|  |  |
| --- | --- |
| Изображение выглядит как эмблема, герб, нашивка, символ  Контент, сгенерированный ИИ, может содержать ошибки. | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ «Специальное машиностроение»

КАФЕДРА «Робототехнические системы и мехатроника»

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ***

***НА ТЕМУ:***

***Разработка мобильной платформы для обучения***

Студент СМ7-74Б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Волков А.О.**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Калиниченко С.В.**

(Подпись, дата) (И.О.Фамилия)

*2024 г.*

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_

(Индекс)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(И.О.Фамилия)

« \_\_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**

**на выполнение научно-исследовательской работы**

|  |  |
| --- | --- |
| По теме: | Разработка мобильной платформы для обучения |
|  | |
|  | |

Студент группы \_\_\_\_СМ7-74б\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Волков Андрей Олегович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

\_\_\_\_\_\_\_\_практическая\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Источник тематики (кафедра, предприятие, НИР) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

График выполнения НИР: 25% к 5 нед., 50% к 7 нед., 75% к 10 нед., 100% к 14 нед.

|  |  |
| --- | --- |
| ***Техническое задание*** | Разработать мобильную платформу для обучения |
|  | |

***Оформление научно-исследовательской работы:***

Расчетно-пояснительная записка на \_\_\_\_\_ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания « 17 » сентября 2024 г.

**Руководитель НИР**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Калиниченко С.В.**

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Волков А.О.**

(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

# РЕФЕРАТ

Объект выпускной квалификационной работы: Разработка мобильной платформы для обучения.

Расчетно-пояснительная записка к выпускной квалификационной работе содержит … страницы машинописного текста, … рисунков, … таблиц. При выполнении дипломного проекта использовались следующие программы: Microsoft Word, Компас 3D V20, Matlab, Mathcad, Inventor 2025.

В данной расчетно-пояснительной записке приведены:

1. Анализ задачи и техническое задание;
2. Структура системы управления;
3. Подбор комплектующих элементов;
4. Расчет системы управления приводом;
5. Разработка алгоритма управления;
6. Моделирование алгоритма управления;
7. Конструирование узла линейного привода

Ключевые слова: мобильная платформа, привод, электродвигатель, система управления, лабиринт, алгоритмы, моделирование, SLAM-навигация.

СОДЕРЖАРНИЕ

[РЕФЕРАТ 3](#_Toc192481065)

[ВВЕДЕНИЕ 5](#_Toc192481066)

[1. ОБЗОР ПРОЕКТА 6](#_Toc192481067)

[1.1 Актуальность задачи 6](#_Toc192481068)

[1.2 Анализ существующих мобильных платформ 7](#_Toc192481069)

[2. ТЕОРЕТИЧЕСКОЕ ПРОЕКТИРОВАНИЕ 11](#_Toc192481070)

[2.1 Постановка задачи 11](#_Toc192481071)

[2.2 Демонстрационная задача для робота 11](#_Toc192481072)

[2.3 Технические требования 13](#_Toc192481073)

[2.4 Кинематика робота 14](#_Toc192481074)

[2.5 Структура системы управления роботом 23](#_Toc192481075)

[3. ВЫБОР КОМПОНЕНТОВ МОБИЛЬНОЙ ПЛАТФОРМЫ 25](#_Toc192481076)

[3.1 Подбор компонентов для нижнего уровня управления 25](#_Toc192481077)

[3.1.1 Электродвигатель 25](#_Toc192481078)

[3.1.2 Энкодер 26](#_Toc192481079)

[3.1.3 Драйвер привода 27](#_Toc192481080)

[3.2 Подбор компонентов для среднего уровня управления 28](#_Toc192481081)

[3.3 Подбор компонентов для верхнего уровня управления 29](#_Toc192481082)

[3.4 Подбор вспомогательный компонентов 30](#_Toc192481083)

[3.4.1 Аккумулятор 30](#_Toc192481084)

[3.4.2 Колеса Илона(меканум) 31](#_Toc192481085)

[3.4.3 Лидар 32](#_Toc192481086)

[3.1.4 Датчик ориентации 33](#_Toc192481087)

[3.1.5 Камера 35](#_Toc192481088)

[4. КОНСТРУКЦИЯ ПРИВОДА И МОБИЛЬНОЙ ПЛАТФОРМЫ 36](#_Toc192481089)

[5. ОБЗОР ВОЗМОЖНОГО ПОДХОДА К ОБУЧЕНИЮ НА МОБИЛЬНОЙ ПЛАТФОРМЕ 38](#_Toc192481090)

[ЗАКЛЮЧЕНИЕ 43](#_Toc192481091)

[Список литературы 44](#_Toc192481092)

[ПРИЛОЖЕНИЕ А 46](#_Toc192481093)

[ПРИЛОЖЕНИЕ Б 47](#_Toc192481094)

# ВВЕДЕНИЕ

Несмотря на значительный прогресс в области робототехники, проблема подготовки специалистов, обладающих не только теоретическими знаниями, но и практическими навыками работы с реальным оборудованием, остаётся актуальной. Существующие образовательные платформы зачастую либо слишком просты и не позволяют изучать сложные алгоритмы, либо слишком дороги и сложны в освоении. В связи с этим замысел данной дипломной работы является разработка и создание доступной и функциональной мобильной платформы, предназначенной для обучения студентов основам робототехники. Платформа позволит студентам на практике изучить вопросы построения систем управления движением, навигации и взаимодействия с окружающей средой, тем самым обеспечивая более качественное освоение информации.

Задачей работы является разработка мобильной платформы для образовательных целей, включающая анализ существующих решений, проектирование конструкции, создание системы управления и разработку алгоритмов ориентации в пространстве на основе SLAM-навигации.

# ОБЗОР ПРОЕКТА

## Актуальность задачи

Актуальность данной дипломной работы обусловлена критической необходимостью улучшения подготовки специалистов в области робототехники, особенно в контексте ограниченности доступных и эффективных образовательных платформ и недостатка практических навыков у выпускников.

Несмотря на экспоненциальный рост интереса к робототехнике и её широкое распространение в различных отраслях промышленности, логистики и науки, существующие образовательные подходы зачастую не соответствуют современным требованиям рынка труда. Многие образовательные платформы для робототехники имеют ряд существенных ограничений, а именно:

**Высокая стоимость:** значительная часть существующих платформ является дорогостоящей, что делает их недоступными для многих учебных заведений, особенно в условиях ограниченного финансирования.

**Сложность освоения:** некоторые платформы требуют глубоких знаний в области программирования и электроники, что создает барьер для студентов с разным уровнем подготовки.

**Отсутствие поддержки современных технологий:** некоторые платформы не поддерживают современные технологии, такие как ROS (операционная система для роботов), SLAM (одновременная локализация и картографирование) или продвинутые методы управления движением.

Данная дипломная работа направлена на решение указанных проблем путем разработки доступной, функциональной и гибкой мобильной роботизированной платформы, специально разработанной для образовательных целей.

Разработанная платформа позволит значительно повысить уровень практической подготовки студентов, обеспечив их необходимыми навыками и знаниями для успешной работы в быстро развивающейся области робототехники.

