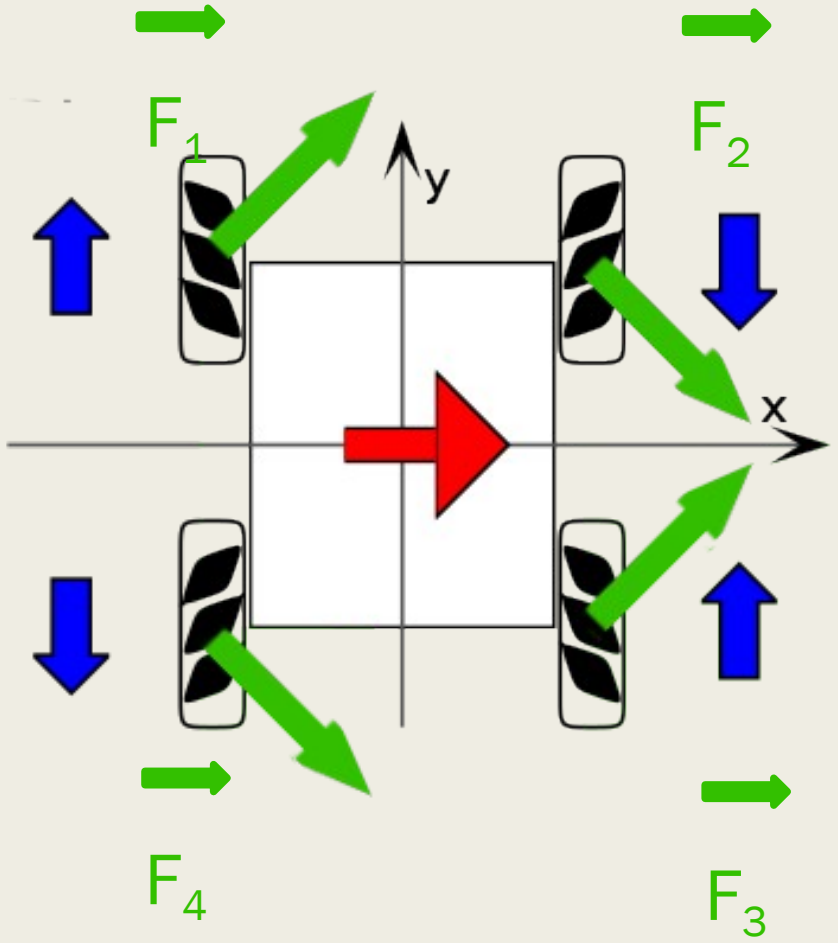
ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ ДВИЖЕНИЯ

