**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение высшего образования «Казанский национальный исследовательский

технический университет им. А.Н. Туполева-КАИ»

(КНИТУ-КАИ)

Институт \_\_\_\_\_\_\_\_Автоматики и электронного приборостроения\_\_\_\_\_\_\_\_\_

Кафедра \_\_\_\_\_\_\_\_\_\_\_\_Автоматики и управления\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направления подготовки: \_\_\_\_\_\_\_\_\_\_\_\_27.04.04 Управление в технических системах\_\_\_\_

Образовательная программа: Управление и информатика в интеллектуальных технических системах\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| **К защите допустить** |
| **Зав. каф. \_\_\_\_\_\_\_\_\_** |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Маливанов Н.Н.** |
| **«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г.** |

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

магистерская диссертация

на тему: «Интеллектуальная система управления мобильным колесным роботом»

ОБУЧАЮЩИЙСЯ Хамидуллин Адель Ильсурович \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(фамилия, имя, отчество) (подпись)*

РУКОВОДИТЕЛЬ к.ф.м.н., доцент, Лазарева Полина Александровна  \_\_\_\_\_\_\_\_\_\_\_\_\_

*(ученая степень, звание, фамилия, имя, отчество) (подпись)*

Казань 2023 г.

# Аннотация

Системы автоматики и автоматизация промышленности занимают важнейшую роль в жизни современного общества и государства. Светофоры, пешеходные переходы, компьютерные сети, промышленные станки и т.д. – огромный шаг в развитии науки, социальной жизни каждого жителя планеты.[4]

Одним из наиболее актуальных научных направлений является развитие технологий искусственного интеллекта, автоматизации и автономной навигации автомобилей, мобильных роботов и колесных аппаратов в целом. Однако реализация таких масштабных устройств требует больших материальных, трудовых и временных ресурсов.

Одним из решений данной проблемы может стать создания миниатюрных копий устройств. Это позволяет студентам, любителям и профессионалам углубиться в изучение робототехники, мехатроники и систем навигации, вносить вклад в развитие данной отрасли новыми алгоритмами, научными исследованиями и моделированием систем. Такие исследования позволяют использовать опыт, полученный на миниатюрных устройствах, и масштабировать полученные модели и алгоритмы на более габаритных, совершенных и сложных системах.

Цель работы – разработка и моделирование системы управления тележкой в ограниченном пространстве с автоматическим обходом препятствия.

Задачи, поставленные в ходе выполнения курсовой работы:

1. Анализ и разработка математических моделей работы двигателя и движения тележки в пространстве.
2. Моделирования и подбор параметров для фильтра сигналов, получаемых с измерительных приборов.
3. Моделирования полученных математических моделей движения тележки и фильтрации в ПО MATLab и Simulink.

# Abstract

Системы автоматики и автоматизация промышленности занимают важнейшую роль в жизни современного общества и государства. Светофоры, пешеходные переходы, компьютерные сети, промышленные станки и т.д. – огромный шаг в развитии науки, социальной жизни каждого жителя планеты.[4]

Одним из наиболее актуальных научных направлений является развитие технологий искусственного интеллекта, автоматизации и автономной навигации автомобилей, мобильных роботов и колесных аппаратов в целом. Однако реализация таких масштабных устройств требует больших материальных, трудовых и временных ресурсов.

Одним из решений данной проблемы может стать создания миниатюрных копий устройств. Это позволяет студентам, любителям и профессионалам углубиться в изучение робототехники, мехатроники и систем навигации, вносить вклад в развитие данной отрасли новыми алгоритмами, научными исследованиями и моделированием систем. Такие исследования позволяют использовать опыт, полученный на миниатюрных устройствах, и масштабировать полученные модели и алгоритмы на более габаритных, совершенных и сложных системах.

Цель работы – разработка и моделирование системы управления тележкой в ограниченном пространстве с автоматическим обходом препятствия.

Задачи, поставленные в ходе выполнения курсовой работы:

1. Анализ и разработка математических моделей работы двигателя и движения тележки в пространстве.
2. Моделирования и подбор параметров для фильтра сигналов, получаемых с измерительных приборов.
3. Моделирования полученных математических моделей движения тележки и фильтрации в ПО MATLab и Simulink.

# СОДЕРЖАНИЕ

[ВВЕДЕНИЕ **3**](#_Toc127027281)

[1. Анализ и разработка математических моделей работы двигателя и движения тележки в пространстве. **4**](#_Toc127027282)

[1.1 Математическая модель двигателя постоянного тока робота. **4**](#_Toc127027283)

[1.2 Математическая модель управления роботом-тележкой **6**](#_Toc127027284)

[2. Моделирование системы управления роботом. **10**](#_Toc127027285)

[2.1 Моделирование двигателя постоянного тока робота. **10**](#_Toc127027286)

[2.2 Моделирование системы управления роботом без внешнего возмущения. **18**](#_Toc127027287)

[3. Моделирование системы управления роботом с учетом внешних возмущений **20**](#_Toc127027288)

[3.1 Характеристика микро-профиля дороги и его математическая модель. **20**](#_Toc127027289)

[3.2 Моделирование системы управления роботом с внешним возмущением. **24**](#_Toc127027290)

[ЗАКЛЮЧЕНИЕ **28**](#_Toc127027291)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ **29**](#_Toc127027292)

[Приложение А **30**](#_Toc127027293)

[Приложение Б **31**](#_Toc127027294)

[Приложение В **32**](#_Toc127027295)

# ВВЕДЕНИЕ

Системы автоматики и автоматизация промышленности занимают важнейшую роль в жизни современного общества и государства. Светофоры, пешеходные переходы, компьютерные сети, промышленные станки и т.д. – огромный шаг в развитии науки, социальной жизни каждого жителя планеты.[4]

Одним из наиболее актуальных научных направлений является развитие технологий искусственного интеллекта, автоматизации и автономной навигации автомобилей, мобильных роботов и колесных аппаратов в целом. Однако реализация таких масштабных устройств требует больших материальных, трудовых и временных ресурсов.

Одним из решений данной проблемы может стать создания миниатюрных копий устройств. Это позволяет студентам, любителям и профессионалам углубиться в изучение робототехники, мехатроники и систем навигации, вносить вклад в развитие данной отрасли новыми алгоритмами, научными исследованиями и моделированием систем. Такие исследования позволяют использовать опыт, полученный на миниатюрных устройствах, и масштабировать полученные модели и алгоритмы на более габаритных, совершенных и сложных системах.

Цель работы – разработка и моделирование системы управления тележкой в ограниченном пространстве с автоматическим обходом препятствия.

Задачи, поставленные в ходе выполнения курсовой работы:

1. Анализ и разработка математических моделей работы двигателя и движения тележки в пространстве.
2. Моделирования и подбор параметров для фильтра сигналов, получаемых с измерительных приборов.
3. Моделирования полученных математических моделей движения тележки и фильтрации в ПО MATLab и Simulink.

# 1. Литературный обзор и постановка задачи

## 1.1 Краткий обзор литературы по моделям, алгоритмам управления и методам исследования автономных транспортных средств

В течение последнего десятилетия в связи с активным развитием систем искусственного интеллекта набирает большую популярность создание более точных и адаптивных алгоритмов и систем автономного управления транспортом. Современные исследования [1] располагают широким спектр вариантов использования различных алгоритмов: от простых компенсационных схем управления до нейросетевых технологий с использованием элементов нечеткой логики. Общим названием такого вида алгоритмов является SLAM (System of Localization and Mapping). Исследователи в ходе моделирования нескольких алгоритмов выявили критерии оценивания: наличие ошибки рассогласования, точность на различных дистанциях, сложность алгоритма и влияние внешней среды. Перечисленные характеристики позволяют выбрать наиболее оптимальный алгоритм управления малым колесным роботом для поставленной задачи. Прежде всего необходимо учитывать воздействие человека на управление тележкой, поскольку достичь полной автономности транспортных средств без участия человека практически невозможно. Функциональная схема навигационной системы, представленная в публикации, изображена на рисунке 1.1.

Часть представленных алгоритмов базируется на использовании нечеткой логики со схемой Мамдани, которая обеспечивает низкую сложность реализации, простоту настройки. Однако гарантий полного обхода препятствий такой алгоритм не дает.

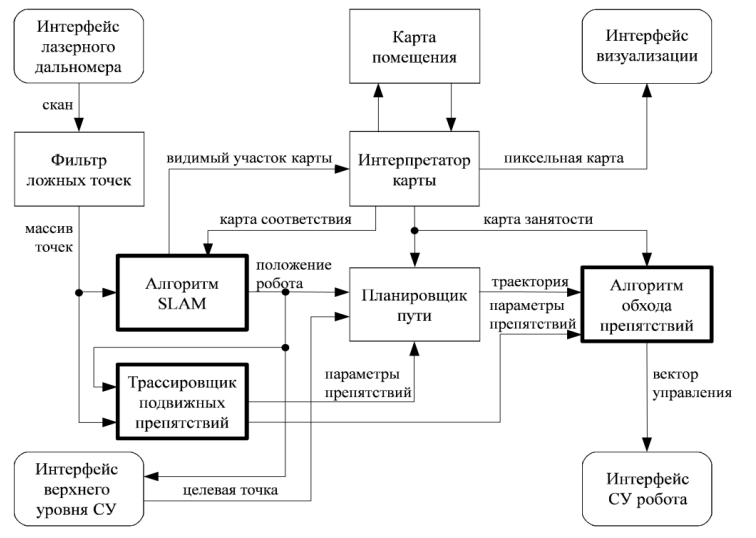


Рисунок 1.1 – Функциональная схема навигации мобильного колесного робота

Большая актуальность данного исследования заключается в описании диалогового управления и нейро-сетевых регуляторов. Системы такого типа имеют обучаемые базы знаний, которые формируют свои знания в динамике. Одним из наиболее значимых элементов диалогового управления является наличие аппарата распознавания речи, лингвистический анализатор. Актуальность данных элементов заключается в интенсивном развитии качества распознавания речи с помощью алгоритмов искусственного интеллекта, что позволяет быстрее адаптироваться под различные интонации, языки. Однако у данной инновации есть отрицательная сторона – она требует большие вычислительные мощности и высокоэффективное оборудование для качественной обработки голосовой информации. В публикации авторы данное качестве не указали.

В статье [2] авторами представлена подробная математическая модель движения автономного робота по нелинейной и изменяемой траектории. Она содержит в себе элементы нелинейного динамического управления движением робота, адаптивных алгоритмов под изменяемую модель внешних воздействий: вода, асфальтированная дорога, грунт. Основная цель данного алгоритма – идентификация внешних воздействий на управлениеаптация под изменяемую среду и стабилизация объекта на трассе, снижая парирование отклонений от исходной траектории движения тележки.

В первую очередь авторы преобразуют нелинейные модели робота к системе задачно-ориентированных координат: выделяются матрицы связи угла вращения колес относительно центра масс, поступательное движение и расчет линейной и угловой скоростей в системе пространства состояний. После чего производится синтез алгоритма управления путем формирования матрицы состояния и управления тележкой, производится моделирование системы с влиянием шумовых характеристик и внешних воздействий. Авторы статьи выделяют желаемые параметры моделирования для достижения наилучших характеристик.

В публикации [3] авторы представляют подробную концепцию компонентно-ориентированной программной платформы для управления и работы автономного мобильного робота. На примере платформы RCE (Robot Components Engine) описаны требования к платформе, функциональные возможности, а также уже реализованные в производстве технологии. Одним из главных требований к разработке платформы являлись модульность, масштабируемость системы, то есть использование открытых протоколов и интерфейсов взаимодействия устройств между собой. Такая особенность, по мнению авторов, позволяет разделить функциональные элементы и структурировать взаимодействие между сенсорами, двигателями и внешней периферией.

Также авторы статьи разделяют архитектуры системы на несколько моделей: поведенческая, «очувствление-планирование-действие» и гибридную. В первом случае все компоненты системы располагаются на одном уровне и работают совместно через аппаратные или программные неразрывные соединения. Модель «очувствление-планирование-действие» предполагает декомпозированную структуру, в которой все элементы выстроены по вертикали: сбор данных сенсоров, формирование представления об окружающей среде исходя из используемой математической модели управления и передачу сигналов управления на исполняющие устройства. Таким образом, все элементы архитектуры имеют на каждом уровне свои функциональные возможности и задачи. Гибридная архитектура сочетает элементы как первой, так и второй модели.

Архитектура платформы RCE построена на базе гибридной архитектуры, в которой часть задач переносится не на элементы робота, а на компьютер-сервер. Такое размещение функций позволяет уменьшить нагрузку на сам робот и упростить как математическую модель управления, так и требования к колесному аппарату. Задачи сервера заключаются в ведении журнала операций, удаленное построение маршрута для возвращения на базовую станцию в случае аварийной ситуации или снижения заряда аккумулятора, диспетчеризация и обработка сигналов для преодоления сложных препятствий с помощью нескольких сенсоров. При отправке сообщений между устройствами используются как синхронные, так и асинхронные способы передачи данных. Асинхронные задачи преимущественно используются для передачи информационных сообщений, не требующих обратного ответа от мобильного робота.

Преимущество данной платформы в том, что она построена на базе наиболее распространенного языка программирования C++, используемого в промышленных встраиваемых устройствах на базе микроконтроллеров. Язык программирования имеет компонентно-ориентированную парадигму, что позволяет разбивать задачи управления на функциональные блоки – компоненты, которые связаны по аппаратных или программным каналам связи. Исходный код программного обеспечения открыт для изучения студентами и инженерами, что позволяет дополнять или модернизировать ПО для различных аппаратных платформ.

Особую позицию занимают алгоритмы распознавания, обнаружения и объезда препятствий. Исследователи по всему миру детально и глубоко изучают данное научное направление. Так, в публикации [[Obstacle avoidance of mobile robots using modified artificial potential field algorithm | EURASIP Journal on Wireless Communications and Networking | Full Text (springeropen.com)](https://jwcn-eurasipjournals.springeropen.com/articles/10.1186/s13638-019-1396-2) ] автор проводит детальный анализ существующих алгоритмов.

