

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение  
высшего образования

«Тольяттинский государственный университет»

Институт машиностроения

(наименование института полностью)

Кафедра

«Промышленная электроника»

(наименование)

11.04.04 Электроника и наноэлектроника

(код и наименование направления подготовки)

Электронные приборы и устройства

(направленность (профиль))

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)**

на тему «Автоматизированный транспортный мини-полигон»

Обучающийся

Д.Г. Токарев

(Инициалы Фамилия)

(личная подпись)

Научный  
руководитель

д-р техн. наук, доцент В.П. Певчев

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2022

## Оглавление

Введение . . . . .	3
Глава 1 Общая компоновка полигона, транспортного средства и возможные принципы его ориентации в пространстве . . . . .	6
Глава 2 Программно-аппаратный комплекс мини-полигона . . . . .	19
2.1 Технические средства мини полигона. . . . .	19
2.2 Основные системные и учебно-тренировочные задачи. . . . .	21
2.3 Корректировка траекторий движения транспортного средства . . . . .	24
2.4 Перспективное развитие технических средств мини-полигона . . . . .	24
2.5 Схемы фрагмента мини-полигона и комплектующие . . . . .	25
2.6 Конструктив программно-аппаратного комплекса мини-полигона. . . . .	33
2.7 Программное обеспечение комплекса мини-полигона . . . . .	34
Глава 3 Развитие беспилотного транспортного средства на основе модели транспортного средства мини-полигона . . . . .	51
3.1 Состав комплекса транспортного средства. . . . .	51
3.2 Структура элементов пульта мануального управления. . . . .	55
3.3 Математическая модель и алгоритм бортового управления транспортным средством . . . . .	57
3.4 Структура пульта управления транспортным средством . . . . .	65
3.5 Управление скоростью транспортного средства . . . . .	65
3.6 Управление направлением движения . . . . .	67
Заключение. . . . .	69
Список используемой литературы и используемых источников . . . . .	70
Приложение А Перечень элементов мини-полигона. . . . .	75

## **Введение**

Подготовка кадров, ориентированных на производственную и творческую проектную деятельность – актуальная задача, решение которой во многом определяет развитие экономики. Данная задача является одним из направлений стратегии перспективного развития России – Национальной технологической инициативы (НТИ), которому прогнозируется наибольший глобальный рынок: «Автонет», беспилотные автомобили, интеллектуальные транспортные системы.

В «дорожной карте» направления, в числе приоритетов, указано: создание системы подготовки профессиональных кадров.

Для обеспечения решения поставленных задач разработаны проектные решения транспортного мини-полигона Автонет – учебно-технической базы ресурсных центров, детских технопарков «Кванториум», учреждений образования. Проект соответствует дорожной карте НТИ, в которой в разделе «Направления реализации плана мероприятий», указано:

- создание сети мероприятий, направленных на решение технологических задач НТИ;
- разработка технологических решений для внутреннего рынка;
- интеграция технологических вызовов НТИ в систему образования.

Транспортный мини-полигон предназначен для решения нескольких задач, соответствующих направлению «Автонет» НТИ.

Основное функциональное назначение мини-полигона:

- демонстратор беспилотного перемещения автомобиля;
- тренажер оператора беспилотного автомобиля;
- тренажер системного интегратора специального программного обеспечения;
- АРМ программиста;
- испытательный комплекс для отработки имитаторов внешней среды и беспилотных транспортных средств.

Комплекс таких глобальных задач, соответствующих назначению разработки при выполнении магистерской диссертации сводится к реализации комплексной задачи, определяющей общую эффективность всех перечисленных функций.

Спроектированная концепция мини-полигона предназначена для практического решения рациональной логистической маршрутизации беспилотных мобильных платформ.

Оттачивание логистических навыков в условиях мини-полигона позволит также значительно минимизировать аварийный результат неверных логистических решений.

Таким образом, целью магистерской диссертации является разработка системы позиционирования модели транспортного робота с логистической маршрутизацией с помощью HMI-интерфейса.

Для реализации цели магистерской диссертации предлагается комплекс реализуемых задач, в состав которого входят:

- проектирование конструктива транспортного средства;
- разработка алгоритма управления средством;
- разработка структурной и электрической принципиальной схем;
- разработка алгоритма управляющей программы,
- конструктивное проектирование макета полигона и модуля управления;
- проведение экспериментальных исследований.

В работе освещены следующие вопросы.

В первой главе рассмотрены варианты общей компоновки полигона, транспортного средства и возможные принципы его ориентации в пространстве. Выявлены требования к конструкции транспортного средства и оборудованию системы ориентации. Сформулированы основные задачи работы.

Вторая глава посвящена анализу управляемости транспортного средства, разработке модели его маневрирования и алгоритма управления.

Разработаны основы и схема управления транспортным средством. Проведен анализ логистической составляющей маршрутизации перемещения модели транспортного средства.

В третьей главе рассматривается конструирование и комплектация ходовой платформы с учетом разработанной модели транспортного средства.

В заключении сформулированы основные выводы, вытекающие из проведенных в диссертационной работе исследований.

К основным научным результатам, полученным впервые, относятся:

1. Разработана модель управления транспортным средством;
2. Выведены зависимости, позволяющие обеспечить точное изменение положение транспортного средства в пространстве;
3. Разработаны алгоритмы управления транспортным средством в различных пользовательских режимах.

## **Глава 1 Общая компоновка полигона, транспортного средства и возможные принципы его ориентации в пространстве**

В настоящее время в различных источниках информации, рассматриваются полноразмерные полигоны, предназначенные для испытания опытных беспилотных автомобилей. Информация о мини-полигонах, позволяющих обеспечить тренировочную и образовательную деятельность, отсутствует. Поэтому тематика настоящего диссертационного исследования является актуальной.

«Автоматизированный транспортный мини-полигон Автонет предназначен для:

- профессиональной ориентации на тематику Автонет;
- практико-ориентированной подготовки кадров по направлению НТИ;
- профильного технического творчества молодежи.

Мини-полигон соответствует «дорожным картам» НТИ.

Функциональное назначение комплекса соответствует задачам Автонет. Мини-полигон может являться:

- демонстратором беспилотного перемещения макета – имитатора автомобиля по заданным маршрутам;
- тренажером оператора макета беспилотного автомобиля, действующего в городской среде;
- тренажером системного интегратора специального программного обеспечения (СПО) – компоновка подпрограмм для управления движением по заданным маршрутам;
- АРМ постановщика задач и программиста СПО МПАН – развитие технологий, аппаратных и программных средств мини полигона;
- испытательным комплексом для отработки аппаратных и программных средств имитаторов внешней среды и беспилотных транспортных средств» [12].

Для позиционирования подвижного объекта рационально использовать позиционирующие системы, например GPS – глобальную систему позиционирования [34].

Данная система использует величину времени задержки сигнала передатчика спутника и принятого сигнала приемником. Учитывая точные координаты спутника, в частности и спутниковой группировки, вообще, возможно определение координат объекта на земной поверхности.

К сожалению, система GPS была разработана для основного применения военным ведомством США. В настоящее время целесообразно исключать риски применения импортных высокотехнологичных систем.

Поэтому необходимо обратить внимание на отечественную разработку – глобальную навигационную спутниковую систему ГЛОНАСС.

Данная система в основном рассчитана на применение в границах территории России. Система ГЛОНАСС, являясь аналогом GPS, выгодно отличается по ряду параметров:

- орбита спутников, поддерживающих систему ГЛОНАСС, имеет разный угол наклона. Это позволяет обеспечить высокую эффективность работы системы в высоких широтах;
- высокотехнологичные объекты, производимые отечественной промышленностью, рассчитаны на поддержку ГЛОНАСС. Это обеспечивает независимость использования GPS;
- спутники системы ГЛОНАСС осуществляют асинхронное перемещение относительно Земли, что упрощает использование системы.

На рисунке 1 показан принцип работы навигационной спутниковой системы. Три спутника из орбитальной группировки, в зоне которых находится искомый объект, позволяют определить его координаты.

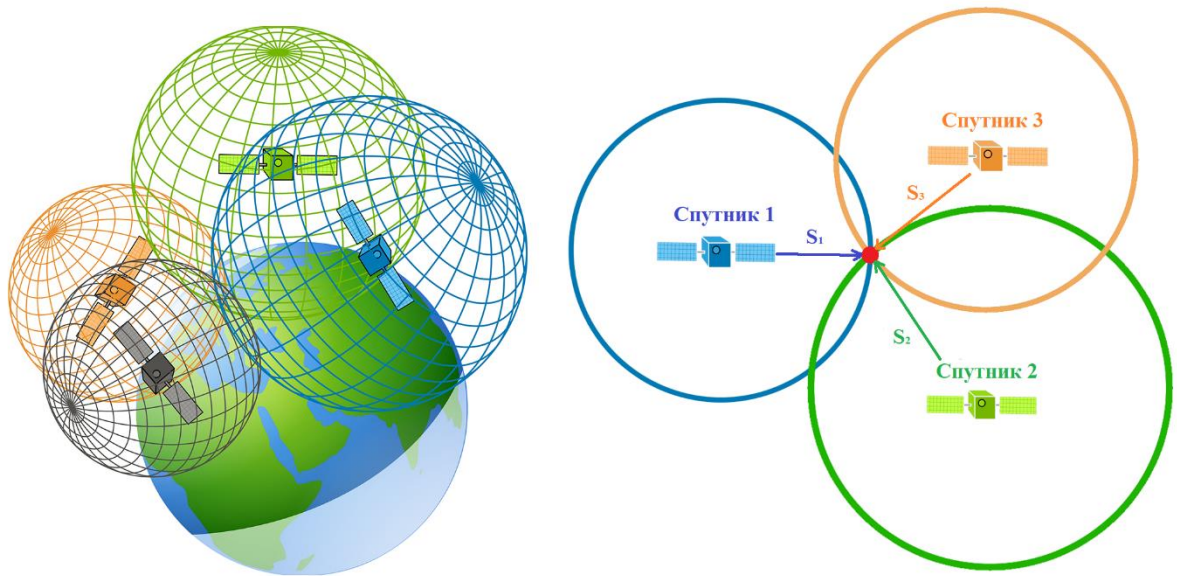


Рисунок 1 – Принцип работы навигационной спутниковой системы

Таким образом, для определения координат искомого объекта, оснащенного приемником сигнала, требуется источник сигнала с известными координатами [22], [25].

Применение таких систем позиционирования характеризуется погрешностями измерения координат, которые составляют в зависимости от атмосферных искажений от 2 до 10 метров.

Проектируемый полигон имеет примерно такие же размеры, поэтому позиционирование транспортных средств в условиях полигонного моделирования с помощью спутниковых систем является нерациональным.

Другим вариантом реализации локальной системы позиционирования является система, основанная на использовании технического зрения.

Задачи ориентации в окружающем пространстве, которые ставятся перед транспортным средством, решаются с помощью датчиков информационно-измерительной системы робота, которые предоставляют информацию о состоянии внешней среды, пройденном пути, углах поворота и других расчётных величинах. Система машинного зрения обладает



наибольшей универсальностью и сравнительно дешевым способом решения сложных задач получения изображения, его математического анализа и принятия решения, на основе которого производится движение [15], [18].

Технология машинного зрения основана на распознавании объекта, воспринимаемого в виде изображения, представленного в виде графических символов. Сканированное изображение делится на пиксели, причем каждому пикселю присваивается некоторое значение. Это значение параметризуется с помощью контрастности. Например, для черно-белого изображения: белый пиксель имеет значение 0, черный пиксель – 255. Остальным пикселям присваиваются значения в соответствующих пределах.

Пример распознавания образа, в данном случае символа – цифры, показан на рисунке 2.

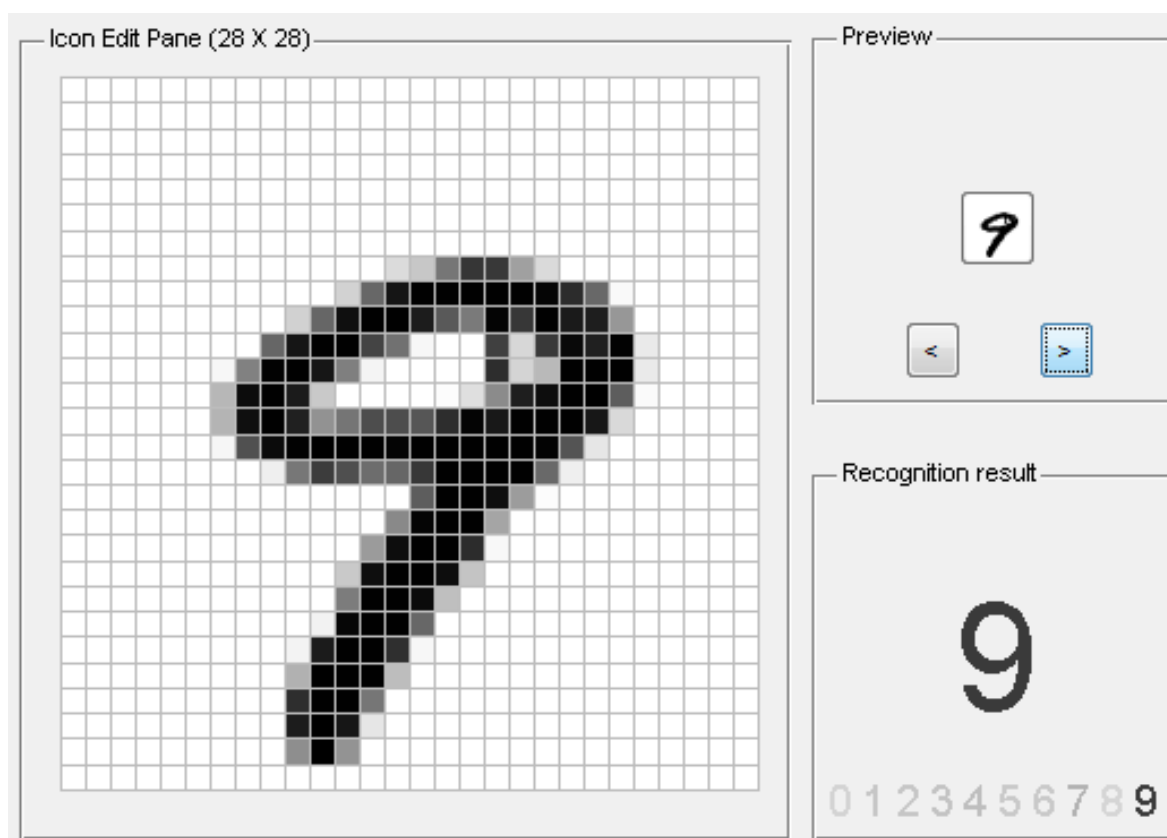


Рисунок 2 – Визуализация принципа работы технологии компьютерного зрения

Для обеспечения функционирования системы позиционирования, основанной на применении компьютерного зрения, в окружающем

пространстве необходимо размещать необходимые метки, которые будут распознаваться оптической камерой.

Кодированное изображение обрабатывается контроллером. Во внешней памяти контроллера хранятся кодированные образы изображений, с которыми сравнивается сканированное изображение. Сравнение с образцами происходит до момента распознавания символа. После завершения сравнения возможно определение положения объекта, связанного с распознанной меткой.

В качестве примера рассмотрим перемещение объекта по лабиринту (рис. 3). Маршрут представлен в виде цифровых символов, расположенных на трассе лабиринта. Причем, не обязательно размещать эти цифры в порядке возрастания или убывания потому, что это всего лишь метки, которые считываются в соответствии с порядком прохождения маршрута, заданного алгоритмом.

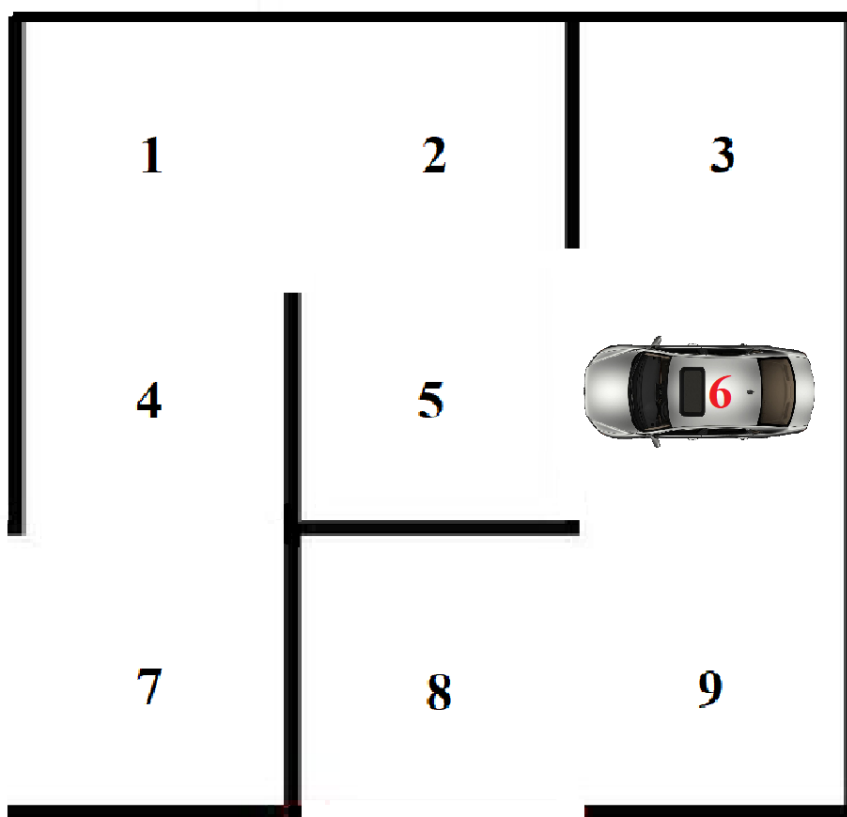


Рисунок 3 – Визуализация принципа работы системы позиционирования с компьютерным зрением

Применение рассматриваемой системы технического зрения не обеспечивает надежность прохождения по заданному маршруту, так как возможно возникновение ошибки при распознавании метки.

Повышение надежности системы возможно путем применения дублирующих камер для распознавания метки или обеспечения постоянства клиренса (дорожного просвета) между объектом и поверхностью полигона, т.е. обеспечения фиксированного расстояния между объективом камеры и распознаваемым символом. И тот, и другой варианты являются достаточно сложными и затратными, так как в первом случае возрастает сложность и стоимость системы, во втором случае требуется обеспечить стабильность положения ходовой части объекта и тщательность обслуживания поверхности полигона.

Кроме того, данный способ предусматривает жесткий алгоритм маршрутизации, основанный на определенной последовательности прохождения символов. Причем, точность позиционирования в данном случае определяется дискретностью шага расположения символов.

Еще один способ, основанный на применении машинного зрения, отличается другой постановкой задачи ориентации.

В данном случае первоочередной задачей ориентации объекта в пространстве является его исследование на наличие непреодолимых препятствий и параллельное составление карты проходимых путей с локализацией своего положения на карте. Таким образом, получаются многоугольники проходимых зон, которые можно добавлять в векторную карту (рис. 4). На такой карте существует бесконечное множество точек для определения маршрута. Использование векторной карты для определения маршрута возможно при условии выбора конечного числа точек с вынужденным незначительным ограничением поиска оптимального маршрута по этим точкам [9].

Для решения задачи поиска пути на карте необходимо использовать растровую карту, учитывая, что маршрут можно представить в виде графов.