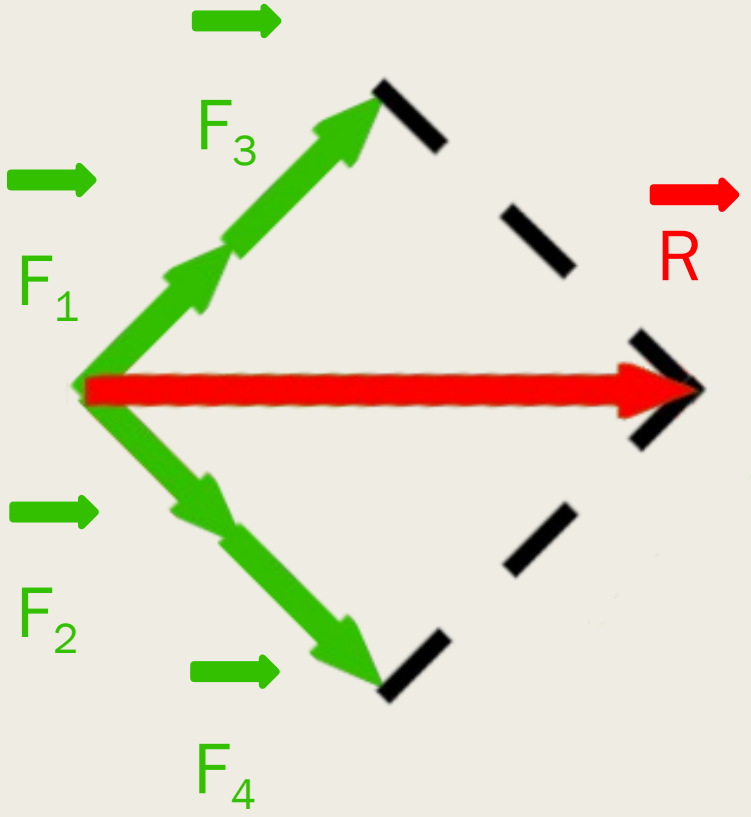
Движение вперед



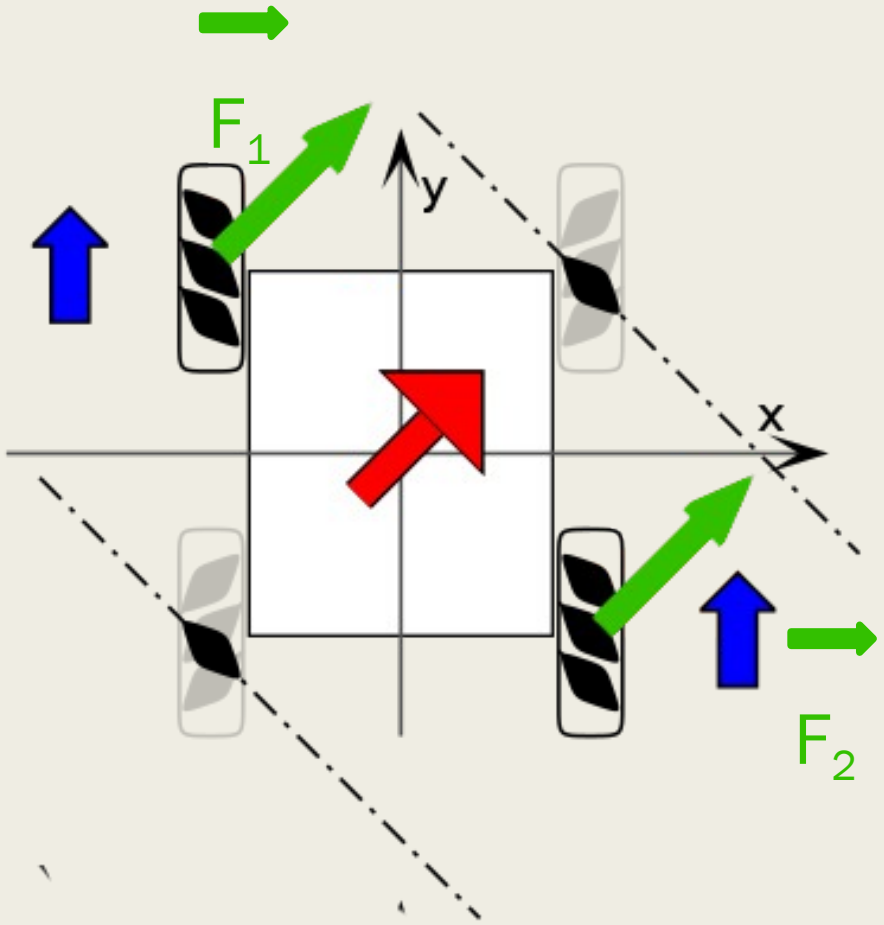


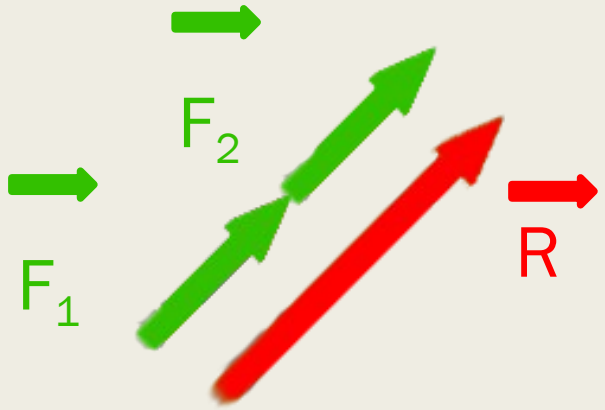
Движение вправо



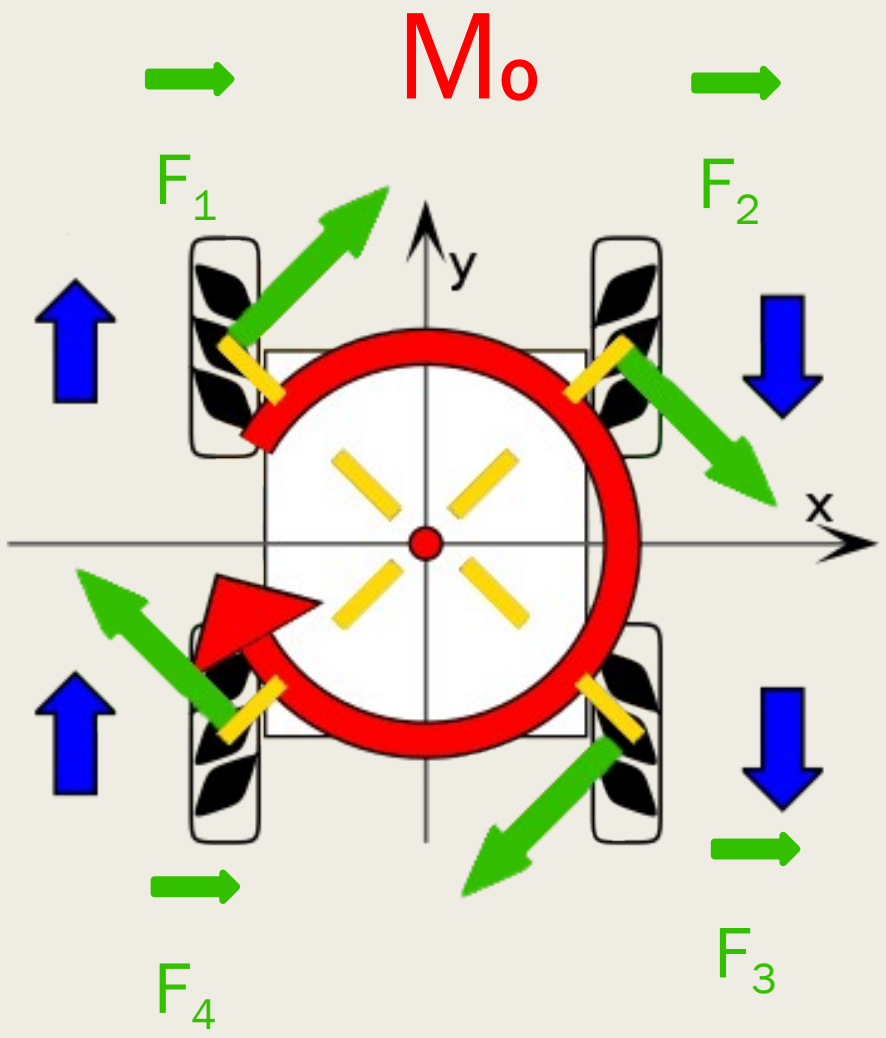