## Анализ существующих мобильных платформ

Современное образование всё чаще обращается к использованию передовых технологий для подготовки специалистов в области робототехники, программирования и искусственного интеллекта. Одним из ключевых инструментов, способствующих эффективному обучению, являются мобильные роботизированные платформы. Они предоставляют студентам и исследователям возможность изучать теоретические основы на практике, разрабатывать и тестировать алгоритмы управления, навигации и взаимодействия с окружающей средой. Такие платформы становятся незаменимыми в образовательных программах, направленных на развитие навыков проектирования, программирования и работы с современным оборудованием. Рассмотрим несколько примеров таких платформ.

**KUKA youBot —** Мобильная платформа KUKA youBot представляет собой высокоманёвренное роботизированное решение, оснащённое четырьмя независимо управляемыми меканум-колёсами. Она обладает максимальная скоростью 0.8 м/c и грузоподъемностью 20 кг, достаточными для работы с легкими грузами в помещениях с ровным покрытием. Платформа обладает открытой архитектурой и совместима с ROS (Robot Operating System), что упрощает разработку и интеграцию алгоритмов управления. Она поддерживает подключение различных сенсоров, таких как лидары, камеры и IMU, для навигации и локализации в пространстве с использованием алгоритмов SLAM. Питание осуществляется от литий-ионного аккумулятора (напряжение 24 В), обеспечивающего несколько часов автономной работы. Платформа отличается модульностью, что позволяет адаптировать её под различные задачи, а также легко отсоединять манипулятор для использования в качестве самостоятельной мобильной единицы. Основные области применения включают образование, исследования и прототипирование автономных систем. Размеры ДхШхВ 500х400х200 мм.

Однако платформа имеет ряд недостатков. Во-первых, её высокая стоимость (на данный момент составляет примерно 2 млн рублей), во-вторых, отсутствие встроенного лидара, что вызывает дополнительные сложности при его интеграции в систему, в-третьих, как отмечают пользователи существует затруднительностьработы для неопытных пользователей из-за сложной системы управления.



1. KUKA youBot

Учебный робот DJI RoboMaster S1 EP представляет собой инновационное образовательное решение от компании DJI, предназначенное для изучения основ робототехники, программирования и искусственного интеллекта. Платформа оснащена всенаправленным шасси с колёсами типа «Илона». Робот также оборудован гибким стабилизатором и системой передачи изображения в режиме «вид от первого лица» (FPV).

RoboMaster S1 EP поддерживает языки программирования Scratch 3.0 и Python, что делает его доступным для обучающихся способствуя развитию навыков программирования. Платформа также совместима с оборудованием с открытым исходным кодом, таким как Micro:bit , Arduino и Raspberry Pi, а также поддерживает платформы DJI SDK и NVIDIA Jetson Nano, что позволяет изучать принципы искусственного интеллекта и компьютерного зрения.

Робот поддерживает подключение сторонних датчиков благодаря четырём сенсорным адаптерам, входящим в комплект. Платформа передвигается с максимальной линейной скоростью 3,5 м/c и имеет грузоподъемность 3 кг. Время работы до 40 минут. Размеры ДхШхВ 320×240×270 мм.

Недостатки робота **DJI RoboMaster S1 EP**, связанные с отсутствием лидара, поддержки ROS и ограниченным временем работы от аккумуляторов, могут существенно повлиять на его функциональность и применение в образовательных задачах.



1. RoboMaster EP

**MYAGV**—это компактная и высокоманёвренная мобильная платформа, разработанная для образовательных целей. Она представляет собой автономного мобильного робота (AMR), способного выполнять задачи по транспортировке грузов, навигации и взаимодействию с окружающей средой. Платформа оснащена всенаправленными колёсами, и имеет максимальную линейную скорость 0,9 м/c. MYAGV поддерживает интеграцию с различными сенсорами, такими как лидары, камеры и ультразвуковые датчики, что позволяет ей эффективно ориентироваться в пространстве и избегать препятствий с использованием алгоритмов SLAM (Simultaneous Localization and Mapping). Платформа совместима с ROS (Robot Operating System), что упрощает разработку и тестирование алгоритмов управления, навигации и взаимодействия с другими системами. MYAGV питается от литий-полимерного аккумулятора (напряжение 12 В), обеспечивающего до 2 часов автономной работы. Платформа имеет грузоподъемность 2 кг и обладает размерами ДхШхВ 311×236×130 мм.

Изображение выглядит как машина, автокомпонент, игрушка, колесо

Контент, сгенерированный ИИ, может содержать ошибки.

1. **MYAGV**

# ТЕОРЕТИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

## Постановка задачи

**Цель:** Разработка конструкции и программных решений образовательной мобильной роботизированной платформ c возможностью совместной работы с манипулятором.

**Задачи:**

1. Исследование и сравнительный анализ существующих мобильных платформ, включая их конструктивные особенности и функциональные возможности.
2. Создание демонстрационного поля и составления технического задания.
3. Разработка структурной схемы мобильной платформы.
4. **Проектирование конструкции мобильной платформы и её компонентов**.
5. Выбор электроники и дополнительных компонентов.
6. **Создание возможного подхода для обучения на мобильной платформе.**
7. Разработка алгоритмов управления для демонстрационной задачи на основе SLAM-навигации и их симуляция в виртуальной среде.
8. Проведение регулировочного расчета для одного привода шасси.

Цель и задачи выпускной квалифиционной работы

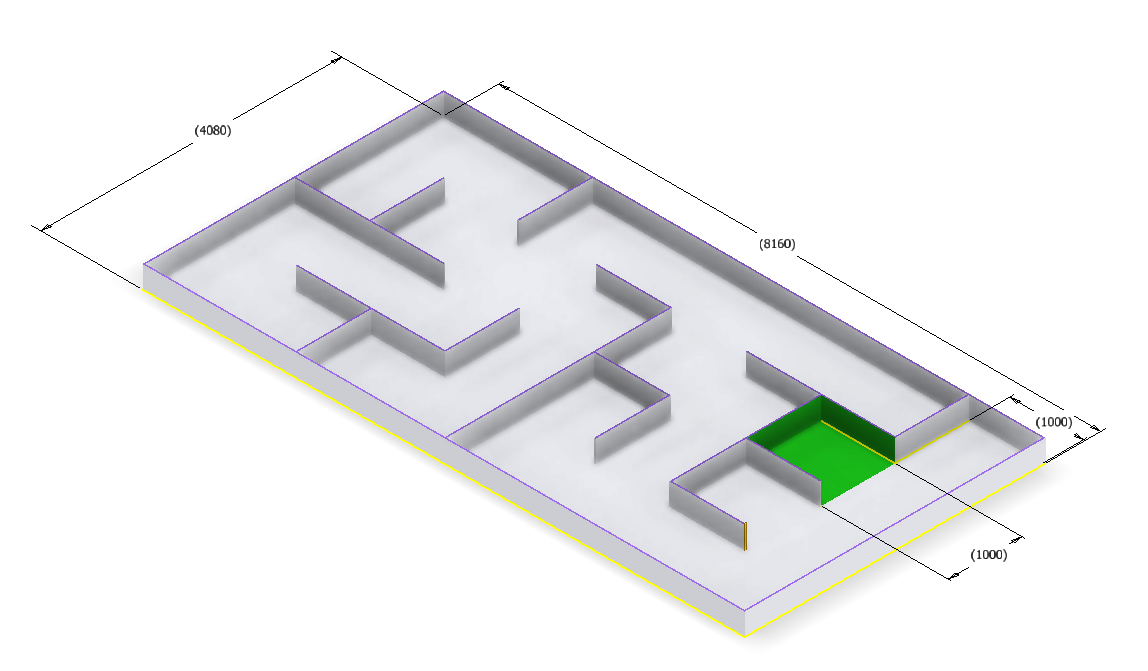
## Демонстрационная задача для робота

В рамках данного исследования, являющегося частью проекта по разработке многофункциональной образовательной платформы для изучения принципов взаимодействия с мобильными платформами, будет рассмотрена и решена конкретная задача, демонстрирующая эффективность применения разработанной системы.