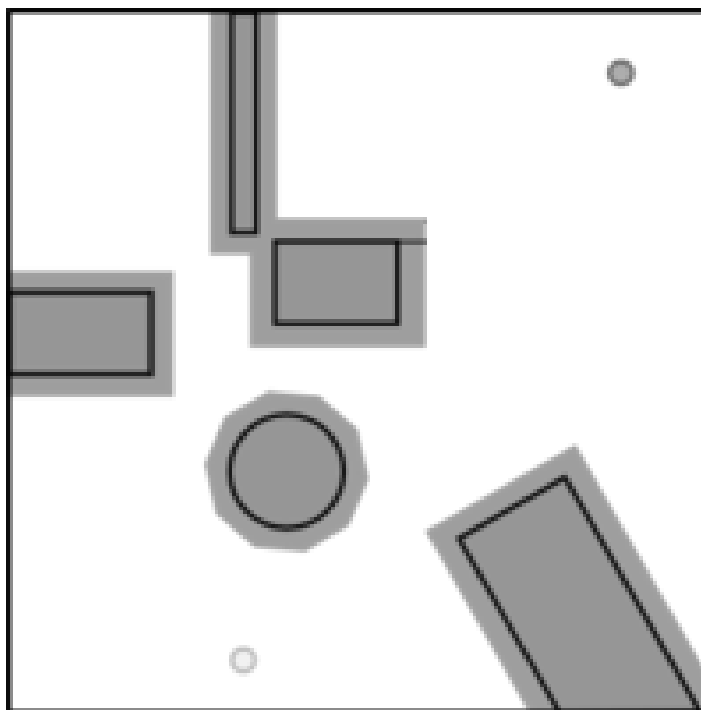


Рисунок 4 – Векторное представление карты

За вершины графа принимаются углы многоугольников, а за ребра – возможность проезда к соседним объектам. Обязательное условие – в список вершин графа должны входить точки старта и финиша. После окончания формирования графа допустимых путей (рис. 5), поиска пути на карте будет производиться во взвешенном графе.

Задача перемещения транспорта в нужную точку карты формулируется следующим образом [21]:

1. Дана карта с указанными препятствиями, текущее положение транспорта и конечное положение (положение цели).

2. Требуется составить маршрут (ломаную от текущей точки до цели), позволяющий избежать всех препятствий и приводящий в нужную точку карты за приемлемое или оптимальное время. При этом под «положением

объекта» могут пониматься как «координаты объекта», так и пара «координаты объекта + направление объекта».

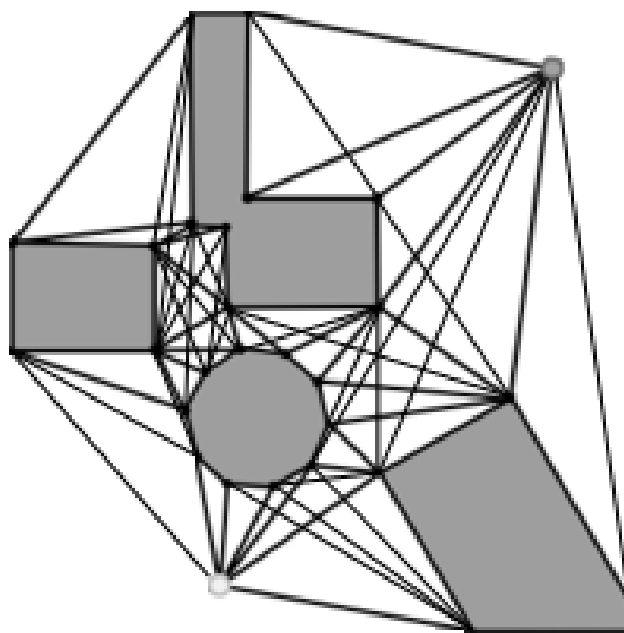


Рисунок 5 – Графы допустимых путей

Для определения взвешенных графов вводится множество вершин, в состав которого не входят начальная и конечная вершины. Причем вершины должны находиться вне препятствий. Положение вершин определяется внешней границей зоны безопасности, которая позволяет при перемещении центра объекта учитывать его габаритные размеры. Размеры зоны безопасности определяются расстоянием от центра объекта до самой выступающей его части или радиусом описанной вокруг объекта окружности с центром, совпадающим с центром объекта. Введение зоны безопасности позволяет выделить на карте множество точек, которое обеспечивает прохождение объекта мимо препятствий без контакта с ними при условии принадлежности центра объекта этому множеству [23].

Расположение множества точек зоны безопасности определяется границами, отстоящими от препятствий на указанное расстояние. Таким образом, формируется дополнительная карта, состоящая из многоугольников, являющихся границами препятствий. Очевидно, что можно балансировать

между формой границ, от которой зависит вид маршрута и количеством полученных вершин, что определяет время поиска оптимального маршрута. При этом необходимо соблюдать правила перехода от карты препятствий к карте-основе для построения графа:

- препятствия, заданные в виде окружностей, приближенно представляются правильными многоугольниками (количество вершин является параметром оптимизации),
- препятствия, заданные в виде отрезков, обводятся прямоугольниками со скругленными с помощью ломаных углами (количество дополнительных вершин для скругления также является параметром оптимизации),
- препятствия, представленные в виде многоугольников, имеют скругления на границах выпуклых углов, представленные ломаными, количество точек в которых также является параметром оптимизации.

После формирования карты препятствий с зонами безопасности строятся графы допустимых путей с учетом начальной и конечной вершин и вершин границ многоугольников зон безопасности. Причем, ребро графа считается недопустимым при пересечении им какой-либо границы препятствия [10].

При традиционном способе определения оптимального пути, например при помощи алгоритма Дейкстры, не учитывается время, затрачиваемое на остановку, развороты объекта в вершинах графа и разгон, требующиеся для движения в следующих направлениях маршрута. Для ввода поправок на время целесообразно ввести «штраф за возможную остановку и возможно требующийся поворот». Штраф должен увеличивать рассчитанную длину минимального пути на расстояние, пропорциональное времени на маневры работа. Сумму штрафа можно точно рассчитать, разложив её на составляющую штрафа за торможение и разгон и составляющую штрафа на

поворот с направления ребра входа в вершину до направления ребра выхода из вершины.

Для начала движения необходимо определить маршрут до цели. Возможны следующие варианты определения маршрута:

- маршрут представляет заранее известную последовательность точек,
- объект последовательно получает информацию о перемещении в определенную точку,
- все возможные точки маршрута объекта изучены, перемещение совершается между этими точками по программе.

При сканировании рабочего пространства транспортного средства с использованием видеокамеры получается изображение с двумерным распределением яркости. «Необходимо определить взаимное положение точек изображения и точек сцены, а также яркость точек изображения.

Пусть точечное отверстие камеры находится на фиксированном расстоянии от плоскости изображения (рис. 6). Предположим, что камера устроена так, что только свет, проходящий через отверстие, может достичь плоскости изображения» [24]. Прямолинейное распространение света обеспечивает отображение каждой точки сцены в направлении луча, началом которого является отображаемая точка и проходящего через точку отверстия камеры.

Множество отображаемых точек формирует центральную (перспективную) проекцию сцены на плоскость изображения. «Определим оптическую ось как перпендикуляр, опущенный из отверстия на плоскость изображения. Введем прямоугольную систему координат с началом в точечном отверстии и осью  $z$ , совпадающей с оптической осью и направленной в сторону изображения» [24]. Система координат считается правой, т.к. ось  $X$  направлена вправо, ось  $Y$  – вверх.

Необходимо «определить положение образа  $P_i$  некоторой точки  $P_o$ , находящегося перед камерой объекта (рис. 6). Будем считать, что никакой другой объект не пересекает луч, идущий от точки  $P_o$  в точку  $O$ » [24].

Пусть  $r = (x, y, z)^T$  – вектор, соединяющий точкой  $O$  с точкой  $P_o$ , а  $r' = (x', y', z')^T$  – вектор, соединяющий точку  $O$  с точкой  $P_i$ .

«Обозначим  $f'$  – расстояние от плоскости изображения до отверстия, а  $x'$  и  $y'$  – координаты точки  $P_i$  на этой плоскости. Два вектора  $r$  и  $r'$  – коллинеарны и отличаются только скалярным множителем.

Если луч, соединяющий точки  $P_o$  и  $P_i$ , составляет угол  $\alpha$  с оптической осью, то длина вектора  $r$  составляет

$$r = -z \cdot \sec(\alpha) = -(r \cdot \hat{z}) \cdot \sec(\alpha), \quad (1)$$

где  $\hat{z}$  – единичный вектор вдоль оптической оси» [24].

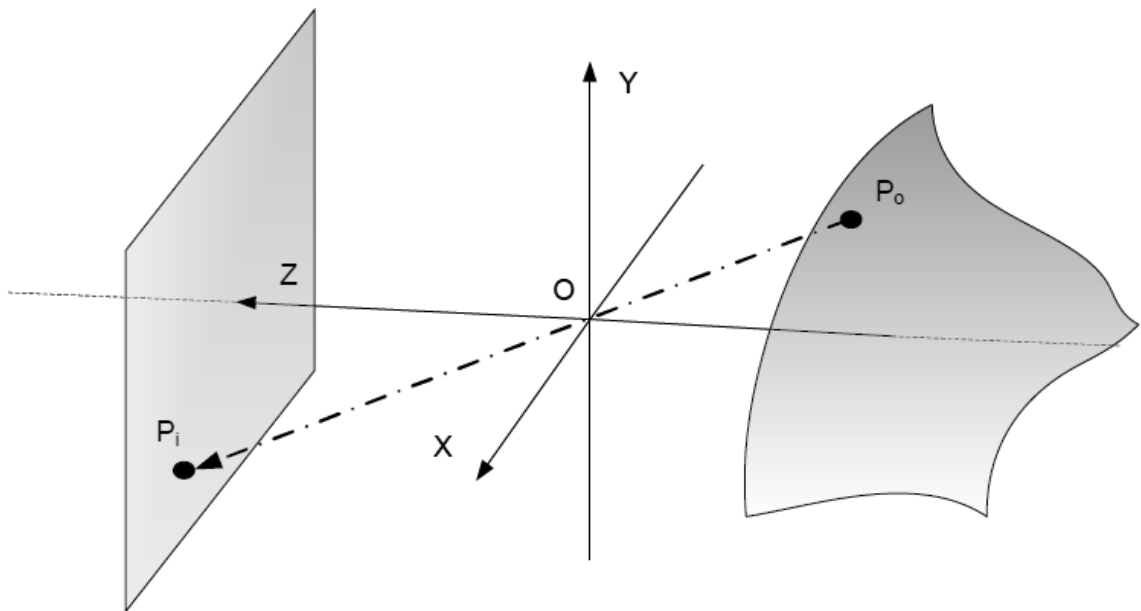


Рисунок 6 – Формирование изображения

«Длина  $r'$  составляет  $r = f' \sec(\alpha)$ , поэтому, учитывая коллинеарность векторов, можно составить равенство

$$-\frac{r'}{f'} = \frac{r}{r \cdot \hat{z}}. \quad (2)$$

Это выражение можно представить в виде компонент» [24]:



$$\begin{cases} \frac{x'}{f'} = \frac{x}{z} \\ \frac{y'}{f'} = \frac{y}{z} \end{cases} \quad (3)$$

Используем гистограммный метод анализа изображения. В соответствии с этим методом изображение (или фрагмент) рассматривается в виде множества независимых элементов (пикселей). Причем расположение пикселей в пространстве не учитывается, и изображение можно интерпретировать как представительную выборку случайного сигнала, каждый отсчет которой статистически независим от других. В соответствии со статистическими методами обработки данных, любой анализ подобной выборки может быть сведен к анализу гистограммы выборки (дискретной оценки плотности распределения случайной величины по данной выборке). При этом способ формирования выборки и интерпретация результатов анализа являются двумерными.

Предполагаем, что сигнал содержит детерминированную полезную составляющую и случайный «шум». Для выделения полезного сигнала используются: оценка среднего (математическое ожидание), ранговые (порядковые) статистики, метод наибольшего правдоподобия, байесовское оценивание и т. п. Применение каждого из подходов основано на некоторых предположениях о характере изображения и априорной информации.

Предположим также, что изображение локально однородно. Это значит, что в пределах фрагмента (апертуры) значение полезного сигнала неизменно. Тогда для фильтрации изображения возможно применение метода средних величин и ранговых (медианных) фильтров. Характеристики фильтрации будут хорошими на однородных областях изображения и неудовлетворительными на границах (размытие границ). Поэтому наличие областей и границ является существенным фактором, требующим внимания при двумерной реализации статистических фильтров [15].

Данный способ достаточно сложен в реализации. Хотя достаточно эффективен при распознавании геометрии объектов и составлении пространственной карты с целью организации и последующей минимизации протяженности маршрутов, что позволяет оптимизировать логистическую составляющую. Его реализация в условиях полигона будет достаточно дорогостоящей [10], [19].

Вышеуказанное приводит к необходимости разработки специализированной системы позиционирования, основанной на применении специальных аппаратных меток, расположенных на трассе полигона, взаимодействующих со считывающим устройством транспортного средства.

В данном случае метка будет являться активным объектом, установленным на транспортном средстве. Устройства считывания целесообразно установить на трассе мини-полигона.

Выводы по главе 1:

- рассмотрены различные варианты позиционирования объектов с использованием глобальных систем позиционирования;
- проведен обзор способов определения расположения объекта с использованием различных датчиков;
- рассмотрена система распознавания положения объекта, использующая техническое зрение;
- сделан вывод о целесообразности использования меток, позволяющих идентифицировать положение объекта в пространстве.

## Глава 2 Аппаратно-программный комплекс мини-полигона

### 2.1 Технические средства мини полигона

«Базовый компонент мини полигона – имитатор фрагмента городской среды разрабатывается как развиваемый динамический архитектурный макет. На рисунке 7 представлен базовый вариант рабочего поля» [12].

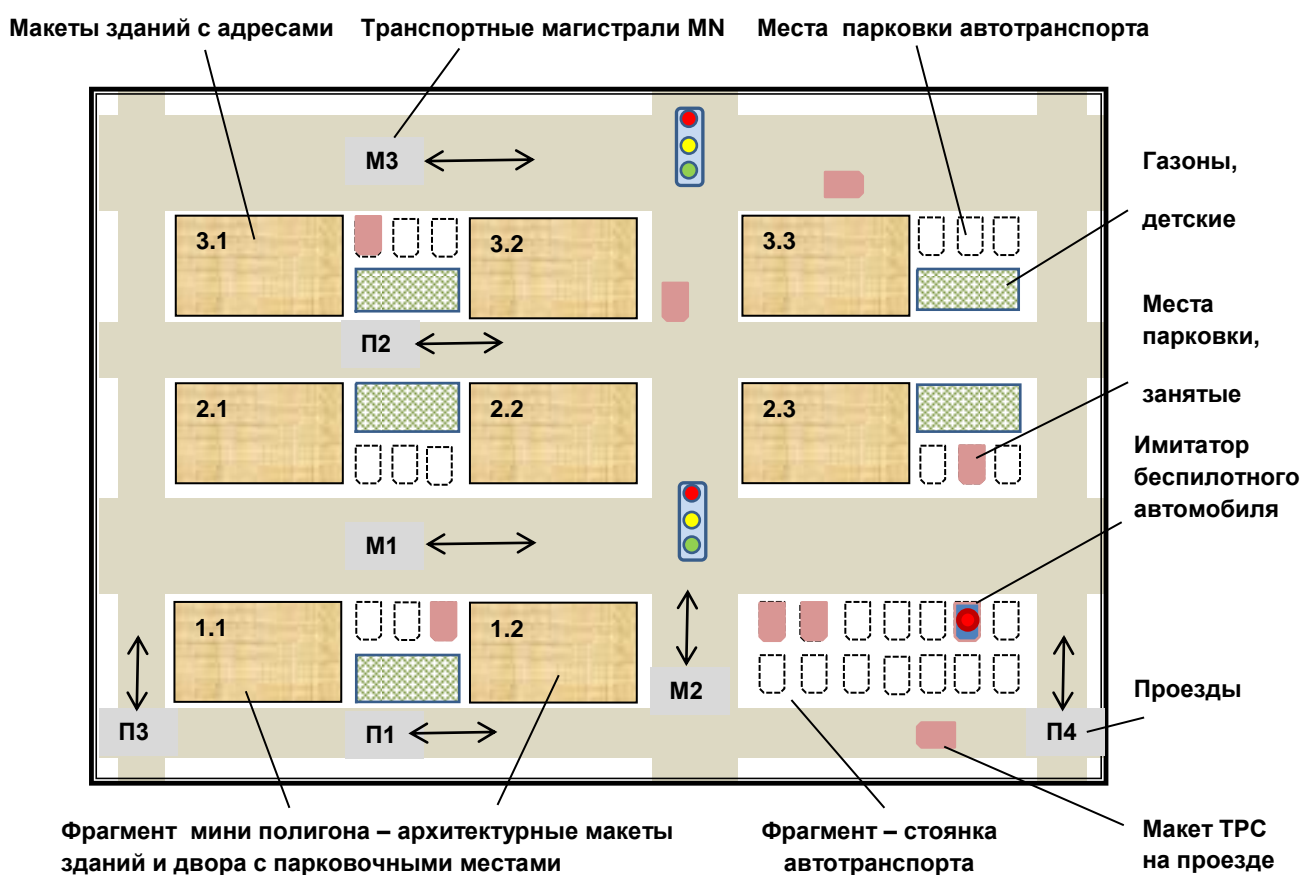


Рисунок 7 – Динамический архитектурный макет городской среды

«Основными режимами применения мини-полигона Автонет являются:  
– дистанционное управление макетом автомобиля – имитатором транспортного средства (ИМ-ТРС), формирование, сохранение в банке данных полигона набора типовых траекторий перемещения транспортного средства (ТП-ТРС);

- формирование маршрута движения беспилотного транспортного средства из исходного положения «старт» в конечное «финиш», с применением набора ТП-ТРС;
- беспилотное движение ИМ-ТРС на мини-полигоне по заданному маршруту;
- беспилотное движение ИМ-ТРС на мини-полигоне с адаптацией маршрута при выявлении помех (нештатных ситуаций);
- беспилотное движение на мини-полигоне по заданному маршруту с поиском свободного места на площадках парковки» [12].

«Для применения в качестве макета беспилотного автомобиля, при выполнении проекта разработана роботизированная платформа – имитатор транспортного средства (ИМ-ТРС). Пространственная ориентация ИМ-ТРС на мини полигоне обеспечивается с применением различных технических средств, базовый вариант – набор лазерных датчиков расстояния, устанавливаемых на имитаторе (рисунок 8) Определение позиции ИМ-ТРС производится по расстояниям от контрольных объектов: стен макетов зданий, боковых панелей мини-полигона» [12].

«При размещении на фрагментах архитектурного макета дополнительных геометрических компонентов, распознаваемых при движении имитатора транспортного средства, идентифицируется номер магистрали или проезда.

Указанная комбинированная идентификация позволяет определять координаты имитатора транспортного средства на мини полигоне, обеспечивать контроль перемещения по заданному маршруту в пространстве и времени. Развитие структуры ИМ-ТРС для расширения набора решаемых на мини полигоне задач производится, в частности, с применением ультразвуковых датчиков расстояния. Ультразвуковые датчики, в отличие от определяющих расстояние до контрольной точки, отражающей видимый луч лазера, определяют наличие отражающей поверхности в угловом диапазоне контроля, ориентировочно,  $\pm 15^\circ$ » [12].

«С применением ультразвуковых датчиков расстояний переднего и заднего обзора, а также лазерных, решаются различные задачи, определяются, уточняются траектории движения, связанные с объездом препятствий, с корректировкой положения имитатора транспортного средства при парковке» [12].



