|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | |
| МИНОБРНАУКИ РОССИИ | | | |
| Федеральное государственное бюджетное образовательное учреждениевысшего образования"Московский технологический университет"МИРЭА | | | |
| Институт кибернетики | | |  |
| кафедра проблем управления | | |  |

|  |  |
| --- | --- |
| **КУРСОВОЙ РАБОТЫ** | |
| **по дисциплине** | |
| **«**Основы программирования систем управления**»** | |
| **Тема курсовой работы «**Информационная поддержка системы управления мобильной роботизированной платформой на базе аппарата нечеткой логики**»** | |
| Студент группы КРБО-03-16 | *Маршев Е.Д..* |
| Руководитель курсовой работы | *асс. Слепынина Е.А.* |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| Работа представлена к защите | «\_\_»\_\_\_\_\_\_\_201\_\_\_ г. |  |
|  |  |  |
| «Допущен к защите» | «\_\_»\_\_\_\_\_\_\_201\_\_\_ г. |  |

Москва, 2017

Содержание

[Введение 3](#_Toc484058357)

[1.Обзор дистанционно управляемых роботов 5](#_Toc484058358)

[2. Система управления мобильным роботом 8](#_Toc484058359)

[2.1 Аппаратные средства систем управления 8](#_Toc484058360)

[2.2 Программные средства систем управления 15](#_Toc484058361)

[2.3 Использование аппарата нечеткой логики для передвижения робота 21](#_Toc484058362)

[2.Тестирование и экспериментальные исследования 23](#_Toc484058363)

[Заключение. 25](#_Toc484058364)

[Список используемых источников 26](#_Toc484058365)

# Введение

Создание робототехники привело к тому, что люди смогли не только автоматизировать большую часть технологических процессов, но и улучшить качество выполнения поставленной задачи. Помимо исполнительных механизмов для физических усилий, роботов стали оснащать датчиками, позволяющие им воспринимать данные об окружающей их среде. Такие датчики особенно важны для мобильных роботов, целью которых является выполнение каких-либо действий через дистанционное управление.

Основные компетенции робототехники связаны с управлением приводами исполнительных устройств (манипуляционной системой) на основании информации, получаемой от сенсорной системы. Для мобильной робототехники к этим компетенциям добавляются новые, связанные с тем, что робот становится подвижным объектом: ориентация (определение собственного положения), навигация (методы управления объектом, маршрутизация, выбор оптимального пути следования объекта в пространстве), позиционирование, избегание препятствий и т.п.

Кроме того, в последнее время к ним добавляются компетенции, связанные с интерактивным взаимодействием с человеком, будь то оператор, клиент или "объект" в зоне действия робота. Существуют случаи, когда робот находится на значительно удаленных расстояниях от управляющей станции и не может управляться в режиме реального времени из-за значительного запаздывания командных сигналов и сигналов обратной связи от аппарата. Возникает задержка, вследствие чего необходимо задать протокол, который бы корректировал действия аппаратной части, функционируя автономно по заложенным программам, получая команды лишь время от времени.

Так как мобильная роботизированная платформа использует колеса в качестве передвижения, то будет целесообразно создать набор нестрогих правил, регулирующий действия робота, на базе аппарата нечеткой логики, вступающих в действие при определенных условиях, особенно в тех ситуациях, когда наперёд неизвестны все возможные ситуации[1].

Целью курсовой работы является создание робота, который будет управляться по сети и где будет использована архитектура *«*Клиент*—*сервер» для обеспечения взаимодействия с оператором, а также создание протоколов действий, при которых робот будет ориентироваться, исходя из данных об окружающей его среде, и сможет действовать, согласно правилам, в которых будут прописаны несколько вариантов действий. Такого робота можно будет применять в различных сферах, начиная от ситуаций опасных для жизнедеятельности человека и заканчивая планетоходами.