Наиболее распространенными считаются алгоритмы Bug 2-го поколения (далее Bug2), VFH. Алгоритм Bug 1-го поколения (далее Bug) в сравнении с 2-м имеет один существенный недостаток – для преодоления препятствия алгоритм предусматривает объезд препятствия по всему периметру, что создает дополнительные временные, ресурсные затраты. Во алгоритме Bug2 данный недостаток решается таким образом: мобильный робот объезжает препятствие по левую сторону, после чего продолжает движение по заданной прямой для достижения цели. Таким образом, если объезд по левую сторону будет иметь более длинный путь в сравнении с объездом по правую, то этот фактор не будет учитываться алгоритмом Bug2. Такое свойство является недостатком алгоритма Bug2. Для его решения был разработан алгоритм VFH на основе вероятностного подхода с использованием полярных гистограмм для построения новой траектории движения робота с объездом препятствия и возвращения на исходную. Алгоритм VFH использует методы математической оптимизации для нахождения локального минимума при расчете расстояния, находясь рядом с препятствием. Точность и качество работы алгоритма VFH заметно повысились, особенно это заметно при передвижении робота между двумя препятствиями.

## 1.2 Патентная проработка видов и конструкций малогабаритных автономных транспортных средств

В ходе исследования зарегистрированных патентов было обнаружено более десятка тысяч исследований на территории Российской Федерации и стран СНГ с помощью сервиса Яндекс.Патенты. С помощью сервиса Google Патенты по всему миру было найдено свыше сотни тысяч патентов. Были исследованы автономные транспортные средства, мобильные колесные роботы и их компоненты, в т.ч. алгоритмы управления движением тележки. Наибольшее количество зарегистрированных работ находятся в КНР, Японии, США и т.д. Большинство было разработано крупными корпорациями. Среди стран СНГ наибольшее число работ зарегистрировано в ВУЗах и НИИ Российской Федерации, Казахстана и Республики Беларусь. Первые патенты были зарегистрированы в 1996 году в США, на данный момент большую популярность данная отрасль получила во второй половине десятилетия 2010-х годов.

## 1.3 Постановка задачи

В результате проведения литературного обзора, патентного исследования в качестве объекта исследования данной работы была выбрана модель двухколесного мобильного транспортного средства типа робот, управление которого производится автоматически без участия человека. Предметом исследования являются математические модели, методы и алгоритмы управления автономного транспортного средства.

Цель исследования – повышение точности и качества алгоритма управления тележкой при объезде препятствий с учетом рельефа дорожного покрытия.

Задача исследования: для указанной модели двухколесного мобильного робота необходимо разработать систему управления тележкой, которая будет манипулировать поворотом колес и скорости движения тележки при объезде препятствий, учитывая качество дорожного полотна.

Данная задача будет решаться в два этапа. В первую очередь будет проводиться синтез управления, который обеспечивает при учете возмущений дорожного покрытия [10] стабилизацию движения робота в заданном направлении с постоянной скоростью. Данная часть задачи решается в 3 главе с использование математической модели робота из публикации [8], не включающей в себя внешние возмущения. Во второй части решаемой задачи в главе 4 будет проведен синтез управления робота для обхода препятствий с помощью измерительных устройств при автономном движении. Для решения задачи в главе 2 будет получена математическая модель движения робота с учетом изменения скорости, направления движения и заданного внешнего возмущения профиля дорожного покрытия.

# 2. Математические модели работы двигателя постоянного тока и движения тележки в пространстве.

Тележка представляет собой робота, собираемого руками человека, размером 20x20 см. Он содержит в себе два электрических двигателя, два ведущих колеса, а также два энкодера для вращения ведущих колес влево-вправо. В качестве измерительных приборов, а именно измерения расстояния до препятствия от робота в нем применяются 3 датчика расстояния, расположенных слева, впереди и справа от центра масс на роботе. Синтез трех датчиков препятствия обуславливается достижением наиболее точной работы алгоритма, а также движения пространства для достижения цели с минимальными столкновениями с препятствиями.

**2.1 Математическая модель двигателя постоянного тока робота.**

Двигатель является главным звеном в работе тележки. Для его точного функционирования следует построить адекватную математическую модель [1].

Составим уравнение Лагранжа-Эйлера, а также учтем влияние противо-ЭДС в обмотке ротора двигателя постоянного тока (ДПТ).

|  |  |  |
| --- | --- | --- |
|  |  | (2.1) |

где - угловая скорость вращения мотора,  - скорость холостого хода,  - момент, развиваемый двигателем,  - пусковой момент и  - момент инерции ротора двигателя.

Для решения данного дифференциального уравнения составим следующее выражение

|  |  |  |
| --- | --- | --- |
|  |  | (2.2) |

где  - электромеханическая постоянная двигателя.

Функция измерения угла во времени будет интегралом функции скорости с учетом начальных условий.

|  |  |  |
| --- | --- | --- |
|  |  | (2.3) |

Для вычисления конструктивных постоянных двигателя, необходимые для управления его моментом, составим систему уравнений. Данная система описывать электрические процессы, протекающие в ДПТ.

|  |  |  |
| --- | --- | --- |
|  |  | (2.4) |

где , - конструктивные постоянные, *U* – управляющее напряжение, *L* – индуктивность обмотки двигателя, *I* – пусковой ток.  - сопротивление обмотки ротора.

Коэффициент  - коэффициент передачи по току. Для его расчета применяется выражение, вычисляющее отношение момента на выходном валу на пусковой ток.

|  |  |  |
| --- | --- | --- |
|  |  | (2.5) |

Коэффициент  - коэффициент передачи по току. Для его расчета применяется выражение, вычисляющее отношение момента на выходном валу на пусковой ток.

|  |  |  |
| --- | --- | --- |
|  |  | (2.6) |

Таким образом, для расчета пускового момента, выдаваемого двигателем постоянного тока (ДПТ), формула примет такой вид:

|  |  |  |
| --- | --- | --- |
|  |  | (2.7) |
|  |  |  |

**2.2 Математическая модель управления роботом-тележкой**

Произведем разработку математической модели мобильного колесного робота с использованием ранее описанной математической модели двигателя постоянного тока. В состав мобильного колесного робота для описания математической модели с дифференциальным приводом входят: два двигателя постоянного тока с установленными к ним колесами, гироскоп и 3 ультразвуковых датчиков [3].

Расчет пройденного расстояния будет производиться с помощью энкодеров, расположенных на валу двигателя, для ориентации робота используются гироскоп, измеряющий угловую скорость и задающий угол поворота колес с помощью измеренных ультразвуковыми дальномерами координат положения робота.

На рисунке 2.1 изображена кинематическая схема мобильного колесного робота.

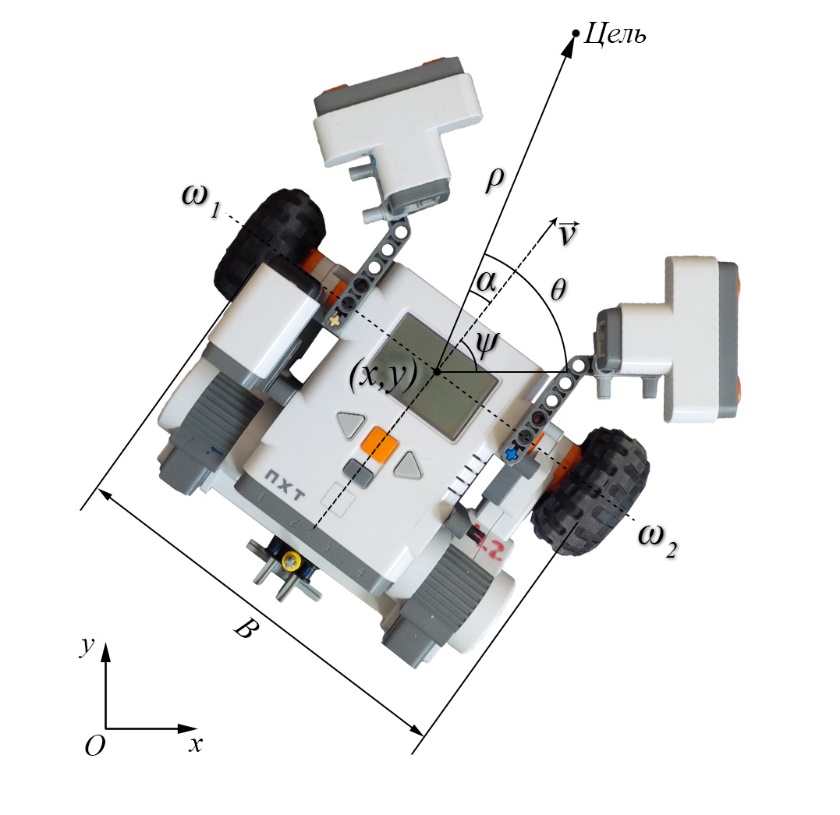


Рисунок 2.1 – Кинематическая схема мобильного колесного робота

На рисунке указаны следующие обозначения: расстояние до целевой точки, азимут, угол между осью OX и направлением на цель, – курс робота, курсовой угол, разность между курсом и азимутом, линейная скорость робота. Движение робота осуществляется за счет двух отдельно управляемых колес. Математическим описанием выше проиллюстрированной кинематической схемы является:

|  |  |  |
| --- | --- | --- |
|  |  | (2.8) |

где , – угловые скорости вращения колес, – координаты движения робота, R – радиус колеса, B – колесная база.

Для решения задачи навигации робота нам необходимо достигать следующих координат: , то есть расстояние до цели и курсовой угол должны достичь нуля. В рамках текущей работы используем локальный подход навигации.

Локальный подход – определение координат устройства по отношению к начальной точке. Планирование задает небольшой отрезок траектории, в конечной точке которого выбирается дальнейшая траектория движения робота [3].

Опишем математическую модель навигации робота к цели в полярных координатах:

|  |  |  |
| --- | --- | --- |
|  |  | (2.9) |

Для управления роботом необходимо найти такие значения угловой и линейной скоростей , при которых будет выполняться условие ранее поставленной задачи, а именно достижение координат: . Для этого воспользуемся функцией Ляпунова. Такая квадратичная функция, включающая в себя расстояние до цели и курсовой угол:

|  |  |  |
| --- | --- | --- |
|  |  | (2.10) |

Производная по времени не положительна, поскольку расстояние до цели и курсовой угол не должны возрастать. Производная имеет следующий вид:

|  |  |  |
| --- | --- | --- |
|  |  | (2.11) |

Выразим производную через математическую модель управления робота:

|  |  |  |
| --- | --- | --- |
|  |  | (2.12) |

Эта производная будет отрицательно определена, если в качестве управляющего воздействия выберем следующие значения скоростей:

|  |  |  |
| --- | --- | --- |
|  |  | (2.13) |

Вышеуказанные значения позволяют достичь роботу своей цели, но без учета внешних препятствий. Для избегания препятствий вводим поправку для вычисления курсового угла:

|  |  |  |
| --- | --- | --- |
|  |  | (2.14) |

где – коэффициент пропорциональной составляющей курсового угла, – расстояние до препятствия, – минимальное расстояние до препятствия.

**2.3 Характеристика микро-профиля дороги и его математическая модель.**

В качестве внешнего возмущения, приложенному к управлению мобильного колесного робота, возьмем влияние случайного микро-профиля дороги на колебания робота во время движения. Охарактеризовать такое возмущение возможно с помощью корреляционной функции и спектральной плотности. Методы статистической динамики позволят охарактеризовать реакцию подвески и элементов управление на это воздействие такими же характеристиками корреляционной функции или спектральной плотности изменения координат системы или их производных [3].

Микро-профиль дороги описывается нормированной корреляционной функцией вида:

|  |  |  |
| --- | --- | --- |
|  |  | (2.15) |

где – время проезда неровности, скорость машины, – коэффициенты корреляционной связи, безразмерные коэффициенты (причем ).

Передаточная функция микро-профиля дороги определяется следующим выражением:

|  |  |  |
| --- | --- | --- |
|  |  | (2.16) |

Ниже в таблице 2.1 представлены рассчитанные значения микро-профиля дороги для трех видов дорожного покрытия: грунтовая, булыжная и асфальтобетонная. Каждая из них отличается качеством покрытия, устойчивостью к изменению погодных условий и колебаниями почвенного слоя [4]. Рассчитанные значения были получены экспериментальным путем при исследовании влияния дорог на подвеску и движение автомобиля [3].

Таблица 2.1 - Рассчитанные параметры микро-профиля дороги

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Вид дороги |  |  |  |  |  |  |
| Грунтовая | 0,634 | 0,366 | 0,017 | 0,153 | 0,144 | 110 |
| Булыжная | 0,8 | 0,2 | 0,32 | 0,2 | 0 | 10,7 |
| Асфальтобетонная | 1 | 0 | 0,14 | 0 | 0 | 0,25 |

# Глава 3. Электрическая схема и элементная база колесного робота

## 3.1 Описание электрических компонентов и соединений блока управления роботом

Малый колесный аппарат представляет собой робот~~-~~автомобиль, имеющий 2 гусеницы колеса, 2 коллекторных мотора постоянного тока, аккумулятор, блок управления колесным аппаратом, датчики приближения [7].

Каждый компонент системы блока управления состоит из ряда базовых электронных компонентов, соединяемых между собой на печатных платах. Рассмотрим наиболее важные узлы подробнее.

1. Адаптер Bluetooth HC-06.

Данное устройство состоит из микросхемы HC-05, 4 триггеров Шмидта 74LVC2G17, буферного элемента 74LVC1G125DBV на базе RS-триггера с 3 состояниями («низкое», «высокое» и Z-состояние) и линейного стабилизатора напряжения TPS73033DBV с фиксированным выводом напряжения 3.3В. Микросхема содержит 34 вывода: первые два – RX/TX-выводы для передачи данных, 11 – вывод на сброс настроек, 12 – питание микросхемы 3.3В, 13, 22 – нулевой потенциал. 24, 31, 32 выводы – управление светодиодной оснасткой. 26, 33 – элементы подключения с внешней антенной. 34 вывод соединен с буферным элементом, который шунтируется резистором на 10 КОм. Для связи с устройствами по технологии Bluetooth в окружающей среде на плате размещается антенна.

