

УДК 373.167
ББК 32.816; 32.97
А79

Серия основана в 2016 г.

Авторский коллектив:

А. А. Салахова, О. А. Феоктистова, канд. пед. наук
Н. А. Александрова, канд. пед. наук М. В. Храмова

Arduino®. Полный учебный курс. От игры к инженерному проекту / А. А. Салахова, О. А. Феоктистова, Н. А. Александрова, М. В. Храмова. — М. : Лаборатория знаний, 2020. — 175 с. : ил. — (РОБОФИШКИ).

ISBN 978-5-00101-250-4

Предлагаемый полный курс познакомит вас с особенностями аппаратного обеспечения и программирования микроконтроллера Arduino Uno®. Рассмотрены графические языки программирования Snap! и ArduBlock, текстовый язык Wiring и производственные языки. Кроме того, вы научитесь читать, составлять и собирать действующие схемы из электронных компонентов. В заключение мы расскажем вам, как правильно оформлять инженерные проекты.

Материал излагается в формате последовательно выстроенных тем, сопровождаемых вопросами, практическими заданиями и проектами.

Проектная часть курса может быть расширена серией книг «РОБОФИШКИ. Конструируем роботов на Arduino®» издательства «Лаборатория знаний».

Для детей среднего и старшего школьного возраста для применения в урочной и внеурочной деятельности и технического творчества дома.

УДК 373.167
ББК 32.816; 32.97

По вопросам приобретения обращаться:
«Лаборатория знаний»

Телефон: (499) 157-5272, e-mail: info@pilotLZ.ru, <http://www.pilotLZ.ru>

ISBN 978-5-00101-250-4

© Лаборатория знаний, 2020

Оглавление

От авторов	5
Глава 1. Прототипирование в робототехнике	7
1.1. Микроконтроллер как основной компонент робота.	7
1.2. Робототехнические платформы открытого типа. Прототипирование	9
Глава 2. Знакомство с Arduino	15
2.1. Платформа Arduino	15
2.2. Контакты платы Arduino Uno	17
2.3. Макетные платы	24
Глава 3. Программное обеспечение Arduino	27
3.1. Среда разработки Snap4Arduino (S4A)	27
Установка S4A	28
3.2. Среда разработки Arduino IDE	32
3.3. Работа в Arduino IDE	33
3.4. Альтернативное программное обеспечение для Arduino	38
Глава 4. Периферия Arduino	47
4.1. Виды периферийного оборудования	47
4.2. Однокомпонентные устройства	48
4.3. Простые упражнения для Arduino и S4A	52
Мигание светодиодом	52
Маячок с убывающей яркостью	54
Светофор, срабатывающий по кнопке	56
Управление движением с помощью потенциометра. Упражнение «Краб»	59
Управление яркостью светодиода с помощью потенциометра	65
Работа с фоторезистором. Упражнение «Робот» ...	67
Терменвокс	73
Ночной светильник	74
RGB-светодиод	76
Сахарница	79
4.4. Модули и сложные датчики	84
4.5. Применение модулей и S4A	88
Сигнализатор затопления	88
Сервопривод	92
4.6. Платы расширения	94

Глава 5. Язык программирования Wiring	103
5.1. Введение в язык Wiring	103
5.2. Программы на языке Wiring: библиотеки и переменные	104
5.3. Основные функции в языке Wiring	107
Функция <i>setup()</i>	108
Функция <i>loop()</i>	108
5.4. Функции Wiring и ШИМ. Работа со звуками	113
5.5. Графические блоки и код в ArduBlock	117
Подключение датчика уровня жидкости	119
Управление потенциометром	121
Работа с LCD-дисплеем	122
5.6. Практические задания по Wiring	127
Фоторезистор	128
Дальномер	129
Шаговый двигатель	131
Датчик температуры и влажности DHT11	134
5.7. Дополнительные задания для самостоятельной работы	135
Шар с предсказаниями	135
Усложнение задачи 1	136
Автоповорот	136
Реклама «Бегущая строка»	137
5.8. Проект «Развитие моторики»	138
Дополнительные задания	150
Глава 6. Применение робототехники в различных сферах ..	151
6.1. Робототехника в современном мире	151
6.2. Arduino и производственные языки	156
6.3. Оформление робототехнических проектов	164
Этап 1. Постановка и осознание проблемы	169
Этап 2. Выбор стратегии решения	169
Этап 3. Требования и ограничения для выбранного решения	170
Этап 4. Формулирование концепции решения	171
Этап 5. Моделирование архитектуры	171
Этап 6. Ресурсная база	172
Этап 7. Техническое задание	173
Заключение	174

ОТ АВТОРОВ

Уважаемые ребята!


