

# Знакомство с Arduino

# 2

Arduino — это простая и производительная микроконтроллерная платформа, предназначенная для создания прототипов робототехнических и электронных устройств (рис. 2.1). Она считается самой простой платформой, с помощью которой можно создавать программируемые электронные устройства, имеющие высокий уровень надежности.

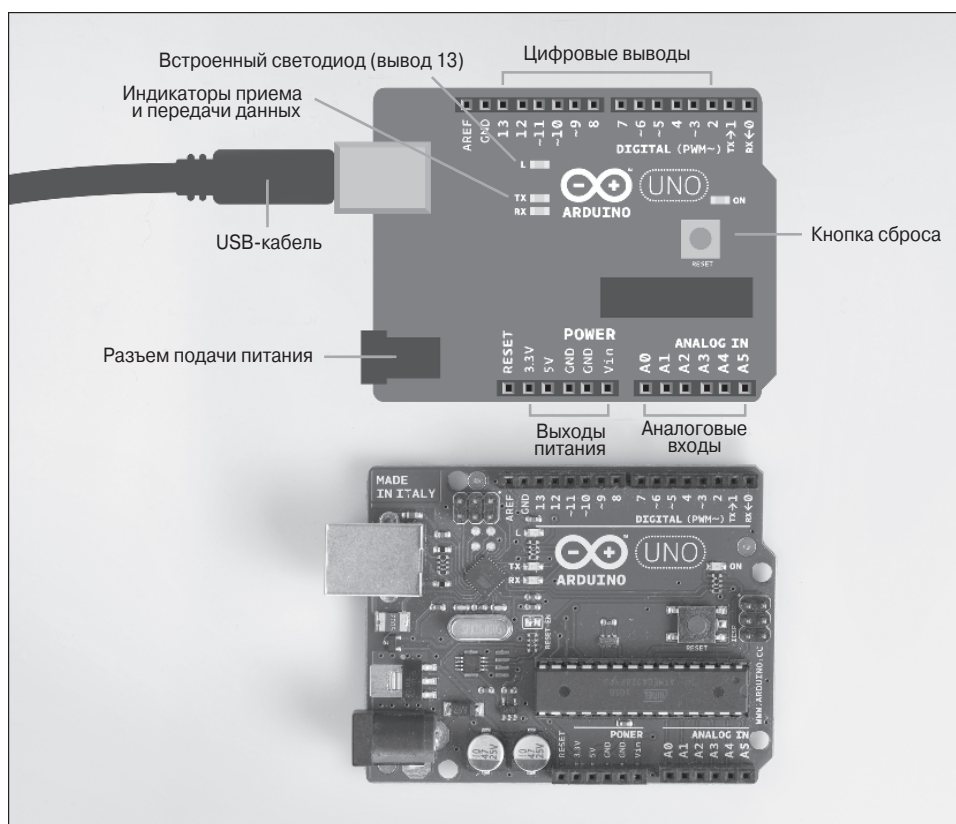


Рис. 2.1. Разъемы Arduino

Начать работу с Arduino очень просто. Все, что вам понадобится, — это самая простая версия Arduino — Uno и кабель USB для подключения платы к компьютеру. Все вместе вам обойдется не более чем в 35–40 долларов. Программное обеспечение для Arduino распространяется бесплатно (исходный код разрешается свободно использовать в обучающих целях, собственных проектах, а также распространять на свое усмотрение).

Первым делом в вашем компьютере нужно настроить интегрированную среду разработки (Integrated Development Environment — IDE) для Arduino. После этого через USB-порт в Arduino можно загрузить программу (на жаргоне Arduino-сообщества она называется “скетчем”). Одновременно в Arduino можно установить только одну программу — ту, которая будет непосредственно выполняться. Вот и все, что вам нужно вкратце знать об Arduino, поскольку, в отличие от Raspberry Pi, у нее нет операционной системы, а потому нечего дополнительно настраивать. Только вы, ваша программа и электрическая цепь, управление которой осуществляется загруженным в Arduino “скетчем”.

