



Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ \_\_\_\_\_ Информатики и систем управления \_\_\_\_\_

КАФЕДРА \_\_\_\_\_ Программное обеспечение ЭВМ и информационные технологии \_\_\_\_\_

## ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

Студент (ФИО)	Группа	Подпись, дата	Оценка
Нахабин Сергей Михайлович	ИУ7-23		
Евтеев Артем Бахтиярович	ИУ7-23		
Колесова Анастасия Владиславовна	ИУ7-23		
Заколесник Максим Владиславович	ИУ7-23		

Ментор команды

\_\_\_\_\_  
*подпись, дата*

\_\_\_\_\_  
*фамилия, и.о.*

Ментор команды

\_\_\_\_\_  
*подпись, дата*

\_\_\_\_\_  
*фамилия, и.о.*

Руководитель практики

\_\_\_\_\_  
*подпись, дата*

\_\_\_\_\_  
*фамилия, и.о.*

2017 г.

## **Введение**

Идеей для нашей практики является робот - гид, который будет сопровождать посетителя до нужной ему цели в музеях, культурных центрах и в любых других сложных локациях, где очень просто потеряться. Система состоит из терминала, с помощью которого пользователь может выбрать маршрут или определенный экспонат, и робота, сопровождающего посетителя. Не исключено, что терминал может быть встроен непосредственно в робота.

## **Аналитическая часть**

На данный момент решением проблемы сопровождения занимаются люди. Наше решение позволяет автоматизировать процесс сопровождения. Также его возможно интегрировать с системой аудиогидов, для получения полноценной экскурсионной системы.

Плюсами данной системы являются:

1. экономия времени для нахождения нужного маршрута
2. пропадает необходимость содержать персонал
3. возможность пользоваться системой людям с ограниченными возможностями
4. возможность интеграции с другими системами, например, аудиогид

К минусам можно отнести:

1. возможная итоговая дороговизна всей системы
2. полное доверие к карте, непредвиденные несоответствия приведут к невозможности работы системы
3. сложности в точности позиционирования роботов

Нашей задачей является создание прототипа данного робота. Нам необходимо создать управляющую программу (УП) для него и реализовать систему связи компьютер - робот через графический интерфейс (GUI).

Для поиска кратчайшего расстояния из одной точки в другую (например, от одного экспоната к другому) мы используем волновой алгоритм, или алгоритм Ли, который принадлежит к алгоритмам, основанным на методах поиска в ширину.

Альтернативой является поиск по правилу правой стены, но путь найденный данным алгоритмом будет одним из самых длинных.

Оборудование для реализации нашего проекта выбиралось, по признакам доступности и цены.

В качестве микроконтроллера робота была выбрана Arduino mega 2560, так как на данной платформе присутствует большое количество контактов для подключения периферийных устройств, у данного решения относительно невысокая цена. Также для данной платформы имеется IDE и развитое комьюнити. Возможной альтернативой является контроллер STM, но для него нет IDE и комьюнити менее развито поэтому реализация окажется сложнее.

Для модулей связи мы выбрали модуль NRF24L01+ по следующим причинам:

1. большой радиус связи - до 2х километров при прямой видимости с использованием антенн.
2. высокая помехозащищенность и большое количество каналов.
3. низкая цена.

Можно было использовать модули Bluetooth, но эффективная дальность у него не более 10 метров, а также его гораздо сложнее подключить к мк ардуино.

Наиболее эффективной для робота является круглая платформа, такая как miniQ, но ввиду высокой цены данного решения, была выбрана платформа прямоугольной формы.

Движение робота организовано на основе драйвера l298n, этот драйвер является самым распространенным, простым и дешевым.

## **Конструкторская часть**

Маршрут будет создаваться по двум двум точкам. УП находит кратчайший путь. между ними, формирует программу движения и отправляет её роботу. Робот получает программу, начинает ей следовать и через определенные промежутки времени/количество операций отправляет УП сигналы о своем состоянии.