Вы держите в руках учебное пособие по курсу, посвященному робототехнике. Эта область технического творчества (именно творчества!) вам знакома по занятиям с робототехническими конструкторами в предыдущих классах. Но уже тогда вы наверняка задумывались: что же дальше? Возможно, вам казалось, что собранные вами роботы больше похожи на увлекательную игру? Поздравляем! Это значит, что ваше мышление — взрослое. В этом курсе мы постараемся ответить на вопросы о том, как робототехника в школе и ваши проекты связаны с робототехникой на предприятиях и инфраструктурой города.


Взросление — это ответственность. Настало время изучить робототехнику как серьезную область, а не только как инструмент моделирования. Вы узнаете, чем отличается открытая архитектура приложений и аппаратного обеспечения от закрытой. Вы познакомитесь с принятыми стандартными обозначениями в электрических схемах и других документах, сможете представлять свои робототехнические проекты в формате, понятном любому инженеру мира, — в соответствии с международными стандартами. Полезным на вашем творческом пути будет и ознакомление с жизненным циклом проекта. Вы научитесь мыслить так, как мыслят инженеры.


Еще один важный навык взрослого человека — умение адаптироваться и выделять основную информацию. Именно он помогает профессиональным программистам создавать программы высокого качества на различных языках программирования. Вы научитесь не писать на определенных языках, а мыслить алгоритмами и условиями. Алгоритмическое мышление позволит впоследствии быстро выучить любой язык программирования, не важно, будет он графическим или текстовым.

Робототехника — важная составляющая жизни современного высокотехнологичного общества. Даже если вы не свяжете свою будущую профессию непосредственно с робототехникой, вы будете сталкиваться с ней и ее принципами повсеместно. Пособие, которое вы держите в руках, призвано помочь вам чувствовать себя комфортно и легко адаптироваться к быстро развивающемуся высокотехнологичному миру.

Для того чтобы вам было удобнее, мы создали систему сигналов-подсказок. На страницах вам встретятся следующие условные обозначения.

 **Запомните** — самые важные понятия, формулировки которых следует выучить наизусть. Они будут использоваться в дальнейшем и являются фундаментальными.

 **Вопросы** — теоретические задания для проверки знаний после прочтения каждого раздела. В них спрашивается только о том, что было изложено в этом разделе.

 **Практические задания** — практические вопросы и простые упражнения для самостоятельной работы по изученной теме. Они требуют дополнительного поиска информации или работы с электронными компонентами.

Это интересно! — любопытные факты, на которые стоит обратить внимание. Они могут пригодиться вам в повседневной жизни и расширить вашу эрудицию.

В книге вы также найдете проект для самостоятельного выполнения. Работа над проектом потребует обобщения навыков и знаний, полученных на основе нескольких тем. В большинстве случаев проекты похожи на лабораторные работы, однако в конце имеют открытую проблему (вопрос), которую необходимо разрешить самостоятельно, применяя творческие решения. Работа над проектом займет больше времени, чем выполнение практического задания.

Мы надеемся, что курс будет не только полезным, но и увлекательным. Если вам потребуется какая-либо помощь, не стесняйтесь спрашивать своих педагогов или писать нам по адресу arduino@pilotlz.ru

И, главное, никогда не сомневайтесь в себе и своих силах! Знаете ли вы, что главная задача инженера — решить поставленную задачу в условиях ограниченных ресурсов. Ваши ресурсы — это в первую очередь ваши знания и время, а остальное легко приобретается в магазине или уже имеется под рукой. Иногда не мешает заглянуть в заветный уголок дома, где хранится всякий хлам — старая электроника, деревянные или пластиковые ящики и многое другое. Остается только включить инженерную фантазию!