Рисунок 8 – Пространственная ориентация имитатора беспилотного транспортного средства на мини-полигоне

## 2.2 Основные системные и учебно-тренировочные задачи

«В режиме «Дистанционное управление автомобилем» оператором на мини-полигоне решаются задачи:

- учебно-тренировочные и демонстрационные перемещения имитатора транспортного средства в городской среде;

- формирование, сохранение в банке данных мини-полигона набора типовых траекторий перемещения, в том числе:
- выезд с парковочного места (по вариантам), въезд на парковочное место;
- выезд на магистрали (двухстороннее движение) и проезды (одностороннее движение),
- съезд с магистралей и проездов, движение по магистралям и проездам;
- корректировка положения автомобиля: смещение направо, налево, вперед или назад;
- старт-стопные перемещения автомобиля с учетом сигналов светофоров;
- контрольно-испытательные перемещения по сохраненным траекториям» [12].

«В режиме «Формирование маршрута движения» системным интегратором управления ИМ-ТРС, с применением АРМ мини-полигона, в частности, решаются задачи:

- формирование базового маршрута движения ИМ-ТРС, с применением набора ТП-ТРС из Банка данных мини-полигона, с учетом заданных положений ИМ-ТРС – исходного и конечного;
- формирование вариантов маршрута движения имитатора, с учетом загруженности стоянок, наличия на парковочных местах макетов транспортных средств;
- формирование вариантов маршрута движения имитатора при загруженности магистралей, проездов;
- выбор, формирование условий выбора маршрутов движения ИМ-ТРС, с учетом загруженности стоянок, магистралей, проездов;
- формирование заданий на расширение набора ТП-ТРС в Банке данных мини-полигона» [12].

«В режиме «Беспилотное движение по маршруту» оператором мини-полигона решаются типовые задачи:

- расстановка макетов транспортных средств для имитации условий окружающей среды;
- запуск движения ИМ-ТРС по маршрутам, с заданными исходными данными:
- базовый маршрут, варианты маршруты, с учетом условий окружающей среды;
- контроль, протоколирование результатов реализации движения имитатора транспортного средства по заданным маршрутам, контрольным позициям (при необходимости, в старт-стопном режиме)» [12].

«Исходное состояние (старт) автомобиля и городской среды задается параметрами:

- координаты, условный адрес парковочного места, на которое установлен ИМ-ТРС;
- координаты, условные адреса парковочных мест, на которые установлены макеты транспортного средства (места заняты);
- координаты, адреса мест размещения на магистралях и проездах, стоянках, макетов транспортного средства (помехи движению)» [12].

«Конечное состояние (финиш) автомобиля и городской среды задается параметрами:

- координаты, адрес основной стоянки, на которую должен прибыть имитатор транспортного средства;
- координаты, адрес парковочного места на основной стоянке, на которое должен прибыть ИМ-ТРС;
- координаты, условные адреса альтернативных стоянок, на одну из которых должен прибыть имитатор транспортного средства, с учетом полной занятости парковочных мест на основной стоянке» [12].

## **2.3 Корректировка траекторий движения транспортного средства**

Корректировка беспилотного движения автомобиля по маршруту, с применением набора ТП-ТРС из Банка данных мини-полигона, производится при выявлении ситуаций:

- смещение имитатора на магистрали или проезде более, чем пороговую величину;
- поворот имитатора при движении по маршруту более, чем пороговую величину, с применением методов локальной ориентации
- контроля положения имитатора, на мини полигоне:
- задержка или опережение прибытия ИМ-ТРС на контрольную позицию, с учетом результатов измерения расстояний до стен зданий с заданными адресами (методы выявления ошибок, сбоев в системе управления);
- выявление наличия помех на исполняемой траектории движения ИМ-ТРС (установленные макеты транспортных средств);
- учет состояния светофоров на маршруте движения.

## **2.4 Перспективное развитие технических средств мини-полигона**

«Развитие структуры ИМ-ТРС для расширения набора решаемых на мини полигоне задач производится, в частности, с применением ультразвуковых датчиков расстояния.

Ультразвуковые датчики, в отличие от лазерных, определяющих расстояние до контрольной точки, отражающей видимый луч лазера, определяют наличие отражающей поверхности в угловом диапазоне контроля, ориентировочно,  $\pm 15^\circ$ .

С применением ультразвуковых датчиков расстояний переднего и заднего обзора, а также лазерных, более эффективно решаются различные задачи управления движением, определяются, уточняются траектории



движения, связанные с объездом препятствий, с корректировкой положения имитатора транспортного средства при парковке.

Для решения расширенного комплекса задач управления ИМ-ТРС на мини-полигоне, выполняются:

- анализ данных, поступающих от лазерных и ультразвуковых датчиков в различных экспериментальных ситуациях движения ИМ-ТРС и окружающей среды;
- разработка методов, алгоритмов формирования траекторий объезда препятствий в различных условиях городской среды;
- экспериментальная отработка средств программного обеспечения реализации алгоритмов формирования траекторий» [14].

## **2.5 Схемы фрагмента мини-полигона и комплектующие**

Для проектирования аппаратно-программного комплекса была сделана попытка разработки одного из вариантов системы позиционирования с маршрутизацией перемещения транспортного средства на трассе (магистраль, проезд) полигона, которая была рассмотрена и выбрана в главе 1.

В состав системы управления позиционированием входит контроллер, который обрабатывает входные сигналы от датчиков позиционирования, являющимися реперными точками маршрутизации, расположенными на трассе.

Для аппаратного взаимодействия с контроллером предусматривается увеличение количества портов ввода-вывода, которое обеспечивает модуль расширения.

Необходима также организация человеко-машинного интерфейса для управления оператором системой позиционирования и маршрутизации.

Оператор обеспечивает маршрутизацию транспортного средства, определяя требуемое маневрирование в пределах трасс мини-полигона.

Человеко-машинный интерфейс должен поддерживать удаленный обмен данными. Для удобства пользователя необходимо предусмотреть электронную индикацию состояния системы.

Для питания компонентов системы в комплекс также входит блок питания.

На рисунке 9 представлена структурная схема мини-полигона.

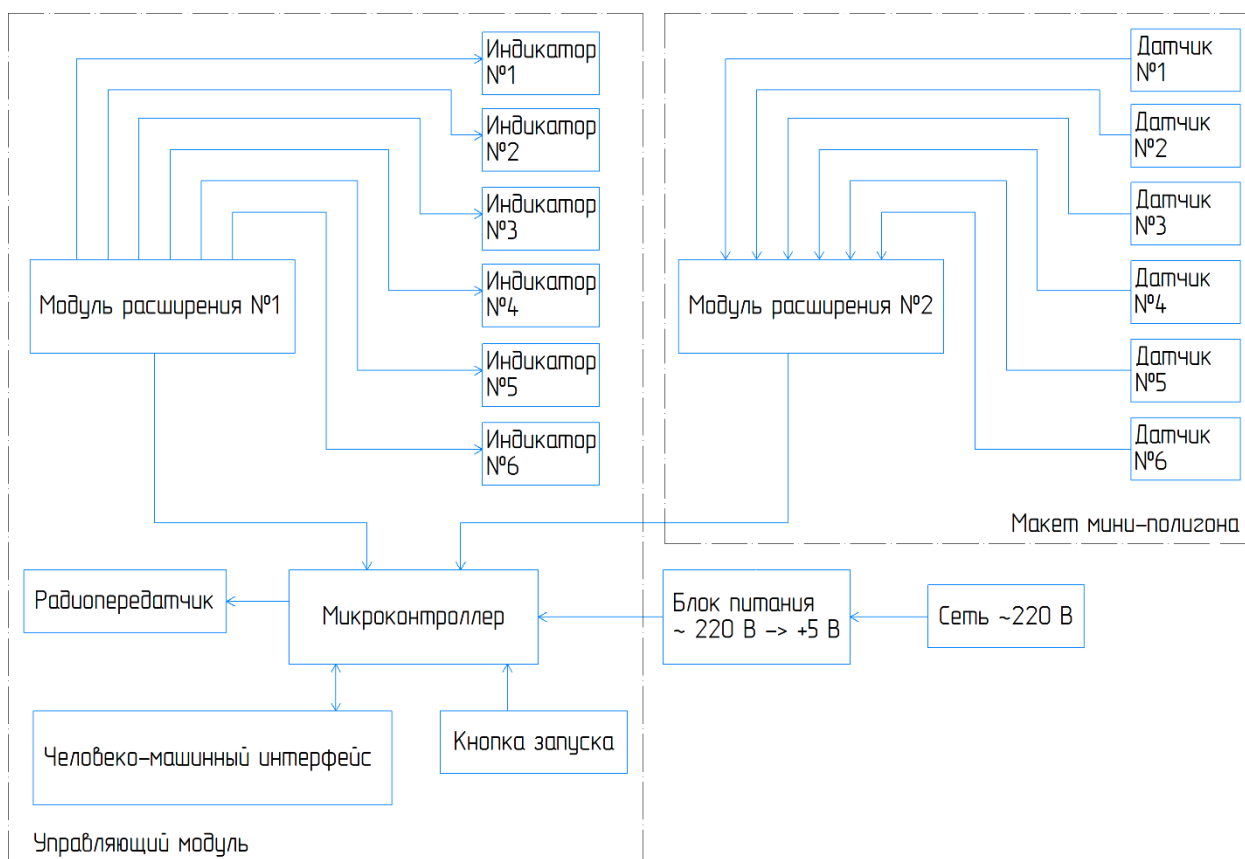


Рисунок 9 – Структурная схема мини-полигона

Блок питания преобразует сетевое переменное напряжение 220 В в постоянное напряжение +5 В, которое предназначено для питания всех электронных компонентов системы.

Кнопка запуска обеспечивает пуск и останов транспортного средства.

Модуль расширения №1 предназначен для отображения номера перекрестка, на котором в настоящее время находится транспортное средство.

Модуль расширения №2 обеспечивает обработку сигналов, идентифицирующих текущее положение транспортного средства, позволяющее определить его приближение к перекрестку.

Для реализации поставленных задач обработки информации и управления целесообразно использовать контроллер Arduino. Выбор обеспечивается наличием большого количества периферийных устройств и устройств сопряжения и низкой стоимостью [3], [4].

Сравнение возможных модификаций контроллеров для использования приведена в таблице 1.

Таблица 1 – Характеристики Arduino Mega, UNO, Nano

Характеристики	Arduino Mega	Arduino UNO	Arduino Nano
Микроконтроллер	ATmega1280	Atmega328	Atmega328
Тактовая частота	16 МГц	16 МГц	16 МГц
Аналоговые контакты	8	8	8
Цифровые контакты	54	22	22
Диапазон аналоговых значений	От 0 до 5 В	От 0 до 5 В	От 0 до 5 В
Есть ли защита от коротких замыканий	Да	Да	Нет

Исходя из необходимых требований возможности и надежности выбираем Arduino UNO, которые обеспечивает достаточное количество цифровых контактов и защиту от коротких замыканий [28].

Для определения текущего положения и ориентации транспортного средства были рассмотрены различные датчики, возможности которых позволяют реализовать указанные функции:

– датчики освещённости (пара – лазерный излучатель (транспортное средство) и фоторезистор (реперная точка трассы)), исключены из-за возможности засветки нескольких близкорасположенных датчиков одновременно [27];

– датчики, основанные на магнитном взаимодействии – герконы, исключены из-за хрупкости корпуса, зависимости срабатывания от ударных и вибрационных воздействий;

– датчики Холла [36], [37], которые могут быть аналоговыми и цифровыми.

С учетом наличия магнита, жёстко закрепленного на транспортном средстве, рациональным является выбор цифрового униполярного датчика SS441A 115G/20G [35].

На рисунке 10 показано подключение датчика Холла.

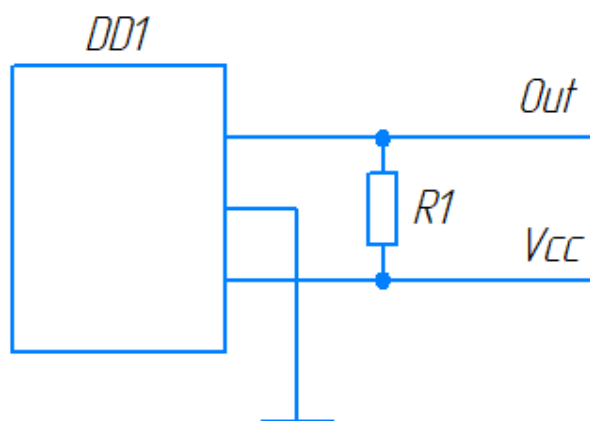


Рисунок 10 – Схема подключения датчика Холла

Для получения перспективной возможности масштабирования мини-полигона предусмотрено применение платы расширения ввода-вывода.

Рассматривалось два варианта:

– платы расширения на сдвиговых регистрах, исключено по причине использования большого количества портов ввода-вывода с необходимым применением многопроводных соединений;

– плата расширения, обеспечивающая поддержку двухпроводного интерфейса I2C [35]. Выбрана и показана на рисунке 11.

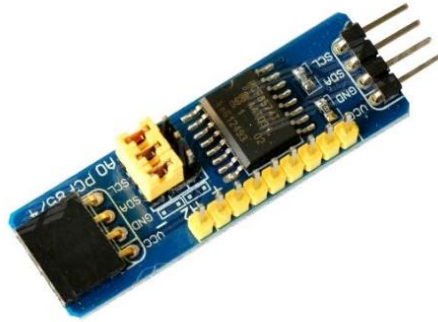


Рисунок 11 – плата расширения PCF8574

Плата поддерживает 8 портов с возможностью расширения до 64 при условии взаимодействия соответствующего количества аналогичных плат.

Для адресации портов используется адресное пространство. По умолчанию устанавливается адрес 0x20. Аппаратно (перемычками) возможно его изменение для получения адресации портов подключения восьми устройств.

Возможные варианты смены адресов приведены в таблице 2.

Таблица 2 – Адресация портов

A0	A1	A2	Адрес
-	-	-	0x20
+	-	-	0x21
-	+	-	0x22
+	+	-	0x23
-	-	+	0x24
+	-	+	0x25
-	+	+	0x26
+	+	+	0x27

Для обеспечения управления транспортным средством необходимо организовать беспроводной обмен данными с контроллером [26], [30].

Для этих целей возможно использование технологий Wi-Fi и bluetooth.

Результатом проведения сравнительного анализа возможностей применения вышеуказанных технологий является таблица 3. При сравнении были учтены наиболее популярные модули.

Таблица 3 – Сравнительные характеристики модулей

Характеристики	ESP-01	HC-05	HC-06	HM-10
Протокол	Wi-Fi 802.11	Bluetooth 2.0	Bluetooth 2.0	Bluetooth 4.0
Режим AT-команд	Да	Да	Нет	Да
Продолжительная работа	Да	Да	Да	Нет
Ток потребления	220 мА	50 мА	50 мА	50 мА
Дальность	40 м	10 м	10 м	10 м

В результате сравнительного анализа сделаны следующие выводы:

1. Модуль ESP-01 потребляет большой ток, исключается, так как не соответствует характеристикам контроллера;
2. Модуль HC-06 не имеет режима AT-команд, исключается, так как отсутствует режим перевода в статус Master;
3. Модуль HM-10 имеет ограниченный ресурс работы, исключен.

Таким образом, требуемыми характеристиками из рассмотренных модулей обладает модуль HC-05, показанный на рисунке 12 [29].

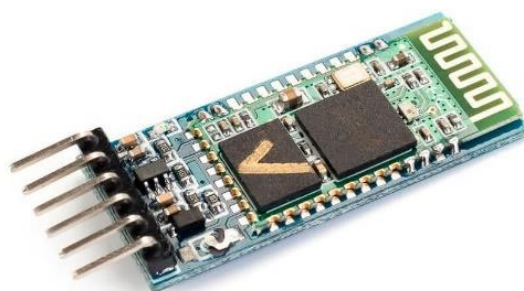


Рисунок 12 – Bluetooth-модуль HC-05

В качестве устройства отображения информации для аппаратной части человеко-машинного интерфейса были рассмотрены несколько вариантов:

- LCD-дисплей в комплекте с джойстиком;
- LCD-дисплей в комплекте с кнопочной клавиатурой;
- комплектный моноблок LCD Keypad Shield.

Устройство комплектного моноблока LCD Keypad Shield удобно при монтаже и программировании (рисунок 13) [7], [11].



Рисунок 13 – Комплектный моноблок LCD Keypad Shield

Данное устройство отличается минимальными аппаратными требованиями, необходимыми для взаимодействия с контроллером [32]:

- пины D4-D10 – работа с LCD-дисплеем;
- пин A0 – работа с кнопками.

Для обработки информации с кнопок используются различные уровни сигнала, значение которых приведены в таблице 4.

Таблица 4 – Уровни сигнала (пин A0)

Выбранная кнопка	Значение уровня сигнала
RIGHT	0-100
UP	100-200
DOWN	200-400
LEFT	400-600
SELECT	600-800
Клавиша не нажата	800-1023

Индикация о прохождении транспортным средством реперных точек полигона спроектирована с использованием светодиодов. Применены светодиоды GNL-5013GT зелёного свечения [33].

В соответствии с исходными данными, структурной схемой и выбранными компонентами представлена принципиальная схема мини-полигона (рисунок 14).

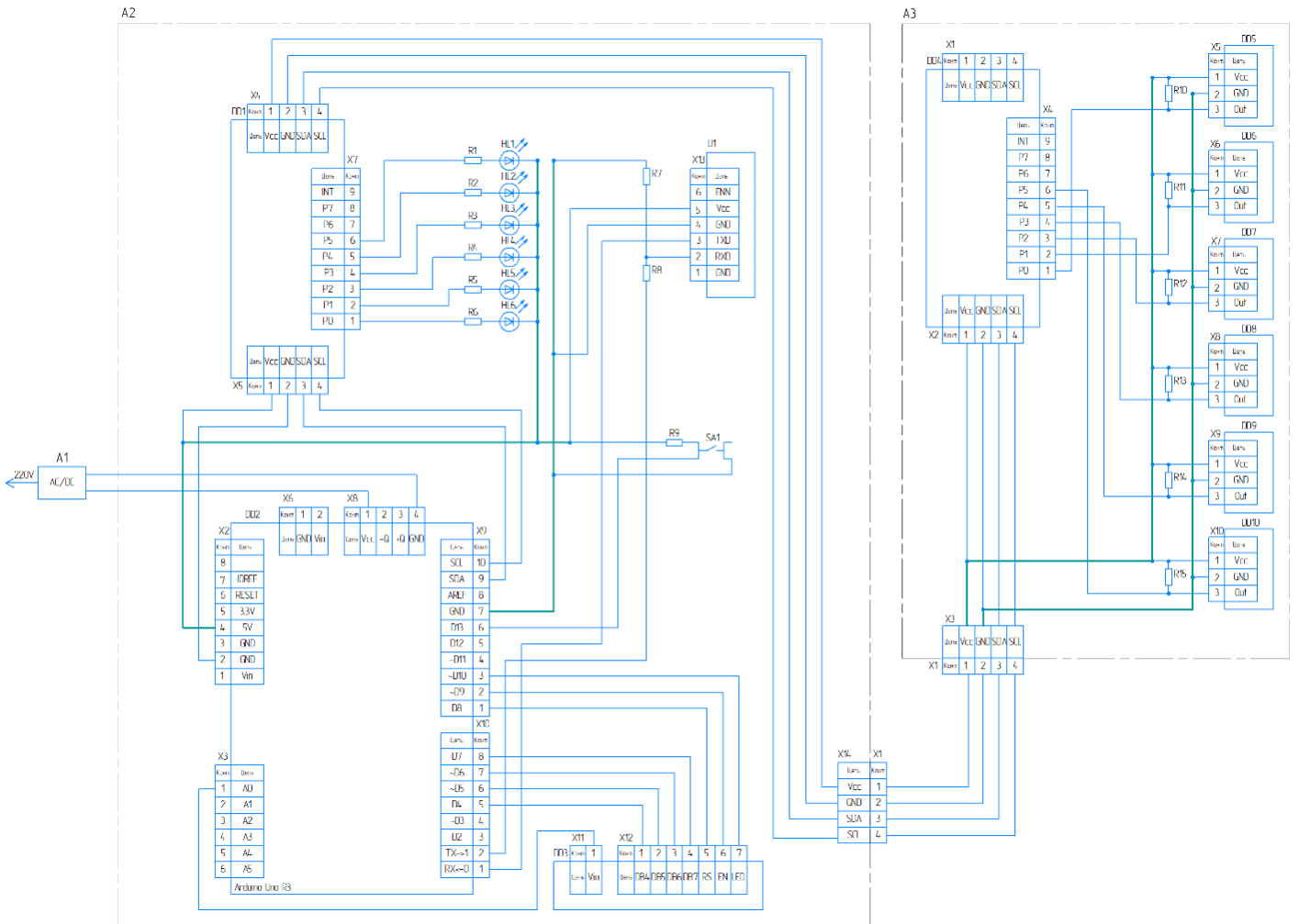


Рисунок 14 – Принципиальная схема мини-полигона

Перечень элементов принципиальной схемы приведён в приложении А.