Задача состоит в следующем:

* Сначала мобильная платформа передвигается по лабиринту, представленному на рисунке, для построения карты. Затем в лабиринт добавляются в случайные места пластмассовые кубы с Aruco маркерами и робота устанавливают в изначальное положение.
* Мобильная платформа перемещается по лабиринту и при обнаружении куба подъезжает к нему таким образом, чтобы он оказался в рабочей зоне манипулятора. После этого платформа посылает сигнал для запуска алгоритма манипулятора.
* Манипулятор захватывает куб и кладет в лоток установленном на мобильной платформе и затем подает сигнал о дальнейшем движении.
* Мобильная платформа передвигается по всему лабиринту исследуя каждую его часть и заканчивает работу по завершению объезда всего лабиринта.

Лабиринт представляет собой замкнутое пространство с минимальном расстоянием между параллельными стенками 1000 мм. В зеленой зоне робот находится в исходном положении.



1. Демонстрационное поле

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

1. Блок схема алгоритма работы робота

Представленная блок-схема описывает взаимодействие мобильной платформы и манипулятора в рамках демонстрационной задачи. Последующие разделы посвящены более подробному описанию работы мобильной платформы.

## Технические требования

На основе анализа существующих мобильных платформ были сформулированы основные технические характеристики, которые необходимо учитывать при разработке новой системы:

* Автономность 2 часа
* Максимальная скорость 1 м/c
* Наличие лидара
* Возможность подключения большого количества периферии
* Напряжение не выше 24 В

Основные технические требования к мобильной платформе, помимо уже перечисленных, в значительной степени определяются характеристиками интегрируемого манипулятора. В частности, для обеспечения надежной работы с манипулятором весом около 5 кг, а также с учетом возможности установки дополнительного оборудования грузоподъемность мобильной платформы должна составлять не менее 8 кг. Габариты основания манипулятора не должны превышать 150x150 мм.

## Кинематика робота

В качестве движителя для реализации дипломного проекта были выбраны всенаправленные колеса типа «Меканум» (колеса Илона). Такое решение обусловлено необходимостью обеспечения высокой маневренности и всенаправленности движения в условиях лабиринта, а также широким распространением колес Меканум в образовательной робототехнике.

Колеса Меканум представляют собой конструкцию, состоящую из центрального диска с расположенными по окружности роликами, ориентированными под углом 45 градусов к оси вращения колеса. Такая конструкция позволяет мобильной платформе, оснащенной четырьмя колесами Меканум, двигаться в любом направлении — вперед, назад, вбок и вращаться вокруг своей оси — без предварительного разворота. Главное преимущество колес Меканум заключается в обеспечении высокой маневренности и все направленности движения, что делает их идеальными для работы в ограниченном пространстве, где требуется точное позиционирование и быстрое изменение траектории. Эта способность позволяет роботу эффективно выполнять такие задачи, как работа в узких коридорах и обход препятствий без необходимости сложных маневров.

При использовании колёс Mecanum необходимо учитывать ряд конструктивных ограничений. Для корректной реализации всенаправленного движения колёса должны быть установлены в соответствии с представленным рисунком, при этом продолжения осей вращения роликов должны пересекаться в центре платформы. Это требование необходимо для точной реализации кинематики движения робота, что будет учтено при дальнейшем проектировании рамы.

Изображение выглядит как снимок экрана, дизайн

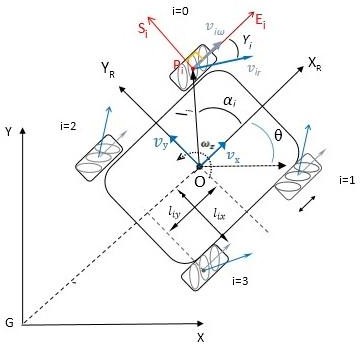
Контент, сгенерированный ИИ, может содержать ошибки.

1. Схема установки Меканум колес

Перейдем к анализу конкретных кинематических соотношений для заданной конфигурации платформы

Параметры и скорости системы определяются следующим образом:

* x, y, θ, положение робота (x, y) и угол его ориентации θ (угол между X и 𝑋𝑅);
* XGY, неподвижная система координат; x, y - координаты опорной точки O в неподвижной системе координат;
* 𝑋𝑅𝑂𝑌𝑅 базовая система координат робота; система координат, связанная с движением центра тела;
* 𝑆𝑖𝑃𝑖𝐸𝑖 система координат i-го колеса в центральной точке колеса 𝑃𝑖;
* 𝑃𝑖 = {𝑋𝑃𝑖 , 𝑌𝑃𝑖 } центр оси вращения колеса i;
* это вектор, который указывает расстояние между центром робота и центром i-го колеса;
* 𝑙𝑖x ,𝑙𝑖y ,𝑙𝑖z половина расстояния между передними колесами и liy половина расстояния между передним колесом и задними колесами;
* 𝑙𝑖 расстояние между колесами и основанием (центр робота 0);
* 𝑟𝑖 обозначает радиус i-го колеса (расстояние от центра колеса до центра ролика);
* 𝑟𝑟 обозначает радиус роликов на колесах;
* 𝛼𝑖 угол между OPᵢ и XR;
* 𝛽𝑖 угол между Sᵢ и XR;
* 𝛾𝑖 угол между 𝑣𝑖𝑟 и Eᵢ;
* 𝜔𝑖 [*рад/с*], угловая скорость колес;
* 𝑣𝑖𝜔 [𝑚/𝑠], (𝑖 = 0,1,2,3) ∈ 𝑅 это вектор скорости, соответствующий оборотам колеса
* 𝑣𝑖𝑟 скорость ролика в колесе i;
* [𝑤𝑠𝑖 𝑤𝐸𝑖 ωi ]T обобщенная скорость точки 𝑃𝑖 в системе координат 𝑆𝑖𝑃𝑖𝐸𝑖;
* [𝑣𝑠𝑖 𝑣𝐸𝑖 ωi ]T обобщенная скорость точки 𝑃𝑖 в системе координат 𝑋𝑅𝑂𝑌𝑅;
* 𝑣x, 𝑣y линейная скорость робота;
* 𝑤z [рад/с]угловая скорость робота;



1. Схема робота на четырех Меканум колесах

Согласно рисунку 7(b), мы можем рассчитать скорость колеса i и тангенциальную скорость свободного ролика, прикрепленного к колесу, касающемуся пола:

Изображение выглядит как диаграмма, линия

Контент, сгенерированный ИИ, может содержать ошибки.