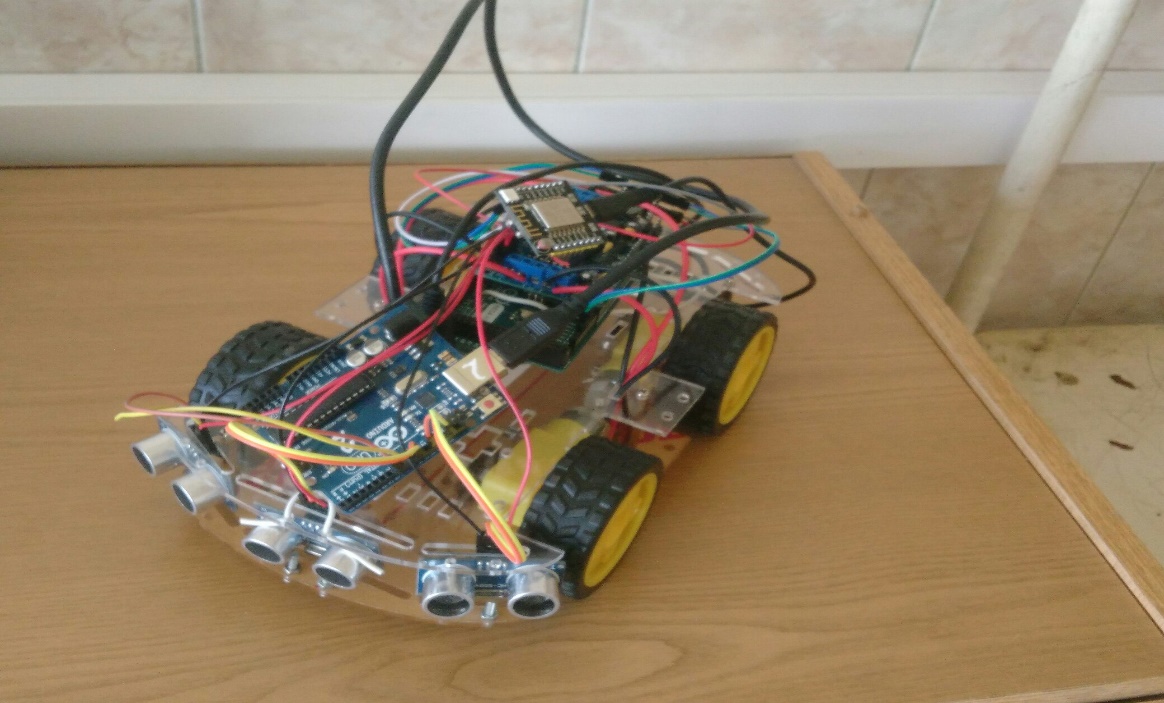


Рис. 1. Мобильная роботизированная платформа

# Обзор дистанционно управляемых роботов

В настоящее время интерес к исследованию и практическому применению дистанционно управляемых роботов растет. Развитие таких технологий дает возможность применения роботов в широком диапазоне задач, в которых требуется удаленное управление. Такие роботы могут применяться в местах экологических катастроф, при работе с радиоактивными веществами, а также в условиях, которых человеку будет трудно или невозможно работать самому.

Дистанционное управление (ДУ) заключается в передаче управляющего воздействия (сигнала) от оператора к объекту управления, которое будет находиться на расстоянии. Есть несколько вариантов использования ДУ. В первую очередь они различаются по типу канала связи[1]. Применительно к данной курсовой работе будет использовано радиоуправление - метод, при котором управляющее воздействие и обратная связь осуществляются с помощью радиоволн.

Одно из самых распространенных на данный момент направлений, при таком подходе, реализовано в планетоходах – аппаратах, которые предназначаются для передвижения по поверхности другой планеты в исследовательских целях.

Кроме того, разработки подобных есть технологии вносят свой вклад в развитие технологий систем дистанционного управления подвижными роботами. Преимущества перед орбитальными аппаратами:

* способность непосредственно исследовать объекты и выполнять эксперименты.

Однако у такого рода роботов есть требования и особенности. Учитывая, что они работают в условиях, которые отличаются от земных, то и устройство должно соответствовать требованиям, чтобы сохранять работоспособность в разных ситуациях. Одна из проблем, при работе с планетоходом – это то, что они находятся на огромных расстояниях от Земли, что в свою очередь вызывает задержку при управлении. Система не сможет управляться в режиме реального времени из-за значительного запаздывания сигналов и сигналов обратной связи от аппарата[1].

В свою очередь, для адекватной работы необходимо задать протокол действий, который будет вступать в силу при определенных условиях, чтобы робот мог функционировать и выполнять заданную ему задачу автономно, получая сигналы от оператора время от времени.

Примеры роботов, управляемых дистанционно:

**Digital Vanguard ROV (DV)** – многоцелевой робот, способный выполнять команды, получаемые дистанционно. Аппарат снабжен манипулятором, телескопической роборукой, а также набором, который необходим при работе с взрывоопасными устройствами. Комплектуется x500 интегрированным порталом управления, обеспечивающим дистанционное управления всеми функциями робота, включая позиционирование руки и захвата, а также камерами. Портал ведет запись аудио, видео в формате HD, позволяет делать снимки, а также выдает на экран данные от набора сенсоров. Графический интерфейс может быть настроен на один из нескольких языков. Работа с пультом требует минимального обучения[1].

