

# **Модель автоматизированной логистической системы.**

Автор: Ковцур Даниил, Ковцур Павел

Государственное бюджетное учреждение дополнительного образования Дом творчества  
«Измайловский» Адмиралтейского района Санкт-Петербурга

Руководитель: Малышев Юрий Владимирович, Нестеров Владимир Викторович.

Санкт-Петербург

2024

# Содержание

## Оглавление

Содержание.....	2
Аннотация.....	3
О команде.....	3
Введение .....	4
Этапы разработки проекта.....	4
Презентация роботизированного решения - модели автоматизированной логистической системы .....	5
Программные алгоритмы .....	11
Эксперимент.....	15
Список используемых источников .....	16
Приложение №1. Чертежи 3D деталей. ....	17
Приложение №2. Программный код складского робота представлен ниже.....	20

## **Аннотация**

В рамках проекта представлена модель автоматизированной логистической системы, состоящая из склада с грузами, складского робота, погрузочного устройства с манипулятором, а также робота доставщика. Особенностью проекта является применение метода ориентирования доставщика при помощи маяков на поле. Объектом исследования является автоматизированная система доставки. Целью является разработка модели автоматизированной логистической системы для доставки грузов. Нашими задачами являются: создание беспилотного робота для склада для автоматического терминала, разработка автоматизированного терминала с манипулятором и с автоматической подачей грузов, настройка его взаимодействия с роботом-доставщиком для осуществления непрерывной погрузки манипулятора, создание беспилотного робота-доставщика, провести испытания модели. Разработана модель автоматизированной логистической системы по доставке грузов.

## **О команде**

Состав команды:

Нестеров Владимир Викторович

Преподаватель ДДЮТ Измайловский по робототехнике. Руководитель команды.

Мальшев Юрий Владимирович

Преподаватель ДДЮТ Измайловский по робототехнике. Руководитель команды.

Ковцур Даниил Максимович

Ученик ДДЮТ Измайловский. Капитан команды. Ответственный за инженерную часть проекта, программирование.

Ковцур Павел Максимович.

Ученик ДДЮТ Измайловский. Помощник капитана команды. Ответственный за конструкцию и создание нетиповых компонентов с использованием пайки.

## **Введение**

### **Актуальность:**

Сегодня логистические системы часто используются. Одним из важных направлений является полная автоматизация таких систем. При этом необходимо учитывать автоматическую погрузку, сортировку, выгрузку, точную остановку, автоматическое распределение по пунктам назначения. При этом одно из направлений – сокращение стоимости и времени доставки за счет использования роботов и систем автоматизации. Такие компании, как СБЕР, обещают доставку от 20 минут.

### **Цель проекта**

Цель - разработка модели автоматизированной логистической системы для доставки грузов

### **Задачи**

В связи с данной целью необходимо решить следующие задачи:

- Создание беспилотного робота для склада для автоматического терминала.
- Разработка автоматизированного терминала с манипулятором и с автоматической подачей грузов, настройка его взаимодействия с роботом-доставщиком для осуществления непрерывной погрузки манипулятора.
- Создание беспилотного робота-доставщика.
- Провести испытания модели.

### **Этапы разработки проекта**

Разработка манипулятора

Разработка робота-доставщика

Разработка складского робота

Разработка программы для каждого из них

Настройка роботов для непрерывной работы проекта

## Презентация роботизированного решения - модели автоматизированной логистической системы

Модель автоматизированной логистической системы состоит из следующих элементов: складского робота, манипулятор с модулем автоподачи, робота доставщика.

### *Манипулятор*

Наш манипулятор состоит из пяти моторов и двух блоков ev3 – первый из блоков отвечает за работоспособность манипулятора, а второй – за загрузку этого же манипулятора. Манипулятор может поворачиваться вправо и влево относительно горизонтали, а также опускаться вверх - вниз по вертикали. Третий мотор управляет захватом робота. Четвертый мотор отвечает за работу конвейера, обеспечивающего непрерывную работу манипулятора. Для более точного поворота стрелы манипулятора на нем закреплены 2 датчика касания и применены автоматические алгоритмы калибровки положений всех моторов. Второй блок EV3 с помощью мотора загружает на конвейер манипулятора грузы, привезенные складским роботом. Обновленная версия манипулятора позволяет перемещать грузы большей массы, до 150 грамм. Также теперь манипулятор может поднимать грузы даже с пола.

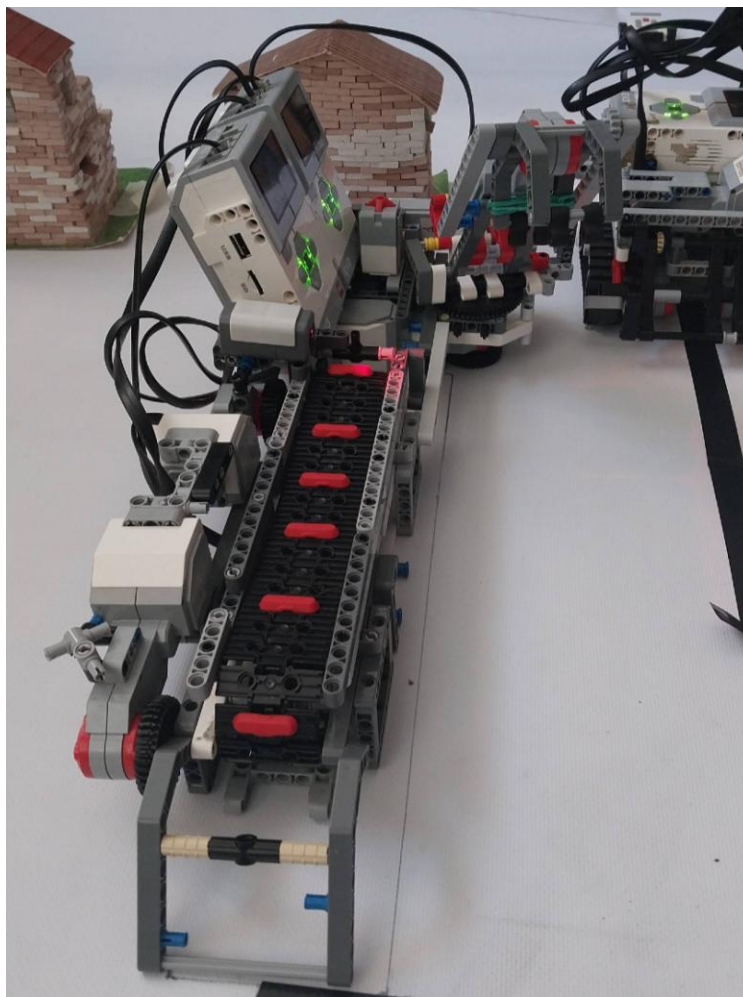


Рисунок 1. Манипулятор.

### ***Робот доставщик***

Робот выполнен в виде тележки на двух моторах и имеет кузов для перевозки грузов. Для ориентирования в пространстве робот использует инфракрасный датчик и установленные на поле маяки.

Робот устанавливает связь с манипулятором и выполняет команды от манипулятора. Также на роботе закреплена FPV камера, подключенная к блоку EV3.

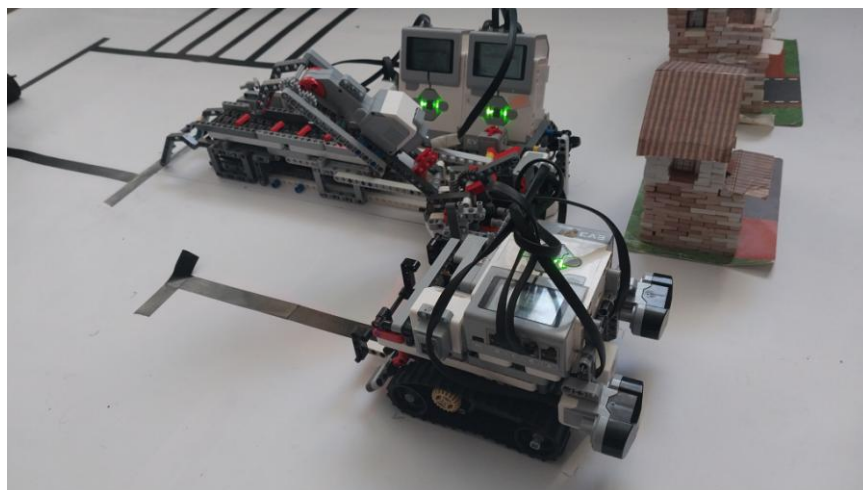


Рисунок 2. Робот доставщик.