Успехов, дорогие друзья!

Глава 1. Прототипирование в робототехнике

1.1. Микроконтроллер как основной компонент робота

Основным компонентом робота является его «мозг» — *микроконтроллер*. Микроконтроллеры применяются всюду для решения совершенно непохожих задач: от автоматизации рутинных процессов в сельском хозяйстве, обслуживании и эксплуатации зданий, электрификации до сложных автономных систем жизнеобеспечения бункеров. Везде, где автоматизация процесса не требует огромных вычислительных мощностей, применяют компьютеры на базе микроконтроллеров. Основными причинами являются их малое энергопотребление, компактные размеры и простота подключения датчиков для взаимодействия с окружающей средой и обработки показаний. Когда необходимо спроектировать систему, решающую многокомпонентную, но все же *единственную* задачу, микроконтроллер становится эффективным и одновременно недорогим решением. Ранее вы изучали робототехнические конструкторы, блок управления которых содержал микроконтроллер и некоторые периферийные устройства.

Существуют две большие группы плат с микроконтроллерами. К первой группе относятся *отдельные платы* с распаянными контактами для подключения периферийных устройств и подачи питания, например Arduino Mega (рис. 1.1). На такой плате в ряде случаев дополнительно устанавливают минимальный интерфейс для передачи и приема информации от внешних устройств, например разъем microUSB. Кроме того, он может одновременно быть и питающим. Платы Arduino используют напряжение 5 В, аналогичное компьютерным стандартам. Именно такое напряжение подается через USB-разъем от компьютерного блока питания.

Платы имеют небольшие размеры, но их можно сделать еще меньше с помощью *внешнего программатора* — устройства для интерпретации микропроцессором данных от компьютера и обратной трансляции команд внешним устройствам. Например, в программаторе нуждается Arduino Pro Micro. Датчики и дополнительные устройства подключаются к платам проводами.

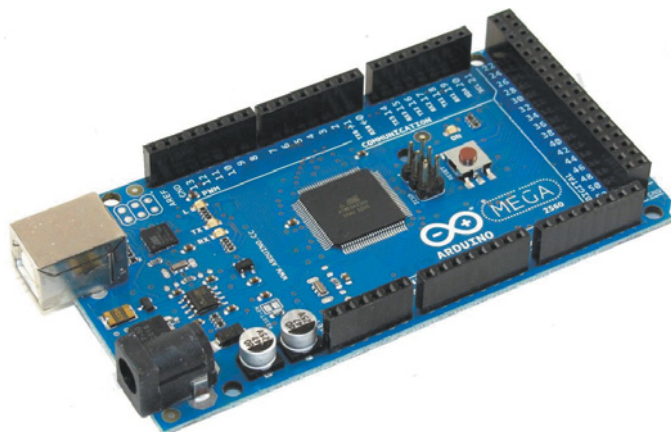


Рис. 1.1. Плата Arduino Mega

Вторая группа плат — это *одноплатные компьютеры*, например Raspberry Pi (рис. 1.2). Как ясно из названия, в этом случае на одной плате располагается не только микроконтроллер, но и часть периферийного оборудования: модули связи, светодиоды и т. д. Благодаря этому одноплатные компьютеры позволяют реализовывать множество разных проектов без каких-либо дополнительных соединений. В результате экономится место и по-



Рис. 1.2. Одноплатный компьютер Raspberry Pi 2 Model B

вышается скорость обмена информацией между компонентами устройства.

Плата микроконтроллера — это основа робототехнической платформы. Платы обеих групп не имеют закрытого корпуса в отличие от готовых программируемых блоков многокомпонентных робототехнических конструкторов, например LEGO NXT, LEGO Mindstorms EV3, VEX.



