МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ

ОЛИМПИАДА ШКОЛЬНИКОВ

Инженерно-конструкторское направление.

Инженерно-конструкторский профиль.

**Документация робота**

Кейс №8 «Сквозь лабиринт»

Школа: №1212

Команда: Platon1212

Начало подготовки: 28.01.2023

Руководитель: Удалов Анатолий Андреевич

Репозиторий проекта на GitHub - https://github.com/whaleOpop/Platon1212

**Оглавление**

[1 Цели и задачи робота 3](#_Toc127723238)

[2 Описание команды, распределение ролей, функций и обязанностей каждого участника команды. 4](#_Toc127723239)

[3 Общее описание функций разработанного решения 6](#_Toc127723240)

[4 Описание используемых аппаратных и программных узлов, модулей, фреймворков и других инструментов. 7](#_Toc127723241)

[5 Функциональное описание разработанного решение в виде UML-диаграмм: 9](#_Toc127723242)

[6 Описание кинематической системы разработанного устройства в виде схемы, диаграмм, подробно описывающие все аспекты кинематического движения, реализующегося во время функционирования устройства. 11](#_Toc127723243)

[7 Алгоритм работы разработанного программного обеспечения в виде блок-схем. 12](#_Toc127723244)

[8 Заключение, результаты работы, анализ функционирования разработанного устройства, предложения по возможному улучшению устройства. 13](#_Toc127723245)

[9 Список литературных источников. 14](#_Toc127723246)

# Цели и задачи робота

Проектируя робота необходимо учитывать особенности работы в условиях, зависящих от поставленной задачи. От поставленной перед робототехнической системой задачи зависит конфигурация и назначение робота.

Задача рассматриваемой робототехнической мехатронной системы — это осуществление перемещение техники и товаров. Роботу необходимо найти и переместить товары по складу.

Для реализации, поставленной задачи робототехническая система должна быть автономной, способной преодолевать препятствия, уметь ориентироваться в пространстве, обладать свойством стабильности.

Обеспечение автономности работы реализуется программой, путем считывания таких параметров как скорость, положение в пространстве, положение колес, линии, расстояния до объекта. Считывая значения параметров, робототехническая система выстраивает оптимальные алгоритмы передвижения. Под обеспечением свойства стабильности подразумевается создание условий корректной работы при любых условиях.

Это достигается за счет реализации правильности написанного алгоритма.

# Описание команды, распределение ролей, функций и обязанностей каждого участника команды.

В команде программисты выполняют задачи такие как:

* Разработка алгоритма
* Написание алгоритма
* Отладка алгоритма
* Исправление ошибок

В команде документоведы~ выполняют задачи такие как:

* Написанием постановки задач
* Написанием диаграмм (use case, state machine diagram, component diagram, sequence diagram)

В команде ответственный за моделирование должен:

* Смоделировать детали робота
* Напечатать смоделированные детали

В команде ответственный за конструирование должен:

* Собрать и настроить робота

**Состав команды Platon1212:**

1. Анисимов Михаил Александрович – программист.
2. Копытько Дарья Владимировна - ответственная за документацию.
3. Коршаков Андрей Алексеевич - ответственный за моделирование.
4. Пономарев Тимофей Русланович - ответственный за дистанционное управление роботом, программист.
5. Громыко Николай Александрович - ответственный за программирование и конструирование
6. Удалов Анатолий Андреевич – руководитель команды.

# Общее описание функций разработанного решения

**О сборке:**

Корпус напечатан полностью на 3d принтере из пластика. Для передвижения робота используется гусеничная база.

**О моторах:**

В нашем роботе используются движки dynamixel которые имеют два режима работы.

Первый это сервопривод, с его помощью мы можем проворачивать колесо на заданный градус.

Второй обычные моторы, с помощью которых колесо крутится с заданной ему скоростью.

**О камере:**

В нашем роботе установлена камера TrackingCam 3, которая упрощает задачу с поиском объектов и линий, путём того, что автоматически выделяет объект и возвращает координаты.

Так же, благодаря ей передается картинка оператору для ориентации.

**Захват**:

На данный момент устройство захвата груза находится на стадии доработки. Однако это не повлияло на результат выполнения демонстрационной работы. Планируется, что с помощью клешни мы зажимаем груз и перемещаем его. Зажим груза происходит с помощью Сервопривода.

