# **Методическое пособие по технологии: «Введение в электротехнику»**

# **Введение**

Одними из разделов в предмете «Технология» являются «Электротехника и автоматика» и «Робототехника», где ученики 8–9 классов должны познакомиться с этими понятиями. Однако эти темы в основном имеют теоретический характер и недостаточно охватывают практические занятия по сборке схем, программированию и другим аспектам. Кроме того, они не используют возможности для интеграции знаний из физики, информатики и физического эксперимента. Предложенный курс устраняет эти недостатки.

На первых уроках ученики знакомятся с основами работы в среде Arduino IDE, изучают понятия отладочной платы и микроконтроллера, учатся управлять простейшими схемами, состоящими из светодиода и одного датчика. Ученики знакомятся с понятиями цифрового сигнала, широтно-импульсной модуляции (ШИМ), аналогового сигнала, АЦП. На этих уроках ученики овладевают навыком чтения принципиальных схем, узнают о принципах работы подключаемых датчиков и других компонентов, углубляясь в физические аспекты, такие как напряжение, ток, закон Ома, последовательное и параллельное соединение, мощность, работа, вольтамперная характеристика.

В следующих уроках ученики погружаются в основы программирования на языке Arduino (C++), изучая фундаментальные концепции программирования. Они знакомятся с типами данных, переменными, областью их видимости, учатся выполнять математические операции и преобразования между типами. Особое внимание уделяется структуре программы Arduino, логическим операторам, условным конструкциям и циклам. Ученики осваивают работу с последовательным портом, учатся выводить отладочную информацию, принимать и обрабатывать входящие данные. Практические задания включают управление светодиодами, RGB-светодиодом, постепенно усложняясь от простых линейных алгоритмов до более сложных программ.

В этих уроках делается акцент на том, как переменные хранятся в памяти контроллера, как обрабатываются вычислительным ядром. Если ученики пишут код, который исполняется на компьютере, то они обычно не задумываются об объеме используемой памяти и времени работы программы, так как ресурсы современного компьютера многократно превосходят те потребности, которые необходимы для решения учебных задач. А микроконтроллер, используемый в Arduino Nano, очень ограничен по вычислительным ресурсам и памяти, поэтому при работе с программой появляется необходимость четко понимать, что такое типы данных, где какой тип использовать, на что влияет оптимизация программы по использованию памяти и времени работы. Таким образом, один из главных недостатков этой платформы (по сравнению с другими микроконтроллерами, такими как Raspberry Pi, ESP) может стать преимуществом для обучения программированию.

Последний раздел комбинирует знания из схемотехники и программирования, позволяя создавать относительно сложные проекты, такие как, например, двухосевой позиционер, состоящий из двух сервомашинок, управляемый при помощи джойстика. Ученики столкнутся с проблемой шумов в полезном сигнале джойстика, должны будут его отфильтровать и далее использовать полученные значения для управления сервоприводами при помощи инкрементов и декрементов. В процессе создания этих алгоритмов они познакомятся с концепцией квазипараллельного программирования микроконтроллеров.

# **Актуальность**

Актуальность разработки учебно-методического комплекса по изучению электротехники, программирования микроконтроллеров обусловлена несколькими факторами. В условиях стремительной цифровизации и повсеместной автоматизации возникает острая необходимость формирования у обучающихся практических навыков работы с электронными системами управления, способности быстро адаптироваться к новым технологическим решениям. Микроконтроллеры Arduino представляют собой отличный инструмент для развития инженерного мышления, позволяя студентам не только изучать теоретические основы электроники, но и создавать реальные работающие устройства, что дополняет методику технического обучения на активной, проектно-ориентированной деятельностью.

# **Методические преимущества**

1. Практико-ориентированный подход
   1. Замена теоретического обучения практическими занятиями
   2. Сборка разнообразных электронных схем
   3. Программирование микроконтроллеров для решения различных задач
2. Междисциплинарная интеграция
   1. Объединение знаний физики, информатики и электроники
   2. Углубленное изучение физических явлений через эксперимент
   3. Связь теоретических концепций с практическим применением
3. Развитие системного технического мышления
   1. Понимание взаимосвязи аппаратных и программных компонентов
   2. Изучение принципов работы электронных устройств
   3. Формирование навыков проектирования технических систем
4. Глубокое погружение в программирование
   1. Изучение фундаментальных концепций программирования
   2. Понимание работы программы на уровне микроконтроллера
   3. Развитие низкоуровневого понимания вычислений (изучение особенностей работы с памятью, понимание влияния типов данных на производительность, навыки оптимизации программы)
5. Формирование исследовательских навыков
   1. Самостоятельное экспериментирование
   2. Отладка и тестирование электронных устройств
   3. Развитие креативного подхода к решению технических задач
6. Профориентационный потенциал
   1. Знакомство с современными технологиями
   2. Получение практических навыков в робототехнике
   3. Возможность профессиональной ориентации в технических областях

# **Обзор пособия «Введение в электротехнику»**

Уроки модулей «Электротехника» и «Робототехника» состоят из лекционной части, которая занимает около 10 минут урока, где преподаватель обозначает основные понятия, которые необходимо усвоить на уроке. Далее ученики собирают схему на плате breadboard и пишут программу в среде Arduino IDE. Одним ученикам дается проще сборка схем, другим – программирование. Краткая шпаргалка к уроку на двух сторонах листа A5 позволяет ученикам работать в комфортном для них темпе.

Методический материал (см. приложение) состоит из 8 уроков (в будущем будет дополнен до 12). Теоретический материал кратко изложен на одной стороне, принципиальная схема, схема (картинка) с breadboard и программа – на второй. Предполагается, что ученики имеют доступ ко всем предыдущим страницам методички.