**Марсоход Curiosity**– планетоход для передвижению по поверхности планеты Марс. Представляет собой автономную химическую лабораторию в несколько раз больше и тяжелее предыдущих марсоходов. На марсоходе установлено два одинаковых бортовых компьютера под названием «Rover Compute Element» (RCE) под управлением процессора [RAD750](https://ru.wikipedia.org/wiki/RAD750) с частотой 200 МГц; они содержат радиационностойкую память. Каждый компьютер включает в себя 256кБ [EEPROM](https://ru.wikipedia.org/wiki/EEPROM), 256 МБ DRAM, и 2 ГБ флэш-памяти. Используется многозадачная [ОСРВ](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%80%D0%B5%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B3%D0%BE_%D0%B2%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%B8) - [VxWorks](https://ru.wikipedia.org/wiki/VxWorks). Компьютер постоянно следит за марсоходом: например, сам может повысить или понизить температуру в те моменты, когда это необходимо. Он даёт команды на фотографирование, вождение, отправку отчёта о техническом состоянии инструментов. Приказы марсоходу передаются операторами с Земли. Имеет две системы связи: В первую входят передатчик и приёмник [X-диапазона](https://ru.wikipedia.org/wiki/X-%D0%B4%D0%B8%D0%B0%D0%BF%D0%B0%D0%B7%D0%BE%D0%BD), с помощью которых робот связывается напрямую с Землёй, со скоростью до 32 кбит/с. Вторая работает в диапазоне дециметровых волн ([ДМВ](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%86%D0%B8%D0%BC%D0%B5%D1%82%D1%80%D0%BE%D0%B2%D1%8B%D0%B5_%D0%B2%D0%BE%D0%BB%D0%BD%D1%8B))  и создана на базе [программно-определяемой радиосистемы](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE-%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D1%8F%D0%B5%D0%BC%D0%B0%D1%8F_%D1%80%D0%B0%D0%B4%D0%B8%D0%BE%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) [Electra](https://ru.wikipedia.org/w/index.php?title=Electra&redirect=no)Lite, разработанной специально для космических аппаратов. ДМВ-радио используется для связи с искусственными спутниками Марса. Несмотря на то, что у «Curiosity» имеется возможность прямой связи с Землёй, большая часть данных будет ретранслироваться орбитальными аппаратами, обеспечивающими большую пропускную способность за счёт большего диаметра антенн и более мощных передатчиков. Скорости передачи данных между роботом и каждым орбитальным аппаратом могут быть 2 Мбит/с («[Марсианский разведывательный спутник](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D1%81%D0%B8%D0%B0%D0%BD%D1%81%D0%BA%D0%B8%D0%B9_%D1%80%D0%B0%D0%B7%D0%B2%D0%B5%D0%B4%D1%8B%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%81%D0%BF%D1%83%D1%82%D0%BD%D0%B8%D0%BA)») и 256 кбит/с («[Марс Одиссей](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D1%81_%D0%9E%D0%B4%D0%B8%D1%81%D1%81%D0%B5%D0%B9)»), каждый спутник имеет возможность держать связь приблизительно 8 минут в день. Также у орбитальных аппаратов заметно больше временное окно, в котором имеется возможность связи с Землёй[1].

# 2. Система управления мобильным роботом

## **2.1 Аппаратные средства систем управления**

В рамках данной курсовой работы были использованы следующие компоненты конструкции робота:

1. Плата Arduino Mega – 1шт.;
2. Плата Arduino Uno – 1шт.;
3. Ультразвуковые дальномеры Ultrasonic HC-SR04 – 3шт.;
4. Монтажный комплект – 1шт.;
5. Плата Motor Shield – 1шт.;
6. Двигатели постоянного тока – 4шт.;
7. Wi-Fi модуль ESP8266 – 1шт.;
8. Преобразователь логических уровней LLC – 1шт.;
9. Датчики замера скорости – 2шт.;
10. Источник питания - 1шт.;

* Плата **Arduino Mega**.

**Arduino Mega** построена на микроконтроллере **ATmega1280.** Платформа содержит 54 цифровых входа/выходов (14 из которых могут использоваться как выходы ШИМ), 16 аналоговых входов,4 последовательных порта UART, кварцевый генератор 16 МГц, разъем USB, силовой разъем, разъем ICSP и кнопку перезагрузки. Для работы необходимо подключить платформу к компьютеру посредством кабеля USB или подать питание при помощи адаптера AC/DC, или аккумуляторной батареей. **Arduino Mega** совместима со всеми платами расширения, разработанными для платформ **Duemilanove** или **Diecimila**[2].

