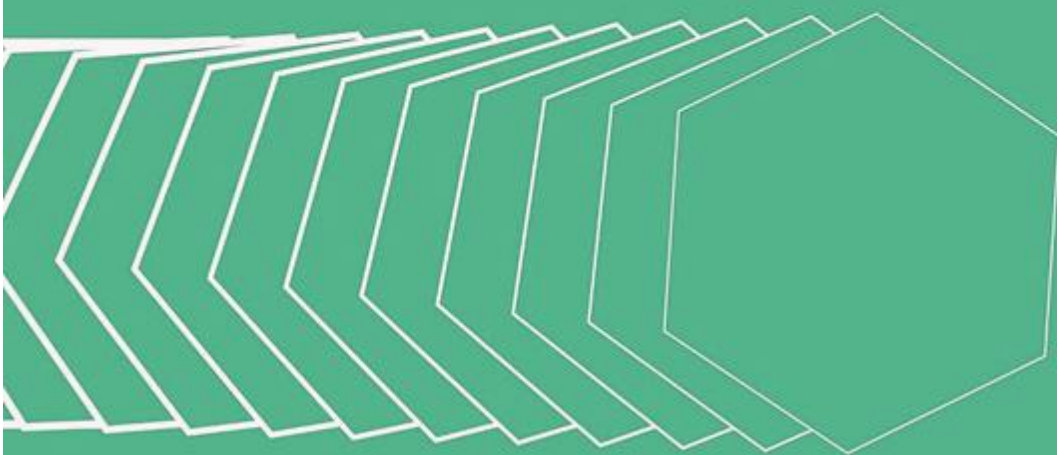




**ФГБОУ ДО
ФЦДО**

Методическая разработка программного модуля

«РОБОТ ПОЧТАЛЬОН»



Направление:
Робототехника

Авторы:

Давыдов А.К., педагог
дополнительного
образования,
Оглезнева А.М., методист,
«Кванториум», г.о. Королев

2022

СОДЕРЖАНИЕ МОДУЛЯ

С каждым годом количество роботов, используемых в разных сферах, непрерывно растет. Существует даже мнение, что к 2030 г. Они смогут полностью заменить людей в ряде профессий. Как Вы думаете, с чем это связано? Робот может постоянно выполнять однообразную и скучную работу, не болеет, не устает, не способен лениться, его можно сделать устойчивым к неблагоприятным факторам внешней среды. Какие профессии может «освоить» робот? Где автоматические помощники будут особенно полезны?

Представим, что нам нужно организовать обмен почтой между несколькими домами на далеком севере. Кругом снег и лютый мороз, а уж если метель, то совсем беда. Живому почтальону добираться будет весьма непросто. Но мы можем сделать сеть небольших тоннелей в снегу, которые защитят нашего посланника от внешней среды, и отправить по ним робота! Вот только сможет ли он проехать по всему маршруту, не забыв ни единого дома? У вас есть базовая модель робота и поле с нарисованным путём в виде линий и перекрестков от места «Почты» до нескольких «Домов». Необходимо запрограммировать робота так, чтобы он смог самостоятельно доехать от «Почты» до всех «Домов», куда должна быть доставлена почта.

Цель модуля: практическое закрепление обучающимися полученных знаний в электронике, программировании при работе с мобильным роботом.

Задачи модуля:

- закрепить навыки сборки мобильного робота;
- научить обучающихся применять навыки программирования в среде Arduino IDE для решения комплексных задач;
- развивать умение работать с датчиками линии.

Планируемые результаты модуля:

Артефакт: Программный алгоритм для мобильного робота базовой конструкции для выполнения задачи автономного перемещения по заранее известным ориентирам (нарисованным линиям) от точки до точки.

Метапредметные навыки (Soft Skills):

- умение принимать и удерживать образовательную задачу;
- умение оценивать и использовать имеющиеся ресурсы;
- умение разбивать комплексную задачу на более простые и последовательно решать их;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата.

Предметные навыки (Hard Skills):

- знание назначения электронных компонентов мобильного робота на основе Arduino;
- понимание механики работы двигателя мобильного робота на основе Arduino;

- умение программировать работу двигателя робота на основе показаний датчика линии;
- разработка алгоритма движения мобильного робота на основе Arduino по заданной траектории;
- осуществлять отладку программы движения мобильного робота в среде Arduino IDE.