### ***Складской робот***

Складской робот состоит из тележки, собранной из конструктора Lego Mindstorms EV3, которая движется благодаря двум моторам. Спереди этой тележки расположен подъёмник, на котором находятся вилы. Они могут опускаться и подниматься для обеспечения фиксации груза при перемещении. Сзади закреплён блок EV3. Для движения по линии используется ПД-алгоритм и 2 датчика цвета. На вилах закреплён датчик цвета, который считывает, находится ли груз на подъёмном механизме. На роботе установлен механизм натяжения ленты, понимающий вилы. Складской робот складировать грузы на складе, который представляет собой 3 полки с 5 ячейками на каждой из них. Склад может программно масштабироваться.

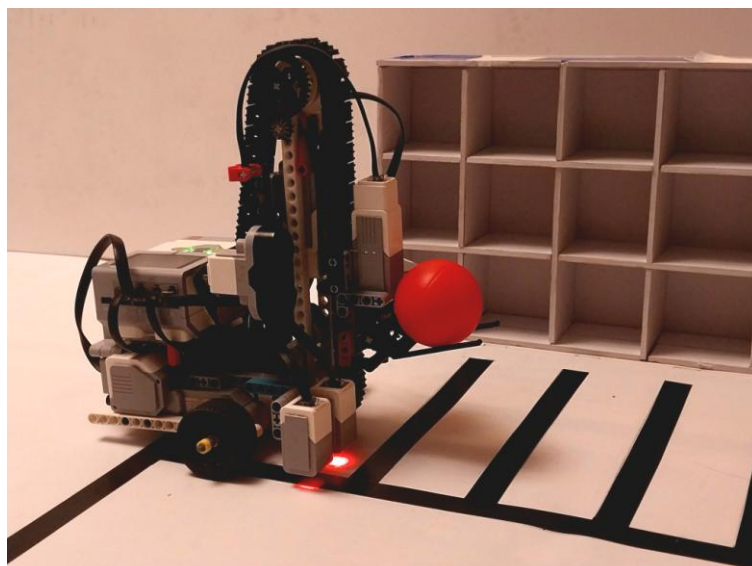


Рисунок 3. Складской робот.(первая версия)

### ***Функции модели***

#### ***Модель обеспечивает следующие функции:***

- Доставка грузов нужного типа складским роботом
- Погрузка груза манипулятором и идентификация груза по цвету
- Навигация робота -доставщика по маякам во время доставки.
- Возможность остановить робота-доставщика оператором,
- Возможность включения и выключения камеры оператором на роботе доставщике, чтобы она не перегревалась
- Автоматическая калибровка мобильных роботов;
- Автоматическая калибровка манипулятора;
- Определение типа груза и установка маршрута в робота-доставщика;
- Выгрузка груза в месте в зависимости от цвета груза;
- Организация взаимодействия между тремя роботами;
- Удаленное управление комплексом (пауза);
- Возможная передача видеоизображения с доставщика.

### ***Принцип действия модели***

Перейду к описанию действий модели представляемого проекта.  
Модель представлена на рисунке ниже.

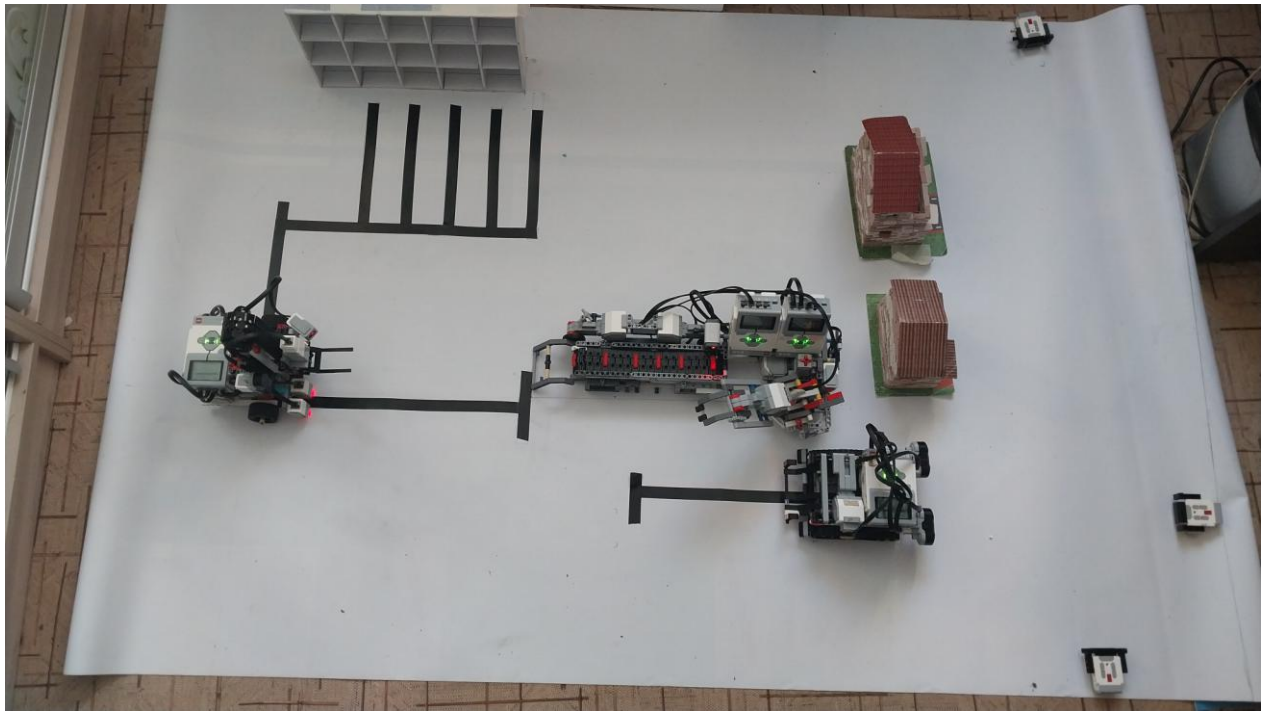


Рисунок 4. Модель системы

Модель создана на основе робототехнического конструктора LEGO Mindstorms EV3

В модели оператор выбирает, надо ли ему загрузить груз на склад, или необходимо его доставить получателю.

Если оператор выбирает режим доставки к получателю (к одному из 4-х домов), то складской робот едет к складу, захватывает нужный груз и доставляет к манипулятору. Груз помещается на конвейер.

Манипулятор загружает груз в робота-доставщика на складе и даёт команду роботу на отправку.

Далее робот-доставщик доставляет груз одному из получателей, при этом ориентируется по маякам.

При загрузке на склад манипулятор берет груз с определенного места на полигоне, и кладет его на конвейер. Далее груз по конвейеру попадает на складского робота. Этот робот едет на склад и загружает груз в свободное место, а затем возвращается на нулевую позицию.

Особенностью проекта является наличие камеры на роботе-доставщике, а также 3d печатных деталей для всех роботов проекта.

Программа написана в среде LEGO MINDSTORMS EV3 Home Edition а также в CLEVR3R.

### ***Конструкторская и технологическая часть модели***

В пунктах, представленных ниже, будет описана конструкторская и технологическая часть.



1) Манипулятор имеет 2 степени свободы и может перемещать достаточно тяжелые грузы ( до 150 грамм) грузы . Датчик цвета зафиксирован на конвейере так, чтобы он работал, пока не появится какой либо груз. После достаточного приближения датчика цвета к грузу происходит считывание цвета шарика. Для точного перемещения стрелы манипулятора установлены датчики касания. Они обеспечивают точный поворот стрелы на 90 градусов.

2) Робот доставщик имеет 3 степени свободы. Для повышения проходимости используются гусеницы, чтобы робот мог преодолевать препятствия. Робот оснащен специальным переходником собственной разработки для подключения FPV камеры и радиопередатчика. **Социальное взаимодействие и инновации**

Решение поможет автоматизировать систему доставки, при использовании навигации по геолокации. Разработанное решение позволит сократить часть людей, а также значительно ускорить доставку.

Мы провели опрос, состоящий из таких вопросов, как:

1. Важна ли автоматизация на складах?
2. Сколько автоматических складов вы знаете?
3. Выгодно ли содержать складского робота(ов) вместо множества сотрудников?
4. Как можно улучшить и ускорить работу складов?
5. Как можно уменьшить время доставки грузов заказчикам?