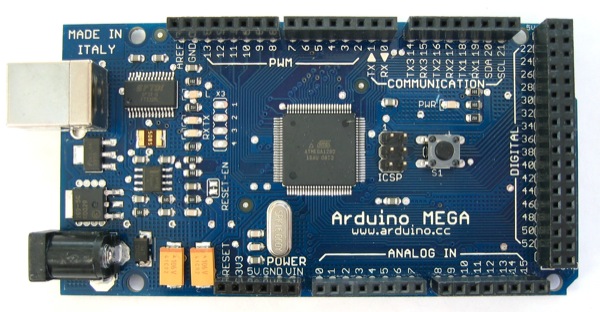


Рис. 2. Плата Arduino Mega

* Плата **Arduino Uno**.

**Arduino Uno** контроллер построен на **ATmega328**. Платформа имеет 14 цифровых вход/выходов (6 из которых могут использоваться как выходы ШИМ), 6 аналоговых входов, кварцевый генератор 16 МГц, разъем USB, силовой разъем, разъем ICSP и кнопку перезагрузки. Для работы необходимо подключить платформу к компьютеру посредством кабеля USB, либо подать питание при помощи адаптера AC/DC или батареи.

В отличие от всех предыдущих плат, использовавших FTDI USB микроконтроллер для связи по USB, новый **Ардуино Uno** использует микроконтроллер **ATmega8U2[2].**



Рис. 3. плата Arduino Uno.

* Ультразвуковые дальномеры **HC-SR04.**

Ультразвуковой дальномер определяет расстояние до объектов

точно так же, как это делают летучие мыши. Он генерирует звуковые импульсы на частоте 40 кГц и слушает эхо. По времени распространения звуковой волны туда и обратно можно однозначно определить расстояние до объекта. На рис. 4 показан внешний вид дальномера, а рис. 5 показан принцип действия датчика[3].



*Рис. 4. Ультразвуковой дальномер*

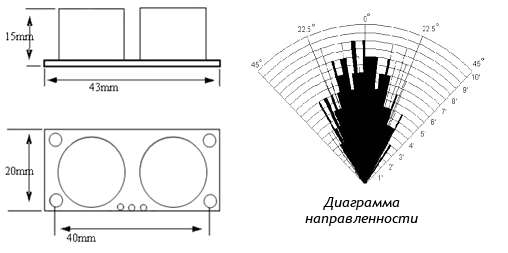


Рис. 5. Принцип действия датчика

* Плата **Motor Shield.**

Плата расширения Motor Shield изготовлена на основе микросхемы L293D, являющейся 4-ех Н-полномостовым драйвером, разработанным для управления индуктивными нагрузками, такими как реле, соленоиды, двигатели постоянного тока и шаговые двигатели. Она позволит вам управлять четырьмя двигателями постоянного тока с помощью платы Arduino, независимо регулируя скорость и направление каждого из них. Кроме всего прочего, вы также можете измерять ток, потребляемый каждым двигателем. Плата совместима с Arduino – платами. Реализована функция торможения и блокировки свободного вращения[4].

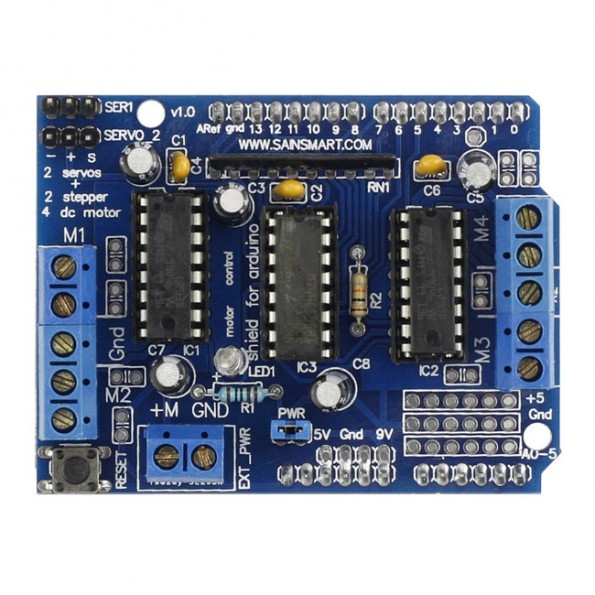


Рис. 6. Плата Motor shield

* Wi-Fi микроконтроллер **ESP8266**.