# Описание используемых аппаратных и программных узлов, модулей, фреймворков и других инструментов.

Модуль управления OpenCR1.0[1] (открытого исходного кода модуля управления для ROS) - программируемый робототехнический контроллер с открытым исходным кодом со встроенным мощным MCU от ARM Cortex M-7.

Arduino IDE[2] — интегрированная среда разработки для Windows, MacOS и Linux, разработанная на Си и C ++, предназначенная для создания и загрузки программ на Arduino-совместимые платы, а также на платы других производителей.

Dynamixel[3] – сервоприводы с обратной связью, которые имеют два режима работы:

* сервопривод, с его помощью я могу поворачивать колесо на заданный градус
* обычные моторы, с помощью которых колесо крутится с заданной ему скоростью.

Модуль технического зрения TrackingCam[4] является сенсорным устройством для исследования окружающего пространства путем обработки и анализа изображения со встроенной видеокамеры. Модуль предназначен для применения с различными образовательными робототехническими наборами и может использоваться для создания роботов, способных распознавать и анализировать объекты по ряду признаков - цвету, размеру, форме и т.д.

Для работы с модулем не требуются фундаментальные знания в области обработки изображений, т.к. настройка и обучение модуля на поиск и сопровождение требуемого объекта выполняется автоматически. Процесс обучения модуля производится с помощью специального графического интерфейса, в котором пользователь выделяет и размечает по признакам анализируемые элементы.

Diagrams.net[5] (ранее draw.io) - это бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом, разработанное на HTML5 и JavaScript. Его интерфейс можно использовать для создания таких диаграмм, как блок-схемы, каркасы, UML-диаграммы, организационные диаграммы и сетевые диаграммы.

# Функциональное описание разработанного решение в виде UML-диаграмм:

Вариант использования (use case)[6] — последовательность действий, которые система или другая сущность могут выполнять в процессе взаимодействия с акторами изображена на рисунке 5.1.