---

Все же существует еще одна деталь, о которой стоит упомянуть, описывая Arduino. Это загрузчик, занимающий небольшой объем памяти микроконтроллера. Загрузчик представляет собой небольшую программу, которая запускается непосредственно при подаче питания или перезагрузке платы контроллера. Эта программа позволяет устанавливать внешнее исполняемое приложение, загружаемое через USB-порт, исключая необходимость применения отдельного аппаратного программатора.

---

Небольшая по размерам Arduino Uno имеет высокую надежность. Вам вряд ли удастся повредить электронику платы вследствие неправильного подключения внешнего оборудования (не стоит умышленно проверять платформу на крепость, ведь при должном старании все же можно “выжечь” один или несколько контактов платы).

Обучаться работе с Arduino очень интересно. Начинающие пользователи будут удивлены широкими возможностями платформы по подключению внешнего оборудования. В отличие от Raspberry Pi, можно напрямую подключать к ней аналоговые резистивные датчики без использования дополнительного оборудования, поскольку Arduino оснащается встроенным аналого-цифровым преобразователем.

## Базовая настройка Arduino

---

Ниже показано, как правильно настроить Arduino в Linux, Windows и Mac.

### Ubuntu Linux

Подключите Arduino к компьютеру с помощью кабеля USB. Питание на Arduino подается непосредственно из USB-порта компьютера, поэтому внешний блок питания для него не потребуется. Запустите терминальное приложение.

---

Командный терминал запускается различными способами. В Xubuntu и других основанных на технологии XFCE дистрибутивах, например Debian, выполните из главного меню команду **Applications** ⇒ **Accessories** ⇒ **Terminal** (**Приложения** ⇒ **Стандартные** ⇒ **Терминал**). Клавиша <Super>, также известная как клавиша <Windows> или "ненужная" клавиша, поддерживается во многих операционных системах. Если в стандартном дистрибутиве Ubuntu вы используете оболочку Unity, то найдете терминал в главном меню (**Dash**), расположенном в левом верхнем углу экрана.

---

Для того чтобы добавить в компьютер среду разработки Arduino, установите пакет `arduino`. Ниже показано, как эта задача выполняется в Ubuntu Linux.

```
$ sudo apt-get update
$ sudo apt-get -y install arduino
```

Откройте своему пользователю доступ к USB-порту (без выполнения этого требования среда разработки Arduino не будет корректно функционировать). Первая из приведенных ниже команд добавляет пользователя в группу `dialout`, а вторая переключает командную среду на эту группу, избавляя вас от необходимости выходить из системы и регистрироваться в ней повторно.

```
$ sudo adduser $(whoami) dialout
$ newgrp dialout
```

Запустите среду разработки Arduino следующей командой:

```
$ arduino
```

Среда разработки готова к использованию. При выходе и последующем входе в систему вы сможете запустить среду разработки Arduino из меню.

Теперь можно приступить к тестированию среды разработки Arduino и запуску первой программы.

## Windows 7 и Windows 8

Загрузите последнюю версию программного пакета Arduino по следующему адресу:  
<http://arduino.cc/en/Main/Software>

Распакуйте загруженный архив в любое удобное для вас место (на рабочий стол или в папку Загрузки, например).

Подключите плату Arduino Uno к компьютеру с помощью кабеля USB. Как уже упоминалось, плата Arduino получает питание из USB-порта, поэтому ей не требуются дополнительные источники электроэнергии. Операционная система Windows автоматически найдет новое оборудование и начнет процесс установки драйверов для Arduino. Однако процесс установки далеко не всегда завершается успешно, и в подобных случаях на экране появляется сообщение об ошибке.

Если драйверы для Arduino автоматически установить не удалось, то выполните следующие действия.

1. Запустите программу Проводник (Windows Explorer), щелкните правой кнопкой мыши на значке Компьютер и выберите в контекстном меню команду Управление.
2. В окне Управление компьютером перейдите в раздел Диспетчер устройств, выберите соответствующую команду на левой панели. Найдите в списке оборудования Arduino Uno, щелкните на нем правой кнопкой мыши и выберите команду Обновить драйверы.
3. Выберите вариант Выполнить поиск драйверов на этом компьютере. Перейдите в папку, в которую ранее распаковывался дистрибутив программного пакета Arduino, откройте папку с драйверами, выделите файл `arduino.inf` и щелкните на кнопке Далее.
4. Windows установит драйверы для Arduino.