Давайте перейдем к описаниям ответов участников опроса.

На первый вопрос большинство опрошенных ответили утвердительно, так как скорее всего эти люди видели в автоматизации ускорение процесса доставки и повышение экономики.

Второй вопрос оказался самым сложным и вызвал затруднения, так как даже самые известные складские компании только начинают внедрять автоматизацию.

На третий вопрос почти все респонденты ответили положительно. Это обусловлено возможностью эксплуатировать робота круглосуточно, так как он не устает, не нуждается в зарплате и медицинской помощи, и может заменять сразу нескольких работников склада.

На четвертый вопрос все участники опроса ответили по-разному, однако присутствовало несколько одинаковых решений:

- 1) Автоматизировать склад.
- 2) Заменить работников склада, выполняющих тяжелый физический труд, на более выносливых роботов.
- 3) Применить системы автоматизации, которые позволят увеличить эффективность работы людей.

Таким образом, автоматизация складов является наиболее подходящим решением по улучшению и ускорению работы складов.

Ответы опрошенных людей на пятый вопрос показали, что автоматизация доставки является основным решением для уменьшения времени доставки грузов заказчикам.

## Программные алгоритмы

В программе нашего проекта используются циклы, подпрограммы, а также ветвления( условия)

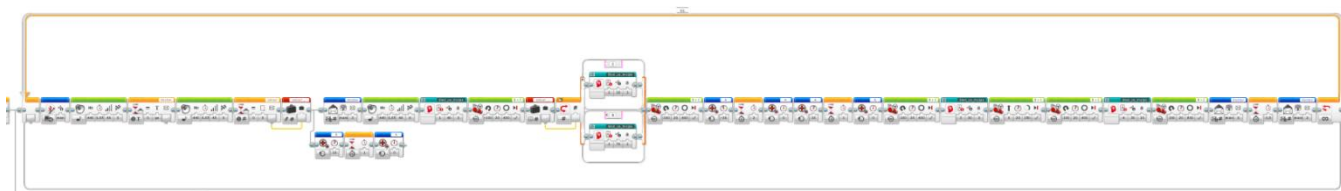


Рисунок5. Главный цикл робота доставщика.

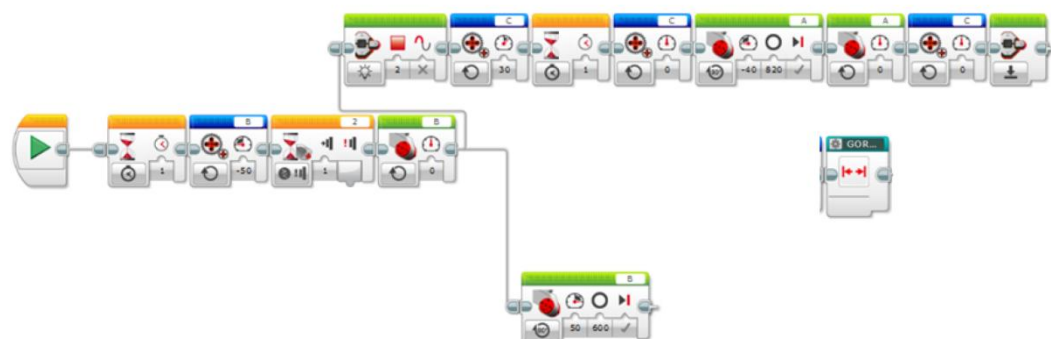


Рисунок6. Подпрограмма поворота направо стрелы манипулятора.

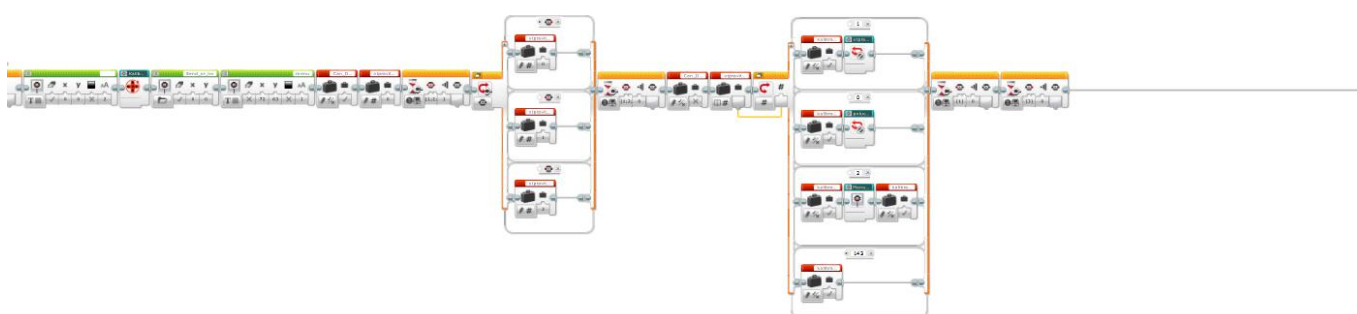


Рисунок7. Ветвления (условия) в главном цикле программы манипулятора.

Схемы выбора представлены для оператора, чтобы выбрать, в какой дом доставить груз, для автоматической остановки робота доставщика, а также для работы других важных функций нашего проекта.

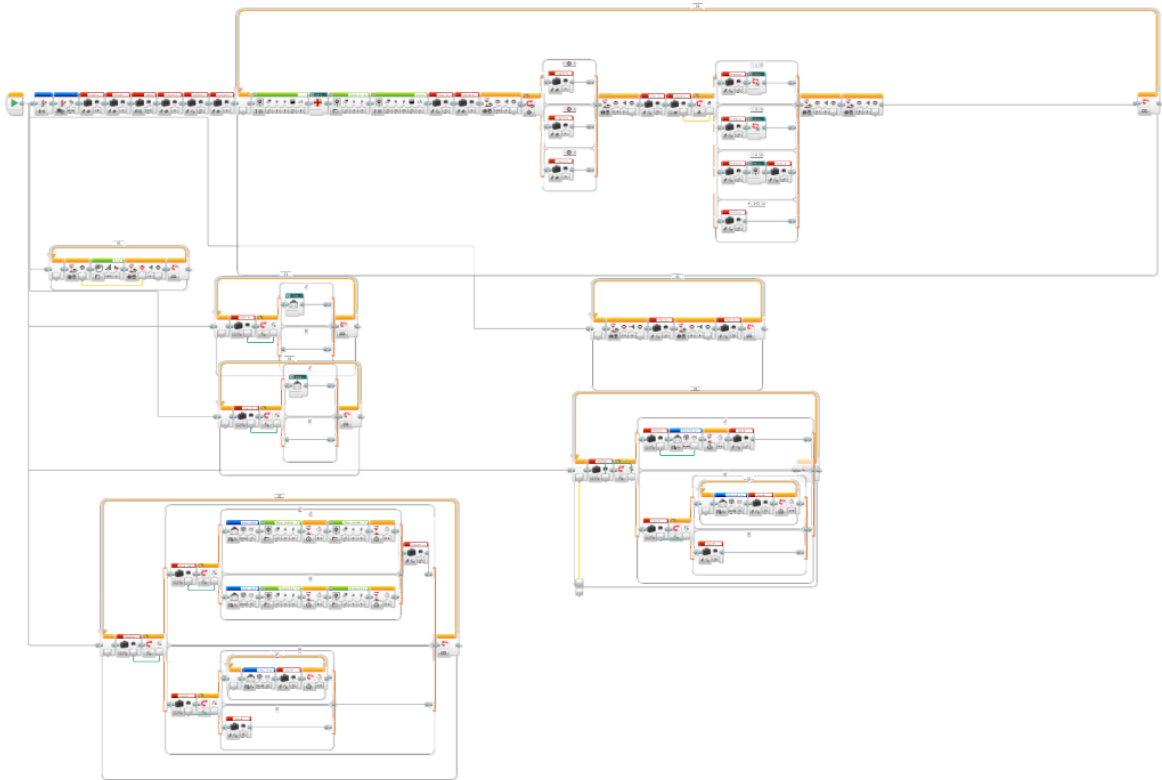


Рисунок9. Программа манипулятора.

Циклы применены для организации движения роботов и непрерывной работы программы всех роботов в целом.

Подпрограммы применены для упрощения структуры программы и для уменьшения ее объема.