Движение по диагонали

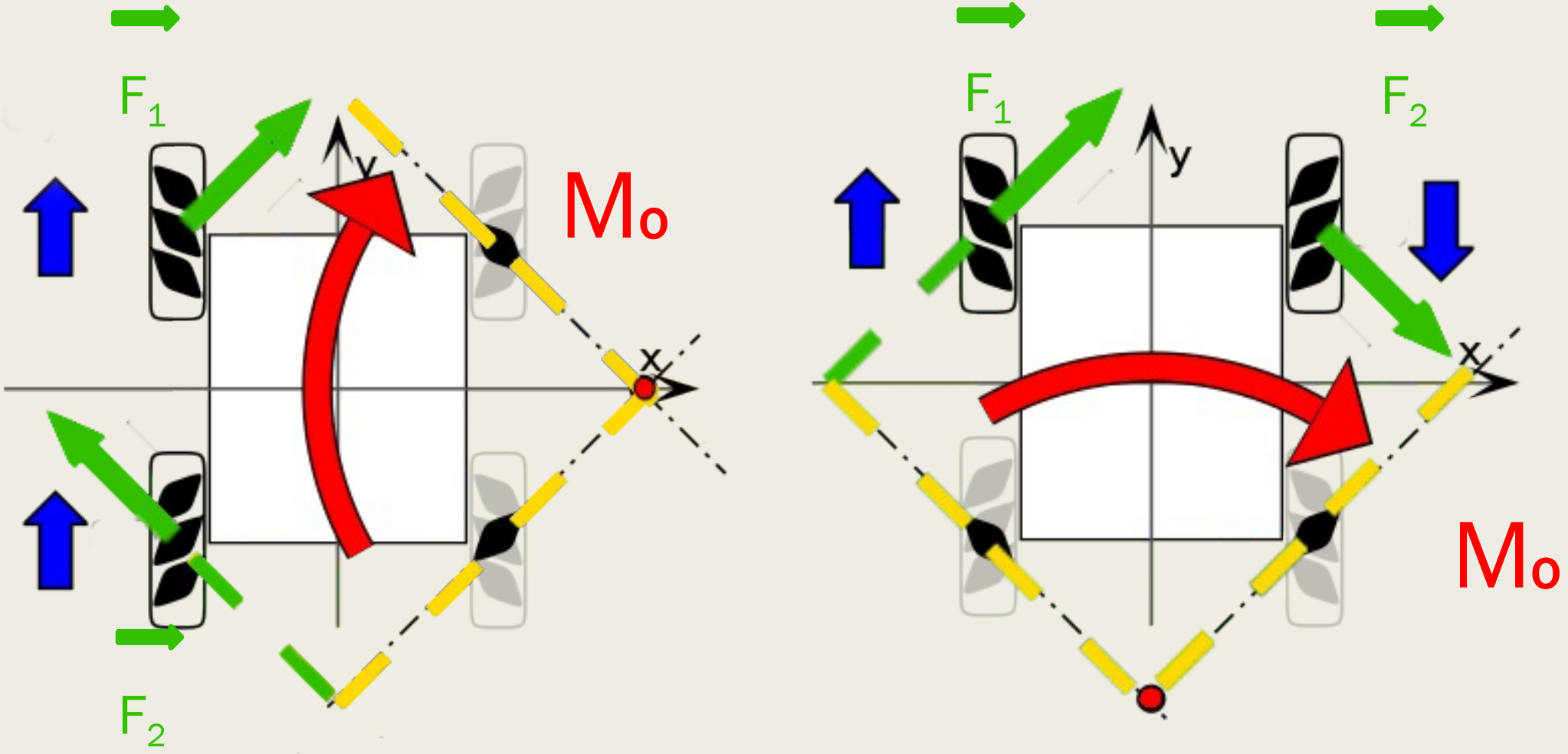




Вращение относительно центра

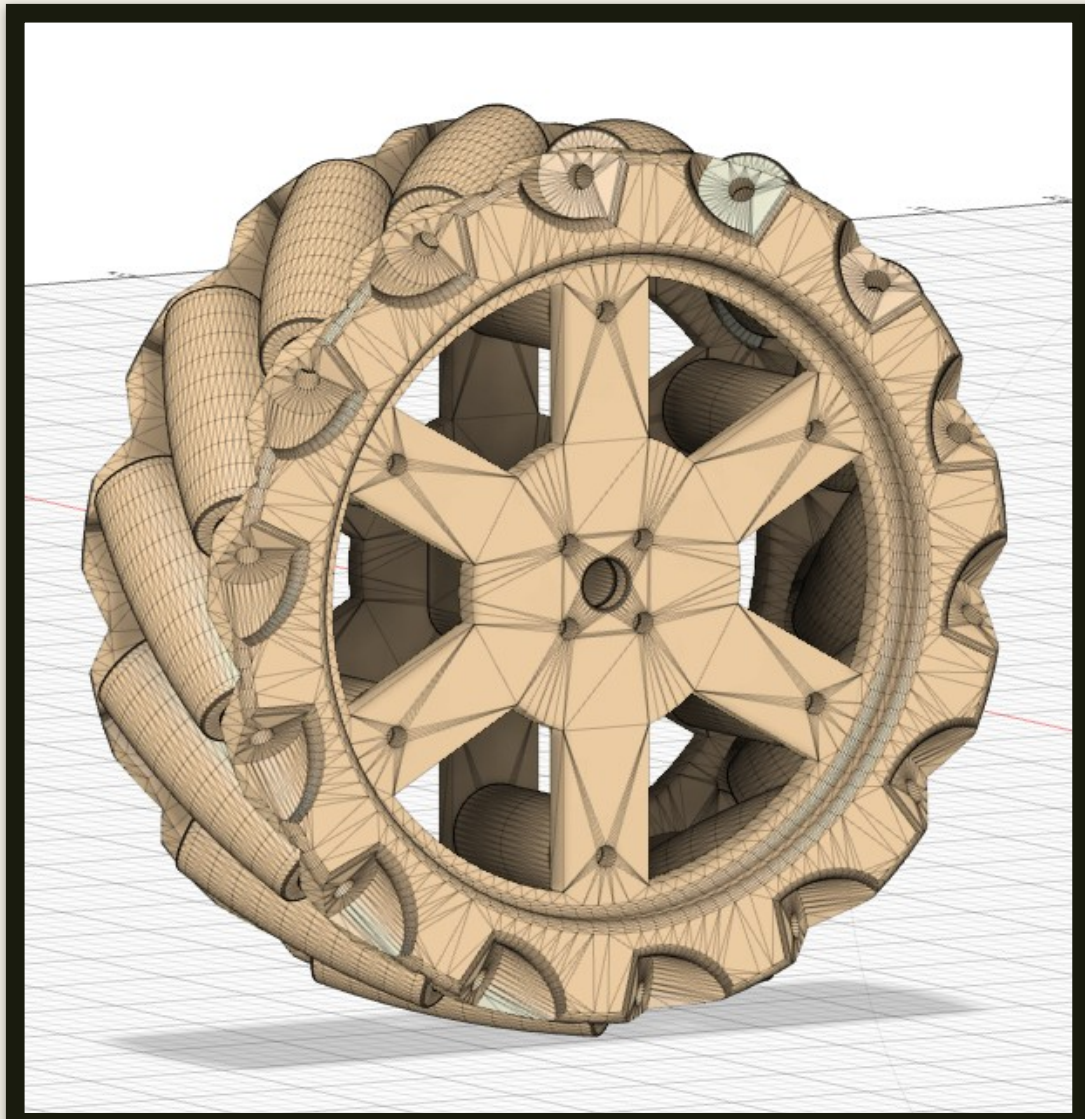


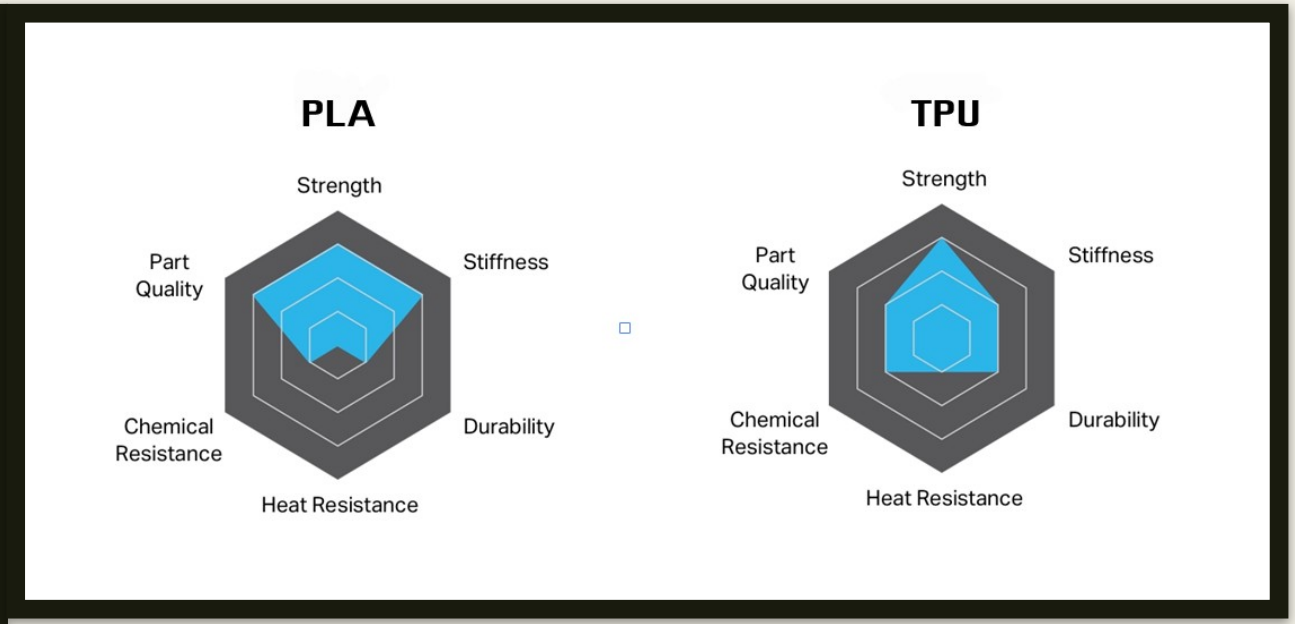
Вращение относительно точки вне платформы