Вопросы

1. Перечислите плюсы применения микроконтроллеров на производстве или, например, при обслуживании зданий.
2. Назовите основные группы плат микроконтроллеров и их особенности.
3. Чем отличается робот, автоматизирующий процессы обслуживания или производства, от производственного автомата? Имеют ли значение для каждого из них постоянно изменяющиеся показатели окружающей среды? Найдите и выпишите определения робота и автомата.



Практическое задание

Зайдите на сайты с проектами на робототехнических платформах Arduino и Raspberry Pi. Познакомьтесь с проектами и выпишите пять идей, которые вам особенно понравились. Чем они полезны?

Сайты с проектами:

1. Каталог устройств и поделок на Arduino (<http://arduino-projects.ru/>).
2. Проекты для Arduino (<http://arduino-diy.com/>).
3. Робототехнические проекты (<https://diyhacking.com/>).
4. Сообщество Hackster.io (<https://www.hackster.io/>).

1.2. Робототехнические платформы открытого типа. Прототипирование

Что же характерно для *робототехнических платформ открытого типа* (например, Arduino Uno, Raspberry Pi или STM32)? Во-первых, в уже знакомых вам робототехнических конструкторах датчики были заключены в закрытые корпуса (как и сам блок управления) и имели ограниченные (количеством деталей

в наборе) вариации сборки. Доступа непосредственно к электронным схемам не было. В практической робототехнике, применяемой на производстве и в других сферах деятельности человека, количество подключаемых готовых компонентов к микроконтроллеру почти не ограничивается. Основная плата, на которой он расположен, должна иметь возможность бесконечного наращивания компонентов без необходимости полного перемонтажа.

Вторая особенность платформ, которые будут рассмотрены в этой главе, вынесена в заголовок и напрямую связана с возможностью быстрого наращивания компонентов. **Прототипирование** (от англ. *prototyping*) — это процесс быстрой «черновой» сборки устройства.

Устройство, собранное в процессе прототипирования, называется **прототипом**. Прототипы изготавливаются штучно и нужны для тестирования и совершенствования идеи перед запуском в массовое производство.

Процесс создания прототипа состоит из четырех шагов.

1. Определение начальных требований.
2. Разработка первого варианта прототипа, который содержит только пользовательский интерфейс системы.
3. Изучение прототипа заказчиком и конечным пользователем, получение обратной связи о необходимых изменениях и дополнениях.
4. Переработка прототипа с учетом полученных замечаний и предложений.

Зачастую при прототипировании соединения выполняются проводами без пайки и отсутствует внешний корпус. На такую сборку требуется минимальное количество времени и усилий при полной работе итоговой системы. Платформы, поддерживающие прототипирование, позволяют собрать достаточно сложные и многокомпонентные схемы, которые в то же время будут обладать возможностью интеграции и (или) взаимодействия с внешним оборудованием.

Третьей важной особенностью является процесс моделирования. Наборы с ограниченной элементной базой, основанные на собственном стандарте физического интерфейса (например, крепления LEGO), не позволяют переносить и применять собранное устройство в реальных условиях.

Конструкторы, которые были рассмотрены ранее в курсе технологии или на уроках робототехники, предполагали моделирование в искусственной среде. Проекты, реализуемые на