Ниже приведена блок схема работы нашего проекта

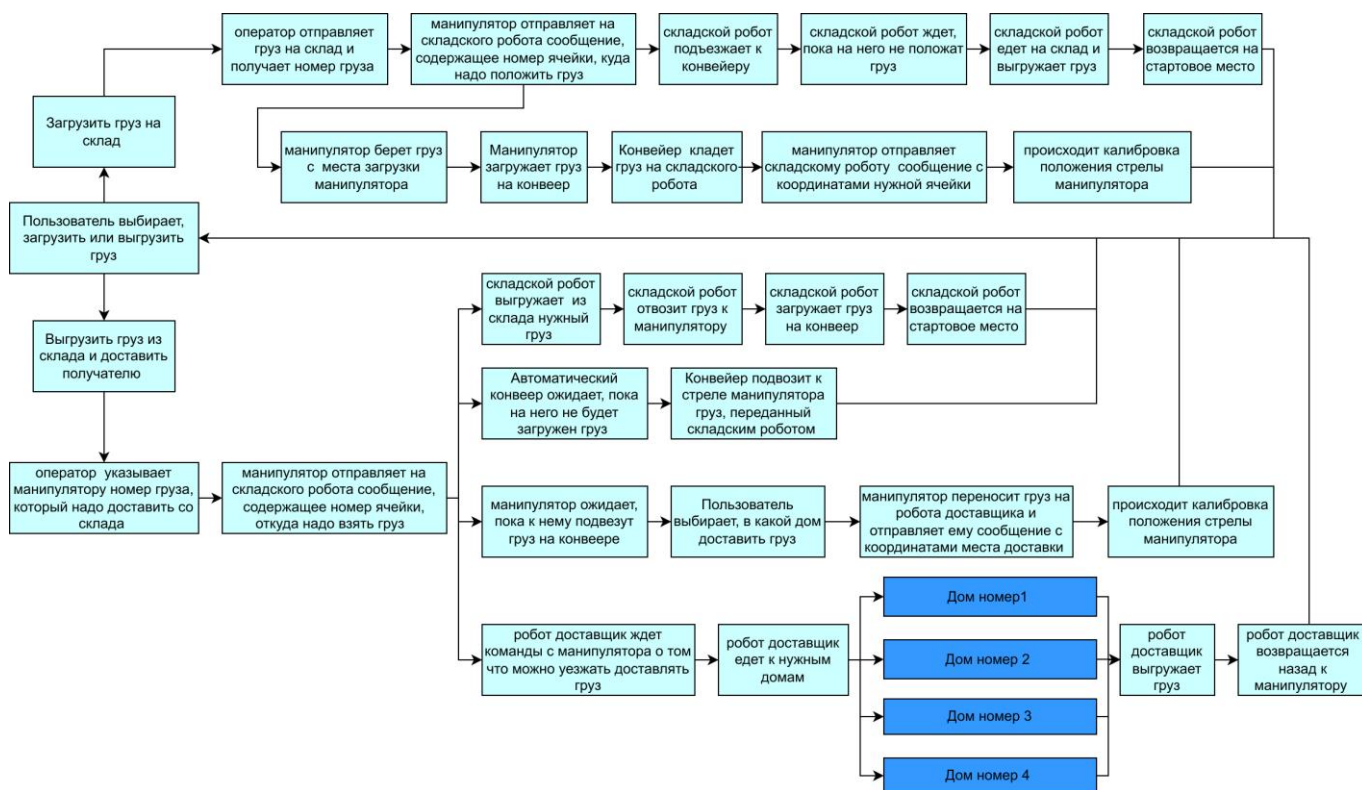


Рисунок 8. блок схема работы проекта

Давайте теперь перейдем к описанию экономической части нашего проекта.

### Экономическая часть

Данную схему возможно использовать в системах автоматизированной доставки.

Таблица №1

### Бизнес -модель проекта

<p><b>Ключевые партнёры</b> Отечественная компания AstaBot</p>	<p><b>Ключевые действия</b> Автоматическая доставка грузов становится более популярной в мире. Компании хотят сократить время доставки и сократить расходы.</p>	<p><b>Ценностное предложение</b> Доставка и сортировка роботами дешевле, чем использование курьеров и рабочих.</p>	<p><b>Сегменты клиентов</b> Компания Яндекс ввела доставку роботами только в одном районе Петербурга, при этом другие кампании только планируют введение беспилотной доставки роботами.</p>
<p><b>Ключевые ресурсы</b> Всё больше количество товаров продаётся через онлайн магазины.</p>			

<b>Структура издержек</b>	<b>Потоки</b>
<p>1. исследования рынка услуг доставки, и других рынков;</p> <p>2. производство, в том числе расходы на электроэнергию, материалы, амортизацию фондов и нематериальных активов, оплату труда портового персонала</p> <p>3. обслуживание производственного процесса и управление им;</p> <p>4. набор и подготовку кадров;</p> <p>5. организации производства, труда, повышение качества продукции порта);</p> <p>6. подготовку и изучение новой продукции порта, ее сбыт.</p>	<p><b>доходов</b></p> <p>Сокращение выплаты расходов на курьеров и рабочих.</p>

## **Эксперимент**

Разработка программы испытаний

Программа испытаний включает:

- Проверка работы мобильных роботов.
- Проверка работы автоматизированного терминала в части автоматической погрузки и разгрузки разных грузов. Модель позволяет осуществить доставку и сортировку грузов в виде шариков в пункт разгрузки.

В соответствии с программой испытаний проведены испытания, которые подтвердили работоспособность модели.

## **Выводы**

В рамках проекта представлена модель автоматизированной логистической системы, состоящая из склада с грузами, складского робота, погрузочного устройства с манипулятором, а также робота доставщика. Особенностью проекта является ориентирование робота-доставщика при помощи маяков на поле.

В модели оператор получает запрос на доставку груза и запускает процесс доставки определённого груза.

Складской робот доставляет нужный груз к манипулятору. Груз помещается на конвейер.

Манипулятор загружает груз в робота-доставщика на складе и даёт команду роботу на отправку. Далее робот-доставщик доставляет груз одному из получателей, при этом ориентируется по маякам. Особенностью проекта является наличие камеры на роботе-доставщике, а также 3d печатных деталей для некоторых компонентов проекта.

Цель достигнута, разработана модель автоматизированной логистической системы по доставке грузов.

Дальнейшие задачи

Дальнейшей задачей является оснащение проекта компьютерным зрением, а также построение робота-доставщика на основе Arduino.

### **Список используемых источников**

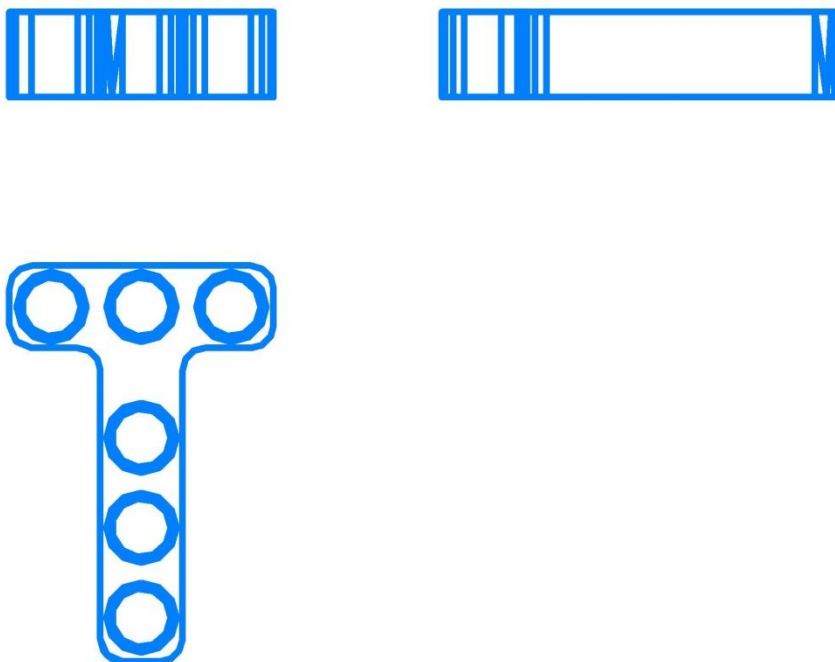
1. Большая книга LEGO MINDSTORMS EV3. 181 / Лоренс Валк ; [пер. с англ. С.В. Черникова]. – Москва : Издательство “э”, 2017. – 408с. : ил – (Подарочные издания. Компьютер).
2. Книга идей LEGO MINDSTORMS EV3. 181 механизм и устройство / Йошихито Исогава ; [пер. с англ. О.В. Обручева]. – Москва : Издательство “э”, 2017. – 232с. : ил – (Подарочные издания. Компьютер).
3. Филлипов С.А. Робототехника для детей и родителей. -СПБ.: Наука.