Место модуля в структуре программы: в основе модуля лежит изучение основных методов программирования и управления базовой моделью мобильного робота. Рекомендуем использовать данный модуль в качестве вводной части основного модуля ДООП по робототехнике на основе Arduino после знакомства с платформой Arduino, базовыми электронными компонентами и основами схемотехники и изучения основ программирования в Arduino IDE. Модуль позволит на практике закрепить полученные знания и научиться применять их в новой для обучающихся ситуации, развивая навыки решения комплексных задач. Наиболее подходящая возрастная категория обучающихся – 10-13 лет. В этом возрасте у детей формируются основы теоретического мышления и произвольность реакций, стремление к общению со сверстниками и появлением в поведении признаков, свидетельствующих о стремлении утвердить свою самостоятельность, независимость, личную автономию. Ведущая деятельность в этом возрасте – учебно-профессиональная, в связи с чем основой модуля являются проблемный и практикоориентированный подход к обучению.

Количество учебных часов/занятий: 8 часов.

Связь с учебными предметами: информатика, физика, математика.

УЧЕБНЫЙ ПЛАН

№ п/п	Тема занятия	Формы работы	Количество акад. часов		
			Всего	Теория	Практика
1.	Знакомство с мобильным роботом.	Практическая работа	1	0	1
2.	Разбор алгоритма движения по линии.	Практическая работа	2	1	1
3.	Разбор фиксации и подсчёта перекрестков во время движения по линии.	Практическая работа	2	1	1
4.	Программирование заданной траектории движения робота.	Практическая работа	3	0	3
	Итого		8	2	6

Занятие 1.

Тема: Знакомство с мобильным роботом.

Практика: Считывание значений датчиков линии в Arduino. Анализ значений с датчиков на линии и фоне. Управление моторами с платы Arduino.

Занятие 2.

Тема: Разбор алгоритма движения по линии. Тестирование на поле.

Теория: Основы движения по линии с несколькими датчиками. Релейный регулятор на двух датчиках линии.

Практика: Программирование, тестирование и настройка движения мобильного робота по линии.

Примечание: В зависимости от успеваемости обучающихся возможен дополнительный разбор других регуляторов движения по линии (например, пропорциональный).

Занятие 3.

Тема: Разбор фиксации и подсчёта перекрестков во время движения по линии.

Теория: Обработка значений датчиков на перекрёстках. Движение робота на перекрёстке: проезд, поворот. Подсчёт перекрёстков в программе.

Практика: Программирование, тестирование и настройка движения мобильного робота по линии с обработкой, проездом и подсчётом перекрёстков.

Занятие 4.

Тема: Программирование заданной траектории движения робота.

Практика: Написание программы для мобильного робота, позволяющую проехать заданную траекторию. Тестирование и отладка на поле.

Примечание: В зависимости от имеющегося поля и успеваемости обучающихся, траектория может иметь разную сложность, но должна содержать минимум один перекрёсток, на котором необходимо сделать поворот.

НЕОБХОДИМЫЕ МАТЕРИАЛЫ И ОБОРУДОВАНИЕ

Аппаратное и техническое обеспечение:

Поле для тестирования движения мобильного робота, содержащее траекторию с контрастными линией (например, черная линия на белом фоне) и перекрёстками. Пример поля представлен на рисунке 1.

Рабочее место обучающегося:

1. Ноутбук, мышь;
2. Собранный базовый мобильный робот, включающий в себя:
 - плату Arduino;
 - 2-мотора постоянного тока (мотор-редуктора) с колёсами;
 - драйвер для управления моторами;
 - 2-х датчика линии;
 - автономный источник питания (аккумулятор, батарея);
 - корпус и соединительные элементы (провода, макетная плата или др.).

*Примеры подходящего мобильного робота представлены на рисунке 2.

Программное обеспечение: Arduino IDE.