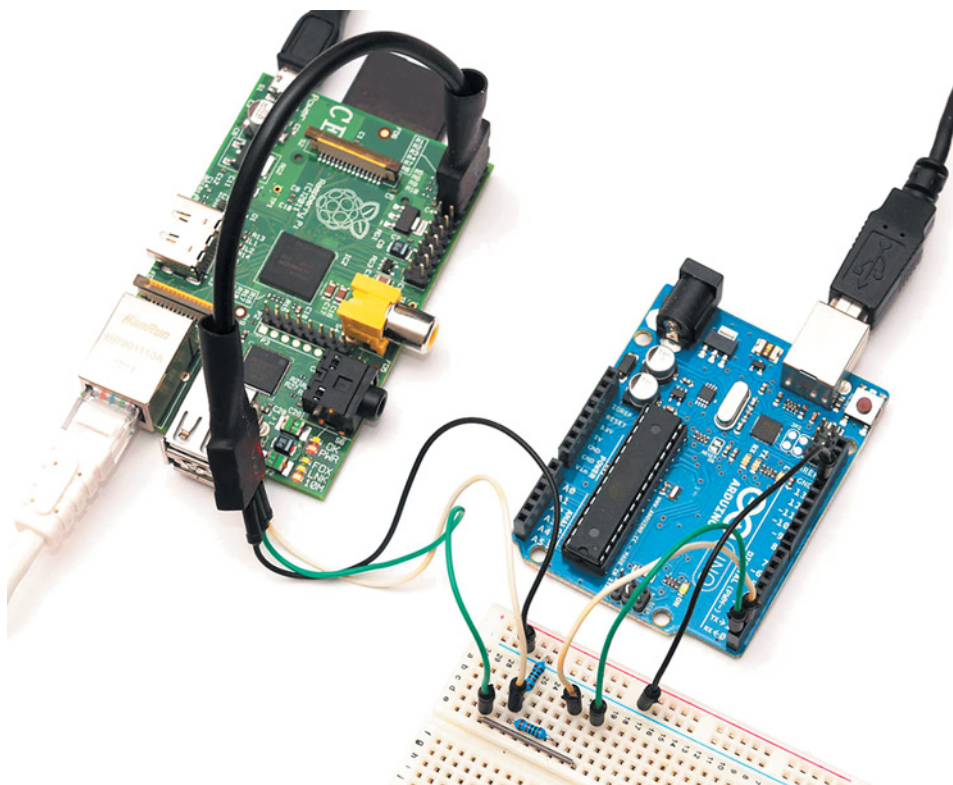


Рис. 1.3. Соединение Raspberry Pi с Arduino Uno через протокол I²C

платформах с прототипированием, носят *прикладной характер* и затрагивают неигровые ситуации.

Применение универсальных интерфейсов управления, питания (например, стандартных для компьютерной техники напряжений 5 В и 3,3 В), передачи данных и крепления позволяет соединять в проектах различные дополнительные компоненты из реального, а не моделируемого мира (рис. 1.3).

Например, установка на прототипе реле (рис. 1.4) позволяет управлять светильниками, электрическими чайниками, кофеварками, вентиляторами, подключенными к обычной бытовой сети. Поэтому платформы, поддерживающие прототипирование, применяют в качестве управляющих устройств в проектах «умного дома» и даже на производстве. Кроме того, универсальность интерфейсов дает возможность приобретать *детали разных производителей*, что практически невозможно для многокомпонентных наборов с собственным интерфейсом («ТРИК», LEGO Mindstorms

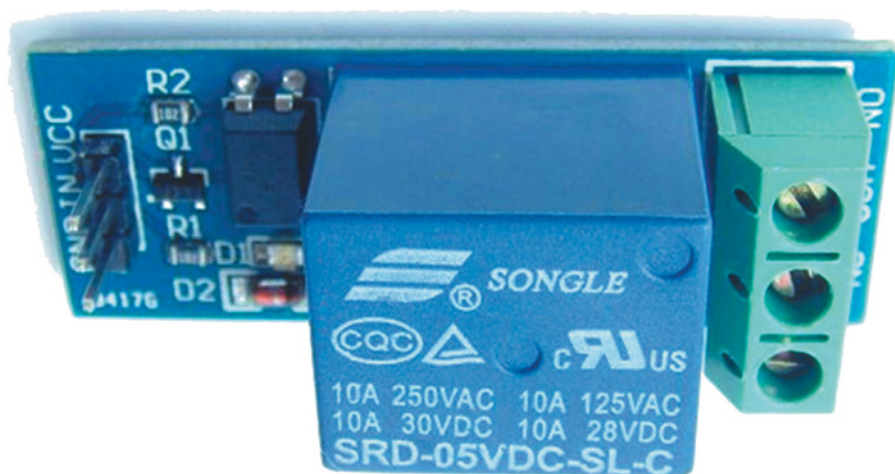


Рис. 1.4. Реле

EV3 и др.). Возможность применения дополнительных компонентов позволяет выбирать для достижения целей проекта самые точные и специфические датчики, например датчик количества газа бутана в воздухе или сканер NFC- и RFID-меток. На основе платформ Raspberry Pi или Arduino можно самостоятельно собрать даже мобильный телефон.