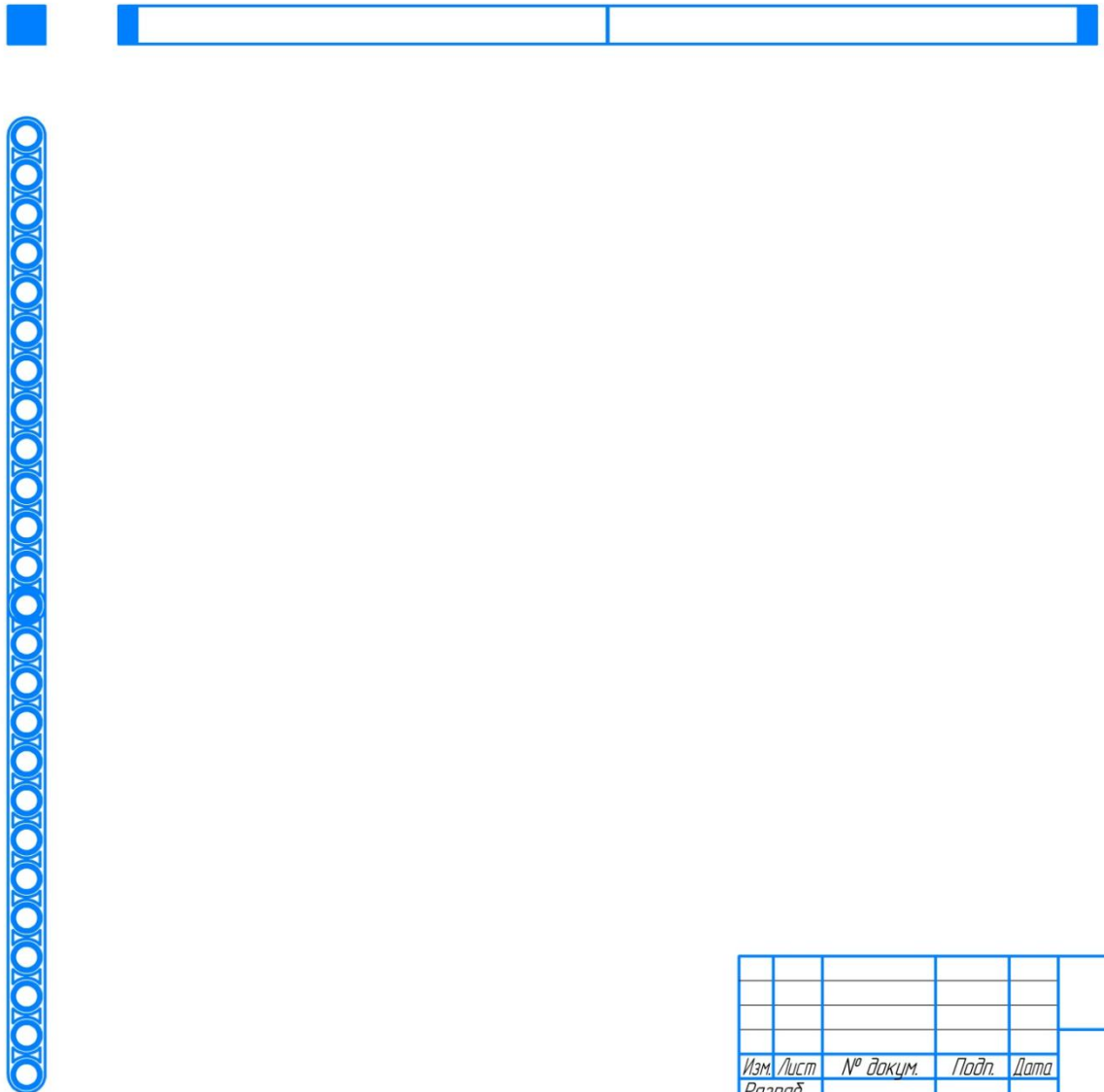
## Приложение №1. Чертежи 3D деталей.

Ниже представлены схематичные чертежи 3д печатных деталей, которые были созданы в КАД-системах Autodesk Inventor и Компас. Модели были напечатаны на 3д принтере flying bear ghost6 (модернизированный)