1. Параметры i-го колеса

Согласно рисунку 7(b) и с учетом уравнений (уравнение 1), скорость колеса i в системе координат 𝑆𝑖𝑃𝑖𝐸𝑖 может быть выведена из:

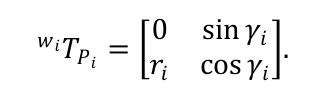
Изображение выглядит как текст, Шрифт, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.

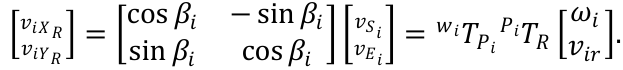
Изображение выглядит как текст, Шрифт, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Матрица преобразования от скоростей i-го колеса к его центру:



Согласно рисунку 7(a) и рисунку 6, скорость центра колеса, переведенная в систему координат 𝑋𝑅𝑂𝑌𝑅 , может быть достигнута с помощью уравнения 4.



Тогда матрица преобразования от центра i-го колеса к системе координат робота может быть получена из уравнения 5.

Изображение выглядит как Шрифт, текст, белый, типография

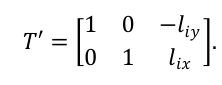
Контент, сгенерированный ИИ, может содержать ошибки.

Поскольку движение робота является плоским, мы также имеем:

Изображение выглядит как текст, Шрифт, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Где:



Изображение выглядит как Шрифт, текст, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.Из (уравнение 3) и (уравнение 5) можно получить обратную кинематическую модель:

Изображение выглядит как текст, Шрифт, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, Шрифт, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.Поскольку rᵢ ≠ 0, 0 < [γᵢ] < π/2, det( ) ≠ 0, det( ) ≠ 0, следовательно, объединяя уравнения 4 и 6, базовая скорость робота (в точке O), связанная с вращательной скоростью i-го колеса, может быть получена из уравнения 9.

Изображение выглядит как текст, Шрифт, белый, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Согласно уравнению 3 и уравнению 4, существует связь между переменными в каждой системе координат колес робота и его центром. И с обратной кинематикой скорость системы может быть получена путем реализации линейной скорости 𝑣𝑖𝑟 и угловой скорости ωᵢ i-го колеса в уравнении 10 и наоборот в уравнении 11.

Изображение выглядит как текст, Шрифт, белый, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, Шрифт, число, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.



Где

Изображение выглядит как текст, Шрифт, белый, типография

Контент, сгенерированный ИИ, может содержать ошибки.

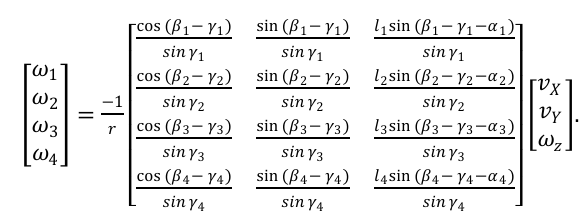
Учитывая тот факт, что 𝑙𝑖x = 𝑙𝑖 cos 𝛼𝑖 и 𝑙𝑖y = 𝑙𝑖 sin 𝛼𝑖, и предполагая, что колеса имеют одинаковый размер, матрица преобразования:

Изображение выглядит как текст, Шрифт, линия, белый

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, Шрифт, линия, рукописный текст

Контент, сгенерированный ИИ, может содержать ошибки.

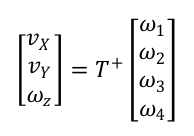
Поскольку существует связь между независимыми переменными 𝑣𝑖𝑟 и ωᵢ в каждом соединении, а также между угловой и линейной скоростью системы, предполагая отсутствие проскальзывания колеса по земле, обратную кинематику системы можно получить из уравнения 14.

В уравнение 15 показана матрица Якоби для обратной кинематической системы:

Изображение выглядит как текст, Шрифт, белый, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Для прямой задачи кинематики и в соответствии с уравнением 10:

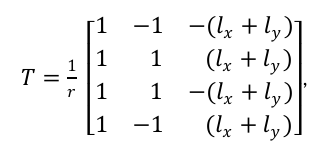


Проанализируем параметры конфигурации робота, представленной на рисунке 6. В дальнейшем данная конфигурация будет использоваться в качестве основы для рассмотрения в рамках дипломного проекта.

Изображение выглядит как число, Шрифт, прямоугольный, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Заменяя параметры из таблицы в матрицу (уравнение 15) и учитывая уравнение 14:



Изображение выглядит как диаграмма, линия, Шрифт, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Согласно уравнениям (10) и (11) для прямой и обратной кинематики, есть:

Изображение выглядит как Шрифт, диаграмма, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как Шрифт, рукописный текст, текст, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как Шрифт, текст, диаграмма, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Продольная скорость:

Изображение выглядит как Шрифт, типография, рукописный текст, каллиграфия

Контент, сгенерированный ИИ, может содержать ошибки.

Поперечная скорость:

Изображение выглядит как рукописный текст, Шрифт, каллиграфия, белый

Контент, сгенерированный ИИ, может содержать ошибки.

Угловая скорость:

Изображение выглядит как Шрифт, рукописный текст, каллиграфия, типография

Контент, сгенерированный ИИ, может содержать ошибки.

Результирующую скорость и ее направление в стационарной координатной оси (x, y, z) можно получить из следующих уравнений (уравнение 25, 26):

Изображение выглядит как Шрифт, белый, текст, символ

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как Шрифт, рукописный текст, линия, число

Контент, сгенерированный ИИ, может содержать ошибки.

## Структура системы управления роботом

Система управления организована по трехуровневой иерархической схеме:

**Верхний уровень (управляющий компьютер):** отвечает за ключевые задачи SLAM-навигации — локализацию и построение карты — с использованием данных лидара и датчика ориентации. Также управляет приводами через микроконтроллер, определяя требуемое движение.

**Средний уровень (микроконтроллер):** выполняет роль связующего звена, обеспечивая взаимодействие между управляющим компьютером и приводами.

**Нижний уровень (аппаратная часть):** состоит из четырех мотор-редукторов постоянного тока с энкодерами, отвечающих за движение, и лидара, обеспечивающего сбор данных об окружающей среде.

Изображение выглядит как текст, диаграмма, снимок экрана, План

Контент, сгенерированный ИИ, может содержать ошибки.

1. Функциональная схема системы управления

# ВЫБОР КОМПОНЕНТОВ МОБИЛЬНОЙ ПЛАТФОРМЫ

Мобильная платформа состоит из рамы, лидара, компьютера, микроконтроллера, аккумуляторного отсека и 4 приводов, которые включают в себе драйвер привода, двигатель постоянного тока с планетарном редуктором и магнитного энкодер установленного на валу двигателя

## Подбор компонентов для нижнего уровня управления

### Электродвигатель

Согласно требованиям технического задания, грузоподъёмность платформы должна составлять не менее 15 кг. В рамках данного проекта используются всенаправленные колёса (колёса Илона). При выборе учтены особенности движения платформы, рассмотренные в разделе теоретического проектирования. В частности, при расчёте параметров электродвигателя был принят во внимание наихудший сценарий, при котором функционируют только два двигателя, что соответствует движению робота по диагонали. Данный режим работы выбран в качестве расчётного, поскольку он создаёт максимальную нагрузку на электродвигатели.