Управляющее устройство общается с ESP8266 через UART(Serial – порт).ESP8266 может работать как в роли [точки доступа](https://ru.wikipedia.org/wiki/%D0%A2%D0%BE%D1%87%D0%BA%D0%B0_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%B0) так и оконечной станции. При нормальной работе в локальной сети ESP8266 конфигурируется в режим оконечной станции. Для этого устройству необходимо задать [SSID](https://ru.wikipedia.org/w/index.php?title=SSID&action=edit&redlink=1) Wi-Fi сети и, в закрытых сетях, пароль доступа[5].



Рис. 7. Модуль ESP8266

* Logic Level Converter(LLC) – преобразователь логических уровней.

Преобразователь логических уровней представляет собой миниатюрное устройство, которое понижает 5В-сигналы до 3.3В, и повышает 3.3В-сигналы до 5В. Преобразователь также работает с устройствами, логический уровень которых составляет 2.8В и 1.8В. Одна такая плата способна преобразовывать напряжения между 4 выводами на одной стороне и 4 выводами на другой.

Устройство неприхотливо, оно может использоваться как с обычными последовательными интерфейсами UART, I2C, SPI, так и для преобразования любых цифровых сигналов. Не работает с аналоговыми сигналами[5].

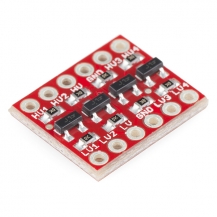


Рис. 8. Logic level converter(LLC)

На рис.9 показано, как каждый аппаратный элемент будет взаимодействовать с другим.

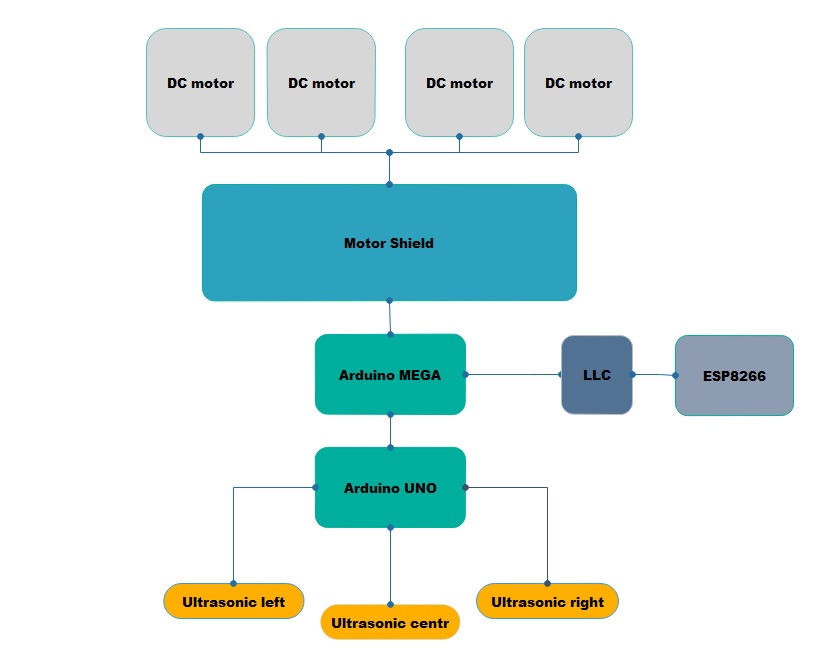


Рис. 9. Структурная схема

## **2.2 Программные средства систем управления**

Для разработки программного обеспечения робота, будет использована среда разработки Arduino IDE, состоящая из встроенного редактора программного кода, панели инструментов и меню. Для загрузки программ и связи вреда разработки подключается к аппаратной части Arduino. Программа, написанная в среде Arduino, называется скетч. Он пишется в текстовом редакторе. Кнопки панели инструментов позволяют проверить и записать программу, создать, открыть и сохранить скетч, открыть мониторинг последовательной шины. При загрузке скетча используется Загрузчик – небольшая программа, загружаемая в микроконтроллер на плате. Она позволяет загружать программный код без использования дополнительных аппаратных средств.

Вся система состоит из компонентов, которые будут взаимодействовать на аппаратном уровне. Так как управление будет по сети, то необходимо разработать программный код, который будет связывать клиента, то есть персональный компьютер, и сервер, в данном случае ESP8266. Взаимодействие будет проходить по протоколу UDP. Графический интерфейс будет составлен так, что при нажатии определенной клавиши будет отправлено сообщение, принимаемое на ESP8266 и исходя из того, что будет в сообщении, распознавать данные и реализовывать в соответствии с кодом.