На схеме представлены триггеры Шмидта для создания канала передачи данных по технологии Bluetooth. Первые два триггера на вход принимают внешний сигнал RX, который распределяется по двум триггерам: первый сигнализирует светодиодом и подает напряжение на вход 3.3В, второй – передает данные на 2 вывод микросхемы HC-06 для приема данных с внешнего устройства. Следующие два триггера принимают на вход сигнал TX с микросхемы HC-06 и выводят их через первый триггер на внешний канал связи, а по второму – сигнализируют светодиодом и подают напряжение на вход 5В.

Также рассмотрим линейный стабилизатор напряжения TPS73033DBV на 3.3В. Он содержит 5 выводов: входные линии – IN (напряжение +5В) и EN, подающие сигнал включения стабилизатора, ножка нулевого потенциала GND, ножка выходного напряжения +3.3В OUT. На вход подается шунтированное конденсатором на 100 нФ напряжение на 5В. Выходное напряжение выравнивается конденсаторами на 2.2 мФ и 100 нФ в пределах 3.3В.

На рисунке 3.1 представлено изображения Bluetooth-адаптера HC-06.

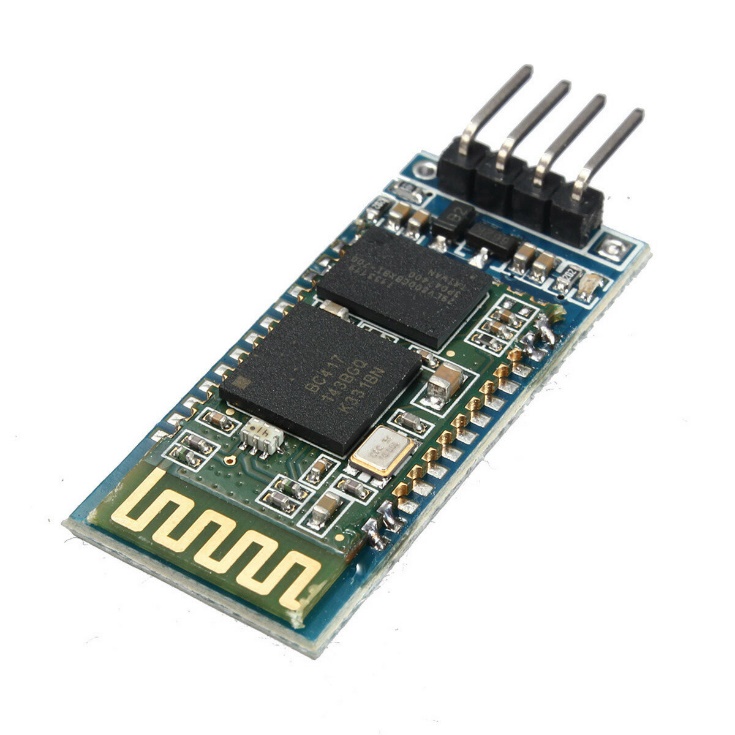


Рисунок 3.1 – Bluetooth-адаптер HC-06

2. Ультразвуковой дальномер HC-SR04.

Данное устройство состоит из набора управляющих микросхем EM78P153S, MAX232, 4-х операционных усилителей LM324, принимающего и передающего звукового устройства.

Микросхема EM78P153S является 8-битным микропроцессором с низким потреблением, изготовленным по технологии КМОП. Содержит 14 контактов, 2 из которых подключаются к шинам питания +5В и GND. На схеме дальномера к выводам P67 и P50 подключаются контакты, синхронизирующие связь с внешним микроконтроллером (в данном случае с Arduino Uno), при этом вывод P67 при длительном отсутствии сигнала переводит дальномер в режим пониженного энергопотребления. Выводы P60, P61 принимают сигналы с операционных усилителей. По выводам P51, P52 передаются сигналы на выводы микросхемы MAX232.

Микросхема MAX232 – преобразователь сигналов порта RS-232 в цифровые сигналы (например, UART) в системах на базе КМОП-технологий. Имеет большое распространение в промышленной микроэлектронике. На схеме дальномера HC-SR04 принимает сигналы с выходов P51, P52 микросхемы EM78P153S на входы интерфейса UART T1IN, T2IN. С помощью управляющих сигналов, поступающих на входы C2+ и VS от компаратора на базе операционного усилителя LM324, который получает сигнал с приемника, микросхема MAX232 преобразует полученный сигнал в сигнал по интерфейсу RS232 на ультразвуковой передатчик, выполненный в виде звукового устройства.

Операционный усилитель LM324 4-канальный на схеме дальномера выполняет следующие функции: усилитель сигнала, фильтр полосы пропускания и компаратор. Полученный с приемника сигнал усиливается, далее с помощью последовательной RC-цепи (конденсатор на 1 нано-фарад и резистор на 6.2 кОм) и операционного усилителя проводится фильтрация низких частот. После чего сигнал вновь усиливается и сравнивается с помощью компаратора для подачи сигнала об обнаружении препятствия на микросхемы MAX232 (выводы VS и C2+) и EM78P153S (выводы P60, P61). На схеме размещается 4 последовательно соединенных операционных усилителя LM324 для обработки входных сигналов с приемника [6].

На рисунке 3.2 показано изображение аппарата – ультразвукового датчика HC-SR04.

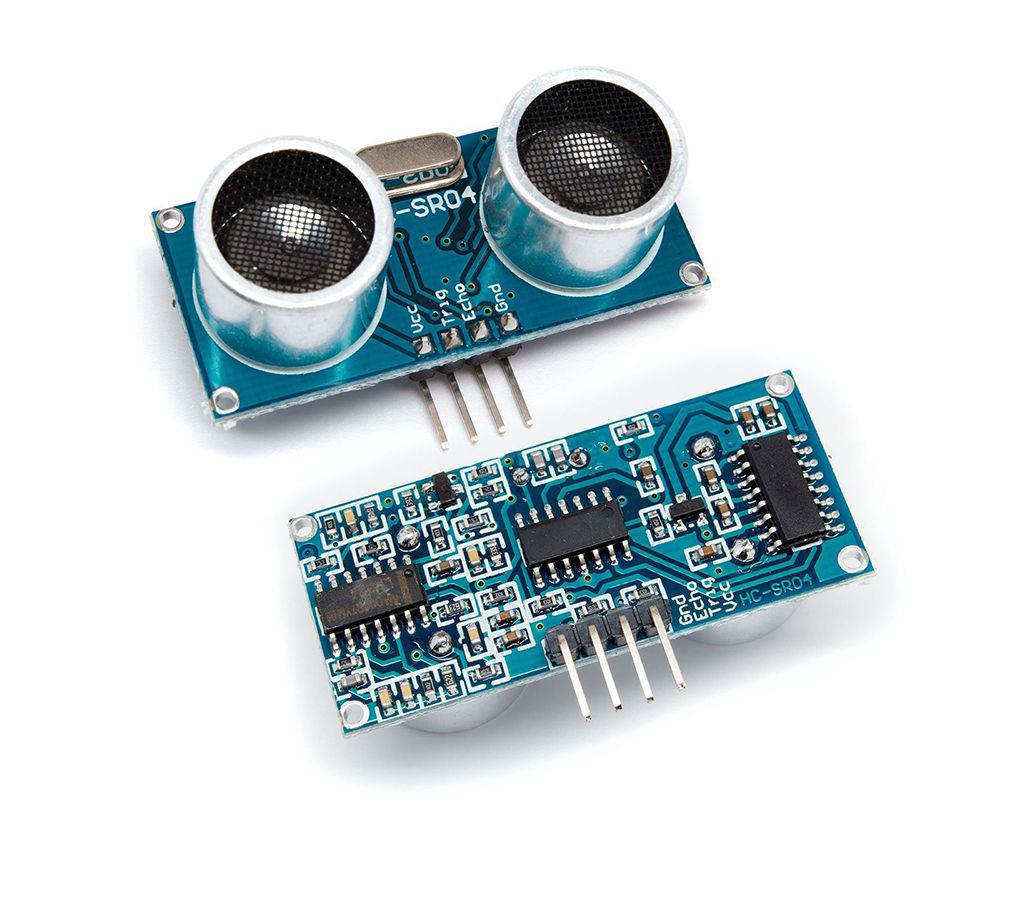


Рисунок 3.2 – Ультразвуковой дальномер HC-SR04

3. Двухканальный модуль управления моторами.

Данное устройство выполнено в виде печатной платы, повторяющей размеры микроконтроллера Arduino Uno с идентично расположенными выводами. Содержит 1 вход для питания платы Motor Shield в диапазоне от +5 до +12В с возможностью параллельного питания платы Arduino Uno, 2 выхода для подключения внешних электродвигателей с напряжением от +5 до +12В. Плата состоит из двойного полномостового драйвера L298P, 2 инвертеров HC1G04, а также 3 транзисторных усилителей и диодного моста из 8 диодов.

Двойной полномостовой драйвер двигателя L298P выполнен в виде микросхемы с 20 выводами, 2 из которых не имеют подключения, 4 вывода – вывод на корпус (GND) и два вывода питания – управляющее питание +5В и +12В для управления внешними подключаемыми устройствами. Диапазон напряжения для управления внешними устройствами лежит от 5 до +12В. При увеличении порогового напряжения есть вероятность выхода микросхемы из строя. К выводам Enable A, Enable B подключены соединения для управления двигателями постоянного тока, к выводам OUT1, 2, 3, 4 подключены по 2 контакта подключаемых электродвигателей [8].

На рисунке 3.3 изображен модуль Motor Shield, подключенный к плате Arduino Uno.



Рисунок 3.3 – Модуль управления внешними моторами Motor Shield

1. Плата-микроконтроллер Arduino Uno.

Данное устройство состоит из набора 8-битных AVR-микроконтроллеров ATMEGA328P и ATMEGA16U2, микросхемы управления питанием NCP1117ST50T3G, 2-х операционных усилителей LMV358IDGKR, стабилизатора напряжения LP2985-33.

Микросхема ATMEGA328P является 8-битным микроконтроллером. На схеме платы-микроконтроллера является основным управляющим контроллером. Содержит 28 контактов, 2 из которых подключаются к шинам питания +5В и еще 2 контакта - к GND, 2 контакта программирования с помощью интерфейса SPI (вход MOSI и выход MISO) и 1 контакт для синхронизирующего входа. К выводам TOSC1 и TOSC2 подключен тактовый генератор c частотой 16 МГц.

Микросхема ATMEGA16U2 является 8-битным микроконтроллером. На схеме платы-микроконтроллера Arduino Uno является микросхемой управления ввода, перепрошивки и преобразователем USB -> UART. Содержит 28 контактов, 2 из которых подключаются к шинам питания +5В и еще 2 контакта - к GND, а 4 контакта для подключения к интерфейсу USB. На схеме микроконтроллера к выводам INT2, INT3 подключены выводы RX и TX микроконтроллера ATMEGA328P синхронизации между микросхемами. К выводам XTAL1 и XTAL2 подключен тактовый генератор c частотой 16 МГц [8].

Операционный усилитель LMV358IDGKR 2-канальный, выполненный в виде микросхемы, на схеме платы-микроконтроллера выполняет следующие функции: индикация состояния платы после синхронизации с загруженной прошивкой и сравнение уровней напряжения между питанием интерфейса USB и подключенным гальваническим элементов (аккумулятор или литиевая батарея) с напряжением +5В. На схеме операционные усилителя размещаются в областях управления питанием платы и сигнализации об успешном запуске с помощью светодиода [5].

Стабилизатор напряжения NCP1117ST50T3G выравнивает напряжение на уровне +5В для подключения с помощью гальванического элемента (аккумулятор или батарея).

На рисунке 3.4 показано изображение аппарата – плата-микроконтроллер Arduino Uno.

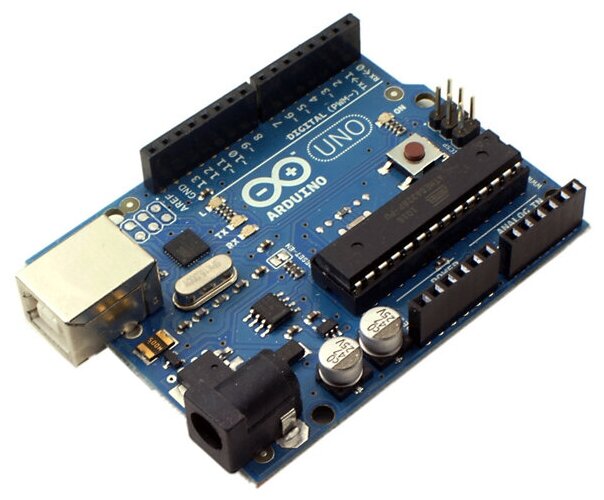


Рисунок 3.4 – Плата-микроконтроллер Arduino Uno

1. Двигатель постоянного тока JGB37-3530.

Электродвигатель с входным постоянным напряжением +12В и скоростью 166 об/мин имеет редуктор с передаточным отношением 56 и длиной 24 мм. Вес двигателя 188 г. На рисунке 3.5 представлено его изображение.



Рисунок 3.5 – Электродвигатель постоянного тока JGB37-3530

## 3.2 Электрическая функциональная схема блока управления

Электрическая функциональная схема (Э2) состоит из четырех функциональных частей: микроконтроллер Arduino Uno, Bluetooth-адаптер HC-06, ультразвуковой дальномер HC-SR04 и модуль управления электродвигателями Motor Shield. Функциональные части подключаются к микроконтроллеру по функциональной цепи [3, 4].

Внешними устройствами, подключаемыми к функциональным частям являются:

1) Батарея с номинальным напряжением +5В – питание микроконтроллера Arduino Uno;

2) Батарея с номинальным напряжением +12В – питание модуля управления электродвигателями Motor Shield;

3) Коллекторный электродвигатель с напряжением +12В и мощностью 166 об/мин. Подключается к плате Motor Shield в количестве 2 шт.

Электрическая функциональная схема представлена в Приложении А.

## 3.3 Электрическая принципиальная схема блока управления

Электрическая принципиальная схема (Э3) содержит подробное представление электрических соединений мобильного робота [3,5]. Она, как и электрическая функциональная схема, состоит из 4 основных блоков: управление микроконтроллером (Arduino Uno), управление электродвигателями (Motor Shield), управление Bluetooth-соединением (HC-06) и измерительное устройство – ультразвуковой дальномер HC-SR04.

1. Блок управления Bluetooth-соединением.

На схеме контакты +5В и GND подключаются к общей шине электропитания с соответствующими потенциалами. Контакты RX и TX подключаются к контактам TXD и RDX блока управления микроконтроллером (Arduino Uno) для связи между устройствами.