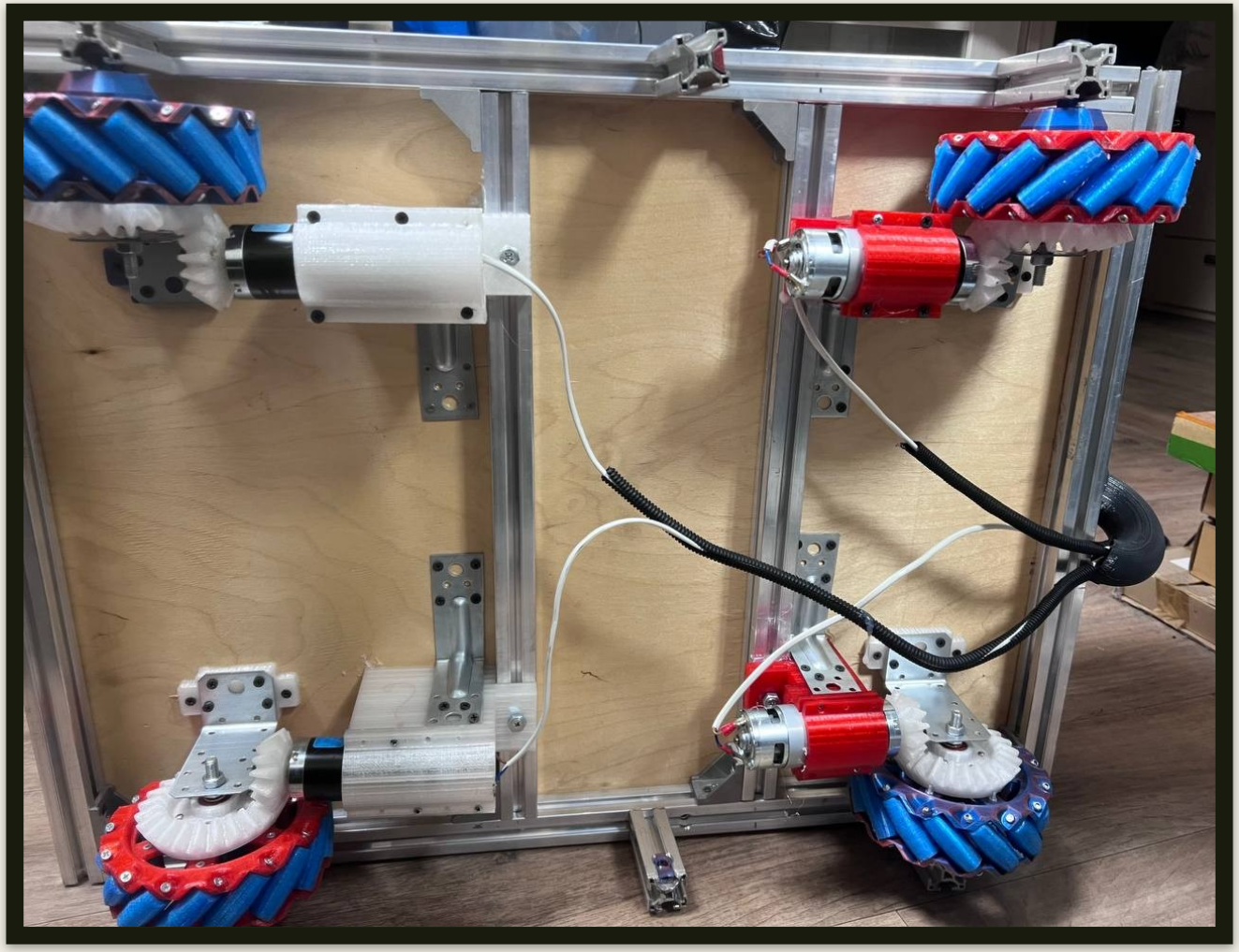
КОНСТРУИРОВАНИЕ ПЛАТФОРМЫ

Моделирование колеса

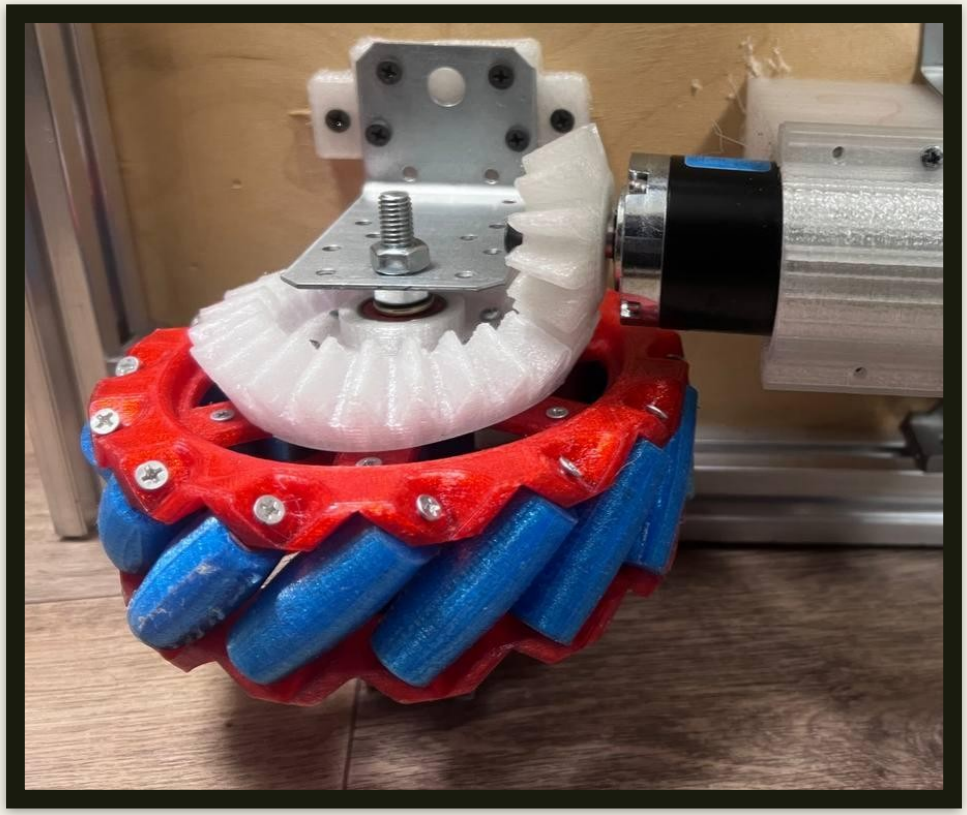




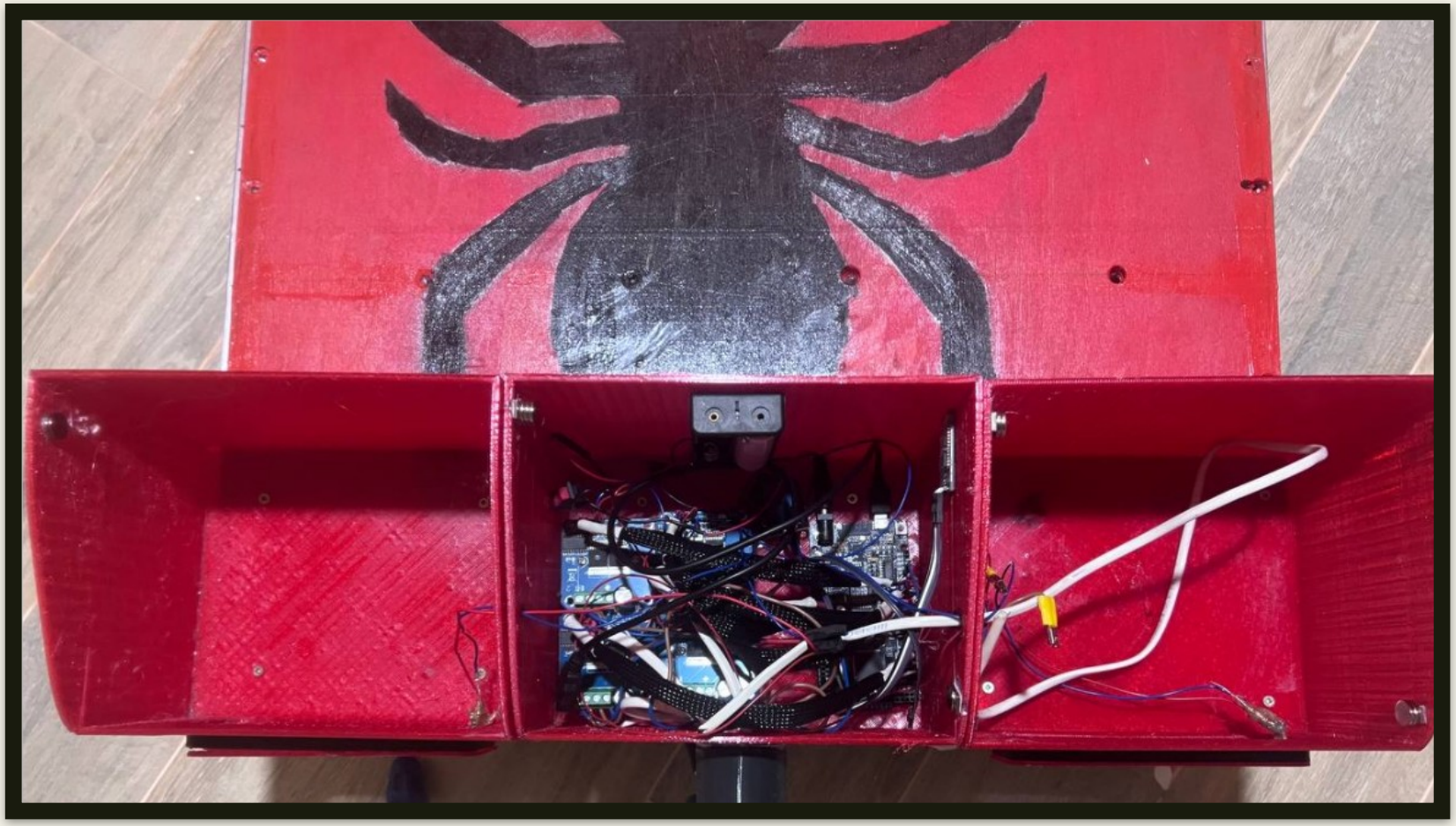
Конструирование платформы



Двигатели и передача

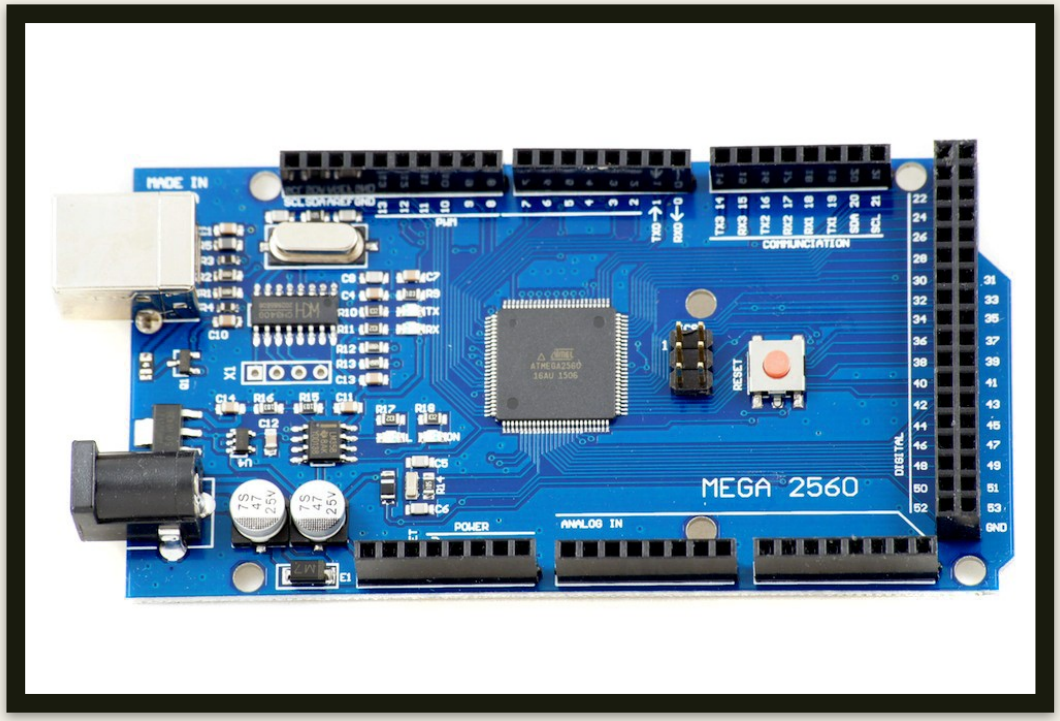


Отсеки для электроники



ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ

Плата Arduino MEGA



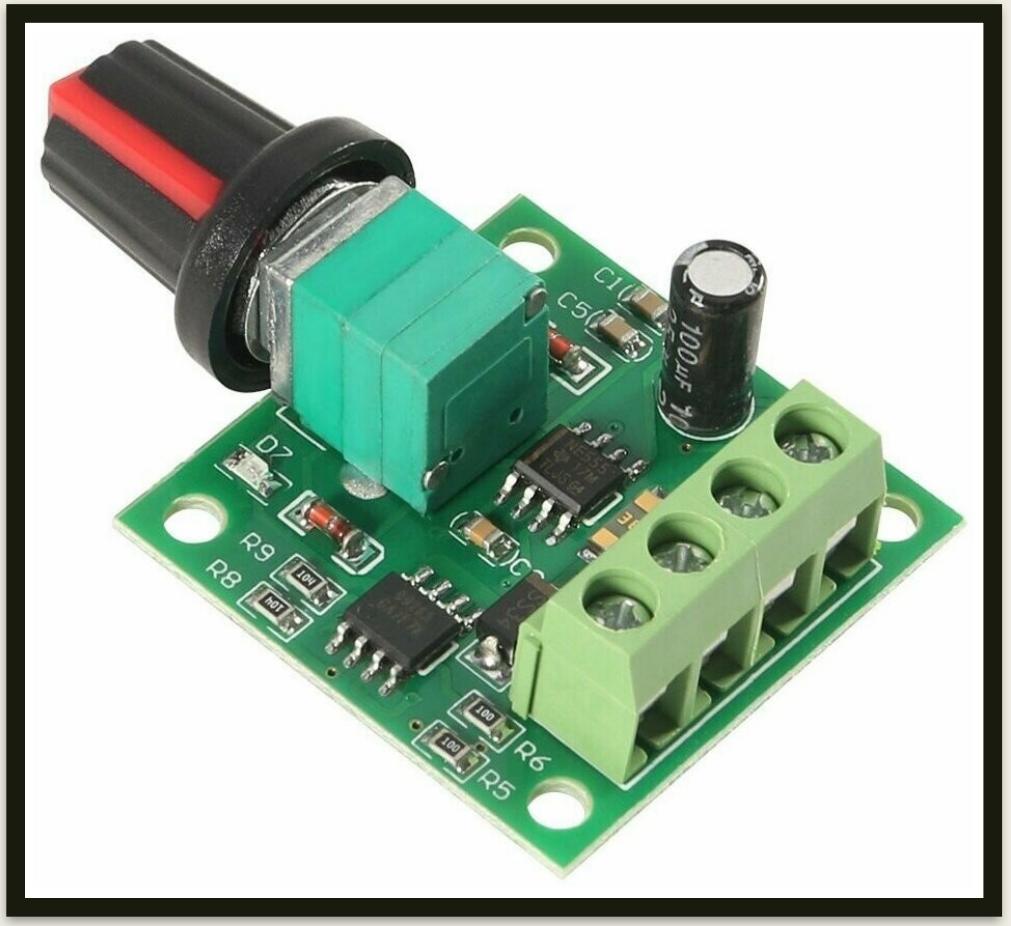
Рабочее напряжение: 5В

Цифровые входы: 54

Аналоговые входы: 16

Входы с поддержкой ШИМ: 14

ШИМ-контроллеры



Номинальное напряжение: 15В

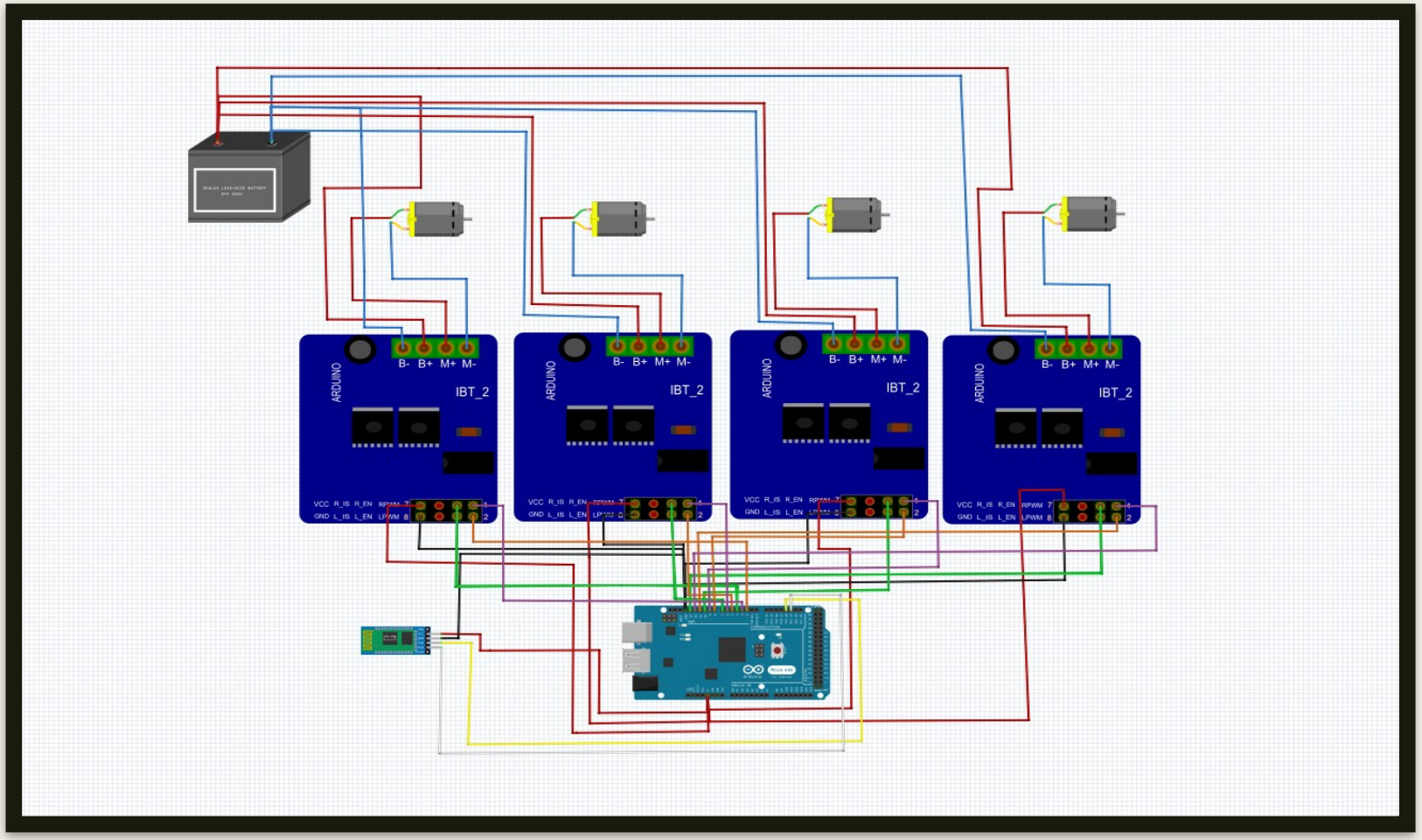
Номинальный ток: 2А

Аккумулятор

Напряжение: 12В

Сила тока: 6А

Принципиальная схема



ПРОГРАММИРОВАНИЕ

Движение вперед/назад

```
////////////////// Движение вперед //////////////////////////////////////////////////////
void forward (){
 ////////// Мотор Front Right/////////////
    digitalWrite(L_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN1,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

 ////////////// Мотор Rear Right //////////////
    digitalWrite(L_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN2,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

 ////////// Мотор Rear Left/////////////
    digitalWrite(L_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN3,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

 ////////// Мотор Front Left/////////////
    digitalWrite(L_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN4,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);


}
```

```
////////////////// Движение назад //////////////////////////////////////////////////////
void back () {
 ////////// Мотор Front Right/////////////
    digitalWrite(L_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
    digitalWrite(R_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN1,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

    digitalWrite(L_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
    digitalWrite(R_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN2,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

 ////////// Мотор Rear Left/////////////
    digitalWrite(L_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
    digitalWrite(R_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN3,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

 ////////// Мотор Front Left/////////////
    digitalWrite(L_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
    digitalWrite(R_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN4,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

}
```