На основании проведённых расчётов, представленных в приложении А, был выбран двигатель постоянного тока CHP-42GP-775 ABHL с планетарным редуктором производства компании CHIHAI MOTOR.

Изображение выглядит как кабель, цилиндр, соединитель, батарея

Контент, сгенерированный ИИ, может содержать ошибки.

1. CHP-42GP-775 ABHL

Характеристики ЭД:

Номинальная мощность ;

Номинальная частота вращения ;

Номинальный момент ;

Напряжение ;

Номинальный ток ;

Максимальный ток ;

Характеристики планетарного редуктора:

Передаточное число: 1:25;

Материал шестерни: аналог стали 20ХНЗА.

### Энкодер

Для обеспечения управления двигателем с заданной скоростью требуется использование энкодера. В данном случае определение точного положения вала не является обязательным условием, поэтому целесообразно применение инкрементного энкодера. Из каталога производителя был выбран инкрементный магнитный энкодер с разрешением 425 импульсов на оборот. Указанный энкодер установлен непосредственно на валу двигателя.

Изображение выглядит как снимок экрана, круг, текст, мультфильм

Контент, сгенерированный ИИ, может содержать ошибки.

1. Энкодер

### Драйвер привода

Драйвер PWM Motor Driver Module – выполняет преобразование низковольтных управляющих сигналов, поступающих от микроконтроллера, в высоковольтные и сильноточные сигналы, необходимые для обеспечения питания электродвигателя. В рамках данной системы используется напряжение 22,2 В с максимальным током 24 А. Для реализации указанных параметров выбран драйвер на основе H-моста производства компании CHIHAI MOTOR, оснащённый датчиком тока. Взаимодействие с микроконтроллером осуществляется посредством широтно-импульсной модуляции (ШИМ).

Изображение выглядит как электроника, Электронный компонент, Компонент схемы, Пассивный компонент цепи

Контент, сгенерированный ИИ, может содержать ошибки.

1. Драйвер привода PWM Motor Driver Module

## Подбор компонентов для среднего уровня управления

Микроконтроллер является управляющим элементом системы. Он обрабатывает данные от энкодера, формирует управляющие сигналы для драйвера и реализует алгоритмы управления, такие как регулировка скорости или выполнение заданного движения.

Для реализации выбран микроконтроллер STM32F407VET6, поскольку он обладает рядом преимуществ, таких как доступная стоимость, широкое распространение на рынке и поддержка подключения до четырёх энкодеров. Это позволяет осуществлять управление четырьмя приводами одновременно, что соответствует требованиям проекта и обеспечивает необходимую функциональность системы.

STM32F407VET6 — это высокопроизводительный 32-разрядный микроконтроллер (MCU) на базе ядра ARM Cortex-M4F (с FPU) от STMicroelectronics. Предназначен для широкого спектра приложений, требующих высокой скорости обработки данных, гибкости и развитой периферии.

Основные характеристики:

Ядро: ARM Cortex-M4F

Тактовая частота: до 168 МГц

Память: 512 КБ флэш-памяти, 192 КБ SRAM.

Периферия:

Таймеры: Широкий набор таймеров общего назначения.

Коммуникации: USART, SPI, I2C, CAN, USB OTG FS, Ethernet MAC.

АЦП/ЦАП: 3 12-битных АЦП (до 24 каналов), 2 12-битных ЦАП.

GPIO: 114 линий ввода/вывода.

Напряжение питания: 1,8 В — 3,6 В.

Изображение выглядит как Электронный компонент, Компонент схемы, Электронная техника, электроника

Контент, сгенерированный ИИ, может содержать ошибки.

1. STM32F407VET6

## Подбор компонентов для верхнего уровня управления

Для управления всей системы был выбран недорогой одноплатный компьютер Raspberry Pi 5. Он представляет собой усовершенствованную версию своих предшественников, предлагая более высокую производительность, улучшенную энергоэффективность и расширенные возможности.

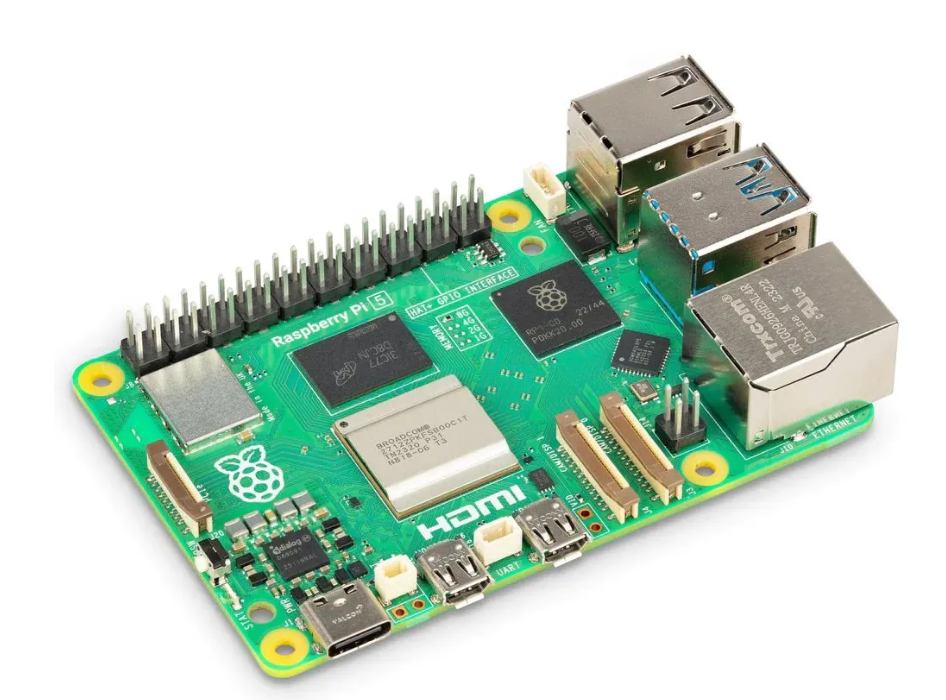
Основные характеристики:

Процессор: 64-битный четырёхъядерный процессор ARM Cortex-A76 с тактовой частотой до 2,4 ГГц.

Оперативная память: 8 ГБ.

Периферия и интерфейсы:

* 2 × порта USB 3.0 (до 5 Гбит/с) и 2 × порта USB 2.0.
* 2 × порта HDMI с поддержкой 4K при 60 Гц.
* Интерфейс GPIO (40 контактов) для подключения датчиков и внешних устройств.
* Поддержка Wi-Fi и Bluetooth 5.0.



1. Raspberry Pi 5

## Подбор вспомогательный компонентов

### Аккумулятор

Питание мобильной платформы будет производиться за счет 2 аккумулятор с номинальным напряжением 22,2 вольта соединенных параллельно.

HRB 6S 6000 — это литий-полимерный (LiPo) аккумулятор.