На рис. 10 показан внешний вид графического интерфейса. При нажатии на кнопку будет происходить передача команды на Wi-Fi модуль по сети, который при принятии сигнала будет передавать значение на плату Arduino Mega через преобразователь логических уровней (LLC), согласовывающий работу Arduino, рабочее напряжение которой 5В, и ESP8266, у которой рабочее напряжение 3.3В. Mega, при принятии сообщения, будет исполнять, в соответствии с действием, определяющем в зависимости от данных в сообщении. Обе платы соединены напрямую, через UART – последовательный порт, служащий для связи устройства Arduino с другими устройствами, и поддерживают последовательный интерфейс обмена данными. Блок-схемы работы клиент-приложения и ESP8266 представлены ниже на рис. 11 и рис. 12. Программный код представлен в приложении 1 и приложении 2.

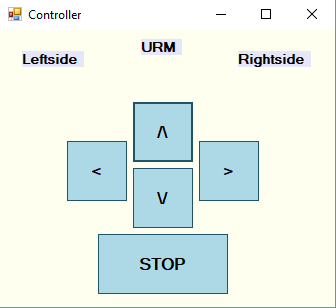


Рис. 10. Графический контроллер

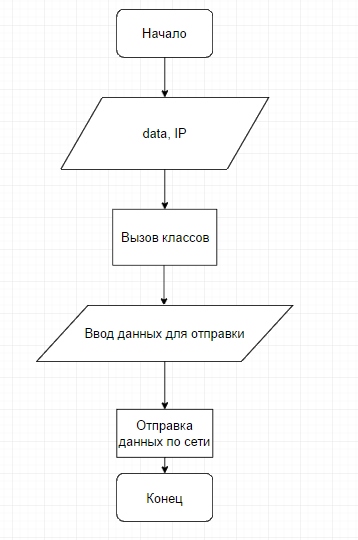


Рис. 11. Блок-схема работы клиент-приложения и сервера.

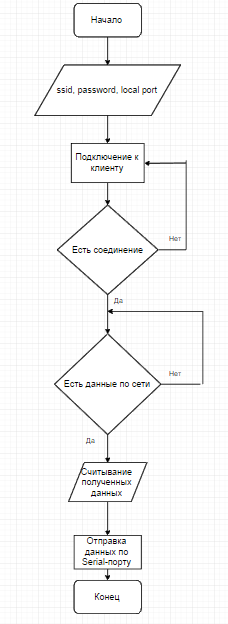


Рис. 12. Работа сервера на ESP8266

Arduino Mega и Motor shield также соединены напрямую по UART-порту. Передача на расстоянии необходима для задания направления роботу, то есть, при получении команды по сети, Mega задает параметры движения для моторов постоянного тока, взаимодействие с которыми происходит посредством платы расширения Motor shield. Управление скоростью происходит при помощи широтно-импульсной модуляции ([ШИМ](http://wiki.amperka.ru/%D0%BA%D0%BE%D0%BD%D1%81%D0%BF%D0%B5%D0%BA%D1%82-arduino:%D1%88%D0%B8%D0%BC)), которое на роботе задается при помощи датчиков замера скорости, в соответствии с определенными правилами его движения. Чтобы задать направление, используется определенный параметр, прописанный в библиотеке, необходимой для работы с Motor shield. Блок-схема работы представлена ниже на рис. 13. Программный код представлен в приложении 3.

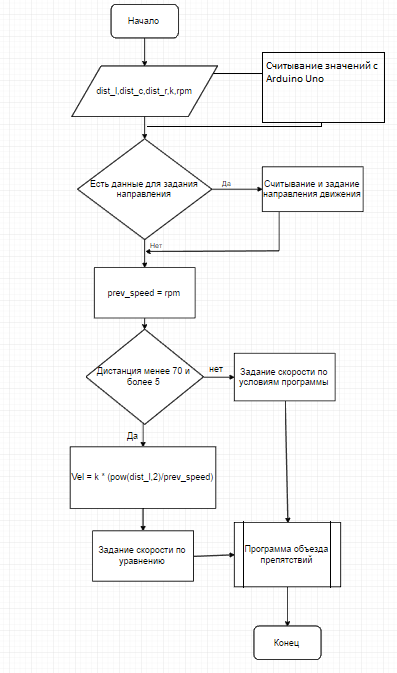


Рис. 13. Взаимодействие Arduino Mega, Arduino Uno и Motor shield

При согласовании работы ультразвуковых датчиков HC-SR04 для замера расстояния перед роботом и Wi-Fi модуля для удаленного управления возникла проблема. При получении значения по сети оно инвертировалось, вследствие чего робот не мог правильно функционировать, из-за чего пришлось использовать плату Arduino Uno. Как и с Motor shield, Arduino Uno поддерживала связь с Mega по UART-порту. Uno считывала значения с дальномера и, производя определенные расчеты для корректного вывода значения, показывала расстояние до объекта в зоне действия. Эти значения передавались в Mega по Serial-порту. Полученные значения используются для задания расстояния и в аппарате нечеткой логики, которые так же реализованы в программном коде из приложения 3.