Движение по диагонали

```
////////////////////////////// Движение по диагонали назад вправо//////////////////////
void backright (){
 ////////// Мотор Front Left//////////////
    digitalWrite(L_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN4,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

    ////////////// Мотор Rear Right //////////////
    digitalWrite(L_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN2,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);


}
////////////////////////// Движение по диагонали назад влево//////////////////////////////
void backleft () {
 ////////// Мотор Front Right//////////////
    digitalWrite(L_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
    digitalWrite(R_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN1,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

 ////////// Мотор Rear Left//////////////
    digitalWrite(L_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM-
    digitalWrite(R_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
    analogWrite (EN3,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

}
```

```
////////////////// Движение по диагонали вперед влево //////////////////////
void forwardleft (){
////////// Мотор Front Left//////////////
    digitalWrite(L_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN4,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

////////////// Мотор Rear Right //////////////
    digitalWrite(L_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN2,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

}
////////////////// Движение по диагонали вперед вправо ////////////////////////
void forwardright(){
 ////////// Мотор Front Right//////////////
    digitalWrite(L_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN1,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

 ////////// Мотор Rear Left//////////////
    digitalWrite(L_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
    digitalWrite(R_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
    digitalWrite(EN3,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

}
```

Движение вправо/влево

```
////////////////// Движение вправо //////////
void right(){
  ////////// Мотор Front Left//////////////
   digitalWrite(L_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN4,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

   ////////// Мотор Rear Left//////////////
   digitalWrite(L_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN3,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

    ////////// Мотор Front Right//////////////
   digitalWrite(L_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN1,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

   ////////// Мотор Rear Right //////////////
   digitalWrite(L_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN2,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

}
```

```
////////////////////// Движение влево ////////////////
void left(){
  ////////// Мотор Front Left//////////////
   digitalWrite(L_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN4,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

   ////////// Мотор Rear Left//////////////
   digitalWrite(L_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN3,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

   ////////// Мотор Front Right//////////////
   digitalWrite(L_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN1,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

   ////////// Мотор Rear Right //////////////
   digitalWrite(L_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN2,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

}
```

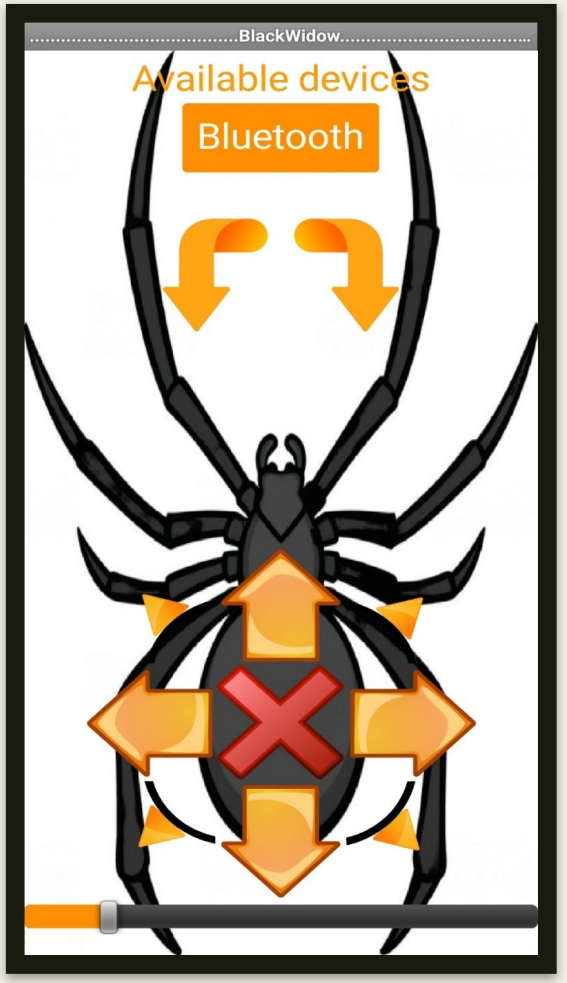
Вращение

```
////////////////// Вращение вправо //////////////
void superright(){
  ////////// Мотор Rear Left//////////////
   digitalWrite(L_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN3,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

  ////////// Мотор Front Left//////////////
   digitalWrite(L_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN4,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

   ////////// Мотор Front Right//////////////
   digitalWrite(L_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN1,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

////////////// Мотор Rear Right //////////////
   digitalWrite(L_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN2,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);
```

```
//////////////////////// Вращение влево ////////////////////////////////////
void superleft(){
  ////////// Мотор Rear Left//////////////
   digitalWrite(L_PWM3, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM3, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN3,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

////////// Мотор Front Left//////////////
   digitalWrite(L_PWM4, HIGH);  // Устанавливаем логическую 1 на входе драйвера L_PWM
   digitalWrite(R_PWM4, LOW );  // Устанавливаем логический 0 на входе драйвера R_PWM
   digitalWrite(EN4,    HIGH);  // Эта функция выполнит те же действия что и функция analogWrite(EN, 255);

   ////////// Мотор Front Right//////////////
   digitalWrite(L_PWM1, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM1, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN1,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN
  ////////////// Мотор Rear Right //////////////
   digitalWrite(L_PWM2, LOW );  // Устанавливаем логический 0 на входе драйвера L_PWM
   digitalWrite(R_PWM2, HIGH);  // Устанавливаем логическую 1 на входе драйвера R_PWM
   analogWrite (EN2,    255 );  // Устанавливаем 100% ШИМ на входах драйвера L_EN и R_EN

}
```