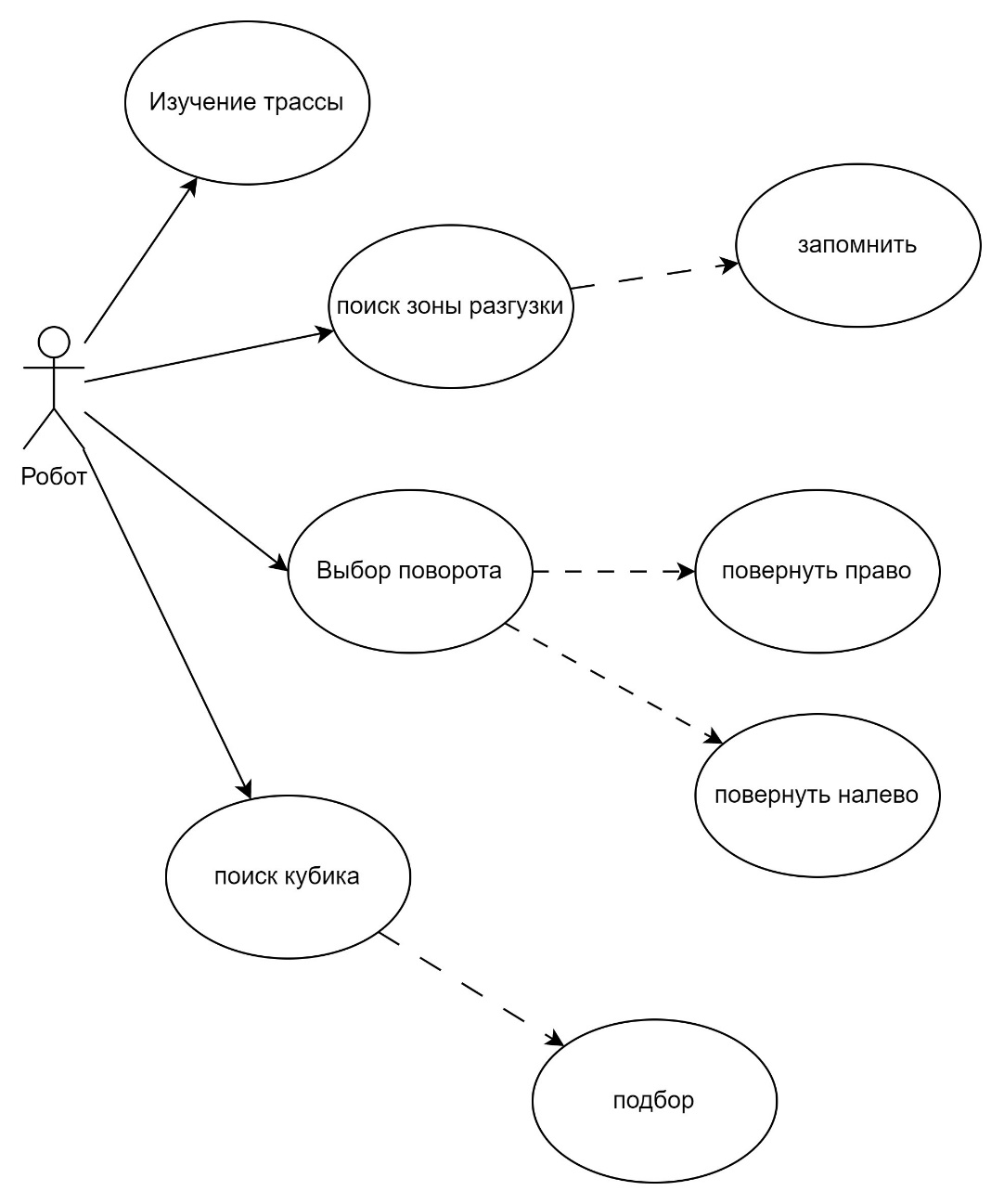
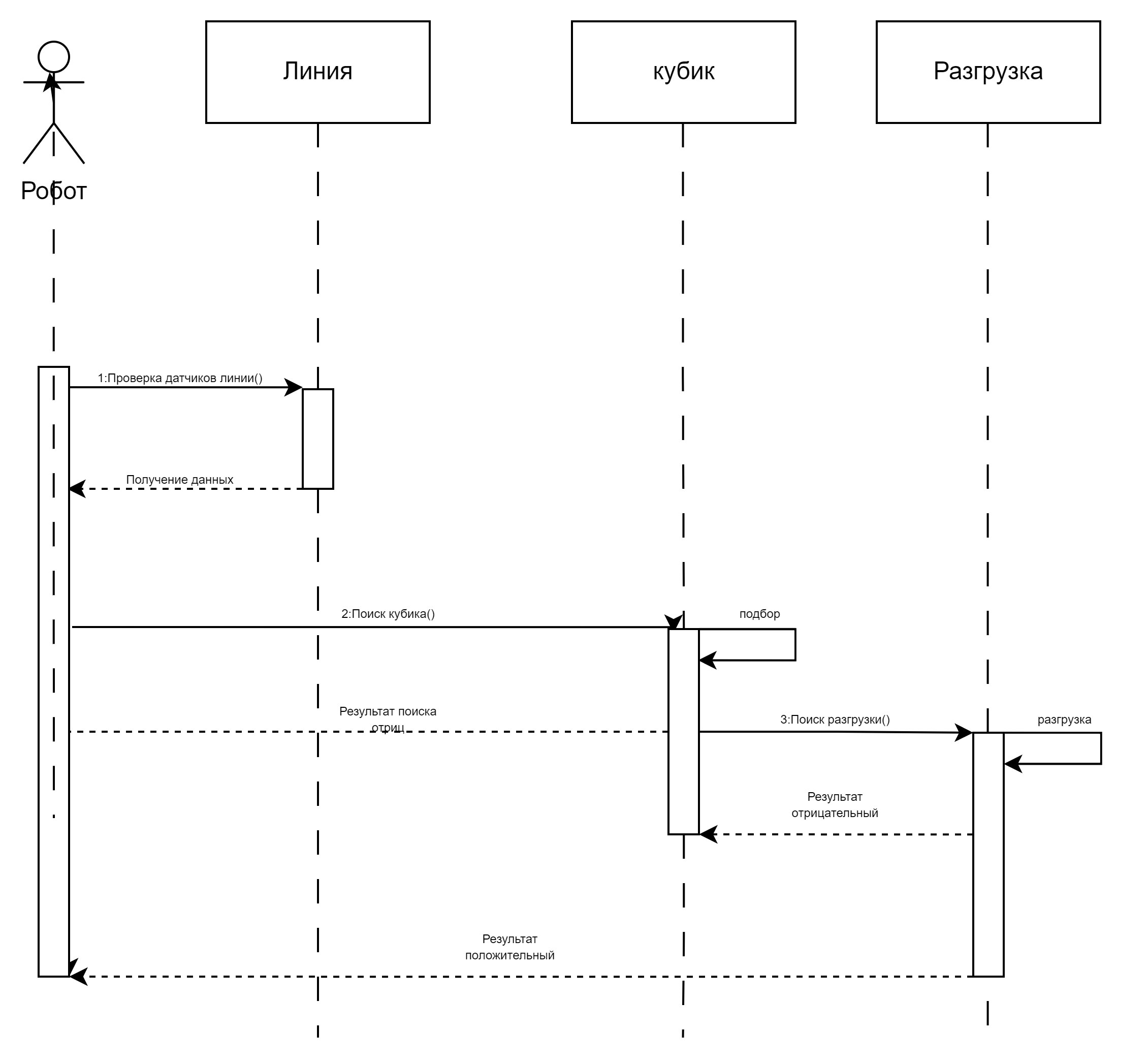