Bluetooth-адаптер состоит из 4 триггеров Шмидта 74LVC2G17, 2 светодиодов желтого цвета с номинальным напряжением +5В, 1 светодиода красного цвета с номинальным напряжением +5В, 1 светодиода зеленого цвета с номинальным напряжением +3.3В, регулятора напряжения TPS73033DBV с выходным напряжением +3.3В, 3 резисторов с сопротивлением 470 Ом, резисторов с сопротивлениями 10 кОм и 1 кОм, внешней антенны и 3 конденсаторов с емкостями 100 нанофарад, 10 нанофарад и 2.2 микрофарад, соответственно.

2. Блок управления электродвигателями.

На схеме к выводам D4-D7 блока управления микроконтроллером подключены выводы платы управления электродвигателями Motor Shield для управления вращением колес (DIR12, DIR34) и поворотом по продольной оси (EN12, EN34). К контактам питания платы +12В и GND подключается внешний гальванический элемент (батарея или аккумулятор) с напряжением от +5 до +12В. Выводы платы M11, M12 и M21, M22 используются для приведения в действие внешних электрических двигателей постоянного тока.

Устройство состоит из 6 резисторов с сопротивлением 1 кОм, 5 резисторов с сопротивлением 4.7 кОм; 7 светодиодов красного цвета, 6 КМОП-транзисторов типа 2N7002, 8 диодов SS14, микросхемы управления двигателями L298P, двух инверторов HC1G04, двух конденсаторов с емкостью 100 микрофарад и одного конденсатора с емкостью 100 нанофарад.

3. Измерительное устройство HC-SR04.

Для подключения ультразвукового датчика HC-SR04 используются контакты D10 и D11 блока. Питание устройства осуществляется с помощью подключения к общей шине питания с разными потенциалами (+5В и GND).

Устройство состоит из 5 резисторов с сопротивлением 10 кОм, 3 резисторов с сопротивлением 75 кОм, резисторов с сопротивлениями 1 кОм, 3.9 кОм, 4.7 кОм, 6.2 кОм, 7.5 кОм, 56 кОм и 300 кОм; 3 конденсаторов с емкостью 10 нанофарад, 2 конденсаторов с емкостью 1 нанофарад; 4 операционных 4-канальных усилителей LM324, а также микросхем управления MAX232, EM78P153S. Излучателем является пьезодинамик, а приемным устройством – пьезомикрофон. К микросхеме EM78P153S подключен кварцевый генератор с частотой 27 МГц.

4. Блок управления микроконтроллером.

Как было ранее описано, блок управления микроконтроллером состоит из двух управляющих микросхем – 8-битных микроконтроллеров ATMEGA16U2 и ATMEGA328P. Первая из них производит управление питанием и шиной USB, а вторая – управление внешними устройствами, подключенными к контактам микросхемы. Питание микросхемами осуществляется либо с помощью гальванического элемента (батарея или аккумулятор), либо за счет подключения питания к разъему USB. Переключение между режимами питания осуществляется с помощью КМОП-транзистора.

Устройство состоит из 4 резисторов с сопротивлением 1 кОм, 3 резисторов с сопротивлением 10 кОм, 2 резисторов с сопротивлением 100 кОм, 2 резисторов с сопротивлением 22 Ом, 2 варисторов CG0603MLC-05E, ферритового сердечника-фильтра BLM21, 2 кварцевых генераторов с частотой 16 МГц, 8 конденсаторов с емкостью 100 нанофарад, 2 конденсаторов с емкостью 1 микрофарад; 2 поляризованных конденсатора с емкости 47 микрофарад, КМОП-транзистор FDN340P, кнопки-переключателя, 3 диодов типа CD1206-S01575, 1 светодиод желтого цвета. На плате имеются два операционных усилителя LMV358IDGKR, микросхемы NCP1117ST50T3G, LP2985-33DBVR для управления питанием микроконтроллера. Управляющими микросхемами микроконтроллера являются ATMEGA328P и ATMEGA16U2. Для подключения к шине USB на плате имеется разъем, обозначенный как X1\_USB.

Общая электрическая принципиальная схема представлена в приложении Б в формате A0. Перечень элементов представлен в приложении В на 3-х листах в формате А4.

# Глава 4. Интеллектуальный нечеткий алгоритма управления роботом

Нечеткая логика – раздел математики, использующий методы классической булевой логики, но расширенный с помощью семантических множеств, называемых лингвистическими переменными. В теории управления использование методов нечеткой логики позволяет разработать гибкий и несложный регулятор. Он широко используется в системах автоматики, робототехнике и приборостроении [5].

Суть данного подхода заключается в использовании вместо понятий «истина» (), «ложь» () из классической математической логики некие заданные множества значений (). Данное свойство алгоритмов нечеткой логики позволяет расширить диапазон возможных величин, подаваемых на вход и выход систем, и убрать возможные ограничения. На входных и выходных лингвистических переменных задаются диапазоны возможных состояний. Состояния выбираются с помощью функции принадлежности. Для формирования логических выводов разрабатывают правила связи входных и выходных величин [4,6].

Ниже в таблице 4.1 представлены состояния входных лингвистических переменных.

Таблица 4.1 – Входные функции принадлежности

|  |  |
| --- | --- |
| Наименование функции | Значения логической переменной |
| – «препятствие очень близко» | {0,…, 20} |
| *–* «препятствие близко» | {21, …, 30} |
| *–* «препятствие отсутствует» | {31, …, +∞} |

В разрабатываемом регуляторе на основе нечеткой логики будет 3 входных функции принадлежности с идентичным набором значений. Они применяются для измерительных устройств – дальномеров, установленных на корпусе колесного робота. Таким образом, количество переменных будет равно 9, они представлены в матрице входных логических переменных (4.1):

|  |  |  |
| --- | --- | --- |
|  |  | (4.1) |

По горизонтали расположены логические переменные каждого дальномера, по вертикали – номер логической переменной по каждому дальномеру.

Функция принадлежности «препятствие очень близко» выполняется в тот момент, когда расстояние от датчика до препятствия находится в диапазоне от 0 до 20 см.

Функция принадлежности «препятствие близко» выполняется в тот момент, когда расстояние от датчика до препятствия находится в диапазоне от 20 до 30 см.

Функция принадлежности «препятствие отсуствует» выполняется в тот момент, когда расстояние от датчика до препятствия более 30 см. Это соответствует тому, что движению робота ничего не мешает.

На рисунке 4.1 представлены входные функции принадлежности и наборы логических переменных для дальномера в виде графика в ПО MATLAB Fuzzy Logic Designer. Значения для остальных дальномеров идентичные.

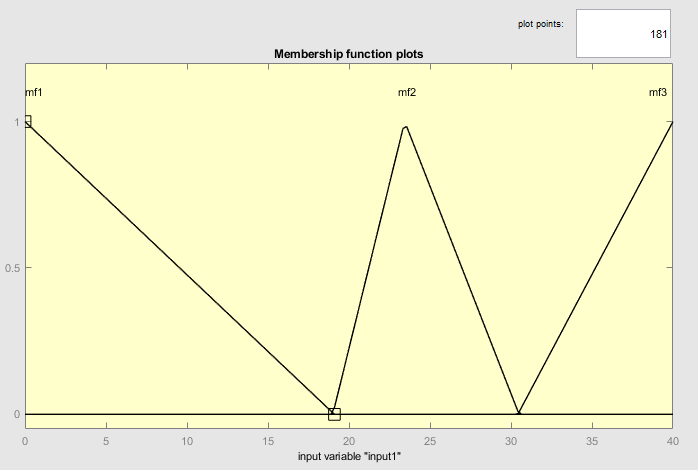


Рисунок 4.1 – Функции принадлежности входных переменных нечеткого регулятора

Далее рассчитаем возможные нечеткие логические выводы регулятора, представленные в таблице 4.2.

Таблица 4.2 – Выходные функции принадлежности.

|  |  |
| --- | --- |
| Наименование функции | Набор значений переменной |
| – «движение в обратном направлении» | {-0.6, -0.5, -0.4} |
| *–* «остановка двигателя» | {-0.05, 0, 0.05} |
| *–* «движение с малой мощностью (1/2 от полной)» | {0.4, 0.45, 0.5, 0.55, 0.6} |
| *–* «движение с полной мощностью» | {0.61, 0.7, 0.8, 0.9, 1} |

В разрабатываемом регуляторе на основе нечеткой логики будет 2 выходных множества с идентичным набором значений. Они применяются для исполнительных устройств – двигателей постоянного тока, установленных на корпусе колесного робота. Таким образом, количество переменных будет равно 8, они представлены в матрице выходных логических переменных (2.2):

|  |  |  |
| --- | --- | --- |
|  |  | (4.2) |

По горизонтали расположены логические переменные одного дальномера, по вертикали – количество дальномеров.

На рисунке 4.2 представлены значения выходных логических переменных для двигателя постоянного тока, расположенного на левой стороне шасси, в виде графика в ПО MATLAB Fuzzy Logic Designer. Значения для двигателя на правой стороне шасси идентичные.

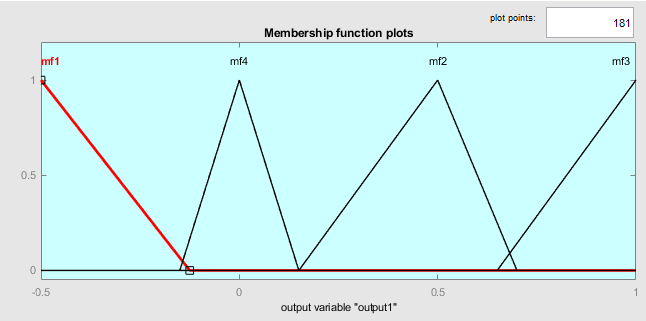


Рисунок 4.2 – Значения выходных переменных нечеткого регулятора

Для формирования связей между входными и выходными логическими терминами разрабатывается база правил нечеткого регулятора в форме «Если – То». Цель правила – создание четкого выбора значений переменных на основе наложенных на входную или выходную переменную ограничений. Между входными и выходными переменными могут использоваться логические операции «И» или «ИЛИ» [7].

У каждой выходной лингвистической переменной должно быть хотя бы одно правило. Каждое правило обозначается символом , где i – индекс правила, R – правило нечеткого регулятора. База правил представлена в таблице 4.3.

Таблица 4.3 – База правил нечеткого регулятора.

|  |  |  |
| --- | --- | --- |
| Входные лингвистические переменные | Выходные лингвистические переменные | Сформированное правило |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Ниже представлено краткое описание каждого правила в соответствии с показаниями датчиков и мощностей двигателей постоянного тока [5,6].

– правило «Задний ход». Если 3 дальномера одновременно получают сигналы о расстоянии менее 20 см, то мы можем предположить, что робот попал в тупик и ему необходимо выехать из него задним ходом.

правило «Препятствие слева». Если дальномер, расположенный на левом боку робота, показывает значение менее 20 см, то система управления блокирует поворот правого колеса двухколесного робота для совершения поворота налево.

– правило «Препятствие впереди». Если дальномер, расположенный на переднем борту робота, показывает значение менее 20 см, то система управления блокирует дальнейшее движение двух колесного робота.

правило «Препятствие справа». Если дальномер, расположенный на правом боку робота, показывает значение менее 20 см, то система управления блокирует поворот левого колеса двухколесного робота для совершения поворота направо.

правило «Препятствия отсутствуют». Каждый дальномер принимает значение более 30 см. Робот движется прямо на полной мощности.

правило «Препятствие слева приближается». Если дальномер, расположенный на левом боку робота, показывает значение менее 30 см, то система управления снижает скорость двух двигателей, чтобы обеспечить плавное вхождение в поворот при приближении к препятствию слева.

правило «Препятствие спереди приближается». Если дальномер, расположенный на передней части робота, показывает значение менее 30 см, то система управления снижает скорость двух двигателей, чтобы обеспечить плавное торможение робота.

правило «Препятствие справа приближается». Если дальномер, расположенный на правом боку робота, показывает значение менее 30 см, то система управления снижает скорость двух двигателей, чтобы обеспечить плавное вхождение в поворот при приближении к препятствию справа.

правило «Препятствие слева и спереди». Если дальномеры, расположенные на левом боку и в передней части робота, показывают значение менее 20 см, то система управления блокирует поворот правого колеса двухколесного робота для совершения поворота налево.

правило «Препятствие справа и спереди». Если дальномеры, расположенные на правом боку и в передней части робота, показывают значение менее 20 см, то система управления блокирует поворот левого колеса двухколесного робота для совершения поворота направо.

На рисунке 4.3 представлена база правил нечеткого регулятора в окне Rule Editor в ПО MATLAB Fuzzy Logic Designer.

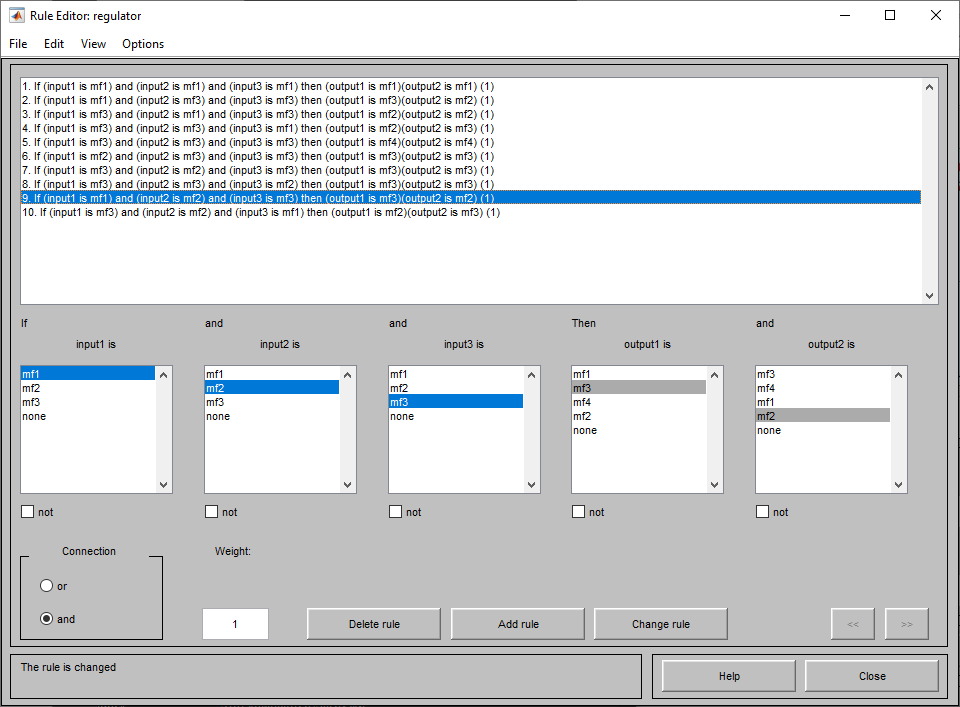


Рисунок 4.3 – База правил нечеткого регулятора

# Глава 5. Моделирование системы управления роботом с нечетким регулятором

Произведем моделирование системы управления мобильным колесным роботом в прикладном ПО MATLAB Simulink. Для этого нам необходимо использовать элементы модели двигателя постоянного тока, представленной в п. 1.1 главы 1, а именно: коэффициент передачи противоЭДС, передаточная функция системы. Модель системы управления представлена в Приложении А.