Четвертая особенность этих платформ — в используемых языках программирования. Наборы, нацеленные на решение задач, моделируемых в искусственной среде (игровых), в основном в стандартных средах («из коробки») используют языки программирования, носящие сугубо учебный характер. К ним относится графическое программирование на Scratch, LabView в среде LME и т. д. Робототехнические платформы, поддерживающие прототипирование, о которых пойдет речь далее, предусматривают *использование языков объектно ориентированного или функционального программирования*, применяемых при написании реальных программных оболочек устройств и для управления компонентами сложных систем. Такими языками являются C и его модификации, например C++ и Wiring, JavaScript, Java, Python и Assembler. Однако Arduino, являясь платформой открытого типа, также поддерживает графические языки программирования, и именно с них мы вскоре начнем знакомство с платой Arduino.

Таким образом, переход на робототехнические платформы открытого типа позволяет значительно расширить возможности конструируемых роботов, использовать их для решения прак-

тических задач и приблизиться к изучению реальных производственных робототехнических комплексов.

Вопросы

1. Что такое прототипирование и прототип?
2. Перечислите характерные особенности робототехнических платформ открытого типа.

Запомните

- ◆ Микроконтроллер ◆ Внешний программатор
- ◆ Прототипирование ◆ Прототип

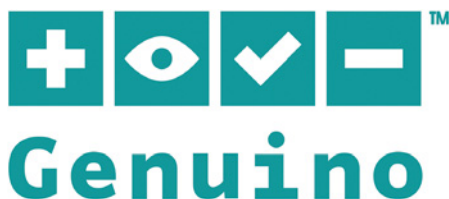
Практическое задание

Найдите информацию о различных платформах открытого типа. Заполните таблицу.

Платформа	Стандартный язык программирования	Чем интересна?
1.		
2.		
3.		
4.		
5.		

Это интересно!

Название «Arduino» согласно официальной документации от разработчиков должно использоваться лишь в США. В остальном мире платформа обязана называться «Genuino». Меняются и логотипы. Несмотря на пожелания разработчиков, по всему миру плата все равно известна как Arduino.



Платформа Raspberry Pi, получившая широкое применение, поддерживает установку операционных систем Linux, Windows 10 и Android. На ее основе можно собрать настоящий смартфон с дополнительными функциями, которых нет даже у самых передовых гаджетов. Например, такой смартфон сможет одновременно служить мультиметром и режущим фанеру лазером. На основе платы Raspberry Pi можно собрать настольный компьютер, подключив к ней монитор, мышь и клавиатуру (рис. 1.5).