## **2.6 Конструктив программно-аппаратного комплекса мини-полигона**

Разумный и простой подход к компоновочному решению конструктива программно-аппаратного комплекса мини-полигона состоит в разделении комплекса на два модуля:

- модуль управления, обеспечивающий функциональное взаимодействие с оператором мини-полигона,
- модуль собственно мини-полигона, данная часть программно-аппаратного комплекса обеспечивает функциональное взаимодействие полигона и транспортного средства.

Такая компоновочная организация обеспечивает относительную независимость конструктивных модулей и хорошую ремонтпригодность.

Конструктив модуля управления (рисунок 15) представляет собой корпус, в котором размещен контроллер и взаимодействующие с ним периферийные устройства.

Отдельно на рисунке показаны (по позициям):

1. Сервисное отверстие для регулирования яркости LCD Keypad Shield;
2. Место расположения LCD Keypad Shield;
3. Компоновочные позиции, соответствующие расположению светодиодных индикаторов, фиксирующих состояния сработавших датчиков;
4. Место расположения тактовая кнопка запуска программы мини-полигона;
5. Место размещения набора кнопок задания маршрута;
6. Место подключения интерфейса USB к контроллеру;
7. Место альтернативного питания;
8. Место для размещения разъема DB9, через который обеспечивается информационное взаимодействие с взаимодействующей частью программно-аппаратного комплекса

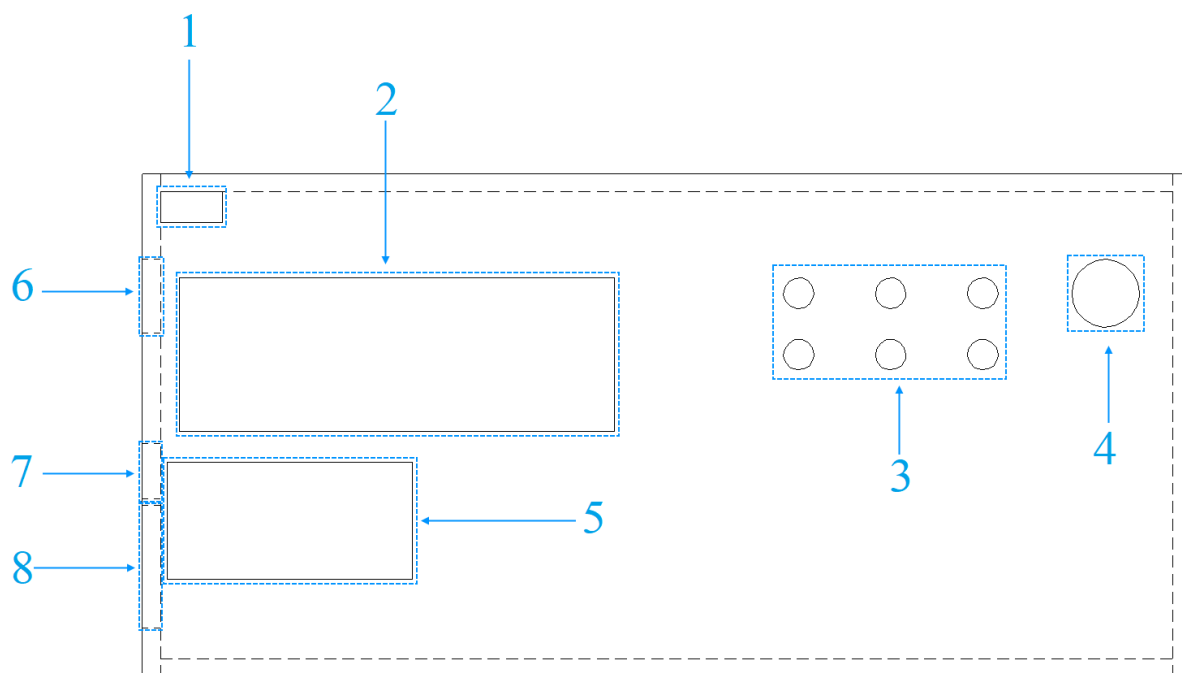


Рисунок 15 – Конструктив модуля управления

## 2.7 Программное обеспечение комплекса мини-полигона

Предпрограммная подготовка заключается в разработке алгоритма для реализации поставленных задач.

Разрабатываемый алгоритм должен учитывать возможность определения и фиксации возможного направления поворота транспортного средства на каждом из перекрестков мини-полигона [31].

Порядок маршрутизации транспортного средства описывается следующими действиями:

1. Предварительное задание маршрута следования транспортного средства, который будет описан последовательностью поворотов на перекрестках;

2. Ожидание мануального использования кнопки «Старт»;

3. Проведение индивидуального циклического опроса датчиков Холла. Изменение состояния одного из них под воздействием магнита транспортного средства соответствует изменению положения транспортного средства, которое информационно сопровождает контроллер;

4. Информация об этом событии отправляется в качестве сигнала управления на транспортное средство. На модуле управления включается светодиод, индицирующий прохождение транспортным средством перекрестка. Такое взаимодействие обеспечивает модуль Bluetooth;

5. Для обеспечения возможности остановки циклической работы транспортного средства необходимо повторное мануальное воздействие на кнопку «Старт», которая по своему назначению является старт-стопной кнопкой запуска и останова мини-полигона. В результате формируется команда завершения цикла, в соответствии с которой транспортное средство возвращается в исходное положение относительно реперных точек мини-полигона.

На рисунке 16 изображена блок-схема разработанного алгоритма.

Разработка управляющей программы учитывает особенности программирования платформы Arduino [1], [2], [13], [16].

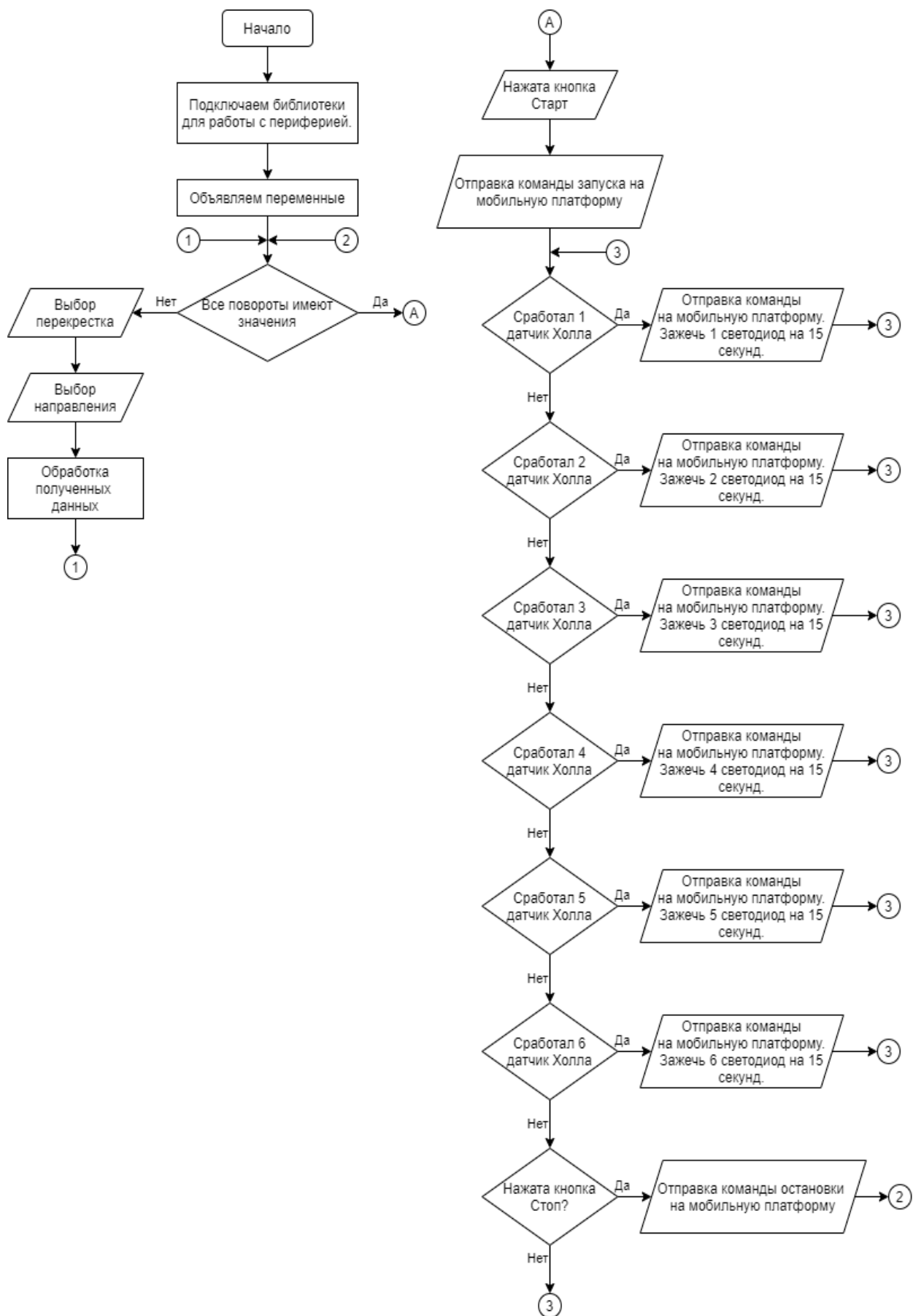
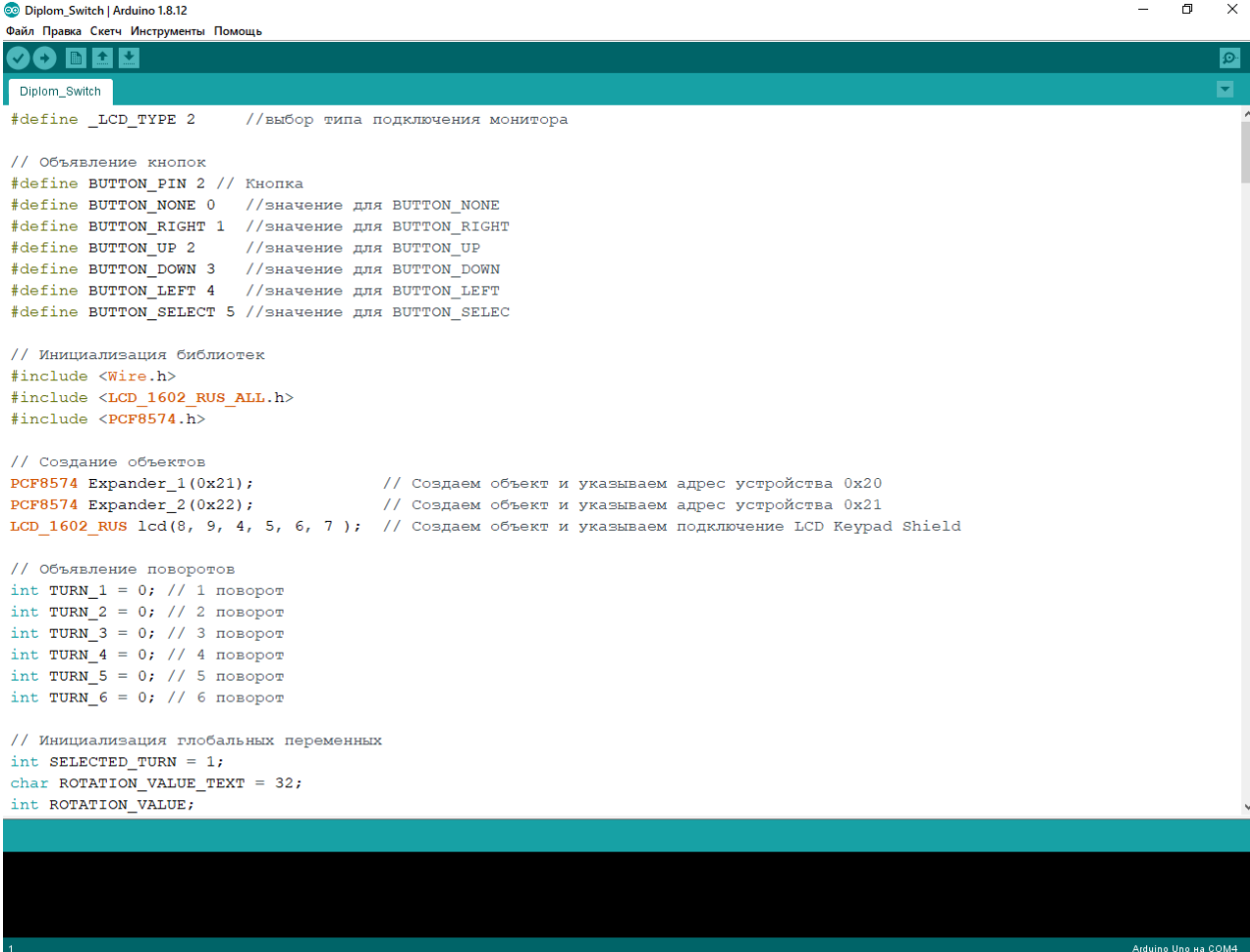


Рисунок 16 – Блок-схема функционирования модуля управления

Для начала определим программные функции для работы LCD-дисплеем (рисунок 17)



```
#define _LCD_TYPE 2 //выбор типа подключения монитора

// Объявление кнопок
#define BUTTON_PIN 2 // Кнопка
#define BUTTON_NONE 0 //значение для BUTTON_NONE
#define BUTTON_RIGHT 1 //значение для BUTTON_RIGHT
#define BUTTON_UP 2 //значение для BUTTON_UP
#define BUTTON_DOWN 3 //значение для BUTTON_DOWN
#define BUTTON_LEFT 4 //значение для BUTTON_LEFT
#define BUTTON_SELECT 5 //значение для BUTTON_SELEC

// Инициализация библиотек
#include <Wire.h>
#include <LCD_1602_RUS_ALL.h>
#include <PCF8574.h>

// Создание объектов
PCF8574 Expander_1(0x21); // Создаем объект и указываем адрес устройства 0x20
PCF8574 Expander_2(0x22); // Создаем объект и указываем адрес устройства 0x21
LCD_1602_RUS lcd(8, 9, 4, 5, 6, 7 ); // Создаем объект и указываем подключение LCD Keypad Shield

// Объявление поворотов
int TURN_1 = 0; // 1 поворот
int TURN_2 = 0; // 2 поворот
int TURN_3 = 0; // 3 поворот
int TURN_4 = 0; // 4 поворот
int TURN_5 = 0; // 5 поворот
int TURN_6 = 0; // 6 поворот

// Инициализация глобальных переменных
int SELECTED_TURN = 1;
char ROTATION_VALUE_TEXT = 32;
int ROTATION_VALUE;
```

Рисунок 17 – Программные функции для работы LCD-дисплеем

Для подключения портов ввода-вывода необходимо инициализировать COM-порты.

Для разработки программного обеспечения (результат называется скетчем) используются необходимые библиотеки:

- Wire.h – обеспечение взаимодействия устройств по шине I2C;
- LCD\_1602\_RUS\_ALL.h – инициализация LCD-дисплея с использованием символов русского языка;

– PCF8574.h – организация взаимодействия ситемы управления с модулем расширения PCF8574.

Объявление требуемых переменных представлено на рисунке 18.

```
#define _LCD_TYPE 2 // Выбор типа подключения монитора

// Объявление кнопок
#define BUTTON_PIN 2 // Тактовая кнопка
#define BUTTON_NONE 0 // Значение для BUTTON_NONE
#define BUTTON_RIGHT 1 // Значение для BUTTON_RIGHT
#define BUTTON_UP 2 // Значение для BUTTON_UP
#define BUTTON_DOWN 3 // Значение для BUTTON_DOWN
#define BUTTON_LEFT 4 // Значение для BUTTON_LEFT
#define BUTTON_SELECT 5 // Значение для BUTTON_SELECT

// Объявление поворотов
int TURN_1 = 0; // 1 поворот
int TURN_2 = 0; // 2 поворот
int TURN_3 = 0; // 3 поворот
int TURN_4 = 0; // 4 поворот
int TURN_5 = 0; // 5 поворот
int TURN_6 = 0; // 6 поворот

// Инициализация глобальных переменных
int SELECTED_TURN = 1; // Выбранный поворот
char ROTATION_VALUE_TEXT = 32; // Вывод значение поворота
int ROTATION_VALUE; // Значение поворота
boolean TURN_SELECTION_MENU_GO = true; // Переменная для запуска меню выбора поворота
boolean ROTATION_VALUE_MENU_GO = true; // Переменная для запуска меню выбора направления поворота
boolean START_MENU_GO = true; // Переменная для запуска меню старта
boolean START_GO = false; // Переменная для запуска меню работы макета мини-полигона
```

Рисунок 18 – Объявление переменных

Разработанные программные объекты должны поддерживать модули расширения и LCD-дисплея (рисунок 19) путем назначения необходимых адресов и контактов (пинов).

```
// Создание объектов
PCF8574 Expander_1(0x21); // Создаем объект и указываем адрес устройства 0x20
PCF8574 Expander_2(0x22); // Создаем объект и указываем адрес устройства 0x21
LCD_1602_RUS lcd(8, 9, 4, 5, 6, 7 ); // Создаем объект и указываем подключение LCD Keypad Shield
```

Рисунок 19 – Создание объектов

Базовая функция Setup (рисунок 20) позволяет организовать запуск последовательного порта, а также установить следующие уровни пинов:

– режим пина 1 для модуля расширения;

- режим пины 2 для тактовой кнопки;
- уровень пинов в состояние 1.