Модель управления колесным мобильным роботом состоит из следующих элементов: 2 двигателя постоянного тока, блок вычисления линейной и угловой скоростей, вычисление координат поворота тележки, курса робота, внешнее возмущение – микро-профиль дорожного покрытия и нечеткий алгоритм управления. Микро-профиль описан в п. 2.3, нечеткий алгоритм управления представлен в главе 4.

В таблице 5 представлены характеристики робота, дополняющие характеристики двигателя и применяемые в ходе моделирования в среде MATLAB Simulink. Файл с параметрами робота представлен в Приложении В.

Таблица 5.1 – Дополнительные характеристики робота.

|  |  |
| --- | --- |
| Наименование параметра (ед. измерения по СИ) | Значение параметра |
| Радиус колеса (метры) | R = 0.028 |
| Целевая координата по оси OX | X = 2 |
| Целевая координата по оси OY | Y = 2 |
| Время выполнения процесса (сек) | T = 20 |
| Минимальная погрешность ошибки рассогласования |  |
| Ширина колесной базы (метры) | B = 0.015 |

В качестве внешнего воздействия для дальномеров зададим следующие массивы данных в блоке повторяющихся исходных данных Repeating Sequence Stair из библиотеки элементов в ПО MATLAB Simulink. Массивы отображены в таблице 6. Каждый элемент массива является предположительным расстоянием до препятствия для каждой из указанных сторон [6].

Таблица 5.2 – Массивы длин расстояний дальномеров.

|  |  |
| --- | --- |
| Расположение дальномера | Массивы значений расстояния |
| На левой стороне | [35 33 40 27 25 22 19 18] |
| В передней части | [35 33 40 41 42 33 22 26 39 40 45 50 47 42 36 31 24 16] |
| На правой стороне | [35 30 25 22 19 15 18 24 25 26 32].' |

Как мы можем видеть, длина массива каждой из сторон имеет разную длину, что в определенных случаях может дать более реальную картину и отработать алгоритмы движения в обратном направлении (тупик перед роботом), остановки, замедления скорости или ускорения для планомерного достижения желаемой траектории движения.

На рисунках 5.1-5.3 представлены графики движения тележки от начала координат () к целевой точке и ошибки рассогласования координат с учетом микро-профиля асфальтированного покрытия. График движения вместе с окном работы нечеткого регулятора отображены на рисунке 3.3. В окне отображены входные и выходные лингвистические переменные, а также работа 10 заданных правил ( в базе правил нечеткого регулятора.

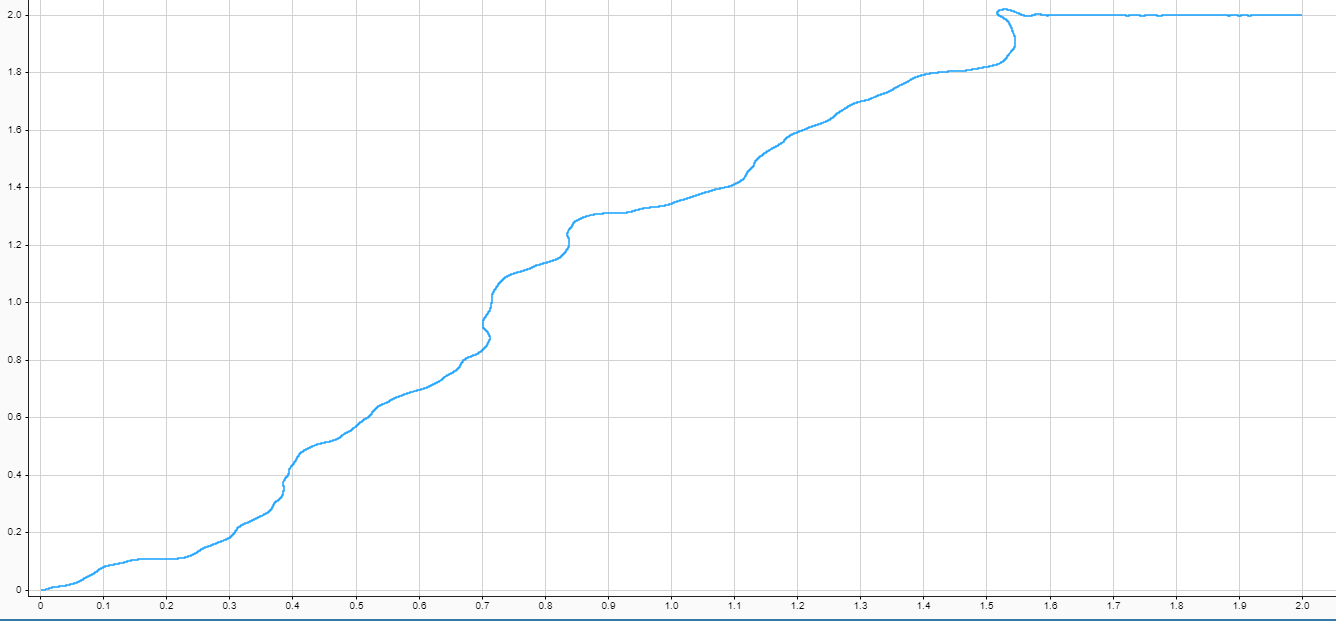


Рисунок 5.1 – Движение робота относительно координатной плоскости OXY.

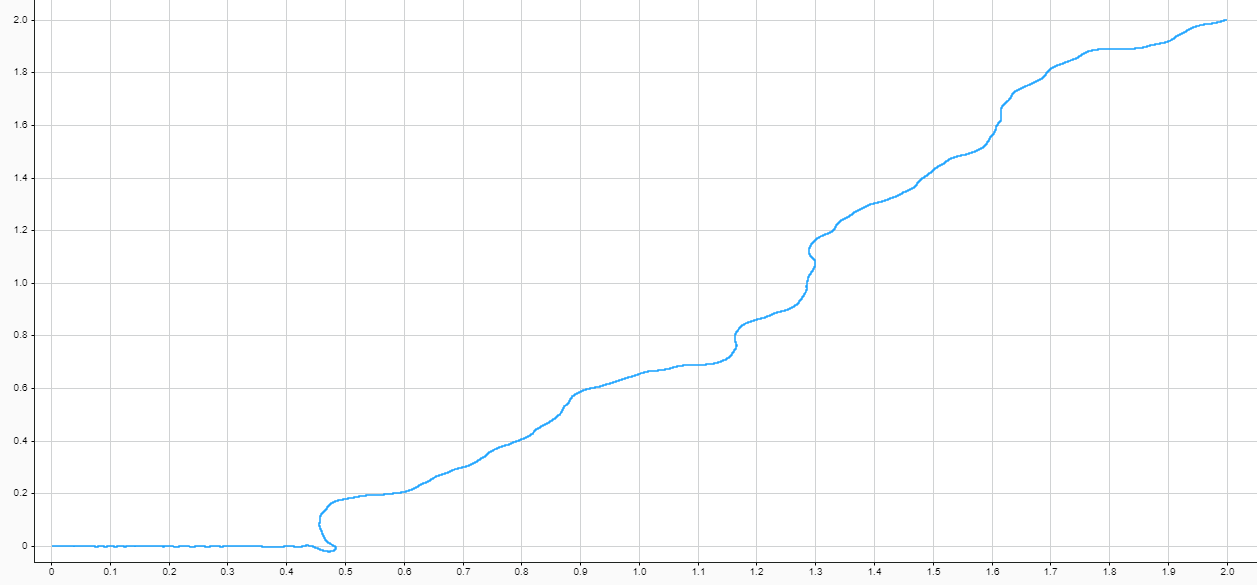


Рисунок 5.2 – Ошибка рассогласования навигации робота относительно координатной плоскости OXY.

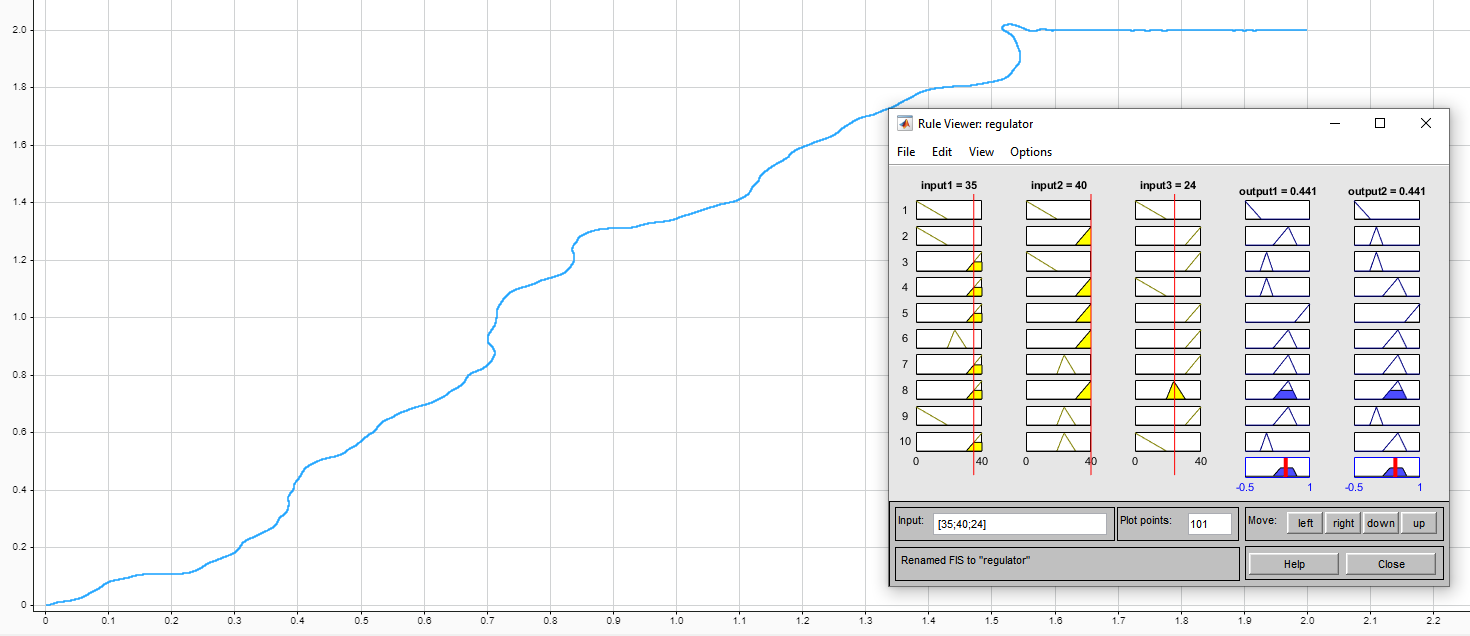


Рисунок 5.3 – Движение робота относительно координатной плоскости OXY с диалоговым окном нечеткого регулятора.

Как мы можем видеть, первые два графика схожи и отличаются лишь проекциями. Таким образом, можем сказать, что ошибка рассогласования – величина обратная координатам движения робота.

На рисунке 5.3 изображен график с регулятором. На нем для 3 входа указано значение 19, что соответствует малому расстоянию до препятствия справа. Таким образом, робот должен совершить поворот налево с помощью остановки двигателя на левой стороне и предварительно снизить скорость. Значение коэффициента мощности левого двигателя , а мощность правого двигателя равна ½ от полной мощности для плавного вхождения в поворот. На пути робота возникало множество препятствий, однако тележка выезжала на желаемую траекторию после объезда преграды [7].

Теперь проведем моделирование системы с микро-профилем дороги с булыжным покрытием. Для этого введем коэффициенты грунтового вида дороги в файл конфигурации робота из таблицы 2.1. Полученные результаты движения мобильного робота показаны на рисунках 5.4-5.6.

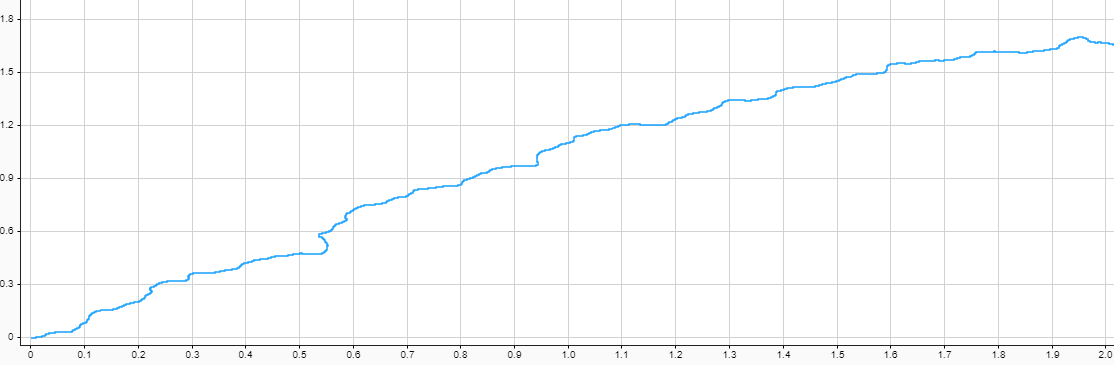


Рисунок 5.4 – Движение робота относительно координатной плоскости OXY.