КАДРОВОЕ ОБЕСПЕЧЕНИЕ

Педагог дополнительного образования, который компетентен в следующих областях: электроника, схемотехника, программирование микроконтроллеров Arduino, знание основных алгоритмов движения мобильного робота.

ФОРМЫ АТТЕСТАЦИИ / КОНТРОЛЯ

Вид контроля – рубежный (модульный). Форма контроля – зачет / контрольный заезд.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Контрольное задание модуля – успешное прохождение мобильным роботом траектории движения (Приложение, рис.1).

Критерии оценивания:

1) Прохождение траектории во время контрольного заезда:	2) Количество затраченных попыток:
Менее 40% - 0 баллов 40% < 1 балл < 70% 70% < 2 балла < 100% 100% - 3 балла	1 попытка = 3 балла 2 попытки = 2 балла 3 попытки = 1 балл Более 3 попыток – 0 баллов. Максимальное количество попыток устанавливает наставник

Соответствие итоговой оценки уровню освоения модуля:

6-5 баллов – высокий
4-3 балла – средний
1-2 балла – низкий

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Для педагогов:

1. Arduino для начинающих: самый простой пошаговый самоучитель / Стюарт Ярнольд; [пер. с англ. М. Райтман]. — Москва: Эксмо, 2017. — 256 с.
2. Блум Джереми «Изучаем Arduino: инструменты и методы технического волшебства: Пер. с англ. - СПб.: БХВ-Петербург, 2015. - 336 с.

Для обучающихся:

1. Аппаратная платформа Arduino – <https://arduino.ru/>
2. База знаний Амперки – <http://wiki.amperka.ru/>

Технологическая карта модуля

Введение в проблему	
Предполагаемая продолжительность	Цель блока
15-20 минут	Целеполагание учебной деятельности и формирование познавательной мотивации
Что делаем: Педагог озвучивает проблемную ситуацию и демонстрирует образец мобильного робота для выполнения задачи. Знакомит обучающихся с основными компонентами, задействованными при его создании, спрашивает об их назначении и принципах работы. Предлагает познакомиться с будущим маршрутом движения робота. Ставит перед обучающимися задачу разработать план решения основной задачи модуля – построения программы движения. Выслушивает варианты обучающихся, предлагает их зафиксировать для дальнейшей реализации и/или корректировки.	
Вопросы для обучающихся: 1) Из каких компонентов состоит данный мобильный робот? 2) Какие знания нам понадобятся для выполнения поставленной задачи? 3) Достаточно ли наших знаний, чтобы успешно выполнить поставленную задачу? Что нужно изучить дополнительно? 4) Каков наш план выполнения задачи?	
Подготовительный этап – выявление и устранение дефицита знаний	
Предполагаемая продолжительность	Цель блока
4,5 акад. часа	Выявление и устранение дефицита знаний, необходимых для успешного выполнения задачи модуля
Что делаем: Обучающиеся знакомятся с конструкцией робота самостоятельно. Вместе с наставником изучают управление моторами с платы Arduino, программируют движение робота по линии. Изучают взаимодействие между датчиками линии и работой моторов. После этого обучающимся предлагается попробовать самостоятельно запрограммировать движение мобильного робота для прохождения маршрута. Лучше всего для этой цели заготовить более простой тренировочный маршрут, чтобы оставить для обучающихся пространство для развития на этапе самостоятельного программирования движения робота. В случае успешного прохождения маршрута с учетом перекрестков, предлагаем обучающимся рассказать, как они додумались до такого решения. В случае, если	

маршрут не пройден, просим выделить затруднение и объяснить, что именно пошло не так. Во втором случае педагог знакомит обучающихся с обработкой значений датчиков на перекрёстках и показывает, как программировать движение робота на перекрёстке: проезд, поворот. После этого обучающимся вновь предстоит пройти тренировочный маршрут с учетом новой информации. На этом этапе происходит отработка и закрепление полученных навыков.

Вопросы для обучающихся:

- 1) Из каких действий состоит алгоритм движения робота для прохождения маршрута с учетом перекрестков?
- 2) Почему первоначальная версия алгоритма не сработала (в случае неуспешного прохождения)?
- 3) Какую предварительную подготовку мы должны провести перед написанием итоговой версии программы?