Допущения:

1. угол поворота робота кратен 90 градусам
2. поверхность, по которой движется робот, гладкая
3. карта, загружаемая программным обеспечением, абсолютно корректна
4. возможны непредвиденные в карте препятствия

## Требования

1. УП должна находить кратчайший путь
2. Система связи стабильна на малых и средних расстояниях
3. Робот должен достаточно точно следовать выбранному маршруту
4. Робот должен обнаруживать препятствия перед ним и подавать звуковой сигнал до тех пор, пока препятствие не будет убрано
5. Робот должен сообщать УП о проделанных действиях.
6. УП в реальном времени отображает состояние робота

## Состав работ:

1. Разработка УП
  - 1.1. разработка пользовательского интерфейса
  - 1.2. разработка алгоритма поиска пути
  - 1.3. разработка алгоритма формирования путевой программы
  - 1.4. разработка серверной части системы связи
2. Разработка робота
  - 2.1. разработка и сборка прототипа
  - 2.2. разработка алгоритма движения
  - 2.3. разработка алгоритма корректировки пути
  - 2.4. разработка клиентской части системы связи
3. Тестирование

Декомпозиция задачи представлена на рис. 1

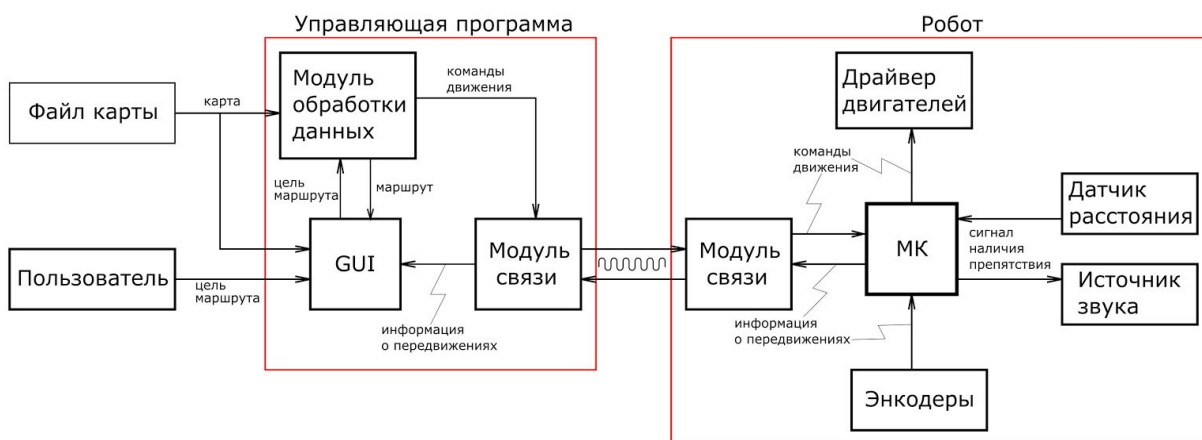


Рис.1 Декомпозиция задачи

Модуль обработки данных, получает карту в виде двумерного массива состоящего из 0 и 1, где 1 обозначает преграду или стену, а 0 говорит нам о свободном проходе. Затем по волновому алгоритму выполняется поиск кратчайшего пути, найденный путь передается в GUI. После этого маршрут преобразовывается в список последовательных команд робота. Список состоит из следующих элементов:  
команда 1 двигаться вперед  
команда 2 повернуть влево  
команда 3 повернуть вправо.  
Этот список передается с помощью модуля связи.  
Волновой алгоритм реализован следующим образом:

### **Распространение волны**

Пометить стартовую ячейку уровнем волны  $wave = 0$

**ПОКА** (финишная ячейка не помечена) **И** (есть возможность распространения волны)

#### **ЦИКЛ**

**ДЛЯ** каждой ячейки, помеченной числом  $wave$

    пометить все соседние свободные непомеченные ячейки числом  $wave + 1$

#### **КЦ**

$wave := wave + 1$

### **Восстановление пути**

**ЕСЛИ** финишная ячейка помечена

**ТО** перейти в финишную ячейку

**ПОКА** текущая ячейка — не стартовая

#### **ЦИКЛ**

    выбрать среди соседних ячейку, помеченную числом на 1 меньше числа

    текущей ячейке

    перейти в выбранную ячейку и добавить её к пути

**ВОЗВРАТ** матрица пути

**ИНАЧЕ**

**ВОЗВРАТ** путь не найден

Модуль связи выполняет отправку списка команд на ресивер по последовательному порту, а также прием сообщений о состоянии отправляемых роботом. Схема ресивера изображена на рис. 2