После запуска программного обеспечения дисплея на его экране выводится информация, заданная пользователем.

```
void setup()
{
  Serial.begin(9600);

  Expander_1.pinMode(P0, OUTPUT);
  Expander_1.pinMode(P1, OUTPUT);
  Expander_1.pinMode(P2, OUTPUT);
  Expander_1.pinMode(P3, OUTPUT);
  Expander_1.pinMode(P4, OUTPUT);
  Expander_1.pinMode(P5, OUTPUT);

  Expander_1.digitalWrite(P0, HIGH);
  Expander_1.digitalWrite(P1, HIGH);
  Expander_1.digitalWrite(P2, HIGH);
  Expander_1.digitalWrite(P3, HIGH);
  Expander_1.digitalWrite(P4, HIGH);
  Expander_1.digitalWrite(P5, HIGH);

  Expander_2.pinMode(P0, INPUT_PULLUP);
  Expander_2.pinMode(P1, INPUT_PULLUP);
  Expander_2.pinMode(P2, INPUT_PULLUP);
  Expander_2.pinMode(P3, INPUT_PULLUP);
  Expander_2.pinMode(P4, INPUT_PULLUP);
  Expander_2.pinMode(P5, INPUT_PULLUP);

  pinMode(BUTTON_PIN, INPUT);

  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("токарев д.г.");
  lcd.setCursor(0, 1);
  lcd.print("элм-2001а");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Мини-полигон");
  lcd.setCursor(0, 1);
  lcd.print("Версия 0.2.1");
  delay(2000);
  lcd.clear();
}
```

Рисунок 20 – Базовая функция Setup

Рисунок 20 содержит переменную, организующую обработку кнопочных сигналов для LCD Keypad Shield.

Обработка заключается в сравнении аналоговых и заданных данных переменной в соответствии с установленными значениями (рисунок 21).

```
int getPressedButton()
{
    int buttonValue = analogRead(0);
    delay(180);

    if (buttonValue < 50) {
        return BUTTON_RIGHT;
    }
    else if (buttonValue < 150) {
        return BUTTON_UP;
    }
    else if (buttonValue < 300) {
        return BUTTON_DOWN;
    }
    else if (buttonValue < 500) {
        return BUTTON_LEFT;
    }
    else if (buttonValue < 800) {
        return BUTTON_SELECT;
    }
    return BUTTON_NONE;
}
```

Рисунок 21 – Переменная обработки сигнала

Базовая функция Loop (рисунок 22) предусматривает возможность организации функции выбора поворота транспортного средства:

```
void loop()
{
    TURN_SELECTION_MENU();
}
```

Рисунок 22 – Функция Loop



Такой подход позволяет описать поворот транспортного средства в виде определенного кода функции (рисунок 23):

```
void TURN_SELECTION_MENU()
{
  if (TURN_1 != 0 && TURN_2 != 0 && TURN_3 != 0 && TURN_4 != 0 && TURN_5 != 0 && TURN_6 != 0)
  {
    START_MENU();
  }
  else
  {
    while (TURN_SELECTION_MENU_GO == true)
    {
      int BUTTON_TURN_SELECTION = getPressedButton();
      switch (BUTTON_TURN_SELECTION)
      {
        case BUTTON_UP:
          if (SELECTED_TURN < 6)
          {
            SELECTED_TURN ++;
          }
          else
          {
            TURN_WRONG_UP();
          }
          break;
        case BUTTON_DOWN:
          if (SELECTED_TURN > 1)
          {
            SELECTED_TURN --;
          }
          else
          {
            TURN_WRONG_DOWN();
          }
          break;
        case BUTTON_SELECT:
          TURN_SELECT();
          BUTTON_TURN_SELECTION = 0;
          break;
        default:
          lcd.setCursor(0, 0);
          lcd.print("Выберите поворот:");
          lcd.setCursor(0, 1);
          lcd.print(SELECTED_TURN, DEC);
      }
    }
  }
  ROTATION_VALUE_MENU();
}
```

Рисунок 23 – Функция выбора поворота транспортного средства

Функция обеспечивает проведение опроса состояния переменных, описывающих повороты. При выборе пользователем маршрута перемещения транспортного средства значения переменных не равны нулю. В этом случае обеспечивается пуск мини-полигона. После выполнения первой функции, описывающей поворот, проверяется значение переменной. Циклическое исполнение операции продолжается до тех пор, пока значение переменной истинно. В противном случае внутри цикла присваиваем значение новой переменной в соответствии с пользовательским нажатием кнопки управления. При проведении операции проверяется непротиворечивость введённых данных (для кнопки «вверх» дополнительное приращение значения на единицу, например двойное нажатие). В противном случае обрабатывается сообщение об ошибке (рисунок 24).

```
void TURN_WRONG_UP ()
{
    lcd.clear ();
    lcd.setCursor (0, 0);
    lcd.print ("Значение");
    lcd.setCursor (0, 1);
    lcd.print ("ВЫШЕ 6!");
    delay (2000);
    lcd.clear ();
}
```

Рисунок 24 – Обработка сообщения при нажатии кнопки «вверх»

В случае повторного нажатия кнопки «вниз» также производится проверка. Однако в этом случае значение переменной убывает на единицу. При этом ошибка обрабатывается по процедуре, показанной на рисунке 25.

```

void TURN_WRONG_DOWN()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Значение ниже 1!");
  delay(2000);
  lcd.clear();
}

```

Рисунок 25 – Обработка сообщения при нажатии кнопки «вниз»

Пользовательское нажатие кнопки обеспечивает вызов функции, обрабатывающей направление поворота (рисунок 26). Результатом обработки является вывод на экран дисплея номера поворота с последующей его очисткой. Далее принимается ложным значение переменной выбора направления поворота и происходит обнуление нажатой кнопки.

```

void TURN_SELECT()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Выбран поворот:");
  lcd.setCursor(0, 1);
  lcd.print(SELECTED_TURN, DEC);
  delay(1500);
  lcd.clear();
  TURN_SELECTION_MENU_GO = false;
}

```

Рисунок 26 – Функция обработки выбора поворота

При отсутствии нажатия на кнопки на дисплее отображается информация о последнем пройденном перекрестке.

Для обеспечения обслуживания очередного поворота вызывается соответствующее меню (рисунок 27).

Алгоритм циклически функционирует до тех пор, пока изменение направление движения отсутствует, т. е. значение переменной поворота истинно. При изменении направления переменной присваивается значение новой нажатой кнопки.

```
void ROTATION_VALUE_MENU()
{
  while (ROTATION_VALUE_MENU_GO == true){
    int BUTTON_ROTATION_VALUE_MENU = getPressedButton();

    switch (BUTTON_ROTATION_VALUE_MENU){
      case BUTTON_DOWN:
        ROTATION_VALUE_TEXT = 95;
        ROTATION_VALUE = 4;
        break;
      case BUTTON_UP:
        ROTATION_VALUE_TEXT = 94;
        ROTATION_VALUE = 1;
        break;
      case BUTTON_RIGHT:
        ROTATION_VALUE_TEXT = 62;
        ROTATION_VALUE = 2;
        break;
      case BUTTON_LEFT:
        ROTATION_VALUE_TEXT = 60;
        ROTATION_VALUE = 3;
        break;
      case BUTTON_SELECT:
        ROTATION_SELECT();
        BUTTON_ROTATION_VALUE_MENU = 0;
        ROTATION_VALUE_MENU_GO = false;
        break;
      default:
        lcd.setCursor(0, 0);
        lcd.print("Сторона поворота:");
        lcd.setCursor(0, 1);
        lcd.write(ROTATION_VALUE_TEXT);
    }
  }
  TURN_SELECT(ROTATION_VALUE);
}
```

---

Рисунок 27 – Функция меню выбора значения поворота

В этом случае в соответствии с программой активизируются несколько блоков значений, обеспечивающих информационное обслуживание нажатой кнопки и вывод соответствующей информации на дисплей.

Нажатие соответствующей кнопки приводит к вызову функции, обрабатывающей выбранный поворот (рисунок 28). При этом информация дисплея содержит выбранное направление. Циклическая работа прерывается, обнуляется состояние значения кнопки.

```
void ROTATION_SELECT ()
{
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Выбрана сторона:");
    lcd.setCursor(0, 1);
    lcd.write(ROTATION_VALUE_TEXT);
    delay(1500);
    lcd.clear();
}
```

Рисунок 28 – Функция обработки поворота

Если никакая кнопка не нажата, выводится текущая информация о последнем повороте транспортного средства.

При нажатии кнопки заканчивается цикл обслуживания последней команды и происходит обработка новых данных, полученных при нажатии кнопки (рисунок 29).

Алгоритмическая конструкция позволяет преобразовывать поворот в значение выбранного направления. Таким образом пользователь может последовательно произвести выбор всех изменений направления движения. Далее комплекс стартует при нажатии стартовой кнопки.

```

void TURN_SELECT(int VALUE)
{
    switch (SELECTED_TURN)
    {
        case 1: TURN_1 = VALUE; break;
        case 2: TURN_2 = VALUE; break;
        case 3: TURN_3 = VALUE; break;
        case 4: TURN_4 = VALUE; break;
        case 5: TURN_5 = VALUE; break;
        case 6: TURN_6 = VALUE; break;
    }

    ROTATION_VALUE_MENU_GO = true;
    TURN_SELECTION_MENU_GO = true;
    TURN_SELECTION_MENU();
}

```

Рисунок 29 – Функция обработки введённых данных

В результате происходит циклический запуск программы, который обеспечивает ее функционирование, пока не изменится значение переменной, соответствующей старту (рисунок 30). Последующие действия связаны с переменной новой кнопки. При этом без нажатия кнопки дисплей показывает ожидание нажатия. При нажатии кнопки дисплей сообщает о пуске мини-полигона. Результатом программной трансформации является запуск меню мини-полигона.

В функции меню старта (рисунок 31) происходит запуск цикла, обеспечивающий запуск меню полигона. Функция отслеживает состояние тактовой кнопки и датчиков Холла. При этом состояния датчиков определяют работу мини-полигона. Функция обрабатывает данные от датчиков.

```

void START_MENU()
{
  while (START_MENU_GO == true)
  {
    int ButtonState = 0;
    ButtonState = digitalRead(BUTTON_PIN);
    switch (ButtonState)
    {
      case HIGH:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("МИНИ-ПОЛИГОН");
        lcd.setCursor(0, 1);
        lcd.print("запущен!");
        delay(2500);
        lcd.clear();
        START_GO = true;
        START_MENU_GO = false;
        break;
      default:
        lcd.setCursor(0, 0);
        lcd.print("Запуск полигона: ");
        lcd.setCursor(0, 1);
        lcd.print("СТАРТ");
    }
  }
  Start();
}

```

Рисунок 30 – Запуск меню мини-полигона

```

void START() {
  Serial.print(0);
  while (START_GO == true) {
    int ButtonState = 0;
    ButtonState = digitalRead(BUTTON_PIN);

    uint8_t val0x22 = Expander_2.digitalRead(P0);
    uint8_t val1x22 = Expander_2.digitalRead(P1);
    uint8_t val2x22 = Expander_2.digitalRead(P2);
    uint8_t val3x22 = Expander_2.digitalRead(P3);
    uint8_t val4x22 = Expander_2.digitalRead(P4);
    uint8_t val5x22 = Expander_2.digitalRead(P5);

    lcd.setCursor(0, 0);
    lcd.print("Работаем...");

    SENSOR_HALL(val0x22, val1x22, val2x22, val3x22, val4x22, val5x22);

    if (ButtonState == HIGH) {
      STOP_START();
      TURN_SELECTION_MENU();
    }
  }
}

```

Рисунок 31 – Запуск меню старта

С помощью функции, показанной на рисунке 32, организуется опрос состояния всех датчиков Холла. При изменении состояния на пульте управления включается связанный с ним светодиод. Транспортному средству выдается команда на поворот, заданного направления.

В течение некоторого времени возникает ситуация ожидания проезда транспортного средства. После проезда в нужном направлении, индикация выключается.



```

void SENSOR_HALL(uint8_t SEN_0, uint8_t SEN_1, uint8_t SEN_2, uint8_t SEN_3, uint8_t SEN_4, uint8_t SEN_5)
{
  if (SEN_0 == 0) {
    Expander_1.digitalWrite(P0, LOW);
    Serial.print(TURN_1);
    delay(15000);
    Expander_1.digitalWrite(P0, HIGH);
  }
  if (SEN_1 == 0) {
    Expander_1.digitalWrite(P1, LOW);
    Serial.print(TURN_2);
    delay(15000);
    Expander_1.digitalWrite(P1, HIGH);
  }
  if (SEN_2 == 0) {
    Expander_1.digitalWrite(P2, LOW);
    Serial.print(TURN_3);
    delay(15000);
    Expander_1.digitalWrite(P2, HIGH);
  }
  if (SEN_3 == 0) {
    Expander_1.digitalWrite(P3, LOW);
    Serial.print(TURN_4);
    delay(15000);
    Expander_1.digitalWrite(P3, HIGH);
  }
  if (SEN_4 == 0) {
    Expander_1.digitalWrite(P4, LOW);
    Serial.print(TURN_5);
    delay(15000);
    Expander_1.digitalWrite(P4, HIGH);
  }
  if (SEN_5 == 0) {
    Expander_1.digitalWrite(P5, LOW);
    Serial.println(TURN_6);
    delay(15000);
    Expander_1.digitalWrite(P5, HIGH);
  }
}

```

Рисунок 32 – Функция опроса датчиков Холла

После проверки состояния всех датчиков Холла обеспечивается анализ состояния кнопки. При ее нажатии происходит остановка меню старта (рисунок 33). На экране дисплея выводится соответствующее сообщение. Переменные, которые описывают направление поворотов, обнуляются. Текущая переменная запуска старта транспортного средства – истина, переменная запуска мини-полигона – ложь. После выполнения поворота программа возвращается в начальное состояние.

```

void STOP_START ()
{
  Serial.print (4);
  lcd.clear ();
  lcd.setCursor (0, 0);
  lcd.print ("Мини-полигон");
  lcd.setCursor (0, 1);
  lcd.print ("остановлен!");
  delay (2500);
  lcd.clear ();
  TURN_1 = TURN_2 = TURN_3 = TURN_4 = TURN_5 = TURN_6 = 0;
  SELECTED_TURN = 1;
  START_MENU_GO == true;
  START_GO = false;
}

```

Рисунок 33 – Остановка меню старта

Выводы по главе 2:

- разработан программно-аппаратный комплекс мини-полигона;
- детально рассмотрена структурная и принципиальная схемы транспортного средства;
- разработано программное обеспечение, позволяющее управлять транспортным средством в пределах мини-полигона;
- разработанный комплекс позволяет обеспечить маршрутизацию и рациональное логистическое обслуживание транспортной системы мини-полигона.

### **Глава 3 Развитие беспилотного транспортного средства на основе модели транспортного средства мини-полигона**

При создании транспортного средства, первым делом встаёт вопрос о реализации подсистем, отвечающих за движение и управление движением [8], [17]. Так как эксплуатация транспорта планируется внутри полигона, было принято решение об использовании для изменения направления движения только бортового поворота. Было принято, что управление платформой должно быть реализовано в двух вариантах, как проводное, так и дистанционное через радиоканал.

В соответствии с заданием, были разработаны два варианта структурных схем подсистемы «Движение», а также структура элементов пульта мануального управления. Структура пульта остается неизменной, как в варианте с проводным, так и с дистанционным управлением.

#### **3.1 Состав комплекса транспортного средства**

Перечень подсистем, необходимых для движения и управления транспортным средством [20]:

1. Подсистема ПДУ-П (Пульт Дистанционного Управления – Проводной) является проводным пультом дистанционного управления, подключенным непосредственно к центральному контроллеру (ЦК), с помощью кабеля связи. Служит для управления движением транспорта, для записи траекторий и пуска платформы по траектории, записанной ранее.

2. Подсистема ПДУ-Р (Пульт Дистанционного Управления – Радио) является беспроводным аналогом ПДУ-П, может выполнять все функции ПДУ-П, но связана с контроллером через радиомодем, что обеспечивает управление роботом на расстоянии.

3. Подсистема «Движение», при проводном управлении – подсистема, обеспечивающая управляемое передвижением транспорта. В подсистему

входят привода мотор-колес и контроллеры, обеспечивающие заданную при помощи ПДУ-П работу мотор-колес.

4. Подсистема «Движение», при радиоуправлении – подсистема аналогичная подсистеме «Движение» при проводном управлении, но управление обеспечивается с ПДУ-Р.

Рассмотрим подсистему движение при проводном управлении. На рисунке 34 представлен вариант управления приводами при помощи проводного пульта.

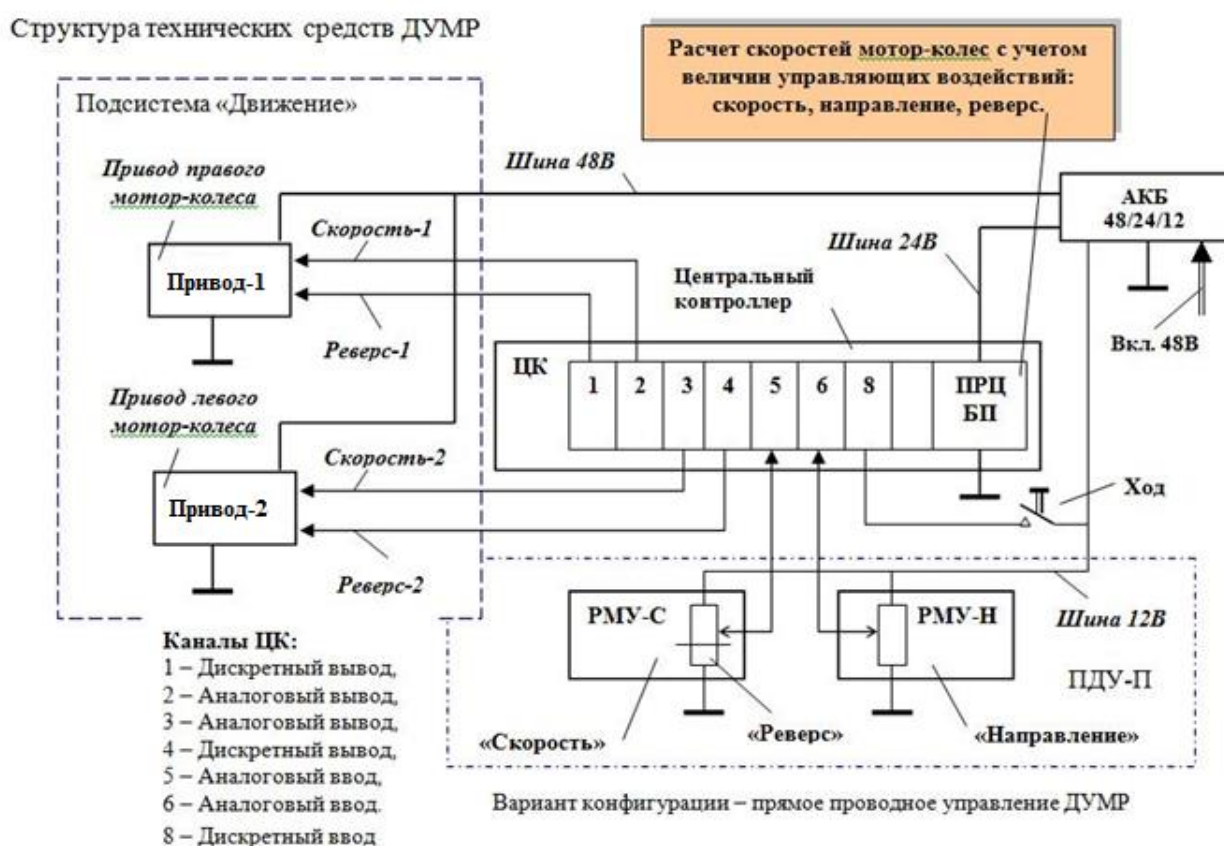


Рисунок 34 – Подсистема «Движение», проводная

На рисунке 34 представлены:

РМУ – ручка мануального управления,

АКБ – аккумуляторная батарея (составная),

ПДУ-П – пульт дистанционного управления, проводной,

ПРЦ – процессор ЦК,

БП – блок электропитания ЦК.