Рисунок 5.1 – Диаграмма использования

Диаграммы последовательностей[7], обычно используемые разработчиками, моделируют взаимодействия между объектами в едином сценарии использования. Они иллюстрируют, как различные части системы взаимодействуют друг с другом для выполнения функции, а также порядок, в котором происходит взаимодействие при выполнении конкретного случая использования. На рисунке 5.2 изображена диаграмма последовательности

Рисунок 5.2 – Диаграмма последовательности

На следующей странице представлена диаграмма компонентов нашего робота рисунок 5.3.

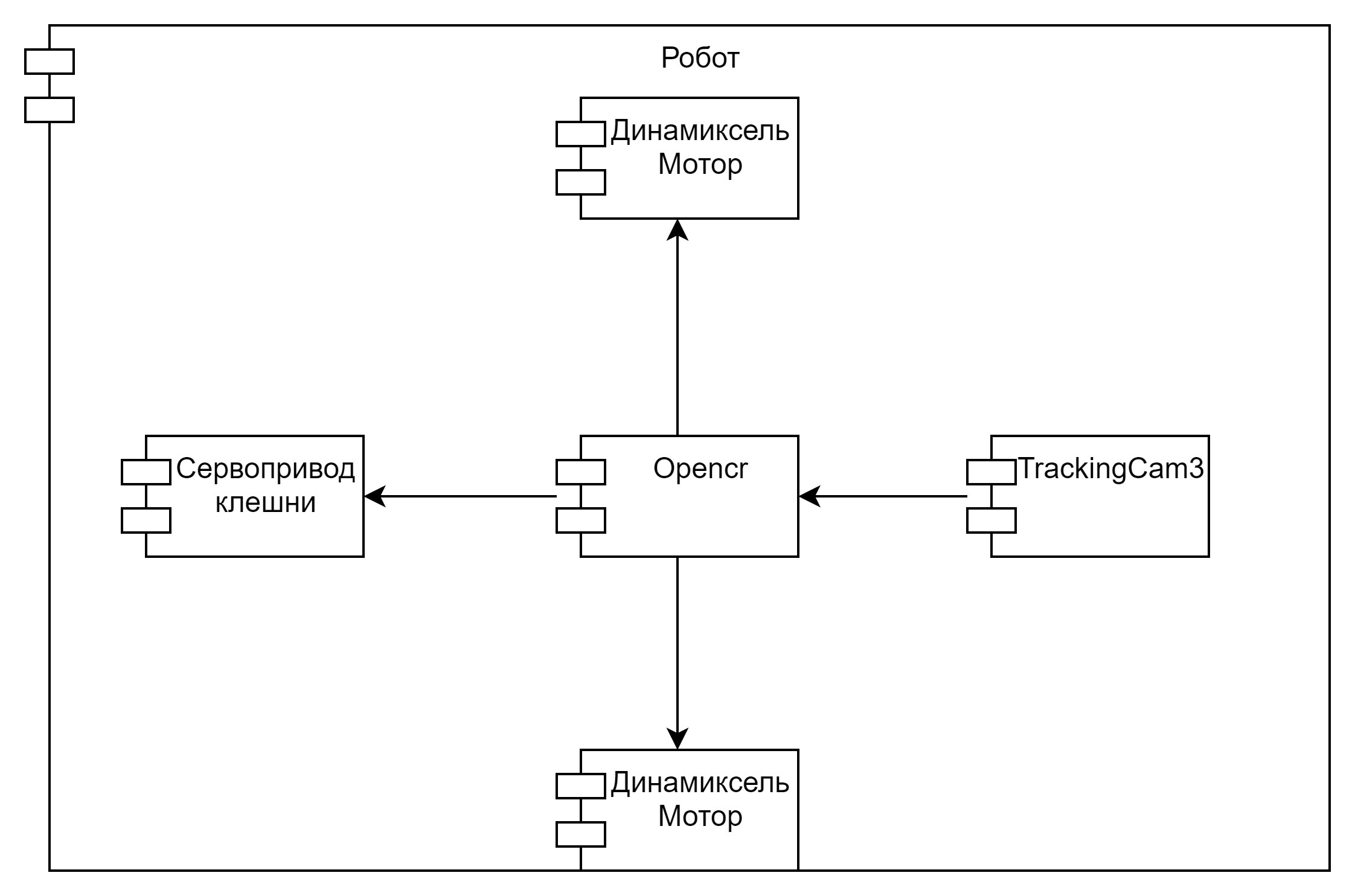


Рисунок 5.3 – Диаграмма компонентов

# Описание кинематической системы разработанного устройства в виде схемы, диаграмм, подробно описывающие все аспекты кинематического движения, реализующегося во время функционирования устройства.

Данная робототехническая система представлена в виде дифференциального привода, использующего в качестве передачи момента и частоты вращения на переднюю ось колёс гусеницы. Кинематическая двухмерная схема представлена на рисунке 6.1:

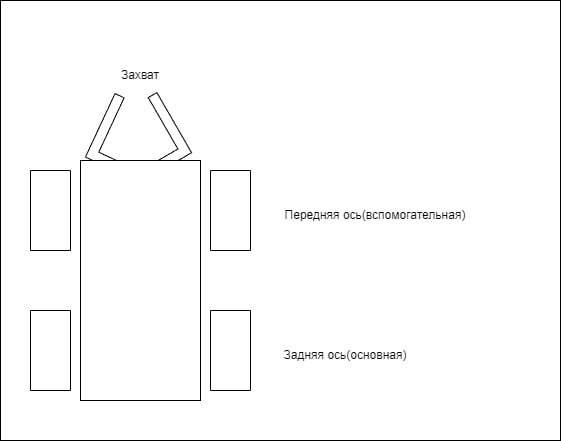


Рисунок 6.1 - Кинематическая двухмерная схема