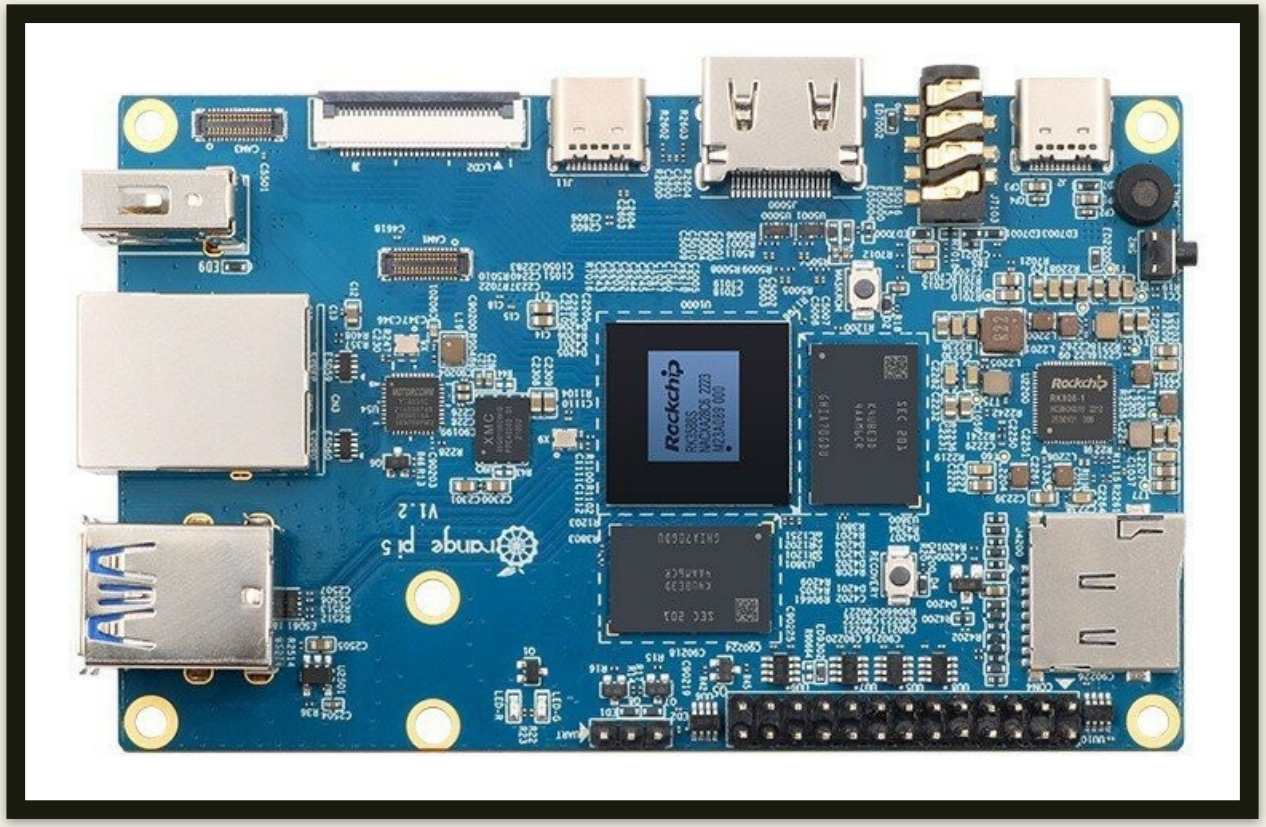
Ввод данных


```
void loop() {
  if (Serial1.available()) {
    command = Serial1.read();
    if (command == 'X') {
      state = 1;
    } else if (command == 'D') {
      state = 0;
    }
    if (command == 'B') {
      back();
    } else if (command == 'F') {
      forward();
    } else if (command == 'SR' ) {
      superright ();
    } else if (command == 'SL') {
      superleft ();
    } else if (command == 'R' ) {
      right ();
    } else if (command == 'L') {
      left ();
    } else if (command == 'G') {
      forwardleft ();
    } else if (command == 'I') {
      forwardright ();
    } else if (command == 'H') {
      backright ();
    } else if (command == 'J') {
      backleft ();
    }  else if (command == 'S'){
      stopRobot();
    }
  }
}
```