В состав проводной подсистемы «Движение» входят приводы левого и правого мотор-колес. Приводы управляются сигналами «Скорость» и «Реверс». Сигнал «Скорость» должен быть рассчитан с учетом величин управляющих воздействий «Скорость» и «Направление». Эти сигналы вырабатываются проводным пультом дистанционного управления (ПДУ-П). При движении «Прямо», одинаковые по величине напряжения сигналы «Скорость», подаются на оба привода (левый и правый). Эти аналоговые сигналы формируются ручкой мануального управления по скорости (РМУ-С), ручка мануального управления по направлению (РМУ-Н) находится в среднем положении («0» градусов).

При выполнении бортовых поворотов налево необходимо понижать скорость левого мотор-колеса, а при бортовых поворотах направо – правого. Степень понижения скорости определяется положением ручки «Направление», то есть величиной напряжения поступающей в ЦК с РМУ-Н. Для расчета величин сигналов скорости используется центральный контроллер. Контроллер также обеспечивает генерацию сигналов «Реверс». Если РМУ-С смещается в крайне нижнее положение, то обеспечивается реверс мотор-колес, а положение ручки РМУ-Н определяет соотношение их скоростей.

Работа подсистемы «Движение» при радиоуправлении, аналогична работе при проводном управлении. Но сигналы на ЦК поступают не с пульта управления напрямую, как при проводном управлении, а с радиомодема – приемника (РМД 2). РМД 2, в свою очередь, принимает сигналы с радиомодема – передатчика (РМД 1), установленного на пульте дистанционного управления.

Пульт дистанционного управления, работает так же, как и пульт проводного управления, только сигналы передаются в радиомодем – приемник с функцией телеметрического контроллера. Перечень и назначения

каналов РМД 1 указаны на рисунке 35. В РМД 2 используется канал RS-232 для связи с ЦК. Канал дискретного вывода РМД 2 используется для управления звуковым сигнализатором. Режим телеметрического контроллера РМД 1 используется для передачи сигналов: «Ход», «Сирена», «Скорость» и «Направление».

Питание РМД 1 обеспечивается встроенным аккумулятором (6В).

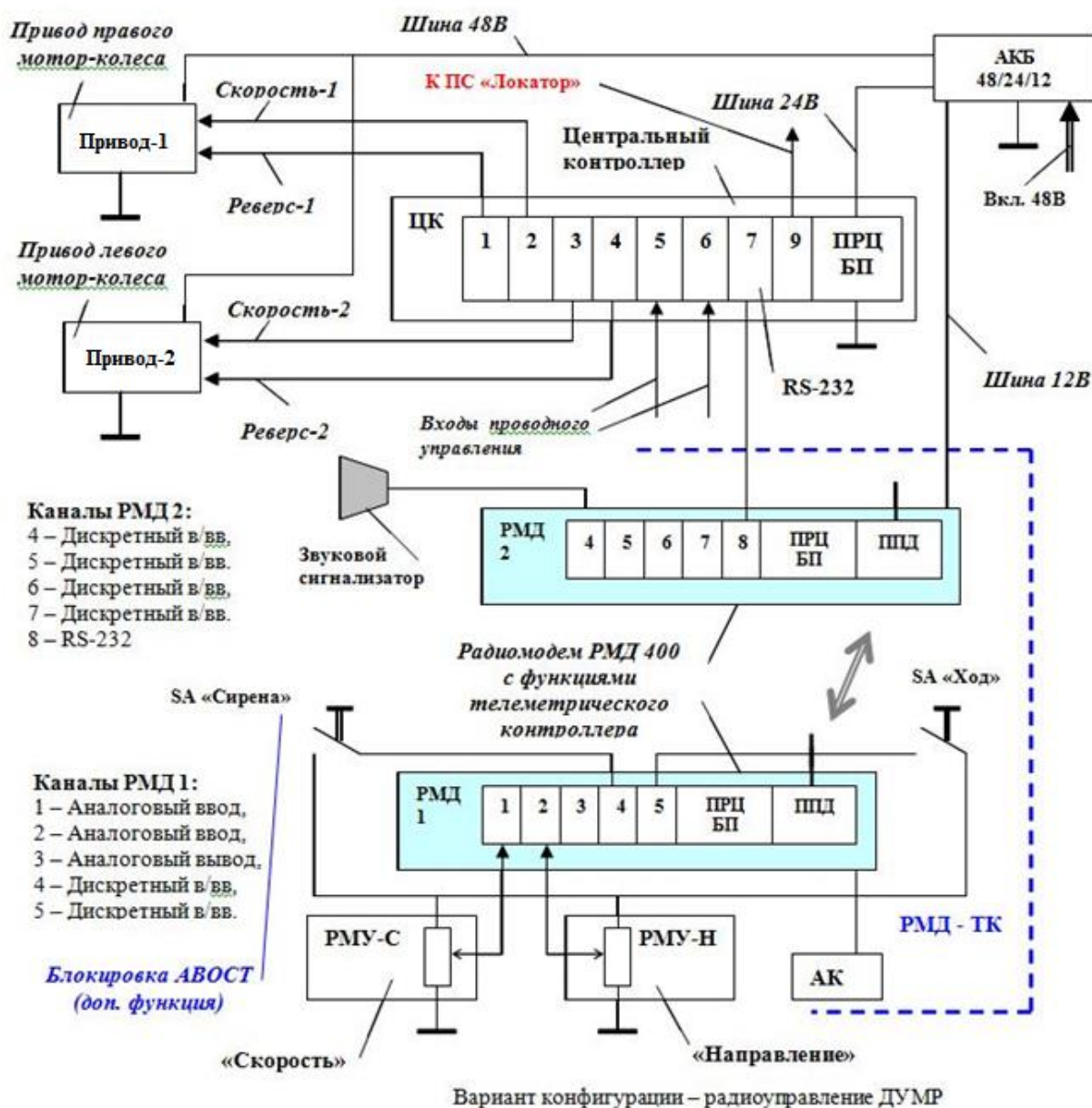


Рисунок 35 – Подсистема «Движение», радиоуправление

Работа подсистемы «Движение» при радиоуправлении аналогична работе при проводном управлении. Но сигналы на ЦК поступают не с пульта управления напрямую, как при проводном управлении, а с радиомодема – приемника РМД 2. РМД 2, в свою очередь, принимает сигналы с радиомодема – передатчика РМД 1, установленного на пульте дистанционного управления.

Пульт дистанционного управления, работает так же, как и пульт проводного управления, только сигналы передаются в радиомодем – приемник с функцией телеметрического контроллера. Перечень и назначения каналов РМД 1, указаны на рисунке 35. В РМД 2 используется канал RS-232 для связи с ЦК. Канал дискретного вывода РМД 2 используется для управления звуковым сигнализатором. Режим телеметрического контроллера РМД 1 используется для передачи сигналов: «Ход», «Сирена», «Скорость» и «Направление».

Питание РМД 1 обеспечивается встроенным аккумулятором с напряжением 6В.

### **3.2 Структура элементов пульта мануального управления**

Пульт управления ДУМР, делится на две основных панели: управление движением (рисунок 36) и управление памятью траекторий (рисунок 37).

На панели управления движением имеется переключатель «Скорость» задающий один из шести скоростных режимов. Также в ПДУ применяется джойстик «Направление», угол наклона которого задает соотношение угловых скоростей мотор-колес. На джойстике управления установлена кнопка «Ход», включающая привода колес.

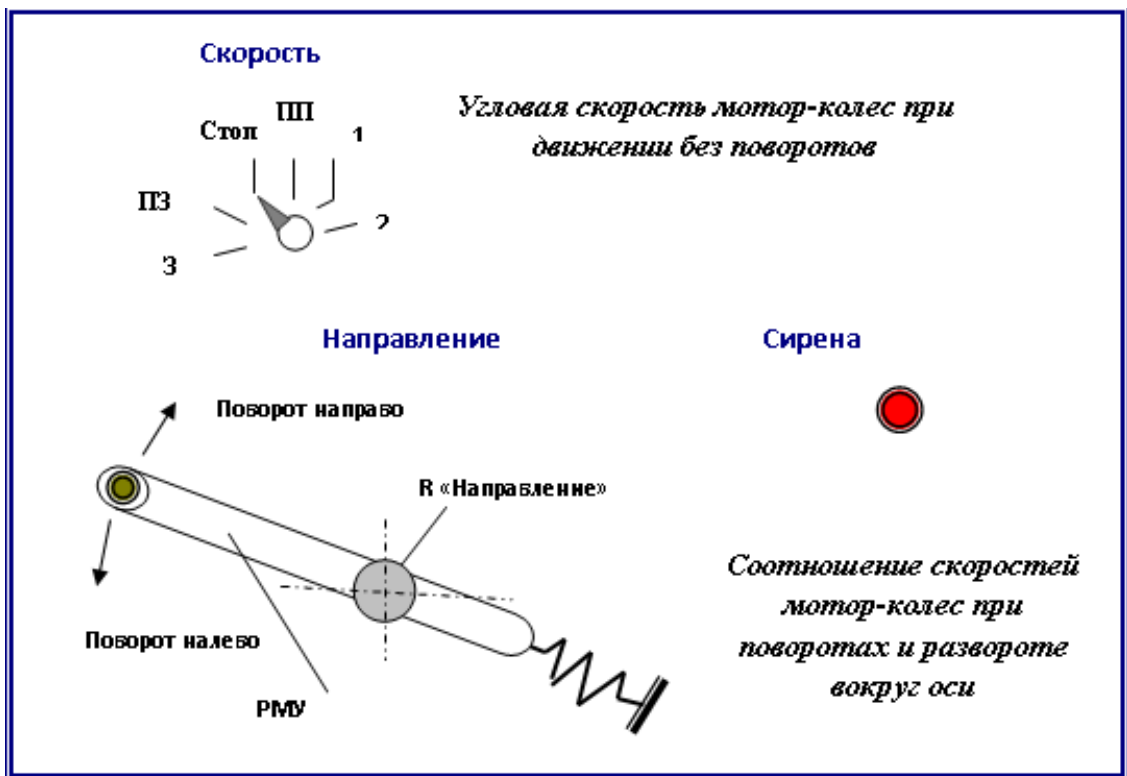


Рисунок 36 – Пульт управления движением

На панели управления движением имеется переключатель «Скорость» задающий один из шести скоростных режимов. Также в ПДУ применяется джойстик «Направление», угол наклона которого задает соотношение угловых скоростей мотор-колес. На джойстике управления установлена кнопка «Ход», включающая привода колес.

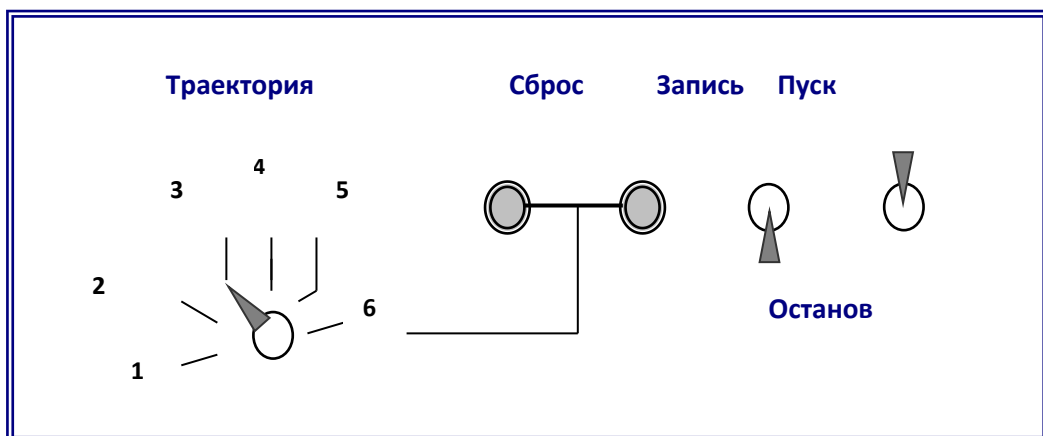


Рисунок 37 – Пульт управления траекториями



На панели управления траекториями имеется переключатель номеров траекторий, а также тумблеры для включения записи траектории или пуска транспорта по записанной траектории. Также имеется кнопки сброса файла ранее записанной траектории.

### **3.3 Математическая модель и алгоритм бортового управления транспортным средством**

При создании модели беспилотного транспортного средства одним из основных вопросов является разработка варианта системы управления этим средством, как ходовой платформой, при помощи пульта дистанционного управления. Было очевидно, что использование бортового поворота более рационально, но это же и усложняет систему управления. Идея бортового поворота заключается в следующем: транспортное средство будет поворачивать в ту сторону, где скорость колес борта меньше [6].

Обычно задача распределения скоростей на бортах решается оператором, имеющим два рычага или джойстика для управления скоростью каждого из бортов. В данной разработке для удобства эксплуатации платформы был выбран вариант с одним наклонным джойстиком, это заставило продумать возможность платформы автоматически просчитывать и задавать скорости бортов в зависимости от наклона джойстика. С данной задачей без труда справляются современные центральные контроллеры с правильно написанной программой. Для реализации такой программы необходимо создать ее алгоритм [5].

Алгоритмы управления движением дистанционно-управляемыми и автономными мобильными роботами в доступных источниках не публикуются. Поэтому потребовалось разработать алгоритм управления направлением движения ходовой платформы.

Управление направлением движения ходовой платформы с бортовым поворотом обеспечивается выбором соотношения угловых скоростей

вращения мотор-колес левого ( $\omega_{\text{ЛМК}}$ ) и правого ( $\omega_{\text{ПМК}}$ ) бортов – расчетом коэффициента направления  $k_{\text{н}}$ , в зависимости от положения датчика направления (джойстик «Направление»).

При движении «прямо»:

$$\omega_{\text{ЛМК}} = \omega_{\text{ПМК}} = \omega_{\text{зад}}, k_{\text{н}} = \omega_{\text{ЛМК}} / \omega_{\text{ПМК}} = +1. \quad (4)$$

Угловая скорость вращения мотор-колес правого и левого борта равна заданной для платформы (переключатель «Скорость», алгоритм обеспечения плавного изменения скоростей).

При поворотах «налево»:

$$\omega_{\text{ПМК}} = \omega_{\text{зад}}. \quad (5)$$

Угловая скорость вращения мотор-колеса правого борта равна заданной для платформы.

Угловая скорость вращения мотор-колеса левого борта определяется расчетным коэффициентом направления:

$$k_{\text{нл}} = f_{\text{нл}}(\alpha_{\text{н}}), \quad (6)$$

где  $\alpha_{\text{н}}$  – угол наклона джойстика «Направление»:

$$\omega_{\text{ЛМК}} = k_{\text{нл}} \cdot \omega_{\text{ПМК}}, k_{\text{нл}} = -1, \dots, +1. \quad (7)$$

При поворотах «направо»:

$$\omega_{\text{ЛМК}} = \omega_{\text{зад}}. \quad (8)$$

Угловая скорость вращения мотор-колеса левого борта равна заданной для платформы.

Угловая скорость вращения мотор-колеса правого борта определяется расчетным коэффициентом направления:

$$k_{\text{нп}} = f_{\text{нп}}(\alpha_{\text{н}}), \quad (9)$$

где  $\alpha_{\text{н}}$  – угол наклона джойстика «Направление»:

$$\omega_{\text{ПМК}} = k_{\text{нп}} \cdot \omega_{\text{ЛМК}}, k_{\text{нп}} = -1, \dots, +1. \quad (10)$$

Преобразование сигналов датчика угла поворота (джойстика «Направление») в сигналы управления угловой скоростью мотор-колес левого и правого борта выполняется центральным контроллером ходовой платформы.

Соотношение угловых скоростей мотор-колес задает радиус поворота (рисунок 38).

Время движения ходовой платформы по окружности ( $360^\circ$ ) можно определить по формулам:

$$t_{\text{окр}} = 2\pi \cdot R_{\text{ЛМК}} / \omega_{\text{ЛМК}} \text{ (для левого мотор-колеса),} \quad (11)$$

$$t_{\text{окр}} = 2\pi \cdot (R_{\text{ПМК}} + B_{\text{МК}}) / \omega_{\text{ПМК}} \text{ (для правого мотор-колеса).} \quad (12)$$

Следовательно:

$$2\pi \cdot R_{\text{ЛМК}} / \omega_{\text{ЛМК}} = 2\pi \cdot (R_{\text{ЛМК}} + B_{\text{МК}}) / \omega_{\text{ПМК}}; \quad (13)$$

$$\omega_{\text{ЛМК}} = \omega_{\text{ПМК}} \cdot R_{\text{ЛМК}} / (R_{\text{ЛМК}} + B_{\text{МК}}). \quad (14)$$

При движении «прямо»

$$\omega_{\text{ЛМК}} = \omega_{\text{ПМК}}, k_{\text{H}} = +1, \quad (15)$$

угол наклона джойстика  $\alpha_{\text{H}} = 0$ .

Получим:

$$R_{\text{ЛМК}} / (R_{\text{ЛМК}} + B_{\text{МК}}) = 1. \quad (16)$$

При  $B_{\text{МК}} > 0$ ,  $R_{\text{ЛМК}} = \infty$ .

При повороте вокруг левого колеса  $\omega_{\text{ЛМК}} = 0$ .

Примем, исходя из эргономических требований, угол наклона джойстика «Направление», равным половине максимального отклонения против часовой стрелки:

$$\alpha_{\text{H}} = -\alpha_{\text{Hmax}} / 2. \quad (17)$$

Получим:

$$\omega_{\text{ПМК}} \cdot R_{\text{ЛМК}} / (R_{\text{ЛМК}} + B_{\text{МК}}) = 0; \quad (18)$$

$$R_{\text{ЛМК}} / (R_{\text{ЛМК}} + B_{\text{МК}}) = 0; \quad (19)$$

$R_{\text{ЛМК}} = 0$ , рисунок 38.

При развороте вокруг оси платформы:

$$\omega_{\text{ЛМК}} = -\omega_{\text{ПМК}}. \quad (20)$$

Примем угол наклона джойстика «Направление» равным максимальному отклонению, против часовой стрелки:

$$\alpha_{\text{H}} = -\alpha_{\text{Hmax}}. \quad (21)$$

Получим:

$$R_{\text{ЛМК}} / (R_{\text{ЛМК}} + B_{\text{МК}}) = -1, \quad (22)$$

$$R_{\text{ЛМК}} = -(R_{\text{ЛМК}} + B_{\text{МК}}), \quad (23)$$

$$2R_{\text{ЛМК}} = -B_{\text{МК}}, \quad (24)$$

$$R_{\text{ЛМК}} = -B_{\text{МК}} / 2, \quad (25)$$

что показано на рисунке 38.