# Алгоритм работы разработанного программного обеспечения в виде блок-схем.

На рисунке 7.1 представлен алгоритм работы робота

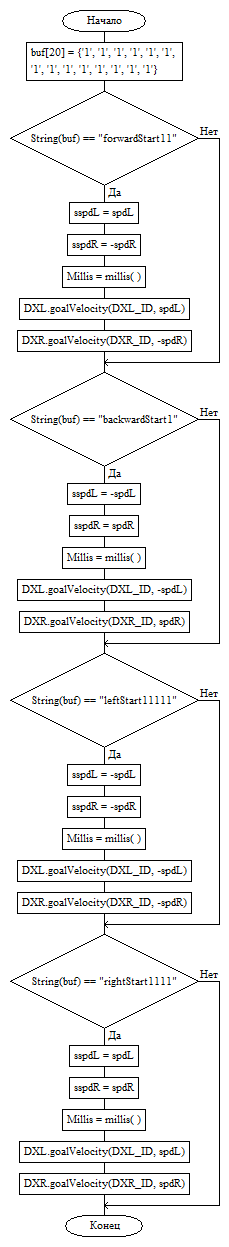


Рисунок 7.1 – Алгоритм работы робота

# Заключение, результаты работы, анализ функционирования разработанного устройства, предложения по возможному улучшению устройства.

Данная робототехническая станция в своём нынешнем виде имеет ряд недостатков, такие как отсутствие соосности (сносности — это расположение осей деталей машины) колесной базы и корректного управления захватом, которые не являются не исправимыми и в дальнейшем будут подвержены тщательной проработке, созданию и внедрению в созданную дифференциальную колесную базу.

# Список литературных источников.

# <https://emanual.robotis.com/docs/en/parts/controller/opencr10/>

1. <https://support.arduino.cc/hc/en-us/articles/360019833020-Download-and-install-Arduino-IDE>
2. <https://emanual.robotis.com/docs/en/dxl/ax/ax-12a/>
3. <https://robotgeeks.ru/collection/komplektuyuschie-dlya-avtonomnyh-robotov/product/modul-tehnicheskogo-zreniya-trackingcam>
4. <https://app.diagrams.net/>
5. https://en.wikipedia.org/wiki/Use\_case\_diagram
6. <https://ru.wikipedia.org/wiki/Диаграмма_последовательности>