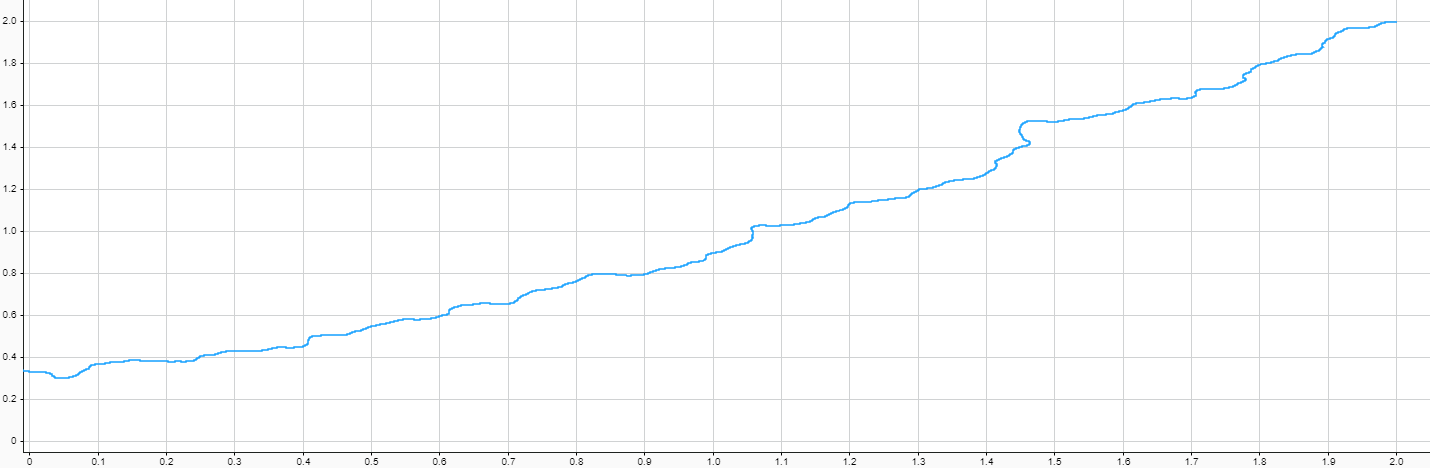


Рисунок 5.5 – Ошибка рассогласования навигации робота относительно координатной плоскости OXY.

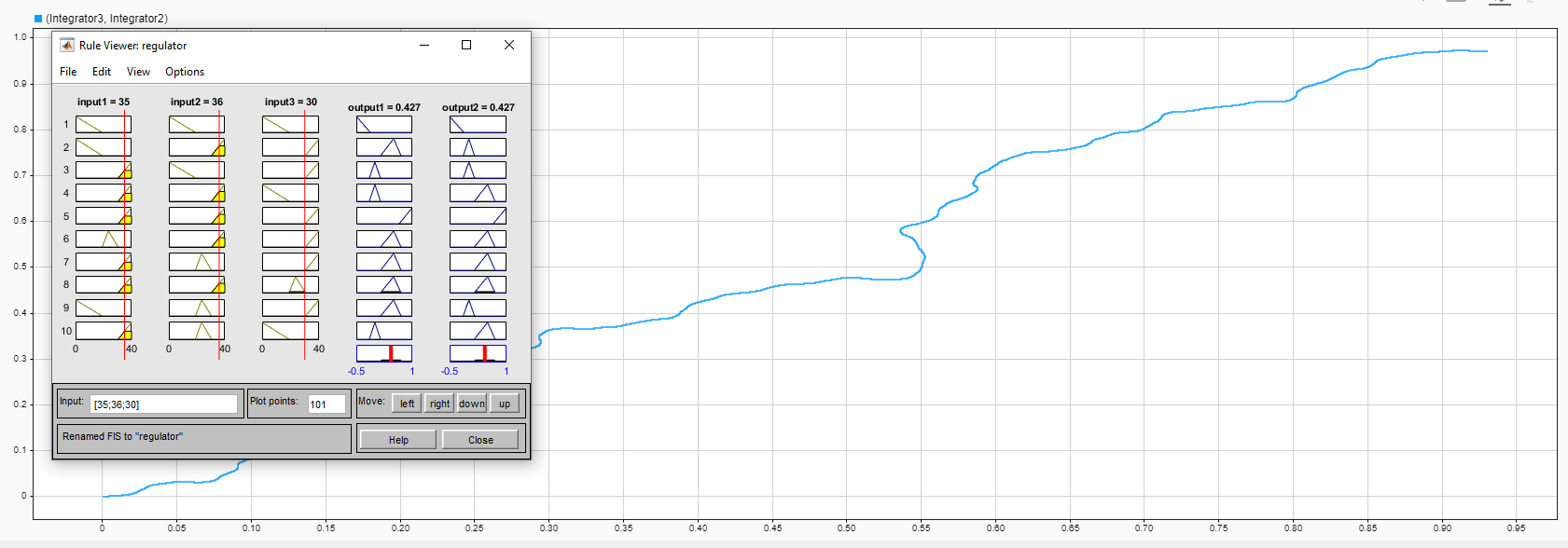


Рисунок 5.6 – Движение робота относительно координатной плоскости OXY с диалоговым окном нечеткого регулятора (булыжное покрытие).

Заметим, что в сравнении с асфальтированным покрытием робот на булыжном дорожном покрытии медленнее приходит к целевой точке из начала координат. Ему требуется больше усилий на преодоление возможных колебаний при движении к линии , однако он не смог ее достичь. Робот доехал к координатам .

На рисунке 5.6 изображен график с регулятором. На нем для каждого входа указано значение в диапазоне от 30 до 35, что соответствует возможному прохождению в узком пространстве с ограничением спереди. Таким образом, робот должен снизить скорость на обоих двигателях, чтобы аккуратно проехать препятствие. Как и на асфальтированном покрытии, робот свободно объезжает возникающие на пути помехи. Однако траектория передвижения сместилась и движение усложнилось из-за проезда по булыжному покрытию.

Также исследуем влияние грунтового дорожного покрытия на управление и навигацию мобильного колесного робота. Для этого вновь введем параметры из таблицы 1 уже грунтового покрытия. Путь робота и ошибка рассогласования изображены на рисунках 5.7 – 5.9.

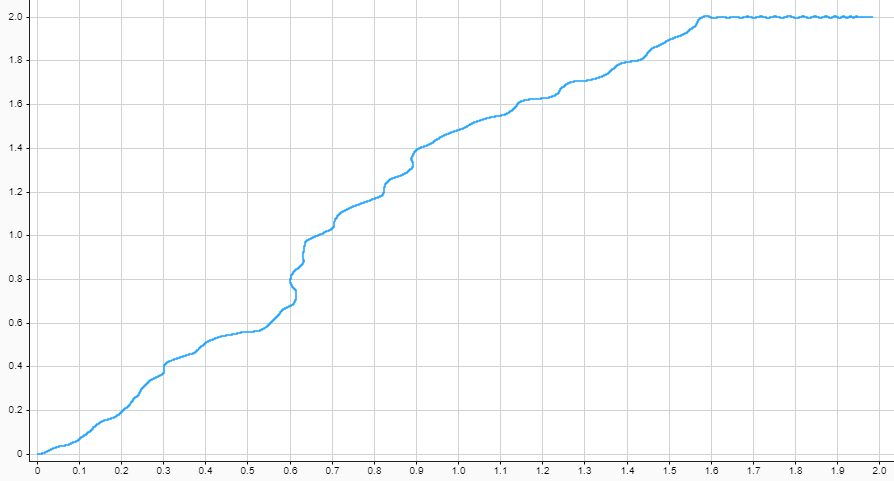


Рисунок 5.7 – Движение робота относительно координатной плоскости OXY (грунтовое покрытие).

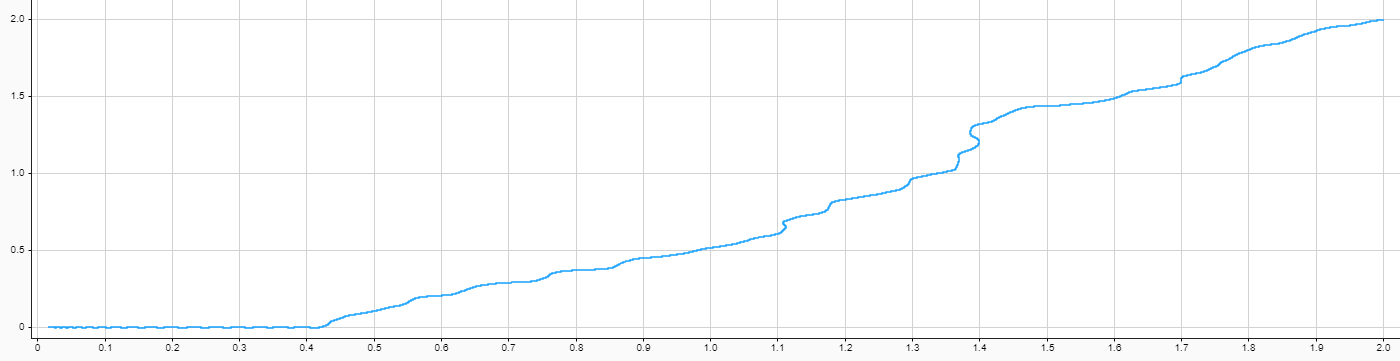


Рисунок 5.8 – Ошибка рассогласования навигации робота относительно координатной плоскости OXY (грунтовое покрытие).

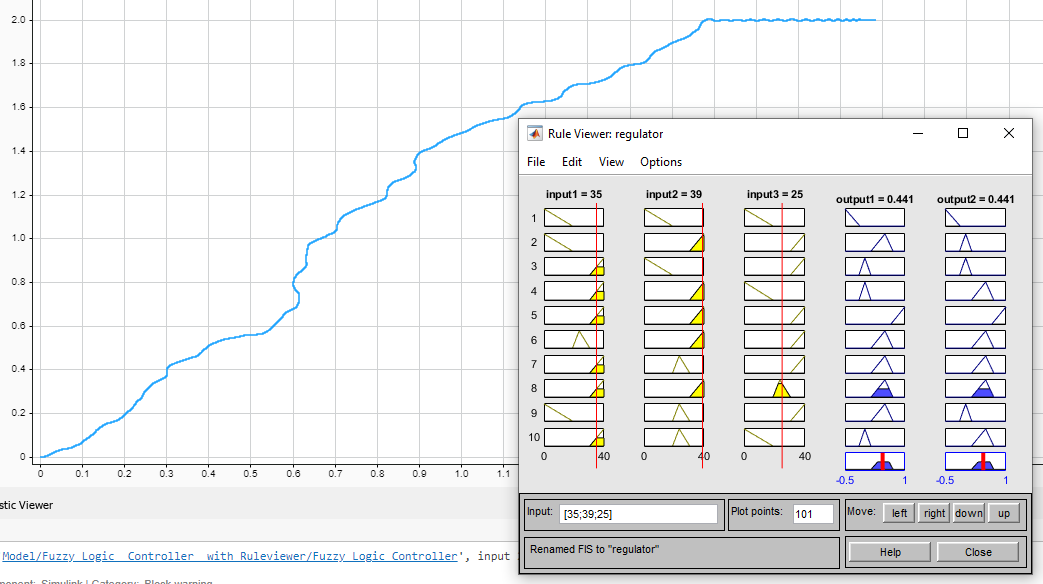


Рисунок 5.9 – Движение робота относительно координатной плоскости OXY с диалоговым окном нечеткого регулятора (грунтовое покрытие).

Сравнивая три вида дорожного покрытия, мы можем прийти к следующим выводам:

1. На асфальтовом покрытии выезд робота к целевой точке происходит с наименьшими регулировкой и формированием траектории движения, что обеспечивает длительное прямолинейное достижение целевых координат и прямолинейный набор скорости для выхода на координату . Оставшийся путь до целевой точки наиболее короткий, т.к. робот по касательной проехал наибольшее расстояние.

2. На булыжном покрытии движение робота к целевой точке проходит с наибольшим временем регулирования и нестабильным формированием траектории движения. Это показывает, что покрытие значительно влияет на управляемость и координацию движения, поскольку лишь малый участок пройденного пути является прямолинейным, в отличие от асфальтового покрытия. Из-за серьезного влияния покрытия роботу сложнее по касательной выехать на желаемую траекторию, что влияет на длину остаточного пути после выхода на целевую координату по оси ОY (). Роботу не удалось достичь желаемой координаты по оси OY ().

3. На грунтовом покрытии робот движется по аналогии с асфальтированной дорогой с толь лишь разницей, что влияние чуть нестабильного покрытия незначительно повлияло на формирование траектории движения робота по касательной. Таким образом время выхода на желаемую координату по оси OY () меньше лишь на 0.5 см.

# 6. Разработка и тестирование программного обеспечения управления и диспетчеризации колесного робота

## 6.1 Функциональные требования к программному обеспечению

При формировании технического задания по разработке программно-аппаратного комплекса большое значение уделяется функциональной части. Она составляет базу, на которой формируются основные требования к функциям и возможностям разрабатываемого прибора: методы управления исполнительными устройствами, приема и передачи сигналов по измерительным датчикам и излучателям, внутренние процессы, протекающие во время работы прибора и т.д.

Разрабатываемый проект состоит из 3 основных частей:

1. Модуль управления электродвигателями

Данный модуль состоит из платы Arduino Uno, к которой подключены электродвигатели. Управление производится с помощью нечеткого регулятора, разработанного в главе 4.

1. Модуль приема-передачи данных на роботе

Блок управления приемом-передачей данных на роботе осуществляется с помощью микросхемы ESP8266. Она подключается к плате Arduino Uno через контакты RX-TX для приема-передачи данных между двумя микроконтроллерами. На плате ESP-01 (минимальная версия микроконтроллера ESP8266) располагается модуль беспроводной передачи Wi-Fi. Через него на сервер поступает информация, собранная при передаче данных между двумя микроконтроллерами.

1. Модуль приема-передачи данных на сервере.

Сервер в данной системе является управляющим звеном. Он имеет графический интерфейс, базу данных с роботами и функциональные компоненты для взаимодействия с роботами. С помощью сервера производится управление парком роботов: подключение к роботу, проверка сигналов с датчиков приближения, перевод в ручной режим управления, разработка маршрутов и формирования траектории движения. Также доступна возможность добавлять новые роботы и удалять из системы вышедшее из строя оборудование.