Реализационный этап

Предполагаемая продолжительность

2-3 акад. часа в зависимости о успеваемости группы.

Цель блока

Практическое закрепление полученных знаний в электронике и программировании при работе с мобильным роботом.

Что делаем:

Обучающиеся программируют движение мобильного робота для прохождения контрольного маршрута. Производят необходимые подсчеты, тестируют и отлаживают программу. По окончании Последние 30-40 минут отводятся на контрольный заезд. Количество попыток для каждого обучающегося устанавливается преподавателем. Полученные результаты фиксируются для определения успешности прохождения образовательного модуля.

Рефлексия

Предполагаемая продолжительность

10-15 минут

Цель блока

Осознание обучающимися метода преодоления затруднений и самооценка ими результатов своей коррекционной (а в случае, если ошибок не было, самостоятельной) деятельности.

Что делаем:

Педагог собирает обратную связь по результатам освоения модуля. Выбор формы проведения рефлексии зависит от состава группы и предпочитаемых методов самого наставника. В случае, если традиционные формы рефлексии отсутствуют, можно провести с обучающимися беседу по нижеследующим вопросам.

Вопросы для обучающихся:

- 1) Насколько вы удовлетворены результатами прохождения модуля?
- 2) С какими затруднениями вам пришлось столкнуться?
- 3) Какие знания и навыки вы сумели приобрести?
- 4) Что было самым запоминающимся в ходе работы над задачей?
- 5) Где еще может пригодиться полученный навык?
- 6) Насколько их видение решения задачи (должно было быть зафиксировано на первом занятии) совпало с реальностью?
- 7) Можно ли улучшить результат из работы? Каким образом?

Приложение 2

Материалы для педагога

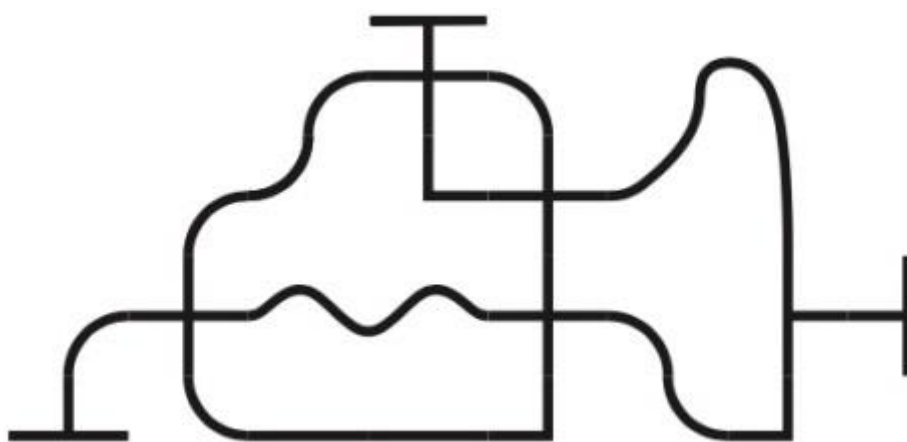


Рисунок 1. Пример поля, подходящего для данного модуля

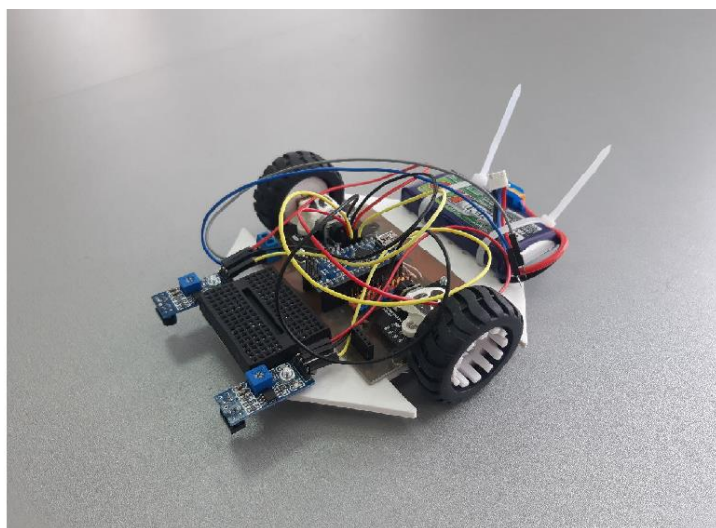
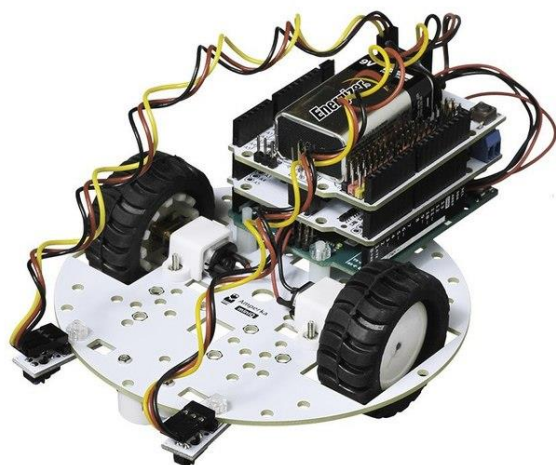