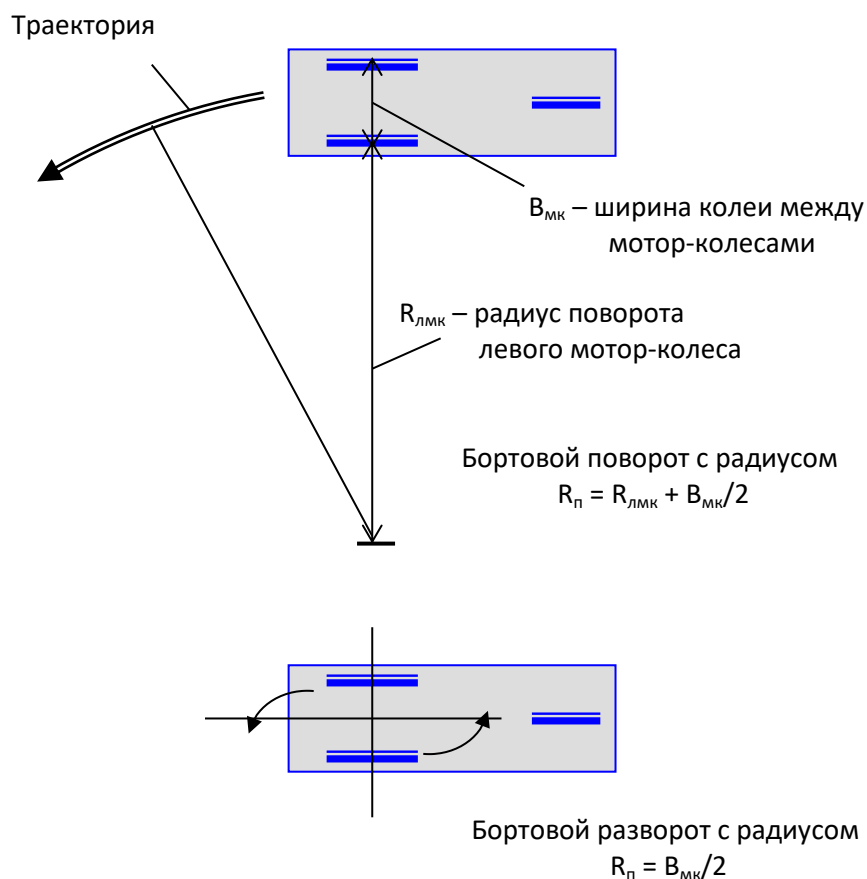


Рисунок 38 – Схемы движения ходовой платформы при расчетном управлении угловыми скоростями мотор-колес

По контрольным точкам зависимости  $k_{\text{НП}} = f_{\text{НП}}(\alpha_{\text{Н}})$ , которую задаем линейной, построим график, приведенный на рисунке 39, а.

Зависимость  $k_{\text{НП}} = f_{\text{НП}}(\alpha_{\text{Н}})$ , также задаем линейной, рисунок 39, б.

Для расчетов получена зависимость:

$$k_{\text{Н}} = 1, \text{ при } \alpha_{\text{Н}} = 0; \quad k_{\text{Н}} = f_{\text{НП}}(\alpha_{\text{Н}}) \text{ при } \alpha_{\text{Н}} < 0; \quad k_{\text{Н}} = f_{\text{НП}}(\alpha_{\text{Н}}) \text{ при } \alpha_{\text{Н}} > 0. \quad (26)$$

Линейные зависимости  $k_{нл} = f_{нл}(\alpha_n)$  и  $k_{нп} = f_{нп}(\alpha_n)$  определим по контрольным точкам построенных зависимостей:

$$k_{нл} = a_{л} \cdot \alpha_n + b_{л}; \quad k_{нп} = a_{п} \cdot \alpha_n + b_{п}. \quad (27)$$

Движение «прямо»:

$$1 = a_{л} \cdot 0 + b_{л}; \quad b_{л} = 1; \quad 1 = a_{п} \cdot 0 + b_{п}; \quad b_{п} = 1. \quad (28)$$

Бортовой разворот «налево»:

$$-1 = a_{л} \cdot (-\alpha_{нмаx}) + 1; \quad a_{л} = 2 / \alpha_{нмаx}. \quad (29)$$

Получаем зависимость:

$$k_{нл} = (2 / \alpha_{нмаx}) \alpha_n + 1, \quad \text{при } \alpha_n < 0. \quad (30)$$

Бортовой разворот «направо»:

$$-1 = a_{п} \cdot (\alpha_{нмаx}) + 1; \quad a_{п} = -2 / \alpha_{нмаx}. \quad (31)$$

Получаем зависимость:

$$k_{нп} = (-2 / \alpha_{нмаx}) \alpha_n + 1, \quad \text{при } \alpha_n > 0. \quad (32)$$

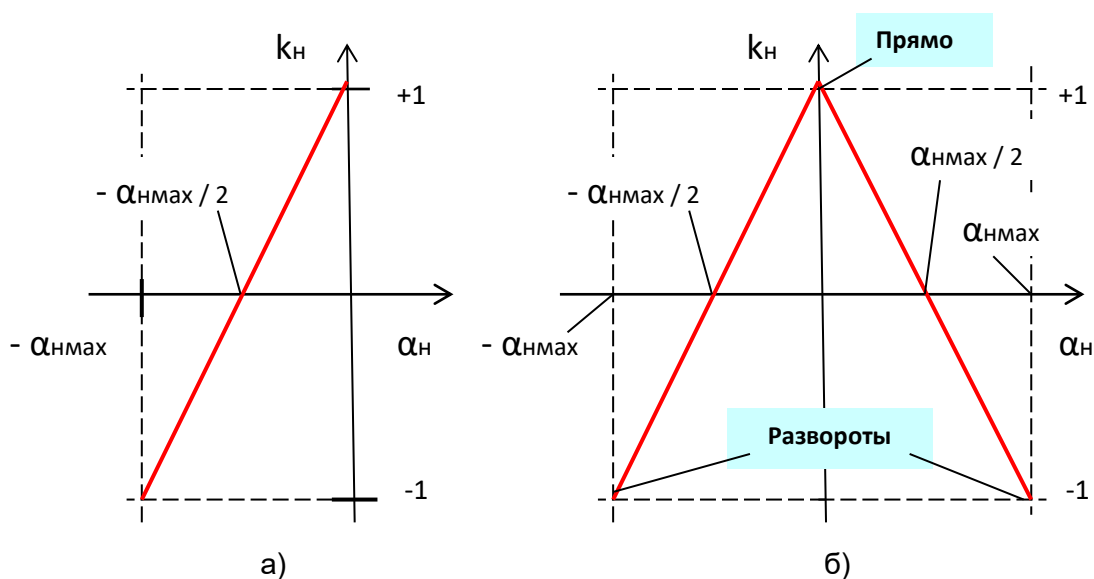


Рисунок 39 – Зависимости коэффициентов направления, определяющих соотношение угловых скоростей мотор-колес от угла наклона джойстика «Направление»

Кинематическая схема органа управления направлением движения (джойстика «Направление») приведена на рисунке 40.

Принято, исходя из эргономических требований:  $|\alpha_{нмаx}| = 50^\circ$ .

Соответствующий алгоритм определения скоростей мотор-колес, обеспечивающий управление направлением движения ходовой платформой приведен ниже, на рисунке 41.

Для корректного управления необходимо использование сигналов датчиков угловой скорости мотор-колес ( $\omega_{\text{ЛМК}}^*$  и  $\omega_{\text{ПМК}}^*$ ), которые формируются из импульсных сигналов ДПР (датчиков положения ротора двигателей).

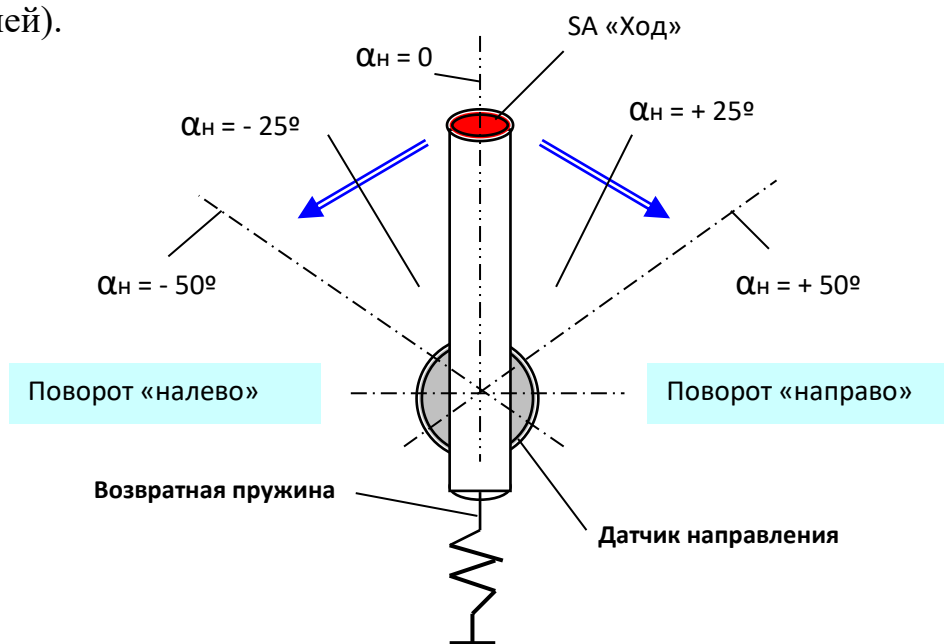


Рисунок 40 – Кинематическая схема ручки мануального управления направлением движения ПДУ (джойстик «Направление»)

Формирование сигнала «Сбой» вызывает аварийную сигнализацию. Вариант: мигающие индикаторы ТПУ, звуковой сигнал заданной тональности.

Далее (рисунки 41, 42), при наличии сигнала «Ход»:  $V_{\text{хп}} = 1$ , центральный контроллер приступает к реализации найденных величин угловых скоростей мотор-колес.

Определяются величины:

$$\Delta \omega_{\text{ЛМК}} = \omega_{\text{ЛМК}} - \omega_{\text{ЛМК}}^*, \Delta \omega_{\text{ПМК}} = \omega_{\text{ПМК}} - \omega_{\text{ПМК}}^*. \quad (33)$$

При  $\Delta \omega_{\text{ЛМК}} \leq |\delta|$  (предельное приращение угловой скорости при разгоне или торможении), величина  $\omega_{\text{ЛМК}}$  подается для исполнения на вход привода левого мотор-колеса:

$$\omega_{\text{ЛП}} := \omega_{\text{ЛМК}}. \quad (34)$$

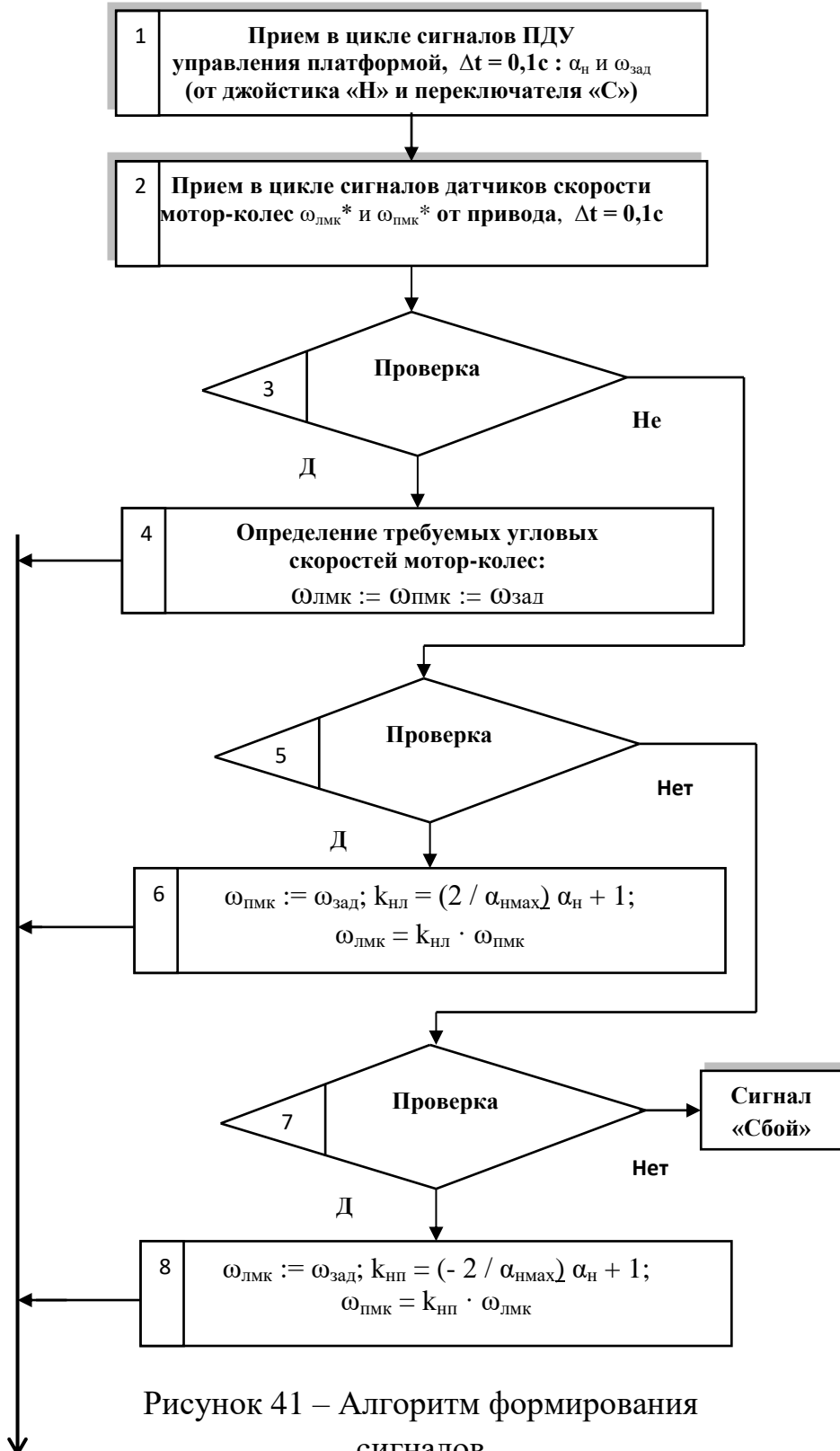


Рисунок 41 – Алгоритм формирования сигналов

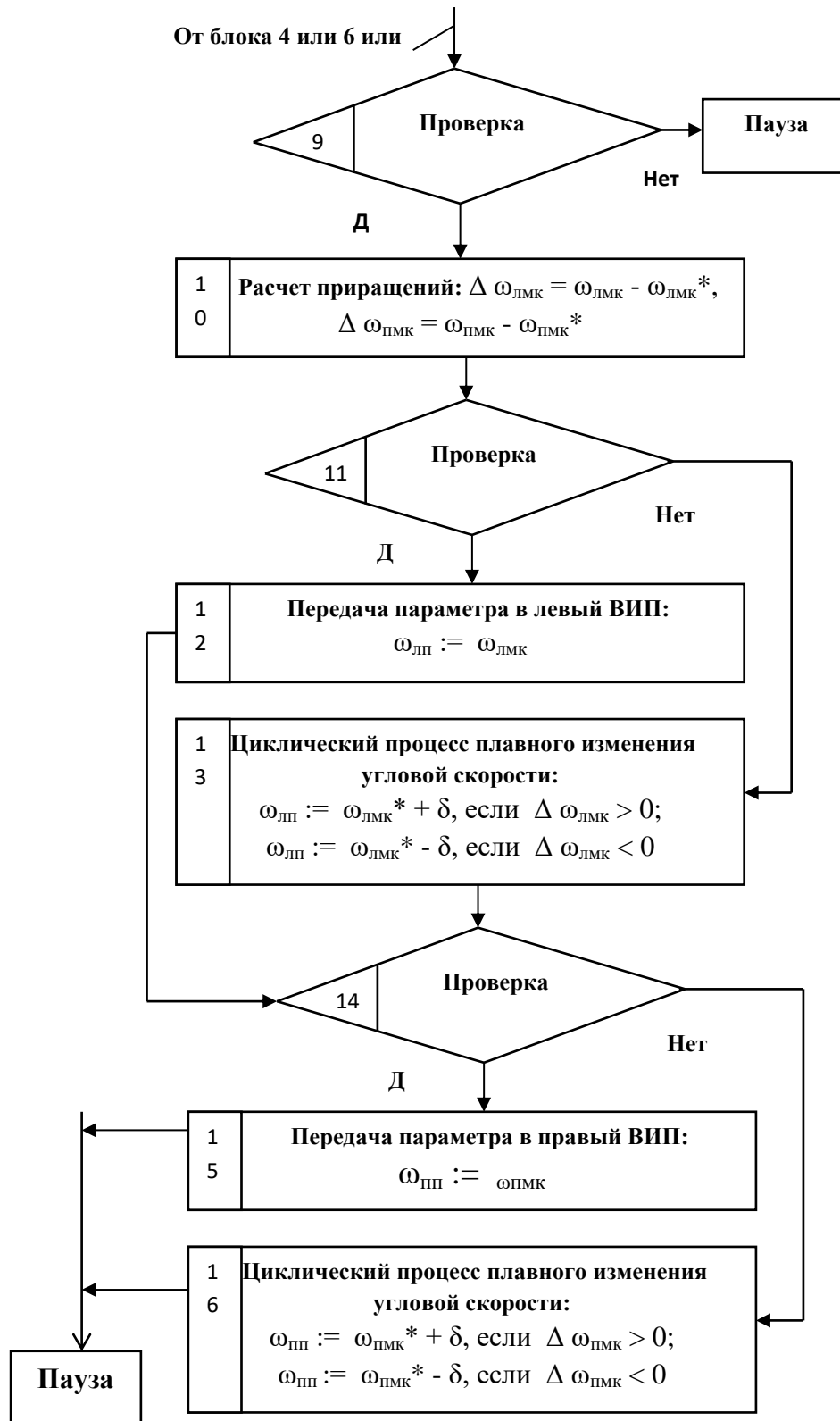


Рисунок 42 – Алгоритм формирования сигналов

Если  $\Delta \omega_{\text{ЛМК}} > \delta$ , то запускается подпрограмма (блок 13) плавного изменения  $\omega_{\text{ЛМК}}$ :

$$\omega_{\text{ЛП}} := \omega_{\text{ЛМК}}^* + \delta, \text{ если } \Delta \omega_{\text{ЛМК}} > 0, \quad (35)$$



$$\omega_{\text{лп}} := \omega_{\text{лмк}}^* - \delta, \text{ если } \Delta \omega_{\text{лмк}} < 0. \quad (36)$$

При  $\Delta \omega_{\text{лмк}} \leq |\delta|$ ,  $\omega_{\text{лп}} := \omega_{\text{лмк}}$ .

Корректировка угловой скорости левого борта завершена.

Аналогично выполняется корректировка угловой скорости правого борта.

### **3.4 Структура пульта управления транспортным средством**

Указанные на рисунке 43 напряжения имеют следующие значения:

$U_{\text{зап}} = 1$  (регистрация сигналов БМУ),

$U_{\text{исп}} = 1$  (блокирование сигналов БМУ),

$U_{\text{зап}} = 1 \ \& \ U_{\text{исп}} = 1, \ U_{\text{исп}} := 0$  (ЦК).

### **3.5 Управление скоростью транспортного средства**

В состав пульта управления движением входит переключатель «Скорость» на 6 положений:

- задняя (З) – на контроллер подается напряжение 6В от источника питания;
- ползучая задняя (ПЗ) – на контроллер с делителя, составленного из равных по величине резисторов, подается напряжение 4,8В;
- нулевая (Стоп) – на контроллер с делителя, составленного из равных по величине резисторов, подается напряжение 3,6В;
- ползучая передняя (ПП) – на контроллер с делителя, составленного из равных по величине резисторов, подается напряжение 2,4В;
- передняя первая (П1) – на контроллер с делителя, составленного из равных по величине резисторов, подается напряжение 1,2В;
- передняя вторая (П2) – на контроллер с делителя, составленного из равных по величине резисторов, подается напряжение 0В.