При формировании функциональных требования в рамках ВКР следует выделить следующие категории:

1. Управление электродвигателями:

– движение в прямом направлении;

– движение с поворотом налево;

– движение с поворотом направо;

– движение в обратном направлении;

1. Управление приемом-передачей информации на роботе:

– прием данных с датчиков приближения;

– прием закодированных управляющих инструкций с сервера;

– расшифрование управляющих инструкций и передача на блок управления двигателями;

– передача данных с датчиков приближения на сервер;

– кодирование результатов работы с блока управления двигателями;

– передача результатов работы двигателей на сервер;

1. Управление приемом-передачей информации на сервере:

– создание записей о роботе;

– показ списка роботов, доступных к подключению;

– подключение к роботу для взаимодействия;

– прием данных с датчиков робота;

– формирование управляющих инструкций и маршрутов для робота;

Кодирование сигналов производится в модуле приема-передачи информации и на сервере. Используемым типом кодировки является стандарт кодирования ASCII символов Base64. Данная система содержит латинские символы A-Z, a-z и цифры от 0-9 и является наиболее распространенной в мире.

Кодирование сигналов позволяет повысить качество передачи информации между роботом и сервером за счет точного указания источника сигнала, например, от робота к серверу и в обратную сторону. Для этого необходимо определить основные характеристики кодируемого сигнала

1) Идентификатор робота;

2) Тип сигнала (датчик, направление движения, обозначение батареи);

3) Текущее значение сигнала (расстояние до препятствия, текущий заряд батареи, заданное движение робота).

4) Подтверждение выполнения операции (только для управляющих сигналов с сервера)

Идентификатор робота задается при загрузке программного обеспечения вместе с IP-адресом устройства для внесения в общую базу данных всех роботов.

В таблице 6.1 представлены все возможные обозначения типов сигналов, которые используются при передаче данных по MQTT протоколу между роботом и устройством.

Таблица 6.1 – Обозначения типов сигналов

|  |  |  |
| --- | --- | --- |
| Обозначение сигнала | Краткое описание | Источник сигнала |
| BATT | Уровень заряда батареи | Робот |
| WF | Режим автоматического управления | Робот |
| BT | Режим ручного управления | Сервер |
| SENL | Датчик препятствия на левом борту | Робот |
| SENR | Датчик препятствия на правом борту | Робот |
| SENP | Датчик препятствия на переднем борту | Робот |
| MOVP | Движение прямо | Робот |
| MOVL | Движение влево | Робот |
| MOVR | Движение вправо | Робот |
| MOVB | Движение в обратном направлении | Робот |
| MOVPS | Движение прямо | Сервер |
| MOVLS | Движение влево | Сервер |
| MOVRS | Движение вправо | Сервер |
| MOVBS | Движение в обратном направлении | Сервер |
| OK | Ответ робота на управляющий сигнал (только при задании движения с платформы) | Робот (конец сигнала) |

С помощью различных типов сигналов формируют комбинации, передаваемые по сети через протокол MQTT и программу-сервер MQTT Broker. Возможные комбинации представлены в таблице 6.2. В качестве идентификатора робота используется кодовое слово ROBOT1.

Таблица 6.2 – Комбинации сигналов

|  |  |  |
| --- | --- | --- |
| Комбинация сигнала | Расшифровка сигнала | Источник сигнала |
| ROBOT1BATT95 | Текущий заряд батареи равен 95% | Робот с идентификатором ROBOT1 |
| ROBOT1WF1 | Включен режим автоматического управления по Wi-Fi | Робот с идентификатором ROBOT1 |
| ROBOT1BT1 | Включен режим ручного управления по Bluetooth | Сервер |
| ROBOT1SENL35 | Расстояние до препятствия по левому борту равно 35 см | Робот с идентификатором ROBOT1 |
| ROBOT1SENR15 | Расстояние до препятствия по правому борту равно 15 см | Робот с идентификатором ROBOT1 |
| ROBOT1SENP29 | Расстояние до препятствия по переднему борту равно 29 см | Робот с идентификатором ROBOT1 |
| ROBOT1MOVP40 | Движение прямо на расстоянии 40 см | Робот с идентификатором ROBOT1 |
| ROBOT1MOVL5 | Движение влево на расстоянии 5 см | Робот с идентификатором ROBOT1 |
| ROBOT1MOVR8 | Движение вправо на расстоянии 8 см | Робот с идентификатором ROBOT1 |
| ROBOT1MOVB3 | Движение в обратном направлении на расстоянии 3 см | Робот с идентификатором ROBOT1 |
| ROBOT1MOVPS100 | Движение прямо на расстоянии 40 см | Сервер |
| ROBOT1MOVLS5 | Движение влево на расстоянии 5 см | Сервер |
| ROBOT1MOVRS10 | Движение вправо на расстоянии 10 см | Сервер |
| ROBOT1MOVBS13 | Движение в обратном направлении на расстоянии 13 см | Сервер |
| ROBOT1MOVPS22OK | Ответ робота на выполненный управляющий сигнал «движение прямо на расстоянии 22 см» | Робот с идентификатором ROBOT1 |
| ROBOT1MOVLS4OK | Ответ робота на выполненный управляющий сигнал «движение прямо на расстоянии 22 см» | Робот с идентификатором ROBOT1 |
| ROBOT1MOVRS | Ответ робота на выполненный управляющий сигнал «движение прямо на расстоянии 22 см» | Робот с идентификатором ROBOT1 |
| ROBOT1MOVBS | Ответ робота на выполненный управляющий сигнал «движение прямо на расстоянии 22 см» | Робот с идентификатором ROBOT1 |

Одним из наиболее важных факторов является частота получения или отправки сигнала. Он может отправляться каждые несколько секунд или по окончании какого-либо действия (например, завершение движения по указанной траектории). Вариативность частот отображена в таблице 6.3.

Таблица 6.3 – Частота сигналов

|  |  |
| --- | --- |
| Обозначение сигнала | Частота отправки |
| BATT | Каждые 30 секунд |
| WF | При переходе в режим автоматического управления |
| BT | При переходе в режим ручного управления |
| SENL | Каждые 2 секунды |
| SENR | Каждые 2 секунды |
| SENP | Каждые 2 секунды |
| MOVP | По окончании движения по заданной инструкции |
| MOVL | По окончании движения по заданной инструкции |
| MOVR | По окончании движения по заданной инструкции |
| MOVB | По окончании движения по заданной инструкции |
| MOVPS | При отправке команды с сервера |
| MOVLS | При отправке команды с сервера |
| MOVRS | При отправке команды с сервера |
| MOVBS | При отправке команды с сервера |
| OK | По окончании движения по заданной инструкции |

Примером закодированного в кодировку Base64 сигнала является комбинация индикации текущего заряда батареи. Сформированный сигнал имеет длину на 1/3 больше исходного.

Исходный сигнал, сформированный роботом:

|  |  |  |
| --- | --- | --- |
|  | ROBOT1BATT95 |  |

Закодированный сигнал, полученный в ходе передачи данных:

|  |  |  |
| --- | --- | --- |
|  | Uk9CT1QxQkFUVDk1 |  |

Для каждой функциональной части формируется перечень технологий разработки программного обеспечения. С помощью него проводится алгоритмизация функциональных требований, разработка программного обеспечения и отладка взаимодействия между устройствами, физическими элементами робота.

## 6.2 Перечень технологий программного обеспечения

Перечень (далее стек) технологий представляет собой совокупность всех применяемых в ходе разработки языков программирования, средств управления базами данных и сред разработки и написания, анализа программного кода [12-13]. Это является основным требованием при разработке программного продукта. Стек технологий позволяет нам подобрать необходимые дополнения, модули, облегчающие и ускоряющие разработку программного продукта. Также он подбирается исходя из параметров и требований задачи.

Данный проект состоит из трех функциональных модулей: Web-приложение для управления роботами с помощью компьютера, СУБД для хранения данных о состоянии датчиков, пройденном расстоянии и общей информации о роботах, а также программного модуля управления блоком приема-передачи информации между роботом и Web-приложением.

Выбранные технологии представлены в таблице 6.4.

Таблица 6.4 – Выбранные технологии для разработки ПО

|  |  |
| --- | --- |
| Компоненты ПО | Наименование технологий |
| Web-приложение | Python 3.10, Django 4.0 |
| Управление двигателями робота | C++, Arduino Library |
| Управление передачей данных по Wi-Fi | C++, ESP8266 Wi-Fi Library |
| Протокол взаимодействия робота и сервера | Message Queuing Telemetry Transport (далее MQTT) |
| СУБД | SQLite3 |
| Среда разработки Web-приложения | JetBrains PyCharm Professional |
| Среда разработки ПО робота | Microsoft Visual Studio Code |
| Операционная система | Ubuntu 22.04, Docker (для MQTT протокола) |

Ниже представлено более подробное описание каждой из выбранных технологий.

1) Web-приложение представляет собой монолитное приложение. Оно базируется на библиотеке Django 4.0 языка программирования Python.

Python – многопарадигмальный интерпретируемый язык программирования с полностью открытым исходным кодом, позволяющим создавать приложения для любых задач, особенно в области математического моделирования, разработки Web-приложений и робототехники (операционная система ROS).

Django – библиотека языка программирования Python для разработки универсальных Web-приложений: монолитных, двухзвенных или трехзвенных. Для отображения интерфейса используется собственная разработка библиотеки, включающая в себя языки разметки HTML, CSS и язык программирования JavaScript для управления кнопками, графиками внутри Web-страниц.

2) Управление двигателями робота

С++ – многопарадигмальный компилируемый язык программирования, имеющий наиболее широкое распространение и применение: от небольших встраиваемых систем, роботов до системы управления крупным промышленным оборудованием, проектирования и моделирования сложных математических, физических расчетов, а также операционные системы для компьютеров и встраиваемых систем. Имеет большое количество дополнений, библиотек по работе с оборудованием.

Arduino Library – библиотека языка программирования C++ для управления микроконтроллерами Arduino. Она позволяет управлять исполнительными элементами микроконтроллера: таймерами, прерываниями, внешними периферийными устройствами, ШИМ и тактовым генератором. Она позволяет создать сбалансированную программу для микроконтроллера, оптимальную по использованию внутренних ресурсов: флеш-памяти, ПЗУ, таймеров и т.д.

В рамках данного проекта производится управление двигателями через выходные внешние контакты, по которым подключены устройства, и поворот колес с помощью прерываний, срабатывающие при появлении препятствий. Микроконтроллер каждые 0,5 секунд обращается к входным внешним контактам и снимает показания с датчиков, расположенных на трех бортах тележки.

3) Управление передачей данных по Wi-Fi

Библиотека ESP8266 Wi-Fi Library позволяет управлять модулем беспроводной связи Wi-Fi микроконтроллера ESP8266 средствами языка C++. Она включает в себя работу с HTTP-соединениями, передаче данных по технологиям WebSocket, MQTT.

В рамках ВКР микроконтроллер ESP-01 (уменьшенная версия ESP8266) подключается к линии RX-TX микроконтроллера Arduino Uno для взаимодействия между сервером и роботом. Микроконтроллер подключается к заданной Wi-Fi-сети и MQTT-серверу для обмена данными. При получении информации от микроконтроллера Arduino Uno модуль ESP-01 шифрует данные и по протоколу MQTT отправляет сообщение на сервер.

4) Протокол MQTT для передачи данных между роботом и сервером

Message Queuing Telemetry Transport (далее MQTT) – легковесный протокол передачи сообщений между IoT устройствами. Преимущества использования этого решения заключается в том, что робототехнические устройства имеют ограниченные ресурсы и пропускную способность передачи данных. Протокол MQTT использует минимальные ресурсы оборудования и требования к передаче данных (минимально 2 байта) между устройствами. Среди робототехнических устройств данная технология является стандартом де-факто по передаче данных в режиме реального времени по проводным и беспроводным соединениям Ethernet, Wi-Fi.

В рамках ВКР по протоколу MQTT передаются данные между устройствами и сервером. В качестве MQTT-сервера используется программа Eclipse Mosquitto, которое выполнено в виде Docker-контейнера.

5) СУБД SQLite3

Данная СУБД является компактной встраиваемой технологией, размещающейся на Web-сервере совместно с веб-приложением. Также данная библиотека позволяет упрощать программу, уменьшать время отклика и доступа между компонентами Web-приложения. Важной особенностью СУБД SQLite является компактность, простота развертывания и полная интеграция с языком программирования Python.

В рамках ВКР в СУБД SQLite3 хранятся данные о роботах, пройденных маршрутах, показаний датчиков каждого из роботов в разный период времени использования. Также в базе данных хранится служебная информация библиотеки Django.

6) Доступные инструменты разработки:

– Инструменты для разработки на языке программирования Python – JetBrains PyCharm Professional, Microsoft Visual Studio 2022, Python IDLE, Microsoft Visual Code;

– Инструменты для разработки на языке программирования C++ – Arduino IDE, Microsoft Visual Studio Code, Programino Arduino IDE, PlatformIO;

– Инструменты для взаимодействия с СУБД SQLite3 – библиотека sqlite для Python, SQLite Administrator.

7) Операционные системы

В качестве операционной системы для развертывания и эксплуатации Web-приложения, MQTT-сервера и средств отладки, разработки на Arduino используется GNU/Linux Ubuntu 22.04.

## 6.3 Разработка программного обеспечения

Платформа управления роботами представляет собой монолитное Web-приложение, которое взаимодействует с устройствами при помощи протокола MQTT. Исходный код Web-приложения представлен в Приложении Б.

Главной страницей платформы является список всех роботов. Он состоит из навигационного меню по всей системе, бокового меню управления роботами. В него входят пункты списка всех устройств на складе и добавления туда нового оборудования. На рисунке 6.1 представлен интерфейс списка всех роботов на складе. Каждая единица списка выполнена в виде отдельной структуры, состоящей из элементов: пиктограмма, ID, IP-адрес устройства и идентификатор робота. В нижней части отображаются текущий заряд батареи робота и кнопка «Подробнее» для перехода к странице управления роботом.