Структура урока выстроена логично и последовательно, начиная с базовых понятий и постепенно переходя к более сложным концепциям, что обеспечивает плавное погружение учащихся в материал. Теоретические знания сразу подкрепляются практическими заданиями, позволяя ученикам немедленно применять полученные знания на практике. Последовательность изложения материала тщательно продумана: от простейших схем с одним светодиодом к более комплексным проектам, что создает ощущение постепенного прогресса и мотивирует учащихся. Методичка удобна для восприятия благодаря четкому разделению на тематические блоки, наличию иллюстративного материала и схем, которые демонстрируют теоретические концепции. Технические термины вводятся постепенно и сопровождаются понятными объяснениями, что делает материал доступным для начинающих. Дизайн пособия способствует удержанию внимания учащихся: важные моменты выделены, информация структурирована, а практические задания четко отделены от теоретического материала, что облегчает навигацию по документу и позволяет быстро находить нужную информацию.

Изображение выглядит как текст, снимок экрана, дизайн

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

Рис. 1. Урок «Аналоговый сигнал».

Методическое пособие дополнено проверочными работами, которые размещены после каждых четырех уроков, что позволяет эффективно оценивать прогресс учащихся и закреплять пройденный материал. Эти самостоятельные работы разработаны с учетом гибкости применения: их можно выполнять как в условиях класса с использованием всего необходимого оборудования (платы Arduino, различные электронные компоненты включая светодиоды, резисторы, датчики), так и в домашних условиях без физической сборки схем. Такой подход делает проверочные задания универсальными и не привязанными к наличию специфического оборудования у учащихся дома. Задания сформулированы таким образом, что позволяют проверить теоретическое понимание принципов работы электронных компонентов, навыки программирования в среде Arduino IDE и умение проектировать схемы, даже если непосредственная загрузка написанной программы в контроллер невозможна.

Изображение выглядит как текст, электроника, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки. Изображение выглядит как текст, снимок экрана, Параллельный, дисплей

Контент, сгенерированный ИИ, может содержать ошибки.

Рис. 2. Проверочная работа

# **Цифровые решения, используемые в разработке методических материалов**

Для упрощения восприятия и визуальной привлекательности иллюстрации должны быть выдержаны в едином стиле. Для построения векторных иллюстраций (графиков, диаграмм) использовались инструменты библиотек Python matplotlib и numpy. Эти инструменты обладают широким набором возможностей для построения диаграмм любой сложности, эти инструменты очень удобны для визуализации математических зависимостей.

Использование Python для создания иллюстраций в учебном пособии предоставляет ряд существенных преимуществ. Во-первых, это обеспечивает точность и научную достоверность представленных данных, так как все графики и диаграммы строятся на основе реальных математических моделей и вычислений. Во-вторых, использование этих инструментов в долгосрочной перспективе приводит к значительной экономии времени: однажды написав код, можно многократно использовать его с разными данными, быстро обновлять материалы при изменении программы и автоматизировать рутинные задачи по созданию иллюстраций. В-третьих, это позволяют создавать качественные наглядные материалы в едином стиле, что придает профессиональный вид всем раздаточным материалам и презентациям. На практике это может применяться для создания анимаций физических процессов, разработки интерактивных моделей электронных схем, визуализации математических зависимостей в доступной для детей форме. Пример использования этих инструментов в среде разработки PyCharm представлен на рис. 4. а.

а) Изображение выглядит как текст, программное обеспечение, Значок на компьютере, дисплей

Контент, сгенерированный ИИ, может содержать ошибки. б) Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.

Рис. 4. а) пример использования Python для построения иллюстраций, б) пример использования EasyEDA для создания принципиальных схем.

Принципиальные схемы удобно создавать, используя специальные среды разработки, например, EasyEDA. Вообще, это инструмент разработки электронных схем и печатных плат для промышленного использования, однако он имеет дружелюбный интерфейс, огромную библиотеку электронных компонентов и возможности симуляции работы электронных схем. Для создания векторных принципиальных схем использовалься именно этот инструмент (рис. 4 б).

# **Заключение**

Курс по модулям «электротехника», «робототехника» интегрирует электронику, физику и программирование. Программа начинается с основ Arduino IDE и микроконтроллеров, развивает навыки сборки и понимания электронных схем и физических процессов, затем переходит к программированию на C++ с акцентом на эффективное использование ресурсов микроконтроллера. Завершается курс созданием комплексных проектов, где учащиеся решают инженерные задачи, включая фильтрацию сигналов и квазипараллельное программирование.

Структура пособия включает учебные занятия и самостоятельные работы для лучшего усвоения материала. Визуальное оформление создано с использованием Python-библиотек (matplotlib, numpy) для графиков и EasyEDA. Такой подход не только обеспечивает единый стиль и профессиональный вид всех материалов, но и значительно оптимизирует процесс их создания и обновления, что упращает дальнейшее развитие этого курса.

**Список литературы**

1. Литература для педагога:
   1. С. Монк. Программируем Arduino. Профессиональная работа со скетчами СПб.: Питер, 2017.
   2. Ревич Ю. В. Занимательная электроника. БХВ-Петербург, 2015. 576 с.: ил.
   3. Сивухин Д. В. Общий курс физики. Учеб. пособие: для вузов. В 5 т. Т.ІІІ. Электричество - 4-е изд., стереот. - М.: ФИЗМАТЛИТ; Изд-во МФТИ, 2004. 656 с.
   4. Бьерн Страуструп. Язык программирования С++.