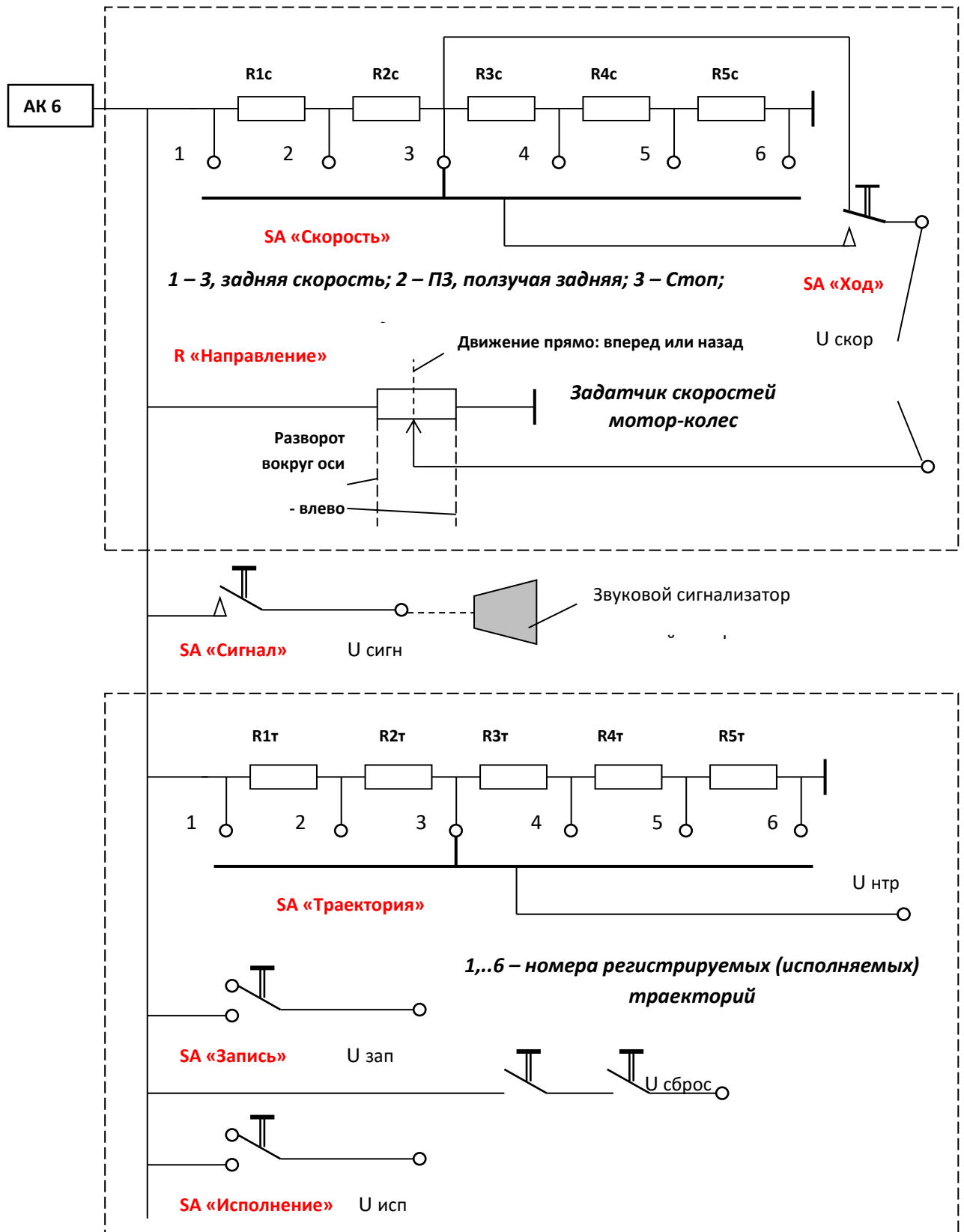


Рисунок 43 – Фрагменты пульта ручного управления

Уровни напряжения распознаются центральным контроллером и преобразуются в величины:

$$\omega_{\text{зад}} = \omega_{\text{пз}}, \omega_3, \omega_{\text{стоп}}, \omega_{\text{пп}}, \omega_{\text{п1}}, \omega_{\text{п2}}. \quad (37)$$

Максимальная скорость оригинала транспортного средства в соответствии с заданием 10 км/ч.

У оригинала, при диаметре колеса 14 дюймов (355,6 мм), за один оборот колесо проходит 1,1 м. Следовательно, за один час, при скорости 10 км/ч, колесо выполнит 9091 оборот. Угловая скорость ( $\omega_{\text{п2}}$ ) равна 2,5 об/сек.

При разработке принято: П1 – 5 км/ч,  $\omega_{\text{п1}} = 1,25$  об/сек;

ПП – 0,5 км/ч,  $\omega_{\text{п1}} = 0,125$  об/сек;

ПЗ – 0,5 км/ч,  $\omega_{\text{пз}} = 0,125$  об/сек;

З – 2,5 км/ч,  $\omega_3 = 0,625$  об/сек.

Кнопка «Ход» применяется для включения мотор-колес. Кнопка подключена к выходу 3,6 В делителя напряжения. В разомкнутом состоянии, этот сигнал передается на ЦК, что соответствует сигналу «Стоп». При нажатии кнопки «Ход» на ЦК подается напряжение, заданное переключателем «Скорость».

### 3.6 Управление направлением движения

На схеме (рисунок 43) показан потенциометр джойстика «Направление». Среднее положение потенциометра соответствует величине угла направления  $\alpha_n = 0$ . Следовательно, ЦК распознает напряжение равное 3В как  $\alpha_n = 0$ . При поворотах джойстика величина напряжения, меньшая, чем 3В, соответствует  $\alpha_n < 0$ . При поворотах джойстика величина напряжения, большая, чем 3В, соответствует  $\alpha_n > 0$ .

Кнопка «Сирена» непосредственно включает звуковой сигнализатор.

Блок программного управления движением содержит переключатель номера траектории, работающий аналогично переключателю «Скорость», с

использованием резисторного делителя напряжения. Переключатель формирует сигнал  $U_{\text{нтр}}$  (номер траектории).

Переключатель SA «Запись» включает регистрацию сигналов «Скорость» и «Направление» с шагом  $\Delta t = 0,1$  сек в файле, соответствующем номеру траектории.

Переключатель SA «Исполнение» отключает ПДУ-П и ПДУ-Р, подает данные из файла траектории для исполнения. Мотор-колеса выполняют записанную ранее траекторию движения. Исполнение траектории должно быть начато в положении платформы, в котором была начата запись.

Кнопки SA «Сброс» дублированы на случай защиты от случайного сброса записанной траектории. Кнопки применяются для удаления записанной траектории, когда запись или считывание не производится.

Если включены оба переключателя «Запись» и «Исполнение», ЦК блокирует оба режима.

Выводы по главе 3:

- создана математическая модель управления бортовым поворотом транспортной ходовой платформой, которая подразумевает зависимость скоростей левого и правого борта от наклона джойстика пульта дистанционного управления;
- на основе математической модели составлена блок-схема, алгоритм программы управления платформой для центрального контроллера;
- разработана структура пульта дистанционного управления для управления скоростью и направлением движения транспортного средства.

## Заключение

В данной работе основное внимание было уделено вопросам, связанным с обеспечением ориентации автономной платформы транспортного средства в пределах мини-полигона.

В результате были получены следующие результаты:

1. Рассмотрены различные варианты позиционирования объектов с использованием глобальных систем позиционирования;
2. Проведен обзор способов определения расположения объекта с использованием различных датчиков;
3. Рассмотрена система распознавания положения объекта, использующая техническое зрение;
4. Сделан вывод о целесообразности использования меток, позволяющих идентифицировать положение объекта в пространстве;
5. Разработан программно-аппаратный комплекс мини-полигона;
6. Детально рассмотрена структурная и принципиальная схемы транспортного средства;
7. Разработано программное обеспечение, позволяющее управлять транспортным средством в пределах мини-полигона;
8. Разработанный комплекс позволяет обеспечить маршрутизацию и рациональное логистическое обслуживание транспортной системы мини-полигона;
9. Создана математическая модель управления бортовым поворотом транспортной ходовой платформой, которая подразумевает зависимость скоростей левого и правого борта от наклона джойстика пульта дистанционного управления;
10. На основе математической модели составлена блок-схема, алгоритм программы управления платформой для центрального контроллера;
11. Разработана структура пульта дистанционного управления для управления скоростью и направлением движения транспортного средства.

## Список используемой литературы и используемых источников

1. Блум Джереми. Изучаем Arduino: инструменты и методы технического волшебства: Пер. с англ. – СПб.: БВХ-Петербург, 2015. – 336 с.: ил. ISBN 978-5-9775-3585-4
2. Глибин Е.С. Программирование электронных устройств: электронное учебное пособие / Е.С. Глибин, А.В. Прядилов, - Тольятти: Издво ТГУ, 2014
3. Густав Олссон, Джангуидо Пиани. Цифровые системы автоматизации и управления Издательство: «Невский Диалект», 2001.
4. Достойные аналоги Ардуино: Teensy, Netduino и другие [Электронный ресурс].URL: <https://arduinoplus.ru/vse-analogi-arduino>.
5. Дубровский, И.Л. Микропроцессорное управление электроприводами промышленных роботов: учебное пособие/ И.Л. Дубровский, А.П. Дамбраускас, А.А. Рыбин / Красноярский гос. технологический ун-т. - Красноярск, 2006.
6. И.В Мирошник, А.Н. Шалаев. Управление траекторным движением автономных роботов. Портал «Информационно-коммуникационные технологии в образовании»: [http://www.ict.edu.ru/ft/001784/sbornik6\\_4\\_5\\_6.pdf](http://www.ict.edu.ru/ft/001784/sbornik6_4_5_6.pdf).
7. Изменение адреса дисплея по шине I2C [Электронный ресурс]. URL: <https://flprog.ru/uchebnyj-centr/articles/podkljuchenie-displeev-i-indikatorov/izmenenie-adresa-displeja-po-shine-i2c/> (дата обращения: 16.05.2021)
8. Ким Д.П. Теория автоматического управления. Том 1. Линейные системы. Издательство: ФИЗМАТЛИТ, 2007.
9. Копша О.Ю. Пространственная ориентация робота / О.Ю. Копша, Г.Н. Абрамов «Молодежь. Наука. Общество» : Всероссийская студенческая научно-практическая междисциплинарная конференция (Тольятти, 5 декабря

2019 года) : электронный сборник студенческих работ . – Тольятти : Изд-во ТГУ, 2019., С.600-602

10. Крутиков, В. Н. Методы оптимизации : учебное пособие / В. Н. Крутиков, В. В. Мишечкин. — 2-е изд., доп и перераб. – Кемерово : КемГУ, 2019. – 106 с. – ISBN 978-5-8353-2437-8. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/135233>.

11. Подключение шилда LCD Keypad Shield 1602 к Arduino [Электронный ресурс]. URL: <https://arduinomaster.ru/platy-arduino/arduino-lcd-keypad-shield/> (дата обращения: 22.05.2021)

12. Полякова О.М., Прентсель А.А., Ройтбург Ю.С. Архитектура аппаратно-программного комплекса для профессиональной ориентации и технического творчества молодежи по направлению национальной технологической инициативы «Автонет». Хуманитарни Балкански изследвания. 2017. № 1. С. 23-26

13. Программирование на языке С для AVR и PIC микроконтроллеров./ Сост. Ю.А. Шпак – К.: «МК-Пресс», 2006. – 400 с., ил. ISBN 966-8806-16-6

14. Ричард К. Дорф, Роберт Х. Бишоп Современные системы управления. Издательство: «Лаборатория Базовых Знаний», 2004.

15. Селянкин, В. В. Компьютерное зрение. Анализ и обработка изображений : учебник для вузов / В. В. Селянкин. – 2-е изд., стер. – Санкт-Петербург : Лань, 2021. – 152 с. – ISBN 978-5-8114-8259-7. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/173806>.

16. Среда разработки Arduino IDE [Электронный ресурс]. URL: <https://arduinoplus.ru/> (дата обращения: 11.04.2021)

17. Тихомиров, Э.Л. Микропроцессорное управление электроприводами станков с ЧПУ / Э.Л. Тихомиров, В.В. Васильев / - М.: Машиностроение, 2005.

18. Токарев Д.Г. Автоматизированное устройство распознавания и сортировки объектов / О.Ю. Копша, Д.Г. Токарев, Ю.А. Францева Современные научные исследования и разработки: Международный электронный научно-практический журнал. – Москва: Олимп, 2018. – №1(18). – С.203-206

19. Токарев Д.Г. Вопросы оперативно-календарного планирования транспортной логистики в интеллектуальных системах поддержки принятия решений / О.Ю. Копша, Д.Г. Токарев «Информационные технологии в моделировании и управлении: подходы, методы, решения» III Всероссийская научная конференция с международным участием: электронный сборник статей . – Тольятти : Изд-во ТГУ, 2020., С.118-125

20. Токарев Д.Г. Бортовой комплекс технических средств дистанционно-управляемой ходовой платформы / «Молодежь. Наука. Общество» : Всероссийская студенческая научно-практическая междисциплинарная конференция (Тольятти, 5 декабря 2018 года) : электронный сборник студенческих работ . – Тольятти : Изд-во ТГУ, 2018., С.670-673

21. Токарев Д.Г. Основные аспекты автоматизированной обработки информации в системе погрешностей положения подвижного объекта / О.Ю. Копша, Д.Г. Токарев «Информационные технологии в моделировании и управлении: подходы, методы, решения» IV Всероссийская научная конференция с международным участием: электронный сборник статей . – Тольятти : Изд-во ТГУ, 2021., С.30-34

22. Токарев Д.Г. Способ ориентации мобильного робота / А.В. Маслова, Д.Г. Токарев, Р.А. Туров Теория и практика научных исследований. Сборник статей по материалам международной научно-практической конференции – 2017. С. 24-28

23. Токарев Д.Г. Повышение эффективности перемещений мобильного робота / А.В. Маслова, Д.Г. Токарев, Р.А. Туров Фундаментальные и прикладные научные исследования. Сборник статей по



материалам международной научно-практической конференции. – Уфа: МЦИИ ОМЕГА САЙНС, 2017. – С. 37-39

24. Хорн Б.К.П. Зрение роботов. (Robot vision) / Бертольд Клаус Поль Хорн (Berthold Klaus Paul Horn). Перевод с английского И.М. Бродской, Е.Ю. Зуевой, А.Ю. Каргашина, под редакцией Е.И. Кугушева, Ю.А. Садова. Научное издание. Художник Г.М. Чеховской. Москва: Издательство «Мир»: Редакция литературы по информатике и робототехнике, 1989

25. Шалыгин, М. Г. Автоматизация измерений, контроля и испытаний / М. Г. Шалыгин, Я. А. Вавилин. – Санкт-Петербург : Лань, 2019. – 172 с. – ISBN 978-5-8114-3531-9. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/115498>.

26. Устройство и принцип работы Wi-Fi сети (преимущества и недостатки) [Электронный ресурс]. URL: <https://hobbyits.com/ustrojstvo-i-princip-raboty-wi-fi-seti/> (дата обращения: 11.05.2021)

27. Фоторезистор: устройство, принцип работы, характеристики [Электронный ресурс]. URL: <https://samelectrik.ru/chto-takoe-fotorezistory.html> (дата обращения: 11.04.2021)

28. Arduino IDE [Электронный ресурс]: «Download the Arduino IDE» URL: <https://www.arduino.cc/en/main/software> (дата обращения: 20.02.2021).

29. Bluetooth модуль HC-05 [Электронный ресурс]. URL: <https://3d-diy.ru/wiki/arduino-moduli/bluetooth-modul-hc-05/> (дата обращения: 22.05.2021)

30. Connecting 2 Arduinos by Bluetooth using a HC-05 and a HC-06: Pair, Bind, and Link [Электронный ресурс]. URL: <http://www.martyncurrey.com/connecting-2-arduinosaurs-by-bluetooth-using-a-hc-05-and-a-hc-06-pair-bind-and-link/> (дата обращения: 10.04.2021)

31. Explain Algorithm and Flowchart with Examples [Электронный ресурс]. URL: <https://www.edrawsoft.com/explain-algorithm-flowchart.html> (дата обращения: 17.05.2021)

32. GitHub - ssilver2007/LCD\_1602\_RUS: Arduino LCD 16x02 display RUSSIAN with NO CYRILLIC symbols set [Электронный ресурс]. URL: [https://github.com/ssilver2007/LCD\\_1602\\_RUS](https://github.com/ssilver2007/LCD_1602_RUS) (дата обращения: 15.05.2021)

33. GNL-5013GT, Светодиод зеленый 20° d=5мм 100-300мКд 565нМ (Green) [Электронный ресурс]. URL: <https://www.chipdip.ru/product/gnl-5013gt>

34. GPS System : Working, Types, Trackers & Its Applications [Электронный ресурс]. URL: <https://www.elprocus.com/how-gps-system-works/#:~:text=The%20working%2Foperation%20of%20the,the%20receiver%20on%20the%20earth.>

35. I2C расширитель портов PCF8574 / Деталька / Сообщество EasyElectronics.ru [Электронный ресурс]. URL: <http://we.easyelectronics.ru/part/i2c-rasshiritel-portov-pcf8574.html>

36. SS441A Datasheet, PDF - <https://www.alldatasheet.com/> [Электронный ресурс]. URL: [https://www.alldatasheet.com/view.jsp?Searchword=Ss441a%20datasheet&gclid=CjwKCAjwhYOFBhBkEiwASF3KGYsNbmKEg5kfANlAWv-MdPPNRn7j0h3WJ371AsCUZC-4c\\_t1YAnrXBoCvP4QAvD\\_BwE](https://www.alldatasheet.com/view.jsp?Searchword=Ss441a%20datasheet&gclid=CjwKCAjwhYOFBhBkEiwASF3KGYsNbmKEg5kfANlAWv-MdPPNRn7j0h3WJ371AsCUZC-4c_t1YAnrXBoCvP4QAvD_BwE)

37. SS441A датчик Холла однополярный цифровой 115G/20G sip3 [Электронный ресурс]. URL: [https://www.voltmaster-samara.ru/catalog/datchiki\\_magnitnogo\\_polya/ss441a\\_datchik\\_kholla\\_odnopol\\_tsi\\_frovoy\\_115g\\_20g\\_sip3/](https://www.voltmaster-samara.ru/catalog/datchiki_magnitnogo_polya/ss441a_datchik_kholla_odnopol_tsi_frovoy_115g_20g_sip3/)

Приложение А

Перечень элементов мини-полигона

Поз	Наименование	Кол	Примечание				
A1	Блок питания B12-1000	1					
A2	Управляющий модуль	1					
A3	Макет мини-полигона	1					
<i>Микросхемы</i>							
DD1, DD4	Модуль расширения PCF8574	2					
DD2	Микроконтроллер Arduino UNO R3	1					
DD3	LCD Keypad Shield	1					
U1	Bluetooth-модуль HC-05	1					
<i>Резисторы</i>							
R1-R6, R8-R15	МЛТ-0,25-10 кОм±10%	14					
R7	МЛТ-0,25-4,7 кОм±10%	1					
<i>Светодиоды</i>							
VD1-VD6	А15013А	6					
SA1	Кнопка тактовая 1571627-1 (FSM103A)	1					
<b>Мини-полигон</b> Перечень элементов							
Изм	Лист	№ докум	Подп	Дата	Лист	Лист	Листов
Разраб							
Проб							
И контр							
Утв							

Рисунок А.1 – Перечень элементов мини-полигона