Stl-файлы доступны по ссылке <https://disk.yandex.ru/d/gMYe6H4NeKxotQ>

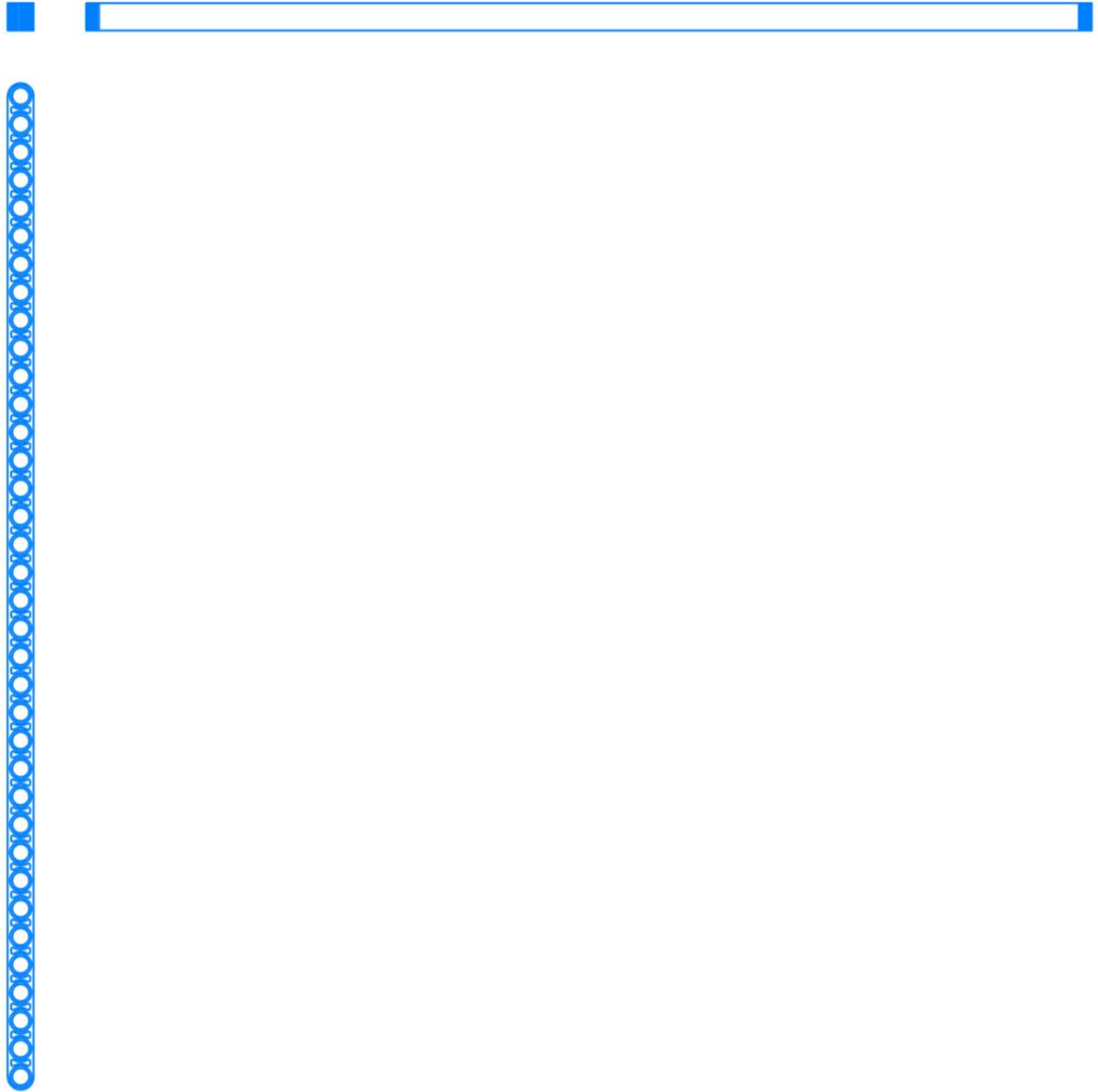


Чертеж1. Т-образная деталь для пульта

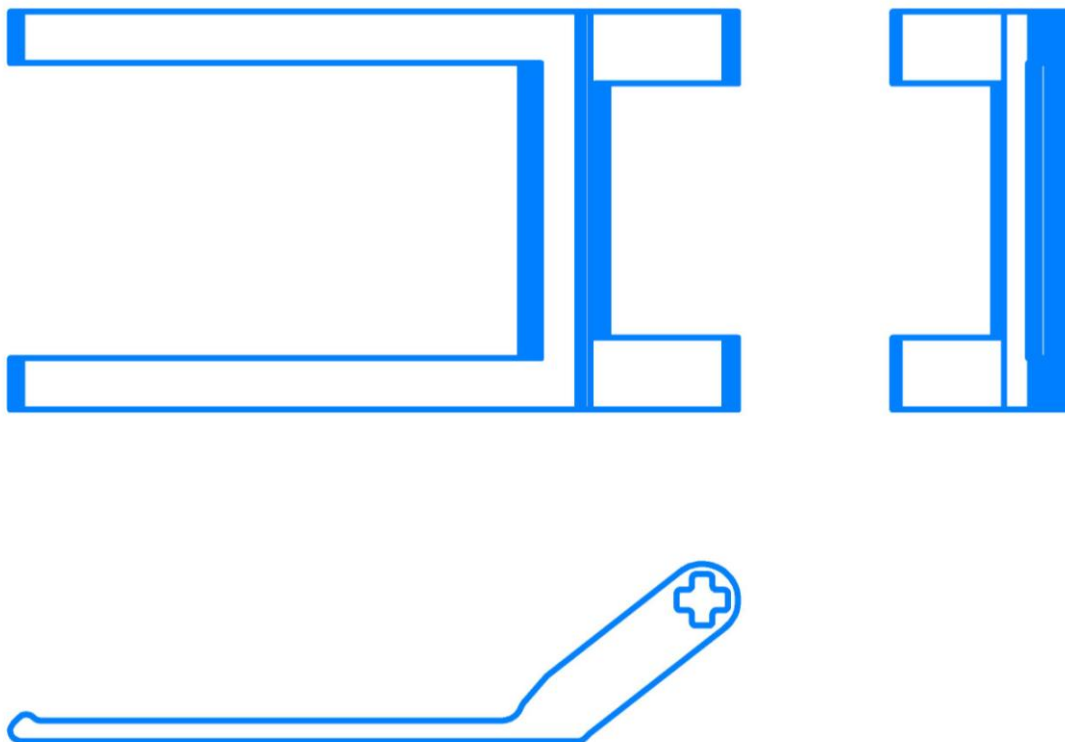


<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>	
<i>Разраб.</i>					
<i>Проб.</i>					

Чертеж2. 25L балки для подъемного механизма у складского робота



Чертеж3. 36L балка для манипулятора



Чертеж4. Вилка складского робота

**Приложение №2. Программный код складского робота представлен ниже.**

```
Speaker.Tone(10,1000,50)
"Program.Delay(2000)
Speaker.Tone(10,1000,50)
MotorA.SetReversPolarity()

'Mailbox.Connect("manipulator")"
```

```

Mail_indentificator =0
Mail_indentificator = Mailbox.CreateForNumber("1")
Mail_indentificator2 = Mailbox.CreateForNumber("2")
flag =0
kp = 0.4
kd = 1.5
kpS2 = 1
kdS2 =2
speed = 30
err = 0
err_old = 0
white = 0
black = 0
grey = 0
gl_white = 61
gl_black = 35
gl_grey = 35
gl_s1_Grey = 25
left_colour_cor = 0
p = 0
d = 0
left_colour =2
right_colour = 3
Rast_up_down_degr =1300
Rast_up_down_degr_vilka = 160
Poworot_90_degrees = 290
grad =100
Zagr_ili_Brat = 1
Porog_stop_goline = 21'20
can_stopp = 1
Go_to_x_manipul = 350
Podnim_do_manipul_put = 700
Podnim_do_manipul_take = 300
Podnim_do_manipul = 700
rast_yach_Y = 595
x = 0
y = 0
speed_go_to_coordinates = 40
nomer_Yacheiki =0
skolko_yacheek_na_polke = 5
visota_sklada = 3
Max_Razmer_sklad = skolko_yacheek_na_polke*visota_sklada
"letter = 115
nomer_Yacheiki = 47
'Calibrate_Zahv2()
'Calibrate_Zahv1()
'Podschet_Yacheiki()
'Thread.Run = Go_to_coordinates_y

Motor.StartPower("A",-10)
Calibrate_Zahv2()
Thread.Run = Calibrate_Zahv1

```

```

goline_do_perekr()
povorot_left()
Thread.Run = Go_Y_to_mid
Motor.ScheduleSync("BC",-15,-15,100,"true")
Motor.Wait("BC")
goline_do_perekr()
povorot_left()
Motor.ScheduleSync("BC",-15,-15,140,"true")
Motor.Wait("BC")
Program.Delay(100)

```

```
Bluetooth_priem_signal()
```

```
Program.Delay(3006486490)
```

```

'Zagr_in_sklad()
'Motor.ScheduleSync("BC",40,0,950,"true")
'Motor.Wait("BC")
'Motor.ScheduleSync("BC",0,-40,250,"true")
'Motor.Wait("BC")
'Dost_iz_sklad()
'Vigruzka()

```

```
Sub Calibrate_Zahv1
```

```

    Motor.StartPower("A",-15)
    Program.Delay(300)
    While Motor.GetSpeed("A")>3
    EndWhile

```

```
Motor.SchedulePower("A",1,Rast_up_down_degr_vilka/4,Rast_up_down_degr_vilka/2,Rast_up_down_degr_vilka/4,"true")
```

```
    Motor.Wait("A")
```

```
EndSub
```

```
Sub Calibrate_Zahv2
```

```

' Motor.StartPower("A",-15)
Motor.SchedulePower("D",100,0,200,0,"true")
Motor.Wait("D")
Motor.StartPower("D",-50)
Program.Delay(200)
While Motor.GetSpeed("D")<-10
EndWhile
'While Motor.GetSpeed("A")<-3
' EndWhiles
Motor.Stop("D","True")
"

```

```
Motor.SchedulePower("A",1,Rast_up_down_degr_vilka/4,Rast_up_down_degr_vilka/2,Rast_up_down_degr_vilka/4,"true")
```

```
    Motor.SchedulePower("D",50,0,200,0,"true")
```

```
    "Motor.SchedulePower("D",-
```

```
80,Rast_up_down_degr/4,Rast_up_down_degr/2+Rast_up_down_degr/4,100,"true")
```

```
    Motor.Wait("D")
```

```
    Motor.StartPower("D",-20)
```

```

Program.Delay(200)
While Motor.GetSpeed("D")<-1
EndWhile
'While Motor.GetSpeed("A")<-3
' EndWhiles
Motor.Stop("D","True")
Motor.SchedulePower("D",50,0,70,0,"true")

Motor.Wait("D")
Motor.ResetCount("D")
EndSub
Sub Grey_calibrate
Buttons.Flush()
Buttons.Wait()
Buttons.Flush()
Speaker.Tone(10,1000,50)
left_colour_cor = Sensor.ReadPercent(right_colour)-Sensor.ReadPercent(left_colour)
gl_white =(Sensor.ReadPercent(left_colour)+ left_colour_cor
+Sensor.ReadPercent(right_colour))/2
Buttons.Wait()
Speaker.Tone(10,1000,50)
Buttons.Flush()
Buttons.Wait()
Speaker.Tone(10,1000,50)
Program.Delay(50)
Speaker.Tone(10,1000,50)
Buttons.Flush()
gl_black =(Sensor.ReadPercent(left_colour)+ left_colour_cor
+Sensor.ReadPercent(right_colour))/2
gl_grey = (black+white)/2
EndSub

Sub goline_pd
err = gl_s1_Grey -Sensor.ReadPercent(right_colour)
p = err*kp
d = (err-err_old)*kd
pd = p+d
If Sensor.ReadPercent(4) > Porog_stop_goline Then
Motor.StartSync("BC",speed + pd,speed - pd)
EndIf
If can_stopp = 0 Then
Motor.StartSync("BC",speed + pd,speed - pd)
EndIf

If Sensor.ReadPercent(4) < Porog_stop_goline and can_stopp = 1 Then
Motor.Stop("BC","True")
EndIf

err_old = err
Program.Delay(10)