## **2.3 Использование аппарата нечеткой логики для передвижения робота**

Аппарат нечеткой логики позволяет определить промежуточные значения для общепринятых оценок, таких как: да или нет, правда или ложь и т.д. Таким образом, нечеткая логика позволяет управлять в моменты, когда не должно быть четко заданных параметров, что является весьма полезным при удаленном управлении роботом.

Для дистанционного робота приводит к необходимости применения аппарата нечеткой логики с целью облегчить процесс управления и контроля работы системы со стороны человека. Применительно к роботизированной платформе, следует задать математическую функцию, которая будет регулировать скорость робота, в зависимости от расстояния. Такая функция позволит координировать движения робота, тем самым облегчит управляемость.

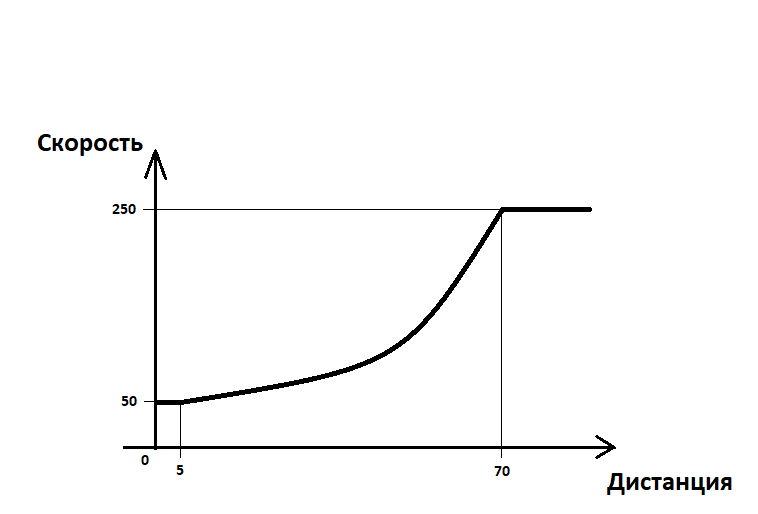
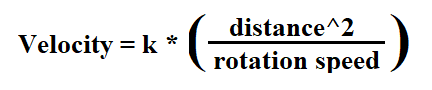


Рис. 14. График зависимости скорости от дистанции

На рис. 14 показана зависимость скорости от расстояния до преграды. Из графика видно, что скорость будет задана по формуле:



В формуле: Velocity – скорость ШИМ’a, то есть, скорость движения робота; k – коэффициент, необходимый для согласования скорости, получаемой с дальномера (rotation speed) и скорости, которую нужно задать для двигателей постоянного тока; distance – расстояние, получаемое с дальномера.

В итоге, скорость будет регулироваться по правилам нечеткой логики, что будет удобнее при управлении, а также при препятствии или ситуации, когда робот заехал в тупик. В таком случае, будет срабатывать протокол для объезда препятствия.

# Тестирование и экспериментальные исследования

Для проверки работоспособности программного обеспечения были проведены экспериментальные исследования, которые заключаются в том, что робот должен показать полную работоспособность и действовать по правилам, заложенным в его программе. Один из таких экспериментов будет описан далее.

Роботу были выставлены препятствия. При приближении к ограждению, на центральном дальномере срабатывает код, который использует формулу для расчета скорости робота, тем самым замедляя его движения. При приближении к расстоянию 5 см робот едет на минимальной скорости и включается алгоритм объезда препятствия. Он выбирает с какой стороны ему будет выгоднее проехать и потому поворачивается налево до тех пор, пока не сможет ехать без помех и продолжает движение до следующего препятствия. На рис. 15 показано, как робот начинал свое движение и препятствия, которые нужно объехать.



Рис. 15. Тест на проверку работоспособности

В итоге, после успешного обхода препятствий, робот продолжал движение и при принятии по сети сигнала “Стоп” от оператора он останавливал свое движение.