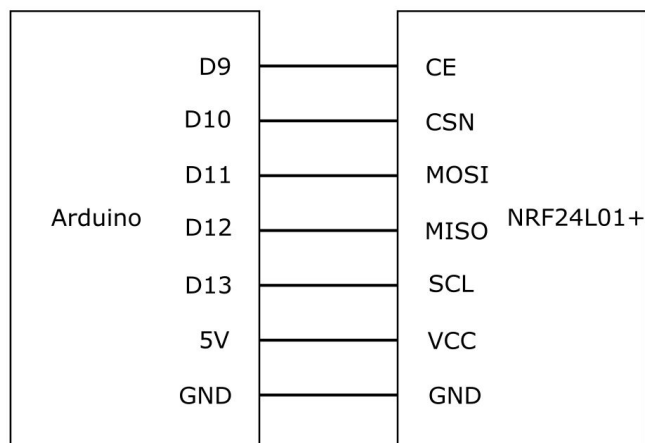


Рис. 2 Схема ресивера

В ходе реализации у нас возникло множество проблем, как конструкторских так и организаторских и проблем снабжения. Итогом стало то что удачно реализовать удалось лишь часть системы связи, отвечающую за передачу команд от УП к ресиверу. Передачу данных между ресивером и роботом реализовать не удалось.

Макет пользовательского интерфейса представлен на рис. 3

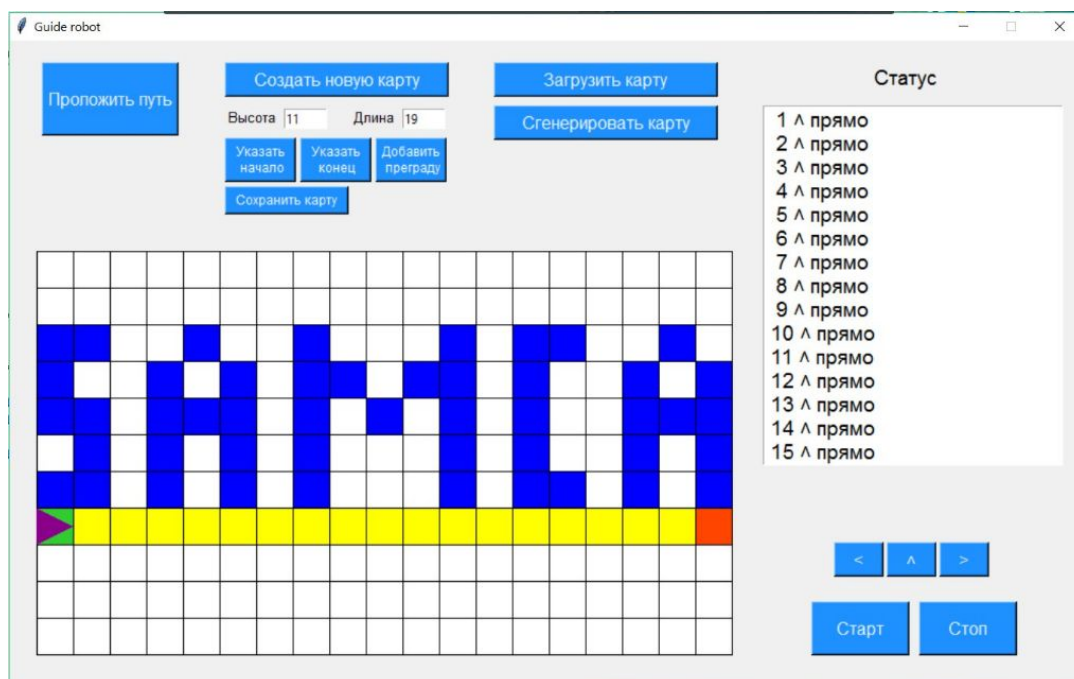


Рис. 3 Макет пользовательского интерфейса

Пользовательский интерфейс выполняет функции по:

1. загрузке карты из файла
2. отрисовке карты
3. созданию новой карты
4. отображению маршрута
5. отображению текущего положения робота
6. отображению списка выполненных команд

Интерфейс выбран в виде диалогового окна ввиду его простоты и удобства.

В течение всего использования программы пользователь видит все возможные этапы работы с программой, следовательно интерфейс понятен на интуитивном уровне и не требует дополнительных объяснений со стороны.

Также такой вид интерфейсов наиболее прост в реализации.