Конфигурация ячеек: 6S — аккумулятор состоит из 6 последовательно соединённых ячеек.

Номинальное напряжение: 22,2 В (номинальное напряжение одной ячейки — 3,7 В; 3,7 В × 6 = 22,2 В).

Ёмкость: 6000 мАч (6,0 А·ч)



1. HRB 6s 6000 мАч

### Колеса Илона(меканум)

Вес: 150 г (одного колеса)

Диаметр внешнего колеса: 97 мм.

Несущая способность: выдерживает 25 кг (четырех вместе)



1. Колеса Илона

### Лидар

RP-Lidar A1 — это недорогой 2D-лидар, предназначенный для сканирования окружающей среды и создания карт местности в реальном времени. Он часто используется в робототехнике, навигации и картографии. Вот основные характеристики RP-Lidar A1:

**Принцип работы:** Time of Flight (ToF) — измерение времени прохождения лазерного луча до объекта и обратно.

**Диапазон сканирования:** 0,15–12 метров (в зависимости от отражающей способности поверхности).

**Частота сканирования:** 5,5 Гц (оборотов в секунду).

**Угловое разрешение:** около 1 градуса (зависит от частоты сканирования и скорости вращения).

**Количество измерений в секунду:** около 2000–4000 точек в секунду (зависит от скорости вращения).

**Длина волны лазера:** 780 нм (инфракрасный, невидимый).

**Скорость вращения:** регулируемая, обычно около 5-15 оборотов в секунду.

**Поле зрения:** 360 градусов.

**Напряжение питания:** 5 В постоянного тока.

**Интерфейс:** USB (для передачи данных и питания).

**Размеры:** диаметр около 98 мм, высота около 60 мм.

**Программное обеспечение и совместимость:**

**SDK:** предоставляется SDK для разработки собственных приложений.

**ROS:** широко поддерживается и имеет готовые пакеты для интеграции.

Основное преимущество данного лидара, что он имеет интеграцию с ROS.

Изображение выглядит как круг

Контент, сгенерированный ИИ, может содержать ошибки.

1. RP-Lidar A1

### Датчик ориентации

В систему SLAM-навигации будет интегрирована дополнительная одометрия на базе датчиков ориентации для повышения точности оценки траектории и минимизации дрейфа.

BNO055 - это 9-осевой (акселерометр, гироскоп и магнитометр). Его ключевая особенность - встроенный микроконтроллер и алгоритм Sensor Fusion, который предоставляет готовые данные об ориентации без необходимости сложной обработки данных с отдельных сенсоров. Вот основные технические характеристики BNO055:

**Общие характеристики:**

**Тип:** 9-осевой абсолютный датчик ориентации (Accelerometer + Gyroscope + Magnetometer + Sensor Fusion)

**Напряжение питания:** 3.3V (рекомендуется) или 2.4V - 3.6V (диапазон)

**Интерфейс:** I2C (до 400 кГц) или UART

**Характеристики акселерометра:**

**Диапазон измерения:** ±2g / ±4g / ±8g / ±16g

**Разрешение:** 14 бит

**Шум:** 150 μg/√Hz

**Характеристики гироскопа:**

**Диапазон измерения:** ±125°/s / ±250°/s / ±500°/s / ±1000°/s / ±2000°/s

**Разрешение:** 16 бит

**Шум:** 0.02 °/s/√Hz

**Характеристики магнитометра:**

**Диапазон измерения:** ±1300 μT (X, Y axis) / ±2500 μT (Z axis)

**Разрешение:** 14 бит

**Частота обновления:** до 100 Гц (зависит от режима работы)

**Режимы работы:**

**Imu:** Только данные акселерометра и гироскопа (без магнитометра).

**Compass:** Только данные магнитометра.

**M4g:** Только данные акселерометра и магнитометра.

**Ndf Fusion:** Fusion всех датчиков (акселерометр, гироскоп, магнитометр) для получения абсолютной ориентации.



1. BNO055

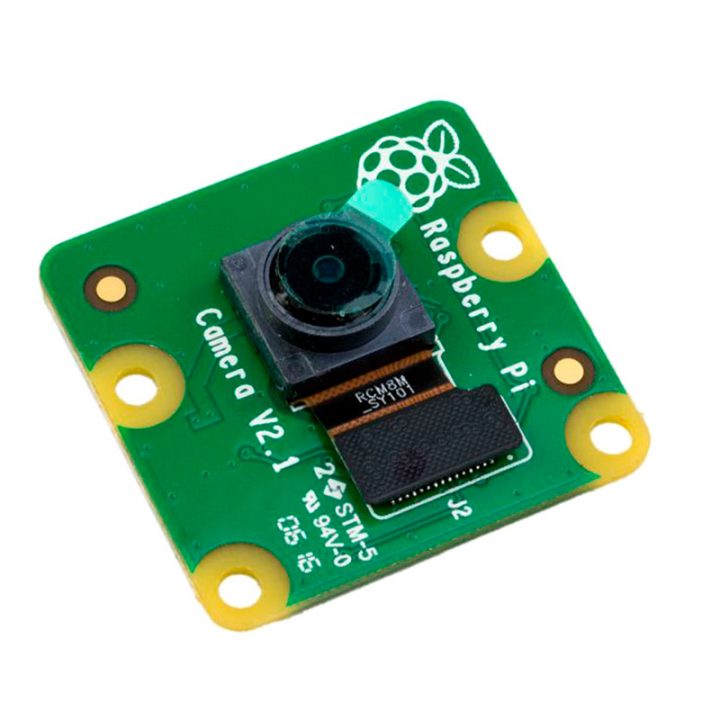
### Камера

Для определения кубиков с Aruco маркерами воспользуемся камерой Raspberry Pi Camera Module 2

**Raspberry Pi Camera Module 2** — это компактная и высококачественная камера, разработанная специально для использования с одноплатными компьютерами Raspberry Pi. Камера подключается к Raspberry Pi через интерфейс **CSI (Camera Serial Interface)**, что обеспечивает высокоскоростную передачу данных с минимальной задержкой.

**Основные характеристики:**

* **Сенсор:** 8-мегапиксельный сенсор Sony IMX219
* **Разрешение фото:** 3280 x 2464 пикселей
* **Разрешение видео:**
  + 1080p30 (1920 x 1080 пикселей, 30 кадров в секунду)
  + 720p60 (1280 x 720 пикселей, 60 кадров в секунду)
  + 640x480p90 (640 x 480 пикселей, 90 кадров в секунду)
* **Размер сенсора:** 1/4 дюйма
* **Угол обзора:** 62.2 x 48.8 градусов (по диагонали 72.4 градуса)
* **Диафрагма:** f/2.0
* **Фокус:** Фиксированный



1. Raspberry Pi Camera Module 2

# КОНСТРУКЦИЯ ПРИВОДА И МОБИЛЬНОЙ ПЛАТФОРМЫ

Основываясь на техническом задании и принимая во внимание особенности колёс Меканум, был разработан робот, конструкция которого обеспечивает расположение центра вращения в геометрическом центре платформы. Основой конструкции является рама, выполненная из алюминиевого конструкционного профиля. Преимущества использования такого профиля заключаются в его лёгкости и высокой прочности, что обеспечивает устойчивость и долговечность конструкции. . Кроме того, выбор алюминиевого профиля обусловлен простотой монтажа дополнительных элементов, а также удобством сборки и разборки рамы, поскольку все компоненты соединяются с помощью Т-образных гаек. Элементы рамы скрепляются между собой при помощи скоб, болтов и Т-гаек, что обеспечивает надёжность и жёсткость конструкции.