Рисунок 2. Примеры мобильных роботов для выполнения задания

Пример кода

// Данный пример кода написан для прохождения траектории мобильным роботом


```

// Основные компоненты робота, на основе которых написан данный код:
// Arduino UNO;
// - Motor Shield производства Amperka, управляющий двумя моторами
постоянного тока
//
http://wiki.amperka.ru/%D0%BF%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82%D1%8B:arduino-motor-shield;
// - 2 аналоговых датчика линии
//
http://wiki.amperka.ru/%D0%BF%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82%D1%8B:analog-line-sensor;

// Правый мотор подключен к клемме: M1+ и M1-
// Левый мотор подключен к клемме: M2+ и M2-
// Motor shield использует четыре контакта 4, 5, 6, 7 для управления моторами
// 4 и 7 — для направления, 5 и 6 — для скорости
#define SPEED_R 5
#define DIR_R 4

#define SPEED_L 6
#define DIR_L 7

// Подключение аналоговых датчиков линии:
// Левый в A0, Правый в A1
// Робот будет ехать по чёрной линии на белом фоне
// Примем следующие значения, выдаваемые датчиками (необходимо
проверить индивидуально для каждого робота):
// Чистое значение белого - 50
// Чистое значение чёрного - 750
// Граничное значение между белым и чёрным (серое) - 400
#define LEFT_SENSOR_PIN A0
#define RIGHT_SENSOR_PIN A1

int num_cross = 0; //переменная для подсчёта перекрёстков

void setup() {
// Настраиваем порты управления MotorShield в режим выхода сигнала
pinMode(SPEED_R,OUTPUT);
pinMode(DIR_R,OUTPUT);

```

```

pinMode(SPEED_L,OUTPUT);
pinMode(DIR_L,OUTPUT);
}

void loop() {
    // Считываем и запоминаем значения с датчиков линии
    int lSens = analogRead(LEFT_SENSOR_PIN);
    int rSens = analogRead(RIGHT_SENSOR_PIN);
    if ((lSens<400)&&(rSens<400)){//если оба датчика видят белое (светлее серого),
т.е. линия между датчиками,
        // то едем вперёд
        digitalWrite(DIR_R,1);      //Запускаем оба мотора вперёд с одинаковой
скоростью
        analogWrite(SPEED_R,100);
        digitalWrite(DIR_L,1);
        analogWrite(SPEED_L,100);
    }

    else if ((lSens>400)&&(rSens<400)){//иначе, если левый на чёрном (темнее
серого), а правый на белом (светлее серого), т.е. линия слева,
        // то едем вперёд с подворотом налево
        digitalWrite(DIR_R,1);    //Запускаем правый мотор быстрее вперёд
        analogWrite(SPEED_R,120);
        digitalWrite(DIR_L,1);    //Запускаем левый мотор медленнее вперёд
        analogWrite(SPEED_L,80);
    }

    else if ((lSens<400)&&(rSens>400)){//иначе, если левый на белом (светлее
серого), а правый на чёрном (темнее серого), т.е. линия справа,
        // то едем вперёд с подворотом направо
        digitalWrite(DIR_R,1);    //Запускаем правый мотор медленнее вперёд
        analogWrite(SPEED_R,80);
        digitalWrite(DIR_L,1);    //Запускаем левый мотор быстрее вперёд
        analogWrite(SPEED_L,120);
    }

    else if ((lSens>400)&&(rSens>400)){//иначе, //если оба датчика видят чёрное
(темнее серого), т.е. наехали на перекрёсток,
        // то считаем данный перекрёсток