Запустите среду разработки Arduino, дважды щелкнув на значке Arduino в папке, в которую был распакован программный пакет Arduino.

Можно приступить к тестированию среды разработки Arduino и запуску первой программы.

## OS X

Загрузите последнюю версию программного пакета Arduino по такому адресу:

<http://arduino.cc/en/Main/Software>

Распакуйте загруженный архив и скопируйте его содержимое в папку `/Applications`.

Подключите Arduino Uno к компьютеру с помощью USB-кабеля. Как уже упоминалось, плата Arduino получает питание от USB-порта, поэтому ей не нужны дополнительные источники электроэнергии. Операционная система OS X не требует установки специальных драйверов для Arduino.

Запустите среду разработки Arduino, дважды щелкнув на значке Arduino в каталоге `/Applications`.

Можно приступить к тестированию среды разработки Arduino и запуску первой программы.

## Приветствие светодиодом

Теперь, когда вы установили в операционной системе среду разработки Arduino, давайте выполним на этой платформе эквивалент программы приветствия, с которой вы уже познакомились при тестировании Raspberry Pi в предыдущей главе.

Для начала удостоверьтесь, что среда разработки правильно определила вашу платформу. По умолчанию в ней указывается вариант Arduino Uno. Если у вас другая версия платформы Arduino, например Mega или Leonardo, то укажите ее в меню `Tools` ⇒ `Board` (Сервис ⇒ Плата).

Загрузите тестовую программу Blink (Мигание). Выполните команду `File` ⇒ `Examples` ⇒ `1.Basics` ⇒ `Blink` (Файл ⇒ Примеры ⇒ 1.Базовые ⇒ Мигание). Щелкните на кнопке `Upload` (Загрузить) или выполните команду `File` ⇒ `Upload` (Файл ⇒ Загрузить), чтобы скомпилировать и загрузить программу в Arduino.

При загрузке первой программы в среде разработки Arduino на экране может отобразиться сообщение об ошибке: `Serial port COM1 not found` (Последовательный порт COM1 не найден). Это происходит потому, что вы не указали, какой именно последовательный порт будете использовать для передачи данных (подключение Arduino к компьютеру представляется как последовательное соединение USB). Выберите необходимый порт в раскрывающемся меню. В Linux это будет `/dev/ttyACM0`, в Mac — скорее всего `/dev/usbmodem1234`, а в Windows — один из COM-портов.

---

Если на экране появилось сообщение об ошибке, отличной от ошибки последовательного соединения, то все равно укажите необходимый порт в меню **Tools** ⇒ **Port** (**Сервис** ⇒ **Последовательный порт**). Если вам сложно определить, к какому порту подключена плата Arduino, то сначала запомните все порты, перечисленные в списке, а затем отключите Arduino от компьютера и посмотрите, какой порт перестал отображаться в списке. Именно через этот порт Arduino и подключается к компьютеру. В OS X каждый порт перечисляется в списке дважды, например `/dev/cu.usbmodem1234` и `/dev/tty.usbmodem1234`. Вам подойдет любой из вариантов.

---

При загрузке программы в Arduino на его плате быстро мигают индикаторы TX (передача данных) и RX (прием данных). А при выполнении программы мигать начинает небольшой индикатор, обозначенный символом L.

Мигание светодиода L извещает об успешной загрузке и выполнении первой тестовой программы.

Примите наши поздравления! Зачем вообще нужно было запускать встроенную тестовую программу? Все просто: в будущем, при реализации сложных проектов, у вас может сложиться ситуация, когда вы засомневаетесь, а выполняет ли Arduino написанный вами исходный код? Вот тогда вспомните свой первый опыт работы с Arduino и повторите его, чтобы удостовериться в работоспособности платформы.

## Структура программы Arduino

В Arduino весь единожды выполняемый программный код находится в теле функции `setup()`. Вместе с этим код функции `loop()` выполняется повторно бесконечное количество раз (до прекращения подачи питания на плату). Рассмотрим листинг 2.1.