Изображение выглядит как Масштабная модель, колесо, Игрушечный транспорт, игрушка

Контент, сгенерированный ИИ, может содержать ошибки.

1. Изометрия мобильной платформы

Все элементы управления, включая управляющий компьютер, микроконтроллер и драйвер электродвигателя, гальванически развязаны от конструкции робота. Это достигается за счёт использования диэлектрических втулок, которые соединяют элементы управления с рамой.

Изображение выглядит как диаграмма, снимок экрана, линия, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

1. Вид сверху

Привод крепится к раме с помощью кронштейна. Поскольку Т-образные гайки не обеспечивают точного позиционирования в горизонтальной плоскости, предусмотрено дополнительное крепление к верхней плите. Электродвигатель соединён с Меканум-колесом посредством переходной втулки. Втулка фиксируется на двигателе с помощью установочного винта, а на колесе — через шлицевое соединение, что исключает возможность прокручивания. Сборочный чертёж данной конструкции приведён в приложении Б.

# ОБЗОР ВОЗМОЖНОГО ПОДХОДА К ОБУЧЕНИЮ НА МОБИЛЬНОЙ ПЛАТФОРМЕ

В данном разделе рассмотрим более детально процесс обучения с использованием мобильной платформы. Основное внимание уделяется изучению алгоритмов SLAM-навигации.

Рассмотрим план обучения для студента, которые не имеет навыков в Linux и ROS, но имеет навыки базового программирования.

**Модуль 1: Основы Linux и командная строка**

* **Цель:** Освоить базовые навыки работы с операционной системой Linux, необходимой для работы с ROS.
* **Темы:**
  + Введение в Linux: История, архитектура, дистрибутивы.
  + Командная строка
  + Управление файлами и каталогами
  + Управление процессами
  + Установка и обновление программ

**Модуль 2: Введение в ROS (Robot Operating System)**

* **Цель:** Понимание архитектуры ROS и умение использовать основные инструменты для создания и запуска ROS-пакетов.
* **Темы:**
  + Введение в ROS: Архитектура, концепции (узлы, топики, сервисы, параметры).
  + Установка и настройка ROS.
  + Работа с ROS-пакетами: Создание, сборка, запуск (catkin).
  + Публикация и подписка на топики
  + Использование сервисов: Определение и вызов сервисов.
  + ROS-инструменты: roslaunch, rostopic, rosnode, rosservice, rosparam.

**Модуль 3: Управление роботом в ROS**

* **Цель:** Научиться управлять реальным роботом с использованием ROS.
* **Темы:**
  + Подключение к реальному роботу: Настройка сети.
  + Чтение данных с сенсоров робота (лидар, энкодеры, IMU).
  + Управление приводами робота: Отправка команд скорости.
  + Использование ROS-драйверов для сенсоров и приводов.
  + Основы кинематики мобильных роботов
  + Реализация простых алгоритмов управления движением

**Модуль 4: Основы SLAM-навигации**

* **Цель:** Получить общее представление о SLAM.
* **Темы:**
  + Введение в SLAM
  + Сенсоры для SLAM: Лидары, камеры, IMU, энкодеры.
  + Основные алгоритмы SLAM (обзорно): Gmapping, Hector SLAM, Cartographer, ORB-SLAM.
  + Метрики оценки качества SLAM.

**Модуль 5: SLAM-навигация и ROS Navigation Stack**

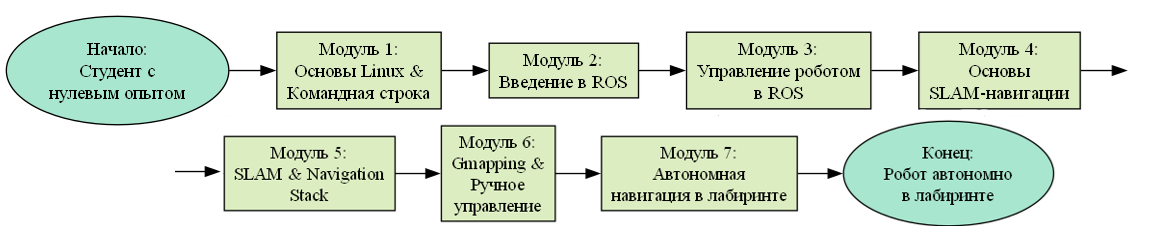
* **Цель:** Изучить ROS Navigation Stack и научиться использовать его для автономной навигации.
* **Темы:**
  + Введение в ROS Navigation Stack: map\_server, amcl, move\_base.
  + Настройка ROS Navigation Stack для работы с картой, построенной с помощью SLAM (Gmapping, Hector SLAM, Cartographer).
  + Алгоритмы обхода препятствий: Dynamic Window Approach (DWA), Timed Elastic Band (TEB).

**Модуль 6: SLAM-навигация в лабиринте с Gmapping и ручным управлением**

* **Цель:** Научиться использовать Gmapping для построения карты лабиринта в ручном режиме.
* **Темы:**
  + Установка и настройка Gmapping.
  + Подключение лидара к ROS и настройка публикации данных.
  + Настройка параметров Gmapping для работы в лабиринте (диапазон сканирования, разрешение карты).
  + Запуск Gmapping и ручное управление роботом для построения карты лабиринта.

**Модуль 7: SLAM-навигация в лабиринте с автоматическим обходом стен и автономной навигацией**

* **Цель:** Использовать SLAM (Gmapping или другие алгоритмы) и Navigation Stack для автономной навигации робота в лабиринте.
* **Темы:**
  + Реализация алгоритма “движения вдоль стены” в ROS.
  + Интеграция алгоритма “движения вдоль стены” с ROS Navigation Stack.
  + Настройка параметров amcl для точной локализации в лабиринте.
  + Настройка параметров локального планировщика (DWA или TEB) для обхода препятствий.
  + Анализ результатов: Покрытие лабиринта, точность навигации, время выполнения.



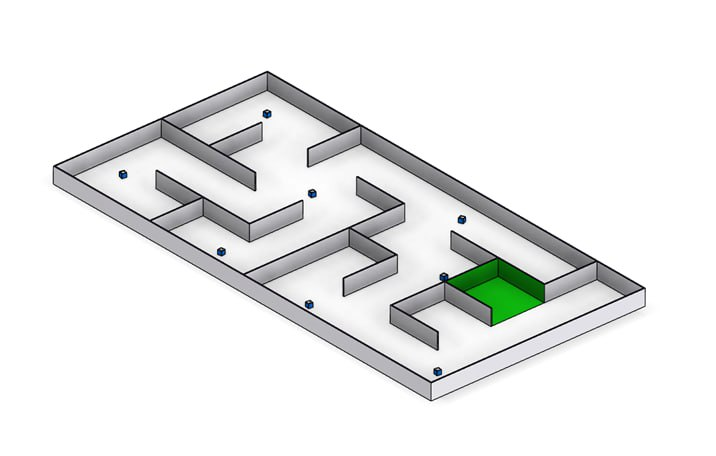
1. Структура обучения

В данной работе акцент сделан на методах, использующих лидар. Все алгоритмы SLAM можно разделить на две категории: те, которые работают с 3D-лидарами, и те, которые используют 2D-лидары. Поскольку в нашем случае выбран 2D-лидар, рассматриваться будут только алгоритмы, адаптированные под этот тип лидаров. Однако для изучения других алгоритмов в будущем можно заменить лидар на 3D-версию и исследовать соответствующие методы.