Несмотря на успешное пройденное испытание, робот не функционировал достаточно хорошо. Были выявлены ошибки, связанные с погрешностью аппаратной части, которые влияли на его работу, искажая данные, необходимые для правильного функционирования. При больших дистанциях дальномер мог показывать неправильные значения, которые, в свою очередь, влекли за собой неправильные расчеты для движения робота и, следовательно, задание скорости для двигателей постоянного тока отличалось от истинных значений. Для улучшения качества работы необходимо использование более стабильных дальномеров. Однако для работы на небольших расстояниях до объекта, датчик работал корректно и алгоритм объезда препятствий сработал так, как и предполагалось. На рис. 16 показана траектория, по которой робот двигался. Красными кругами помечено место, где робот использовал протоколы действия, заданные для объезда препятствия.

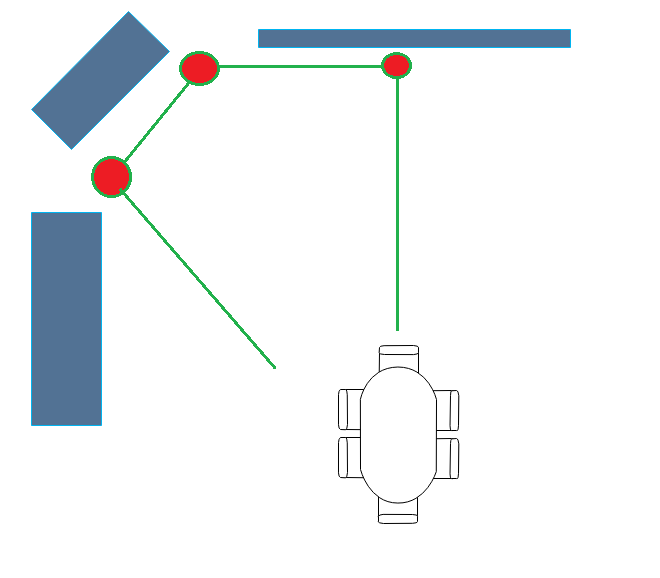


Рис. 16. Траектория объезда препятствий

# 

# Заключение.

В ходе курсовой работы была разработана концепция робота с удаленным управлением, использующего аппарат нечеткой логики. Применение нечеткой логики для управления позволяет сделать существенное продвижение в решении задач удаленного управления роботами. Задача удаленного управления имеет множество проблем, но при удачно выбранном подходе, возможность создавать мобильных роботов реализует возможность добычи полезных ископаемых с других планет, а также позволит действовать оператору на расстоянии без особых трудностей, а при создании достаточно удобного интерфейса, оператору не потребуется иметь глубокие познания в технике, чтобы выполнить свою задачу. Создание систем, управляемых на больших расстояниях, может облегчить решение многих задач, а также открыть новые возможности для робототехники и науки в целом.

Рассматриваемая концепция построения системы управления роботами будет способствовать преодолению перечисленных препятствия и упрощению дистанционного управления, что в свою очередь существенно расширит область применения таких роботов.

# Список используемых источников

1. [Википедия](http://www.wikipedia.ru) [Сайт] : Нечеткая логика. URL: <https://ru.wikipedia.org/wiki/Нечеткая_логика> (дата обращения: 04.05.2017).
2. Википедия [Сайт] : Дистанционное управление. URL: <https://ru.wikipedia.org/wiki/Дистанционное_управление> (дата обращения: 04.05.2017).
3. Википедия [Сайт] : Марсианская научная лаборатория. URL: <https://ru.wikipedia.org/wiki/Марсианская_научная_лаборатория> (дата обращения: 04.06.2017).
4. Мобильный форум [Электронный ресурс] : Digital vanguard. URL: www.mforum.ru/news/article/113718.htm (дата обращения: 04.06.2017).
5. Д. Блум. Изучаем Arduino. Инструменты и методы технического волшебства. – Санкт-Петербург: БХВ – Петербург, 2016. – 336 с.
6. Т. Карвинен, К. Карвинен, В. Валтокари. Делаем сенсоры. Проекты сенсорных устройств на базе Arduino и Raspberry Pi. – Москва: Вильямс, 2016. – 448 с.
7. Just for fun [сайт] : Arduino. Не простой старт с ESP8266. URL: http://tim4dev.com/arduino-esp8266-true-start/ (дата обращения : 04.06.2017)
8. С. Хайкин. Нейронные сети. Полный курс. – Москва: Вильямс, 2016. – 1026 с.
9. У. Соммер - Программирование микроконтроллерных плат Arduino/Freeduino – Санкт-Петербург: БХВ – Петербург, 2016. – 572с.