```

```

num_cross = num_cross + 1;

//и проверяем посчитанный номер
if (num_cross==1){ //если перекрёсток первый, то нужно повернуть на нём
налево
    //проезжаем немного вперёд, чтобы колёсами встать ровно на перекрёсток
(время задержки подбирается экспериментально)
    digitalWrite(DIR_R,1);
    analogWrite(SPEED_R,100);
    digitalWrite(DIR_L,1);
    analogWrite(SPEED_L,100);
    delay(1000);

    //затем поворачиваем налево вокруг своей оси, пока робот не повернётся
до линии продолжения (время задержки подбирается экспериментально)
    digitalWrite(DIR_R,1);
    analogWrite(SPEED_R,100);
    digitalWrite(DIR_L,0);
    analogWrite(SPEED_L,100);
    delay(1000);
}
else if (num_cross==2){ //иначе, если перекрёсток второй, то снова нужно
повернуть на нём налево
    //проезжаем немного вперёд, чтобы колёсами встать ровно на перекрёсток
(время задержки подбирается экспериментально)
    digitalWrite(DIR_R,1);
    analogWrite(SPEED_R,100);
    digitalWrite(DIR_L,1);
    analogWrite(SPEED_L,100);
    delay(1000);

    //затем поворачиваем налево вокруг своей оси, пока робот не повернётся
до линии продолжения (время задержки подбирается экспериментально)
    digitalWrite(DIR_R,1);
    analogWrite(SPEED_R,100);
    digitalWrite(DIR_L,0);
    analogWrite(SPEED_L,100);
    delay(1000);
}

```

```
else if (num_cross==3){ //иначе, если перекрёсток третий (финишный), то на нём надо остановиться
```

```
//Ставим скорости обоих моторов равные нулю и тормозим программу  
здесь на долгое время
```

```
digitalWrite(DIR_R,1);
```

```
analogWrite(SPEED_R,0);
```

```
digitalWrite(DIR_L,1);
```

```
analogWrite(SPEED_L,0);
```

```
delay(100000000000);
```

```
}
```

```
}
```

```
}
```

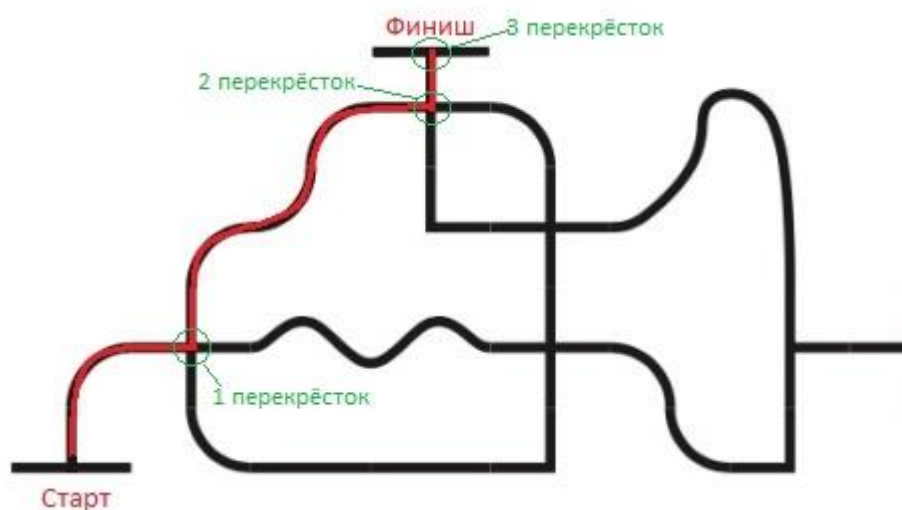


Рисунок 3. Получившаяся траектория движения робота