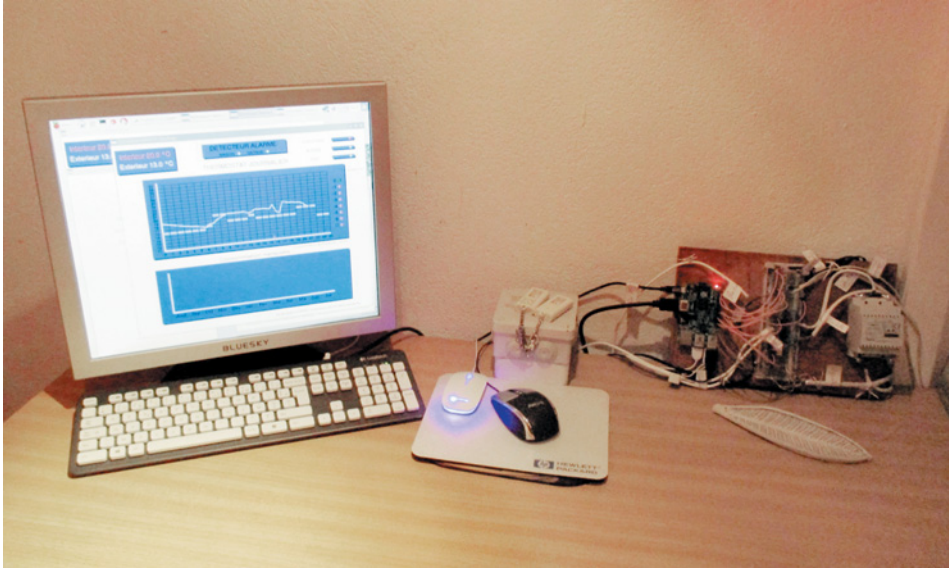


Рис. 1.5. Настольный компьютер на основе платы Raspberry Pi

Глава 2. Знакомство с Arduino

2.1. Платформа Arduino

Arduino — это платформа, состоящая из аппаратно-программных средств для построения систем автоматики и робототехники. *Аппаратной основой* платформы является плата с размещенным на ней микроконтроллером и разведенными (т. е. распаянными) по фиксированной электрической схеме контактами. *Программная часть* представлена средой разработки Arduino IDE.

Платформа обладает несколькими особенностями, делающими ее популярной во всем мире.

Аппаратная часть платформы имеет *открытую архитектуру*. Архитектура аппаратной части устройства называется открытой, если опубликована ее спецификация, т. е. подробное описание, составляющие, схема и применение каждого компонента. Спецификация позволяет любому производителю создать копию продуктов для платформы, тем самым делая их более доступными. Кроме того, появляется возможность создавать улучшенные, более эффективные версии плат и модулей или новые совместимые устройства. Все это влияет на цены, удерживая их на уровне, доступном широкому кругу пользователей. И наконец, открытая спецификация — это возможность самостоятельного ремонта при должных навыках и умениях.

Оригинальные платы Arduino изготавливаются четырех типов в зависимости от габаритов и количества контактов, доступных для подключения (входы и выходы). Приведем самые распространенные из них.

1. *Arduino Uno*: стандартный размер, 20 контактов (рис. 2.1).
2. *Arduino Mega*: увеличенный размер, 70 контактов (рис. 2.2).
3. *Arduino Nano*: уменьшенный размер, 22 контакта (рис. 2.3).
4. *Arduino Micro*: миниатюрная версия, 20 контактов (рис. 2.4), отсутствует DATA-интерфейс USB (требуется внешний программатор, USB обеспечивает только питание платы).

Кроме основных плат с размещенным микроконтроллером, существуют дополнительные платы. Они расширяют возможности Arduino добавлением новых типов интерфейса или дополнитель-