Среди популярных 2D-лидарных SLAM-алгоритмов в ROS выделяются Gmapping, Hector SLAM, Cartographer (в 2D режиме) и Lago-LOAM (2D). В рамках дипломной работы подробно рассматривается один конкретный сценарий обучения. Предполагается, что обучающийся самостоятельно разработает и исследует другие сценарии, в зависимости от своего интереса к изучению конкретных алгоритмов и ситуаций.

В рамках дипломной работы будет рассмотрен исключительно алгоритм Gmapping, при этом построение карты будет выполняться в ручном режиме. Для этого потребуется активировать алгоритм и вручную провести робота по всему лабиринту, после чего повторить этот процесс несколько раз, чтобы добиться высокого качества карты. Далее необходимо будет включить алгоритмы, отвечающие за построение траектории, такие как global\_planner вместе с dwa\_local\_planner, которые позволят роботу объехать весь лабиринт, включая выход из тупиков. Подробное описание алгоритмов построения траектории будет представлено в следующем разделе.

После создания качественной карты и возвращения робота в исходное положение, в лабиринт добавляются кубики с маркерами. Робот начинает движение по заранее подготовленной траектории. Обнаружение маркеров осуществляется не с помощью SLAM-навигации, а с использованием камеры. Поскольку размеры кубиков известны, применяется метод "оценка расстояния до объекта на основе его известного размера". С помощью этого метода робот находит кубики, подъезжает к ним на заданное расстояние, разворачивается на 180 градусов и передаёт сигнал для начала работы манипулятора. После завершения работы манипулятора робот снова разворачивается на 180 градусов и продолжает движение по траектории. По окончании объезда всего лабиринта робот возвращается в исходное положение.



1. Лабиринт с кубиками

Для корректной работы алгоритмов кубики не должны располагаться вплотную к стене. Правильность работы алгоритмов оценивается визуально, а отладка выполняется итерационно. Чтобы избежать столкновений робота со стенами, применяется метод "расширения стен" (инфляция). Это означает, что на основе изначально построенной карты расстояние между параллельными стенами уменьшается, но остаётся достаточным для проезда робота.

В случае возникновения столкновения робота работа немедленно останавливается программно посредством ручного вмешательства оператора, осуществляющего визуальный контроль за процессом.

Изображение выглядит как Масштабная модель, Игрушечный транспорт, игрушка, колесо

Контент, сгенерированный ИИ, может содержать ошибки.

1. Мобильная платформа с манипулятором

# ЗАКЛЮЧЕНИЕ

* Был проведён анализ современных мобильных платформ, на основе которых сформулированы требования к разрабатываемой платформе.
* Реализована трехуровневая система управления, включающая верхний уровень (управляющий компьютер), средний уровень (микроконтроллер) и нижний уровень (аппаратная часть). Это позволяет эффективно управлять движением платформы, обрабатывать данные с сенсоров и реализовывать алгоритмы навигации.
* Разработана конструкция мобильной платформы с использованием всенаправленных колёс типа "Меканум", что обеспечивает высокую маневренность и возможность движения в любом направлении. А также подобраны компоненты для системы управления.
* Создан план обучения, который включает изучение основ Linux, ROS, SLAM-навигации и управления роботом. А также разработана практическая задача для демонстрации возможностей робота.

# ****Список литературы****

1. Р.О. Анисимов, В.С. Бакаев, Т.Б. Бахов, Н.В. Голобурдин, А.М. Марчук, Н.А. Мостаков – МОБИЛЬНЫЙ МАНИПУЛЯЦИОННЫЙ РОБОТ ДЛЯ МЕЖДУНАРОДНЫХ СОРЕВНОВАНИЙ ПО РОБОТОТЕХНИКЕ – 2020 Политехнический молодежный журнал № 11
2. Баженов Е.И., Мокрушин С.А., Охапкин С.И. Разработка системы ориентации мобильного робота в пространстве // Известия Тульского государственного университета. Технические науки. 2021. Вып. 4. С. 354-359.
3. Тарасян В.С., Васильева Г.В., Коварина А.А. Мобильный робот для участия в соревнованиях Worldskills // Управление и обработка информации в сложных системах. 2020. С. 36-41.
4. Чосет, Х. Принципы движения роботов: теория, алгоритмы и реализация / Х. Чосет, К. М. Линч, С. Хатчинсон [и др.]. — Кембридж, Массачусетс: MIT Press, 2005. — 560 с.
5. Зигвар, Р. Автономные мобильные роботы: сенсорика, управление, принятие решений и применение / Р. Зигвар, И. Р. Нурбахш, Д. Скарамуцца. — Кембридж, Массачусетс: MIT Press, 2011. — 576 с.
6. Дюррант-Уайт, Х. Одновременная локализация и картографирование: Часть I / Х. Дюррант-Уайт, Т. Бейли // IEEE Robotics & Automation Magazine. — 2006. — Том 13, № 3. — С. 99-110.
7. Шан, Т. LIO-SAM: плотно связанная лидарная инерциальная одометрия посредством сглаживания и картографирования / Т. Шан, Б. Энглот, Д. Мейерс, У. Ван, К. Ратти, Д. Рус // 2020 IEEE/RSJ Международная конференция по интеллектуальным роботам и системам (IROS). — 2020. — С. 5135-5141.
8. Монк, С. Кулинарная книга Raspberry Pi / С. Монк. — Себастопол, Калифорния: O’Reilly Media, 2016. — 432 с.
9. Новиэлло, К. Освоение STM32 / К. Новиэлло. — Leanpub, 2018. — 500 с.
10. Рохас, Р. Кинематика и динамика четырёхколёсного транспортного средства Mecanum / Р. Рохас. — Берлин : Свободный университет Берлина, 2013. — 25 с.
11. Дунаев, П.Ф., Леликов, О.П. Конструирование узлов и деталей машин – Изд-во МГТУ им. Н.Э. Баумана, 2004. – 400 с.
12. ГОСТ 1.284.3-96. Конструкция и проектирование машин. Стандарты и нормы [Электронный ресурс]. URL: <https://files.stroyinf.ru/Data2/1/4294838/4294838327.pdf> (дата обращения: 25.12.2024)
13. Грищенко, А. Д. Введение в мобильную робототехнику / А. Д. Грищенко. — Москва : Машиностроение, 2006. — 328 с.

ПРИЛОЖЕНИЕ А

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, чек, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

# ПРИЛОЖЕНИЕ Б

Изображение выглядит как текст, Параллельный, линия, черно-белый

Контент, сгенерированный ИИ, может содержать ошибки.