```

```

EndSub
Sub goline_pdS2
  err = Sensor.ReadPercent(left_colour)-gl_s1_Grey
  p = err*kpS2
  d = (err-err_old)*kdS2
  pd = p+d
  If Sensor.ReadPercent(4) > Porog_stop_goline Then
    Motor.StartSync("BC",speed + pd+5,speed - pd)
  EndIf
  If can_stopp = 0 Then
    Motor.StartSync("BC",speed + pd,speed - pd)
  EndIf

  If Sensor.ReadPercent(4) < Porog_stop_goline and can_stopp = 1 Then
    Motor.Stop("BC","True")
  EndIf
  err_old = err
  Program.Delay(10)

EndSub
Sub povorot_right
  Motor.ScheduleSync("BC",15,-15,Poworot_90_degrees,"true")
  Motor.Wait("BC")

  Motor.Stop("BC","True")
EndSub
Sub povorot_left
  Motor.ScheduleSync("BC",-15,15,Poworot_90_degrees,"true")
  Motor.Wait("BC")

  Motor.Stop("BC","True")
EndSub
Sub goline_do_perekr

  While Sensor.ReadPercent(left_colour)+ left_colour_cor < gl_grey+10
    goline_pd()
  EndWhile
  While Sensor.ReadPercent(left_colour)+ left_colour_cor > gl_grey
    goline_pd()
  EndWhile
  Speaker.Tone(10,2000,50)
EndSub
Sub goline_enc

  Motor.ResetCount("BC")
  While (Motor.GetCount("B")+Motor.GetCount("C"))/2<grad
    goline_pd()
  EndWhile
  Motor.Stop("BC","True")
EndSub
Sub goline_enc_s2
  Motor.ResetCount("BC")

```



```

While (Motor.GetCount("B")+Motor.GetCount("C"))/2<grad
  goline_pdS2()
EndWhile
Motor.Stop("BC","True")
EndSub
Sub goline_do_perecr_s2

While Sensor.ReadPercent(right_colour) < gl_grey+10
  goline_pdS2()
EndWhile
While Sensor.ReadPercent(right_colour) > gl_grey-5
  goline_pdS2()
EndWhile
Speaker.Tone(10,2000,50)

EndSub

Sub Go_to_coordinates_x
speed = speed_go_to_coordinates
'grad = 100
'goline_enc()
xn = x
While x>0
  x= x-1
  goline_do_perekr()
EndWhile
x = xn
Motor.ScheduleSync("BC",15,15,20,"true")
Motor.Wait("BC")
povorot_left()
can_stopp = 0
kp = 1
kd = 3
grad = 100
goline_enc()
  kp = 0.5
  kd = 1.5

EndSub
Sub Go_to_coordinates_y
Rast_podiom = 0
flag =0
'if Motor.GetCount("D") <>0 then
'Calibrate_Zahv2()
'endif

If y = 1 Then
  Rast_podiom = 20
EndIf
If y = 2 Then
  Rast_podiom =rast_yach_Y +70
EndIf

```

```

If y = 3 Then
  Rast_podiom =rast_yach_Y*2+70
EndIf
Rast_podiom = Rast_podiom - Motor.GetCount("D")

Motor.SchedulePower("D",speed_go_to_coordinates*1.5,0,(Rast_podiom/5)*2+Rast_podiom*0.6
,Rast_podiom*0,"true")
  Motor.Wait("D")
  flag =1
EndSub
Sub Zahvat
  While flag =0

  EndWhile
  speed = 10

  grad = 100
  goline_enc()
  Motor.ScheduleSync("BC",10,10,120,"true")
  Motor.Wait("BC")
  Program.Delay(400)
  Motor.SchedulePower("A",-50,0,15,0,"true")
  Motor.Wait("A")
  Motor.ScheduleSync("BC",-15,-15,660,"true")
  Motor.Wait("BC")
  povorot_left()

EndSub
Sub Rashvat
  While flag =0

  EndWhile

  speed = 10
  grad = 100
  goline_enc()
  Program.Delay(400)
  Motor.SchedulePower("D",80,0,90,0,"true")

  Motor.Wait("D")
  Motor.SchedulePower("A",10,5,20,5,"true")
  Motor.Wait("A")
  Program.Delay(1000)
  Motor.ScheduleSync("BC",-15,-15,480,"true")
  Motor.Wait("BC")
  Motor.SchedulePower("D",-80,0,90,0,"true")

  Motor.Wait("D")
  Motor.StartPower("A",0)

  povorot_left()

```

```

EndSub
Sub Dostavka
    Motor.ScheduleSync("BC",-15,-15,100,"true")
    Motor.Wait("BC")
    Thread.Run = Go_Y_to_mid
    can_stopp = 1
    grad = (x-1)*150-200
    speed = 30
    goline_enc()

    goline_do_perekr()
    povorot_left()
    Motor.ScheduleSync("BC",-15,-15,100,"true")
    Motor.Wait("BC")
    goline_do_perekr()
    povorot_left()
    Motor.ScheduleSync("BC",-15,-15,100,"true")
    Motor.Wait("BC")

EndSub
Sub Go_to_sklad

    Thread.Run = Go_Y_to_mid

    Motor.ScheduleSync("BC",-15,-15,140,"true")
    Motor.Wait("BC")
    povorot_left()
    Motor.ScheduleSync("BC",-15,-15,100,"true")
    Motor.Wait("BC")
    grad = 200
    goline_enc_s2()
    goline_do_perecr_s2()
    Motor.SchedulePower("BC",40,0,350,0,"true")
    Motor.Wait("BC")
    povorot_right()
    Thread.Run = Go_Y_to_down
    Motor.SchedulePower("BC",40,0,100,50,"true")
    Motor.Wait("BC")
    "grad = 50
    "goline_enc()
EndSub
Sub Go_to_manipul
    Mailbox.SendNumber("manipulator","3",0)
    Motor.ScheduleSync("BC",-15,-15,100,"true")
    Motor.Wait("BC")
    grad = 600
    goline_enc()
    goline_do_perekr()
    Motor.Stop("BC","True")
    'Motor.ScheduleSync("BC",-15,0,Go_to_x_manipul,"true")
    'Motor.Wait("BC")
    'Motor.ScheduleSync("BC",0,-15,Go_to_x_manipul,"true")

```

```

'Motor.Wait("BC")
'Motor.ScheduleSync("BC",15,15,300,"true")
'Motor.Wait("BC")
Go_Y_to_man_take()
Mailbox.SendNumber("manipulator", "3", 1)
Program.Delay(100)
Mailbox.SendNumber("manipulator", "3", 0)
EndSub
Sub go_from_manipul
'Motor.ScheduleSync("BC",-15,-15,200,"true")
'Motor.Wait("BC")
'Motor.ScheduleSync("BC",0,-15,Go_to_x_manipul,"true")
'Motor.Wait("BC")
'Motor.ScheduleSync("BC",-15,0,Go_to_x_manipul,"true")
'Motor.Wait("BC")

Motor.ScheduleSync("BC",-40,-40,100,"true")
Motor.Wait("BC")

Motor.ScheduleSync("BC",40,0,950,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",0,-40,250,"true")
Motor.Wait("BC")
goline_do_perecr_s2()

Motor.ScheduleSync("BC",40,0,950,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",0,-40,250,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",40,40,100,"true")
Motor.Wait("BC")

EndSub
Sub Speedy_go_to_sklad_from_manipul
Go_Y_to_man_put()
'Motor.ScheduleSync("BC",-15,-15,200,"true")
'Motor.Wait("BC")
'Motor.ScheduleSync("BC",0,-15,Go_to_x_manipul,"true")
'Motor.Wait("BC")
'Motor.ScheduleSync("BC",-15,0,Go_to_x_manipul,"true")
'Motor.Wait("BC")
Motor.ScheduleSync("BC",-40,-40,400,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",40,0,950,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",0,-40,250,"true")
Motor.Wait("BC")
goline_do_perecr_s2()
Motor.ScheduleSync("BC",40,40,100,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",40,0,400,"true")