### Листинг 2.1. `blink.ino`

```
// blink.ino - мигание светодиода L при проверке среды
// (c) BotBook.com - Karvinen, Karvinen, Valtokari
void setup() { // ❶
  pinMode(13, OUTPUT); // ❷
}

void loop() { // ❸
  digitalWrite(13, HIGH); // ❹
  delay(1000); // мс // ❺
  digitalWrite(13, LOW);
  delay(1000);
}
```

- ❶ При загрузке программы в Arduino код, содержащийся в теле функции `setup()`, выполняется один раз.
- ❷ Установка вывода D13 в режим вывода сигнала (OUTPUT); предоставляется возможность управлять им из программы.
- ❸ После завершения функции `setup()` вызывается функция `loop()`. После выполнения всех включенных в нее операторов функция `loop()` вызывается повторно. Потом еще раз. И так до бесконечности.
- ❹ Выводу D13 назначается сигнал высокого уровня (HIGH), что соответствует подаче на этот контакт платы Arduino напряжения +5 В.
- ❺ В течение указанной временной задержки состояние вывода не изменяется. Поскольку выводу D13 присвоено значение HIGH, встроенный в Arduino индикатор L продолжает светиться. В течение следующей задержки на вывод передается значение LOW (сигнал низкого уровня), поэтому индикатор L гаснет. Поочередно: светится одну секунду (100 мс) и не светится — тоже одну секунду. И так вечно (шутка)!

## Дополнительные модули: простота и удобство использования

Дополнительный модуль, или плата расширения<sup>1</sup>, — это плата, устанавливаемая поверх платы Arduino и расширяющая функциональные возможности последней или повышающая удобство работы с ней (рис. 2.2). Существует огромное количество самых разнообразных плат расширения для Arduino, начиная с модулей макетирования (прототипирования) и заканчивая сложными устройствами, такими как адаптеры Ethernet или Wi-Fi. Главное достоинство дополнительных модулей для Arduino заключается в их, как бы странно это ни звучало, беспроводном монтаже; они буквально надеваются на плату Arduino, подключаясь своими разъемными колодками к штекерным колодкам платы контроллера. Конечно, далеко не все задачи можно решать с помощью плат расширения, но и забывать об их существовании тоже неправильно.

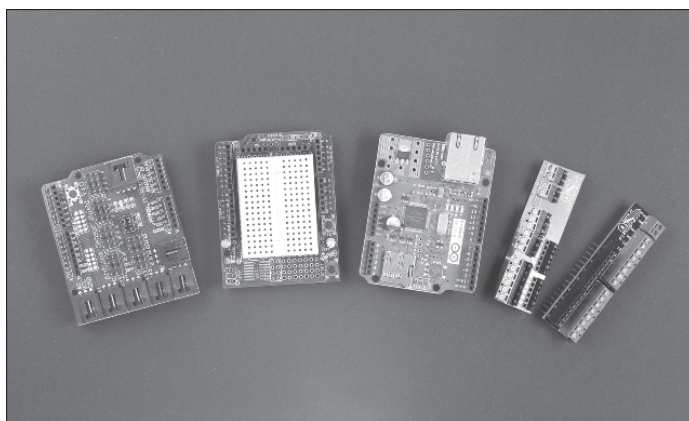


Рис. 2.2. Платы расширения

<sup>1</sup> Известная также как шилд, от англ. “shield” — щит. — *Примеч. ред.*

Некоторые модули вообще не содержат электронных компонентов, а предназначены для удобства подключения макетной платы: они снабжают Arduino штыревым разъемом, позволяющим надевать на него макетную плату безопасного монтажа, что избавляет от необходимости применять проволочные перемычки. Нашим любимым дополнительным модулем стал бесценный Screw Shield, снабжающий плату Arduino (с обеих сторон!) колодками с зажимными клеммами. Он предотвращает выпадение проводов из гнезд, что является самым раздражающим моментом при создании прототипов устройств на макетной плате.

Можете самостоятельно заняться разработкой и созданием плат расширения, внося посильный вклад в развитие платформы Arduino (рис. 2.3). Самое главное — это припаять к своей печатной плате разъем, совпадающий по форме и размеру с контактной колодкой Arduino.



**Рис. 2.3.** *Дополнительные модули, созданные Андреасом Зингерле*