Схема работы изображена на рис. 4

Робот выполняет следующие функции

1. движение по прямой
2. поворот
3. Отсылка сообщений о текущем состоянии
4. определение наличия препятствий

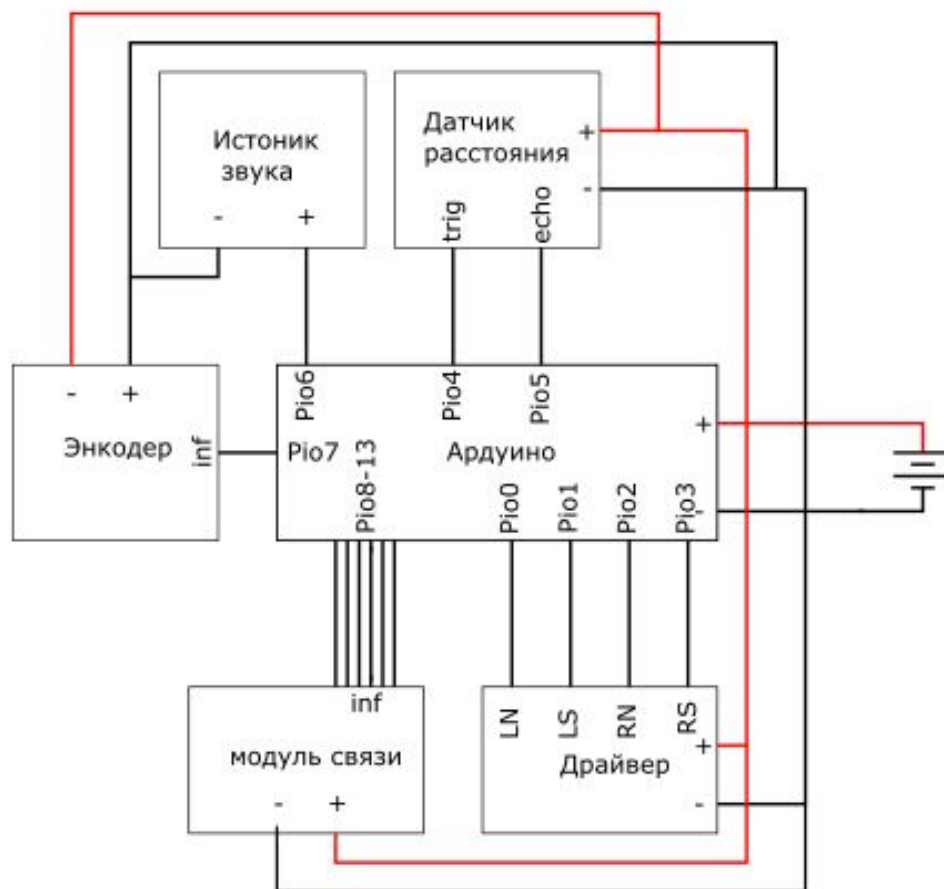


Рис.4 Схема подключения робота.

В ходе разработки робота у нас возникло несколько проблем

1. 4х батареек типа AA оказалось недостаточно для питания всех модулей робота. поэтому было принято решение использовать Li-Po аккумулятор.
2. несовершенство драйвера и колесной платформы привело к тому, что траектория движения робота оказалась непредсказуемой. от запуска к запуску смещения робота могли отличаться длительностью и направлением. Мы принимали попытки по корректировке мощности двигателей с помощью драйвера, а также отслеживания перемещений с помощью энкодеров. Особого успеха мы в этом не добились ввиду нехватки времени на эксперименты.

## Технологическая часть.

Для реализации различных частей проекта мы использовали следующие инструменты. Управляющая программа реализована на языке Python, данный язык был нами выбран потому, что:

1. Все члены команды знакомы с этим языком
2. До начала практики имелся опыт разработки графического интерфейса на этом языке.



Для реализации пользовательского интерфейса использовалась библиотека tkinter. так как членам команды уже доводилось с ней работать.

Для общения с ресивером используется библиотека pySerial, как единственное возможное решение для работы с последовательным портом.

Для реализации программ микроконтроллеров использовалась среда IDE arduino в качестве языка выступает диалект языка C. это решение является самым оптимальным ввиду своей простоты, и наличия большого количества примеров. Из специфических библиотек нам понадобилась библиотека RF24, предназначенная для работы с радиомодулями.

## **Заключение**

В ходе практики мы получили опыт командной разработки, опыт объектно ориентированного программирования и программирования микроконтроллеров. Мы столкнулись с большим количеством проблем, большинство удалось решить, но некоторые оказались для нас непреодолимыми в рамках установленных сроков. Мы пришли к выводу что командная разработка - крайне трудоемкая задача, требующая высокой дисциплинированности и обязательной иерархии. Хотя нам и не удалось довести свою задачу до конца, мы достигли основной цели практики.