ПЕРСПЕКТИВЫ

Машинное зрение



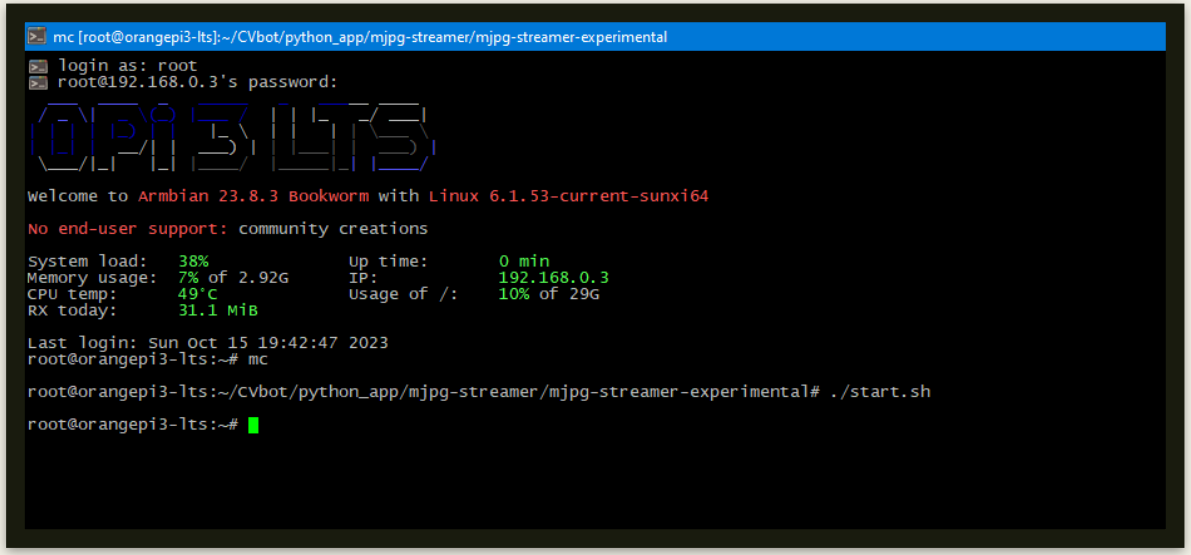
Плата Orange Pi 3 LTS

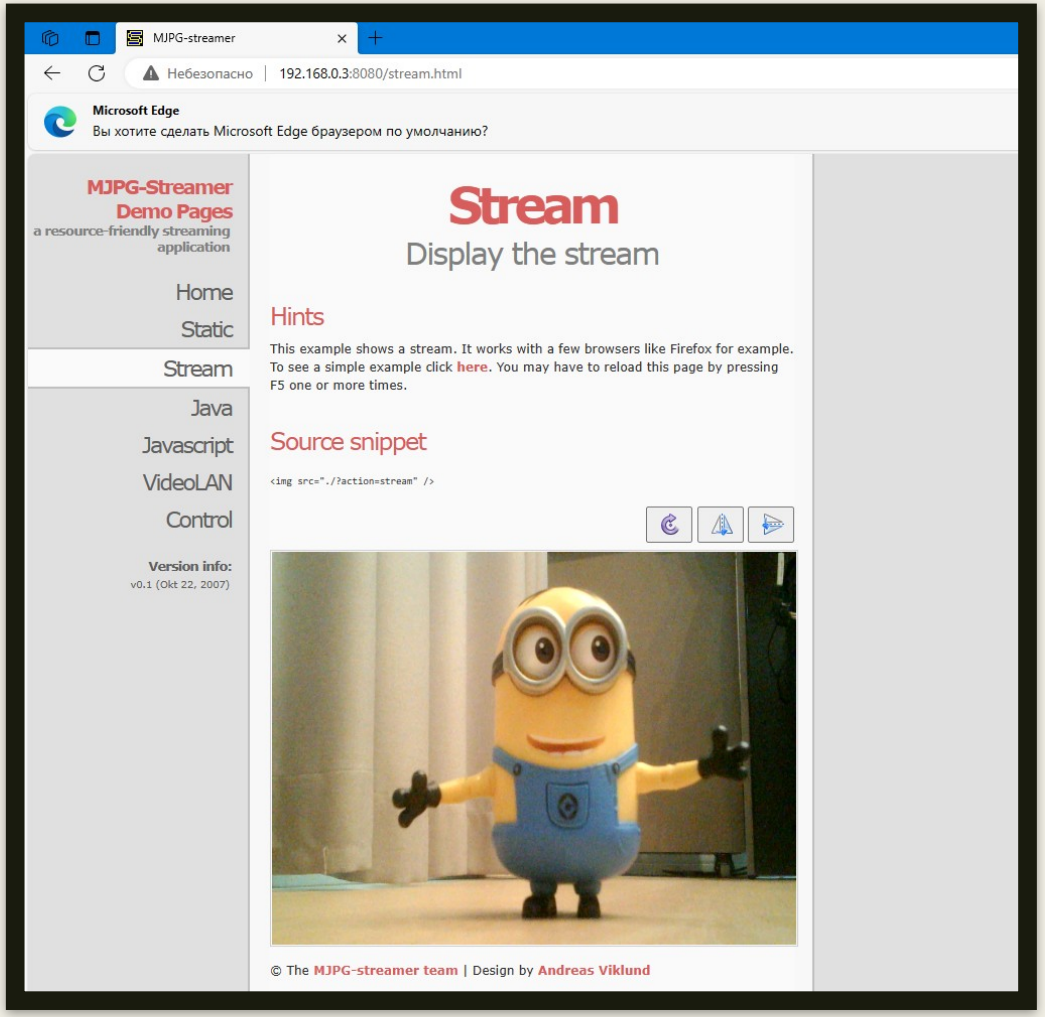
WI-FI и Bluetooth модули

HDMI-вывод

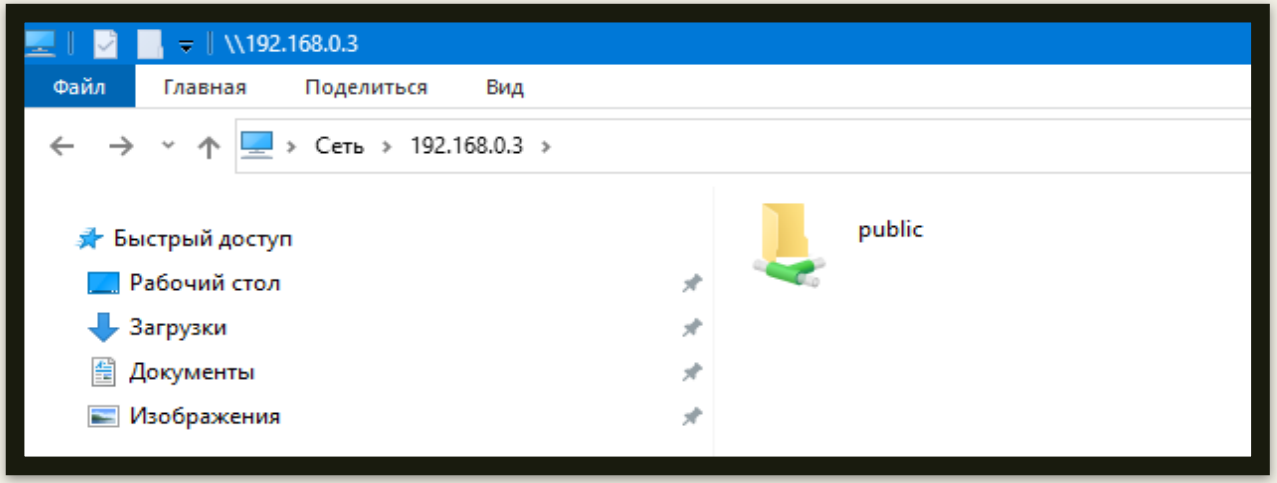
3 USB-порта

Видеотрансляция

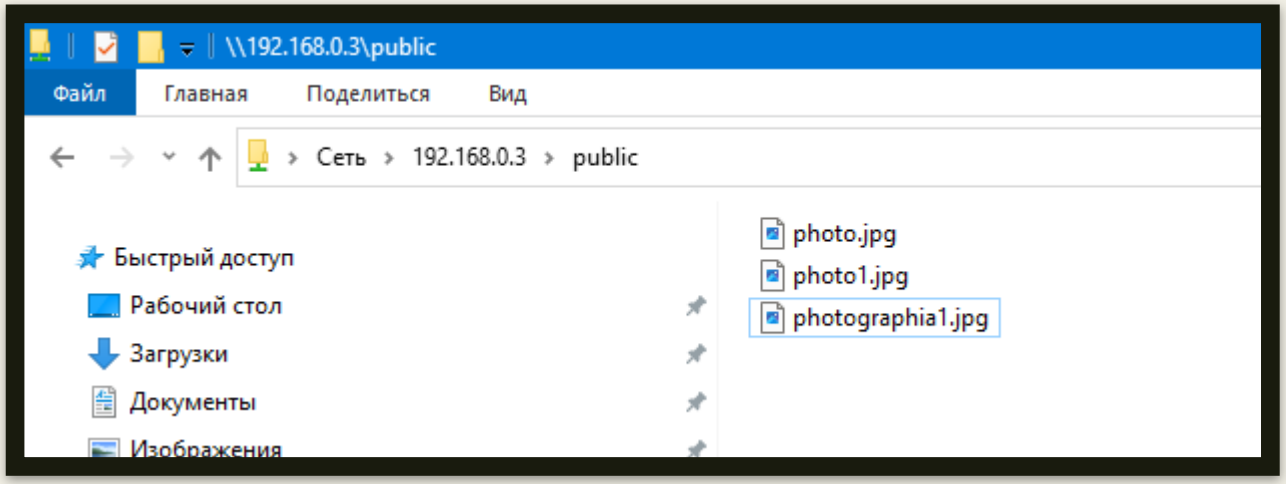


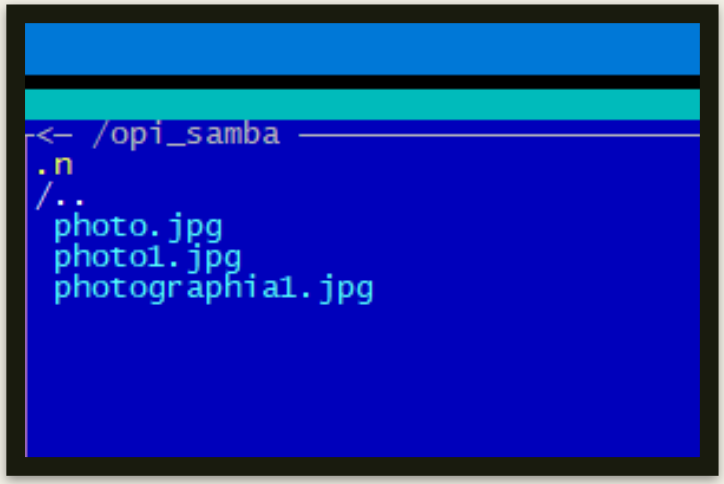


Сетевая папка



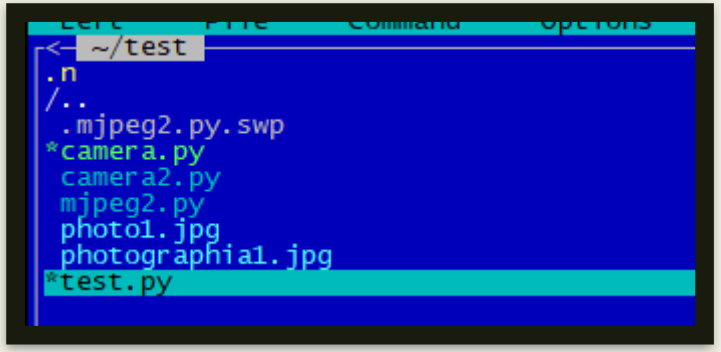
Путь к папке



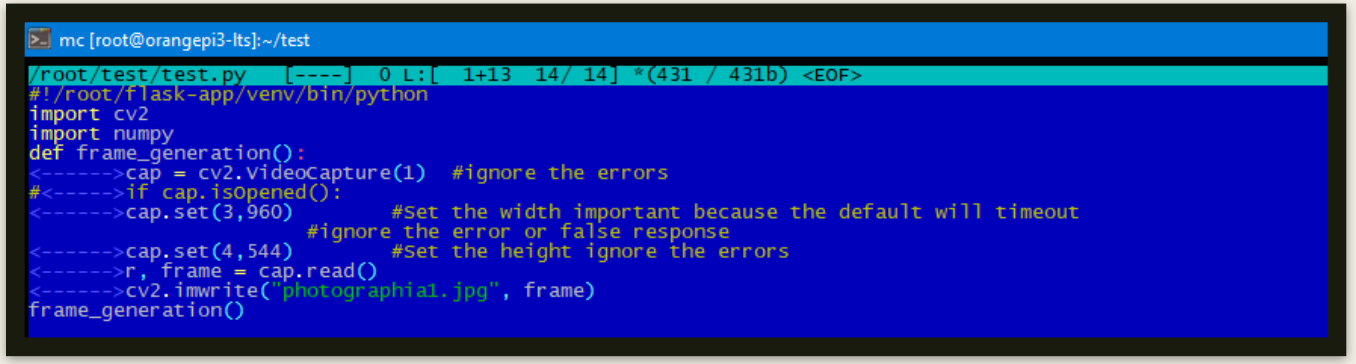


Файлы папки на микрокомпьютере и ПК

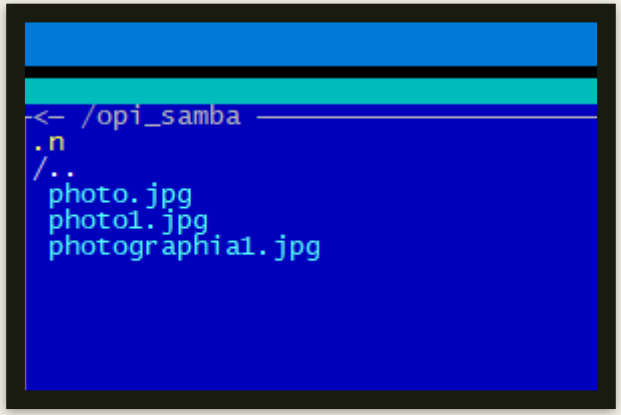
Снимок



Расположение скрипта и фотографий



Скрипт



Сетевая папка

Снимок

СПАСИБО ЗА ВНИМАНИЕ