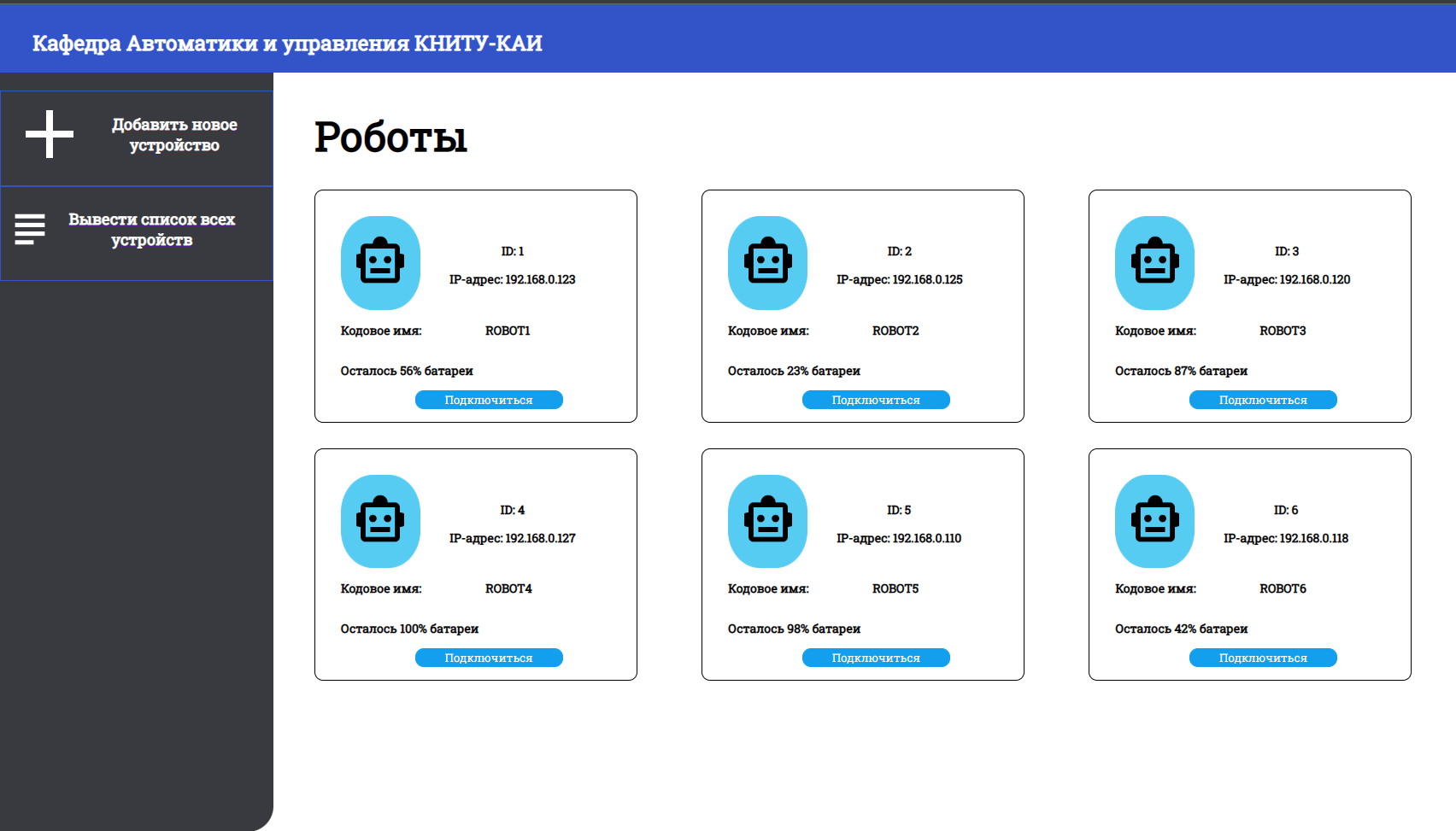


Рисунок 6.1 – Главная страница платформы

Добавление робота в систему проиллюстрировано на рисунке 6.2. Оно предлагает пользователю ввести основные данные: IP-адрес устройства и идентификатор робота. ID устройству присваивается автоматически при добавлении объекта в список всех роботов.

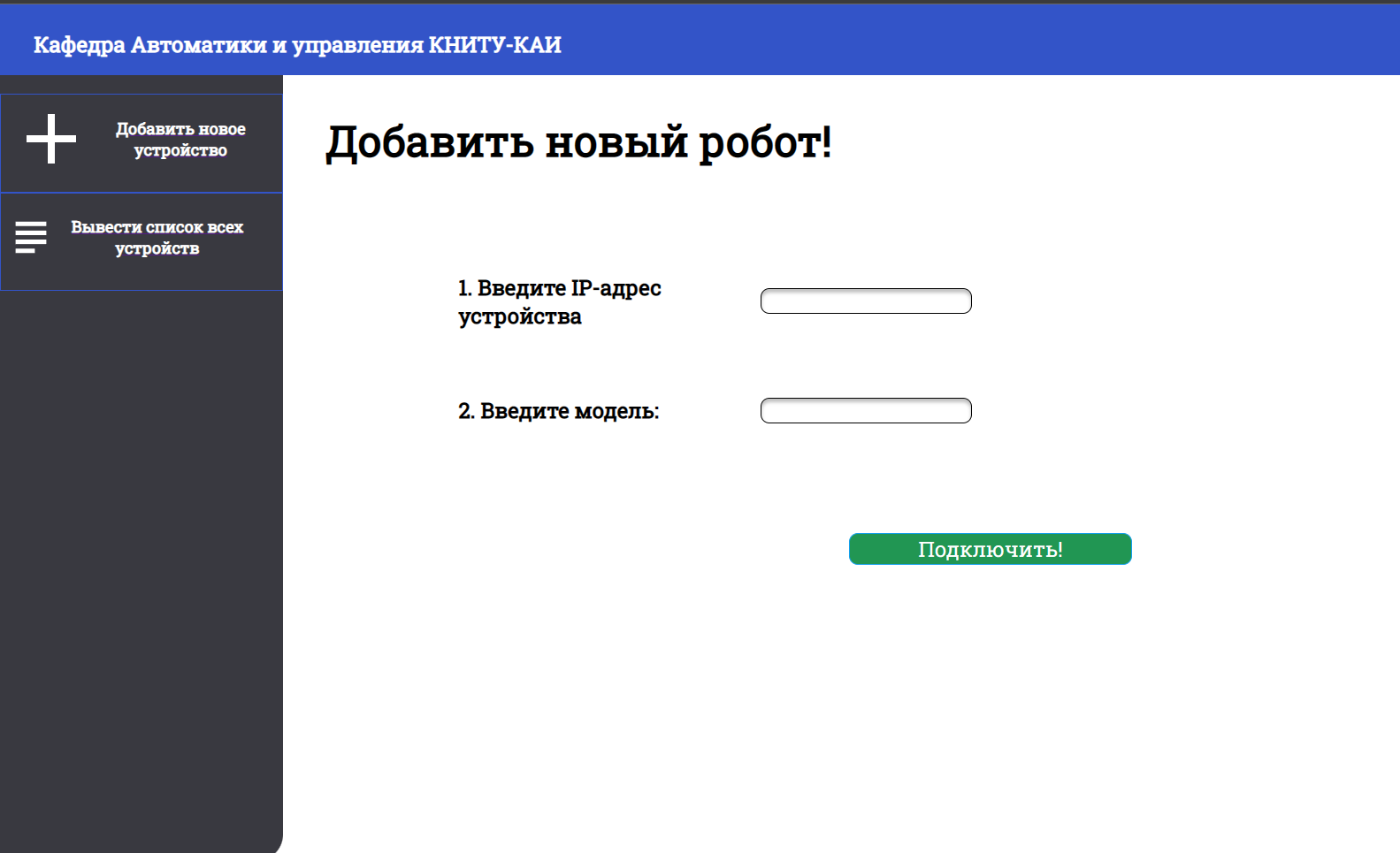


Рисунок 6.2 – Страница добавления робота

На рисунке 6.3 представлена страница выбранного робота, к которому пользователь подключается. Страница поделена на три блока: показания датчиков расстояния, основная информация о роботе и кнопки взаимодействия с роботом.

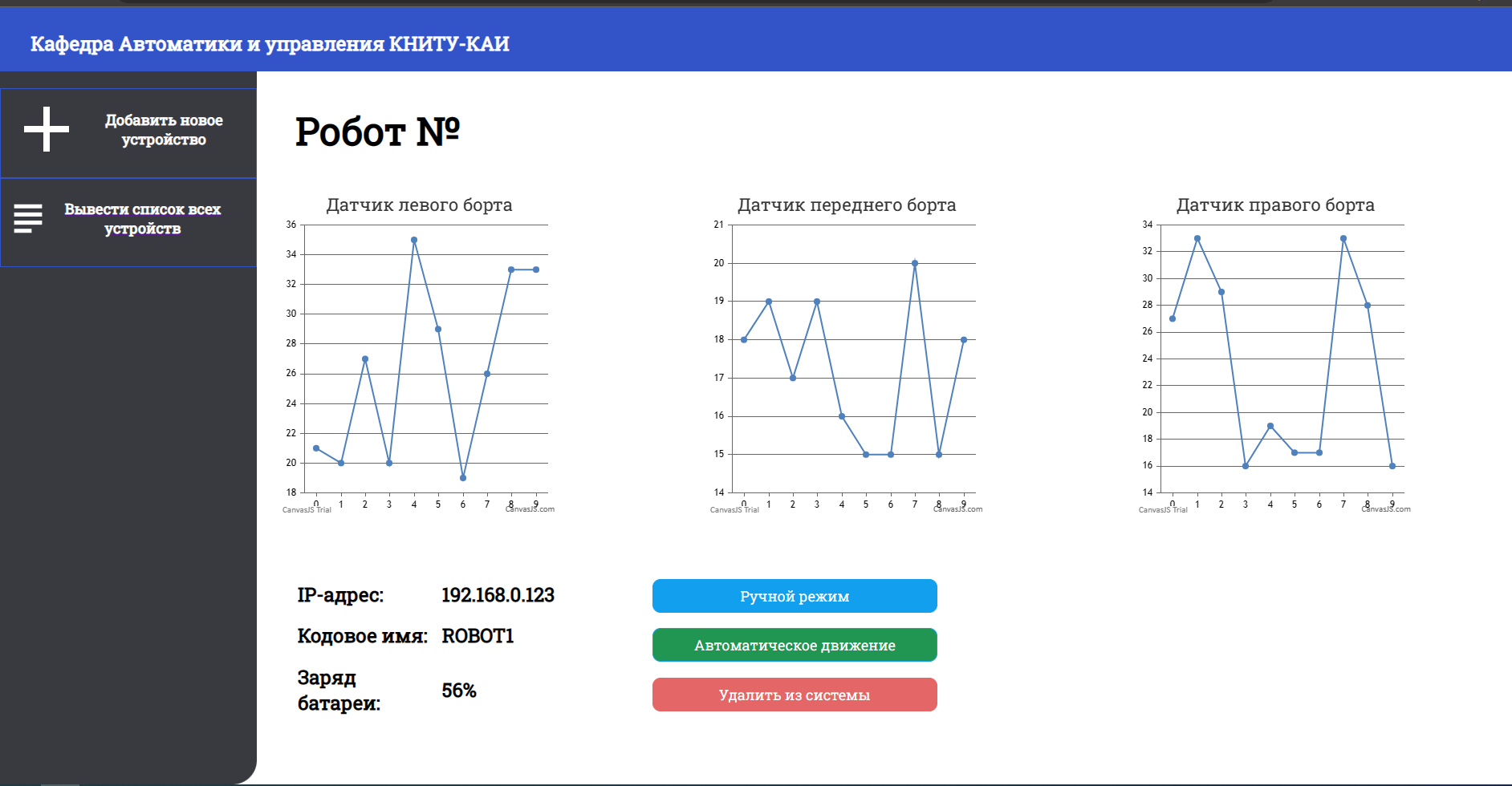


Рисунок 6.3 – Подробная информация о роботе

В блоке показаний датчиков расстояния пользователь видит графики по каждому из датчиков, установленных на роботе. Графики обновляются каждые 0.5 секунд по мере получения новых показаний с сенсоров робота.

Раздел с основной информацией о роботе позволяет пользователю ознакомиться с характеристиками робота: идентификатор, IP-адрес и уровень заряда. В правом блоке расположены 3 кнопки: ручной режим, автоматическое движение и удаление робота из системы. Рассмотрим каждую из них подробнее.

Ручной режим управления позволяет перевести робота в режим управления с помощью Bluetooth-контроллера, расположенного на шасси робота. Робот автоматически переводит Wi-Fi-контроллер в режим пониженного напряжения для снижения потребления энергии. Возврат к управлению с помощью Wi-Fi производится нажатием на тактовую кнопку на корпусе робота. При переключении в данный режим работы у пользователя появляется страница, представленная на рисунке 6.4.



Рисунок 6.4 – Страница ручного режима работы

Автоматическое движение – режим работы робота, в котором маршруты и движения робота задаются с помощью платформы. Перед пользователем появляется график и таблица с координатами X, Y, действиями и кнопками «Добавить» и «Удалить».

Кнопка «Добавить» увеличивает количество строк в таблице в зависимости от количества нажатий. Кнопка «Удалить» удаляет все созданные строки таблицы, кроме первой. Среди действий можно выбрать движение в прямом направлении, поворот налево, поворот направо и движение задним ходом.

Под таблицей расположена кнопка «Начать движение». Она сохраняет заданных маршрут в базу данных, после чего поочередно отправляет каждую управляющую инструкцию на приемник робота. Как только платформа получает положительный результат выполнения команды на роботе, она высылает следующую команду. На графике отображаются заданные пользователем и полученные с робота координаты. Их траектория может отличаться, поскольку робот объезжает препятствия в ходе прохождения заданного маршрута.

**Выводы по главе**

В данной главе был проведен подробный анализ имитационных моделей. Была представлена имитационная модель бизнес-процесса, разработанного в программной среде AnyLogic.

Также представлен стек технологий (инструментарий) для разработки ПО. Представлены аргументы по основным критериям, объясняющие выбор данных решений.

Была рассмотрена автоматизированная система управления материально-техническим обеспечением. Продемонстрирован интерфейс разработанного программного продукта. Показаны функциональные особенности и возможности данного решения.