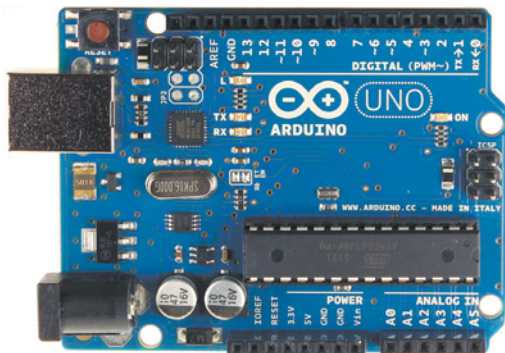


Рис. 2.1. Плата Arduino Uno

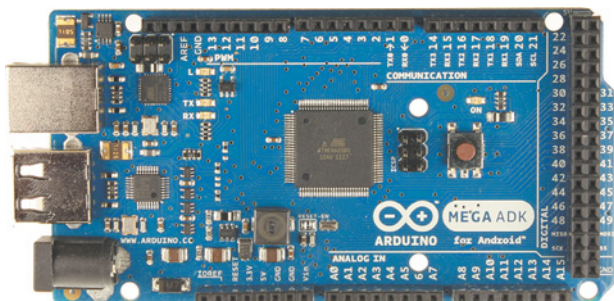


Рис. 2.2. Плата Arduino Mega

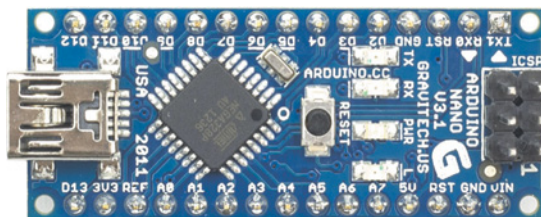


Рис. 2.3. Плата Arduino Nano

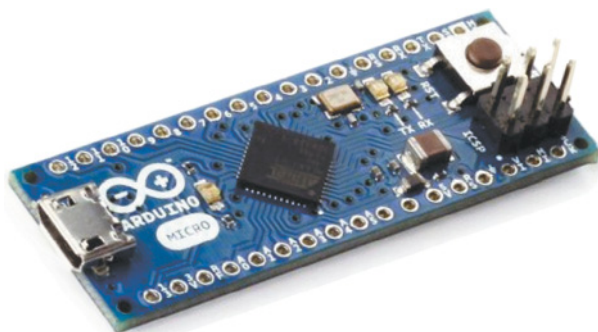


Рис. 2.4. Плата Arduino Micro

ных микросхем. Иногда это зависимые контроллеры, обеспечивающие обмен данными по другим протоколам и стандартам, чем основной микроконтроллер. Такие платы называются *платами расширения* или *щитами (Shield)*. Они устанавливаются непосредственно на плату Arduino Uno или Arduino Mega, занимая все или почти все контакты основной платы, которые дублируются на плате расширения. Подробнее с платами расширения вы познакомитесь в разделе 4.6.

Вопросы

1. Что такое Arduino? Из каких частей состоит данная платформа?
2. Что такое открытая архитектура аппаратной части? Чем полезна открытая архитектура для конечного пользователя?
3. Какие бывают форматы Arduino? Перечислите особенности каждого вида плат.

Это интересно!

Название платформы Arduino Uno связано с местом ее рождения. Официальные платы разрабатываются и производятся в Италии. В переводе с итальянского языка uno означает «один». Arduino Uno — эталонная плата, базовая версия для остальных модификаций.

2.2. Контакты платы Arduino Uno

В книге здесь и далее рассматривается плата *Arduino/Genuino Uno*. Это самая распространенная плата стандартного размера. Она управляется микроконтроллером ATmega328, имеющим характеристики: 2 Кб оперативной памяти, 1 Кб памяти на энергонезависимом носителе (EEPROM, аналог BIOS у персонального компьютера), 32 Кб памяти для загрузки программ, из которых 0,5 Кб занято самим загрузчиком. Входное напряжение от источника питания составляет 7–12 В. Подключенные датчики и модули управляются током 40 мА для основного рабочего напряжения 5 В и 50 мА для дополнительного напряжения 3,3 В. Тактовая частота микропроцессора равна 16 МГц — это почти в 4 раза меньше, чем тактовая частота процессоров первого поколения Pentium для персональных компьютеров!

Юные инженеры и их преподаватели!

Наш полный учебный курс, посвященный робототехнике и платформе **Arduino**[®], познакомит вас с особенностями аппаратного обеспечения и программирования **Arduino Uno**[®]. Кроме того, вы приобщитесь к электронике – научитесь читать, составлять и собирать действующие схемы из электронных компонентов. Наконец, мы покажем вам, как правильно оформлять инженерные проекты.

- ◆ В пособии много внимания уделено языкам программирования. Вы научитесь программировать на графических языках **Snap!** и **ArduBlock**, изучите текстовый язык **Wiring**, познакомитесь с производственными языками программирования.
- ◆ Пособие выполнено в формате последовательно выстроенных тем, сопровождаемых вопросами, практическими заданиями и проектами.

В первую очередь данный курс может быть встроен в урочный процесс в качестве модуля **«Робототехника»**. Также он будет полезен для внеурочной деятельности или самостоятельного изучения.

Курс может рассматриваться в качестве третьей ступени изучения робототехники после курсов С. А. Филиппова **«Уроки робототехники. Конструкция. Движение. Управление»** и Ю. А. Винницкого и К. Ю. Полякова **«Конструируем роботов на ScratchDuino[®]. Первые шаги»** издательства «Лаборатория знаний». Проектная часть курса может быть расширена серией книг **«РОБОФИШКИ. Конструируем роботов на Arduino[®]»**.

info@pilotLZ.ru

arduino@pilotLZ.ru

www.pilotLZ.ru

 vk.com/roboLz