```

```

Motor.Wait("BC")
Motor.ScheduleSync("BC",40,40,50,"true")
Motor.Wait("BC")

grad = 200
goline_enc_s2()
goline_do_perecr_s2()
Motor.SchedulePower("BC",40,0,350,0,"true")
Motor.Wait("BC")
povorot_right()
Thread.Run = Go_Y_to_down
Motor.SchedulePower("BC",40,0,100,50,"true")
Motor.Wait("BC")
"grad = 50
"goline_enc()
endsub
Sub Vigruzka

Motor.StartPower("A",-10)

Program.Delay(1000)
Thread.Run = Calibrate_Zahv1

Motor.ScheduleSync("BC",-40,-40,500,"true")
Motor.Wait("BC")

Motor.ScheduleSync("BC",40,0,950,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",0,-40,250,"true")
Motor.Wait("BC")
goline_do_perecr_s2()

Motor.ScheduleSync("BC",40,0,950,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",0,-40,250,"true")
Motor.Wait("BC")
Motor.ScheduleSync("BC",40,40,100,"true")
Motor.Wait("BC")
EndSub
Sub Dost_iz_sklad
Go_to_sklad()
speed_go_to_coordinates = 30
Podschet_Yacheiki()
Thread.Run = Go_to_coordinates_y
Go_to_coordinates_x()

Zahvat()
Dostavka()
Go_Y_to_man_put()
Go_to_manipul()

```

```

Mailbox.SendNumber("manipulator", "2", 546)
Program.Delay(10)
Mailbox.SendNumber("manipulator", "2", 1)
  Program.Delay(10)
Go_Y_to_man_put()
Motor.SchedulePower("A", 10, 5, 30, 5, "true")
Motor.Wait("A")
Program.Delay(1000)
Motor.SchedulePower("A", -10, 5, 30, 5, "true")
Motor.Wait("A")
go_from_manipul()
Calibrate_Zahv1()

  Thread.Run = Calibrate_Zahv1
EndSub
Sub Zagr_in_sklad
  Thread.Run = Go_Y_to_man_put

  Go_to_manipul()
  Go_Y_to_man_take()
  Wait_for_colour()
  Motor.SchedulePower("A", -10, 0, -20, 0, "true")
  Motor.Wait("A")
  Wait_for_colour()
  Speedy_go_to_sklad_from_manipul()
  speed_go_to_coordinates = 30
  Podschet_Yacheiki()

  Thread.Run = Go_to_coordinates_y

  Go_to_coordinates_x()

  Rashvat()

  Thread.Run = Calibrate_Zahv1
  Dostavka()

EndSub
Sub Go_Y_to_down
  Y_dvig = Motor.GetCount("D")
  Motor.SchedulePower("D", -50, Y_dvig/4, Y_dvig/2+ Y_dvig/4, 0, "true")
  Motor.Wait("D")
EndSub
Sub Go_Y_to_mid

  if Motor.GetCount("D") > 300 then
    speed2 = -50
    Y_dvig = Motor.GetCount("D") - 300
  EndIf
  if Motor.GetCount("D") < 300 then
    speed2 = 50
    Y_dvig = 300 - Motor.GetCount("D")

```

```

EndIf

Motor.SchedulePower("D",speed2, Y_dvig/4, Y_dvig/2+ Y_dvig/4,0,"true")
Motor.Wait("D")
EndSub
Sub Go_Y_to_man_take

if Motor.GetCount("D")> Podnim_do_manipul_take then
    speed2 = -50
    Y_dvig = Motor.GetCount("D")-Podnim_do_manipul_take
EndIf
if Motor.GetCount("D")< Podnim_do_manipul_take then
    speed2 = 50
    Y_dvig = Podnim_do_manipul_take-Motor.GetCount("D")
EndIf
if Motor.GetCount("D") = Podnim_do_manipul_take then
    speed2 = 50
    Y_dvig =0
EndIf

Motor.SchedulePower("D",speed2, Y_dvig/4, Y_dvig/2+ Y_dvig/4,0,"true")
Motor.Wait("D")
EndSub
Sub Go_Y_to_man_put

if Motor.GetCount("D")> Podnim_do_manipul_put then
    speed2 = -50
    Y_dvig = Motor.GetCount("D")-Podnim_do_manipul_put
EndIf
if Motor.GetCount("D")< Podnim_do_manipul_put then
    speed2 = 50
    Y_dvig = Podnim_do_manipul_put-Motor.GetCount("D")
EndIf
if Motor.GetCount("D") = Podnim_do_manipul_put then
    speed2 = 50
    Y_dvig =0
EndIf

Motor.SchedulePower("D",speed2, Y_dvig/4, Y_dvig/2+ Y_dvig/4,0,"true")
Motor.Wait("D")
EndSub
Sub Wait_for_colour
    While Sensor.ReadPercent(1) <5

    EndWhile

endsub
Sub Podschet_Yacheiki
    nomer_Yacheiki_Zapas = 0
    nomer_Yacheiki_Zapas_b = nomer_Yacheiki
    If nomer_Yacheiki_Zapas_b>200 Then

```

```

    nomer_Yacheiki_Zapas_b = 200
EndIf
If nomer_Yacheiki_Zapas_b<0 Then
    nomer_Yacheiki_Zapas_b = 1
EndIf
While nomer_Yacheiki_Zapas_b >0
    nomer_Yacheiki_Zapas_b = nomer_Yacheiki_Zapas_b-1
    nomer_Yacheiki_Zapas++
EndWhile
If nomer_Yacheiki_Zapas>Max_Razmer_sklad Then
    nomer_Yacheiki_Zapas = Max_Razmer_sklad
EndIf
If nomer_Yacheiki_Zapas<1 Then
    nomer_Yacheiki_Zapas = 1
EndIf
y=0
x=0
While nomer_Yacheiki_Zapas >0
    nomer_Yacheiki_Zapas-= skolko_yacheek_na_polke
    y++
EndWhile
nomer_Yacheiki_Zapas+= skolko_yacheek_na_polke
While nomer_Yacheiki_Zapas >0
    nomer_Yacheiki_Zapas--
    x++
EndWhile
If y>visota_sklada Then
    y = visota_sklada
EndIf
If y<1 Then
    y = 1
EndIf
If x<1 Then
    x = 1
EndIf
If x>Max_Razmer_sklad/3 Then
    x = Max_Razmer_sklad/3
EndIf
LCD.Clear()
Rast_linii = 35
LCD.Line (1, Rast_linii,0, Rast_linii,180)
LCD.Line (1, Rast_linii* 2 ,0, Rast_linii* 2 ,180)
LCD.Line (1, Rast_linii* 3 ,0, Rast_linii* 3 ,180)
LCD.Line (1, Rast_linii* 4 ,0, Rast_linii* 4 ,180)

Rast_linii_gor = 42
LCD.Line (1, 0,Rast_linii_gor* 0.05 ,180 , Rast_linii_gor* 0.05 )
LCD.Line (1, 0,Rast_linii_gor* 1 ,180 , Rast_linii_gor* 1 )
LCD.Line (1, 0,Rast_linii_gor* 2 ,180 , Rast_linii_gor* 2 )
LCD.Line (1, 0,Rast_linii_gor* 3 ,180 , Rast_linii_gor* 3 )

LCD.FillCircle(1, x* Rast_linii - Rast_linii/2 , 127 -(y* Rast_linii_gor-Rast_linii_gor/2),15 )

```



```

LCD.Update()
Program.Delay(1000)

EndSub
Sub Bluetooth_priem_signal

While 1=1

Mailbox.SendNumber("manipulator", "2", 1)""""""""""
Lcd.Clear ()
LCD.Update()
letter = Mailbox.ReceiveNumber(Mail_indentificator)""""""""""
letter ++

Mailbox.SendNumber("manipulator", "2", 45)
Speaker.Tone(10,1000,50)
Speaker.Tone(1,1000,50)
Speaker.Tone(10,1000,50)
If letter>50 Then
Zagr_ili_Brat = 1
nomer_Yacheiki = letter-100
Zagr_in_sklad()

EndIf

If letter<50 Then
Zagr_ili_Brat = 0
nomer_Yacheiki = letter
Dost_iz_sklad()

EndIf
Lcd.Clear()
LCD.Update()
Motor.Stop("BC", "True")
Speaker.Tone(10,1000,1000)
Mailbox.SendNumber("manipulator", "2", 546)""""""""""
EndWhile
EndSub

```