

УДК 004+61+796
ББК 32.973.26+28.708
Я92

Яценков В. С.

Я92 Здоровье, спорт и окружающая среда в проектах Arduino. — СПб.: БХВ-Петербург, 2020. — 336 с.: ил. — (Электроника)

ISBN 978-5-9775-4068-1

Описаны новые проекты на платформах Arduino и ESP32/8266 и увлекательные опыты по исследованию человеческого организма и окружающей среды с использованием самых современных и доступных сенсоров и модулей профессионального уровня. Каждый проект начинается с описания основ изучаемого явления и завершается опытами и заданиями для самостоятельной работы. Рассказано, как выбрать плату Arduino, создать домашнюю лабораторию, измерять частоту пульса и содержание кислорода в крови, проверять гальваническую реакцию кожи, снимать электрокардиограмму и регистрировать мышечные токи, контролировать чистоту окружающего воздуха и измерять интенсивность ультрафиолета в разных диапазонах, обрабатывать данные и работать с онлайн-сервисом визуализации Adafruit IO. На сайте издательства помещен архив файлов с исходными кодами программ и цветными иллюстрациями.

*Для широкого круга любителей электроники, школьников,
студентов и преподавателей*

УДК 004+61+796
ББК 32.973.26+28.708

Группа подготовки издания:

Руководитель проекта	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Екатерина Сависте</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Дизайн серии	<i>Марины Дамбиевой</i>
Оформление обложки	<i>Карины Соловьевой</i>

"БХВ-Петербург", 191036, Санкт-Петербург, Гончарная ул., 20.

ISBN 978-5-9775-4068-1

© ООО "БХВ", 2020
© Оформление. ООО "БХВ-Петербург", 2020

Оглавление

Глава 1. Ознакомительная	1
1.1. О чем расскажет эта книга?	1
1.2. Электрическая безопасность — это важно!	4
1.3. Обратная связь	4
ЧАСТЬ I. НЕОБХОДИМОЕ ОБОРУДОВАНИЕ И ПРОГРАММЫ	5
Глава 2. Платформы Arduino и ESP8266	7
2.1. Платы контроллеров Arduino.....	7
2.1.1. Arduino Nano.....	8
2.1.2. Arduino Mini.....	9
2.1.3. Arduino Uno R3	10
2.1.4. Arduino Mega 2560 R3.....	11
2.2. Однокристалльная система ESP8266	12
2.2.1. Основные технические характеристики микросхемы ESP8266	12
2.2.2. Модули и платы ESP8266	13
2.2.3. Особенности эксплуатации ESP8266.....	15
2.3. Среда разработки Arduino IDE	16
2.3.1. Установка Arduino IDE	16
Установка для ОС Windows	16
Установка альтернативных версий IDE	16
Установка для ОС Linux.....	16
Установка для macOS	17
2.3.2. Установка библиотек Arduino	18
Автоматическая установка библиотеки	18
Установка библиотеки вручную	18
2.3.3. Установка расширения ESP8266 для Arduino IDE	19
2.3.4. Особенности программирования ESP8266	19
Порты и прерывания.....	19
Организация задержек.....	20
Поддержка интерфейсов I ² C и SPI	20

Глава 3. Интерфейсы обмена данными	21
3.1. Согласование логических уровней.....	22
3.2. Последовательный интерфейс UART	23
3.2.1. Конвертер интерфейсов USB-UART	24
3.2.2. Встроенный класс <i>Serial</i>	26
Проблемы и ошибки при работе с последовательным портом	27
3.2.3. Программные порты <i>SoftwareSerial</i>	27
3.2.4. Терминал последовательного порта <i>Termite</i>	28
3.3. Последовательная шина I ² C.....	29
3.3.1. Библиотека <i>Wire</i>	30
3.4. Последовательная шина 1-Wire.....	33
3.4.1. Библиотека <i>OneWire</i>	34
3.5. Последовательный интерфейс SPI	36
3.5.1. Библиотека <i>SPI</i>	37
3.6. Подключение по Bluetooth.....	39
3.6.1. Модули Bluetooth HC-06 и HC-05.....	40
Если модуль не отвечает на команду AT.....	42
Настройка модуля HC-05	42
3.6.2. Настройка подключения Bluetooth в Windows 10	43
Адаптер USB-Bluetooth	44
Включение адаптера Bluetooth и добавление устройства.....	45
3.6.3. Утилита Bluetooth Serial Terminal	48
Глава 4. Подключение Arduino к сети Интернет.....	50
4.1. Подключение к проводной сети Ethernet.....	50
4.1.1. Модуль Ethernet ENC28J60.....	51
4.1.2. Шилд Ethernet для Arduino Uno и Arduino Mega	55
4.2. Беспроводное подключение по Wi-Fi	58
4.2.1. Подключение контроллера ESP8266 к сети Wi-Fi.....	59
4.2.2. Сетевой шилд Dragino Yún	61
Питание шилда Dragino Yún	62
Функции кнопки сброса Dragino Yún.....	62
Добавление платы Dragino Yún в Arduino IDE.....	62
Подключение шилда к компьютеру для настройки	63
Определение типа базовой платы	65
Загрузка скетча через сеть из Arduino IDE	66
Подключение к Wi-Fi и вывод сообщений в консоль Linux	67
Глава 5. Визуализация данных.....	70
5.1. Построение графиков на компьютере.....	70
5.1.1. Встроенный плоттер Arduino IDE.....	71
5.1.2. Графопостроитель Serial Port Plotter.....	73
5.1.3. Графопостроитель FlexiPlot.....	75
Дополнительные настройки рабочего окна графика	77
Встроенный терминал	78

Рисование столбчатых диаграмм.....	78
Динамические диаграммы.....	80
5.1.4. Arduino плюс Excel — сбор и хранение данных.....	81
Установка расширения на компьютер.....	82
Запуск расширения.....	82
Рабочее окно и органы управления.....	83
Формат строки данных Arduino.....	84
Команды настройки и передачи данных.....	85
Специальные команды и управление.....	85
Рабочая книга Excel.....	87
Прочие команды.....	87
Демонстрационный скетч PLX-DAQ.....	88
5.2. Онлайнная приборная панель Adafruit IO.....	91
5.2.1. Учетная запись и потоки данных.....	92
Получение главного ключа.....	92
Создание групп и потоков.....	93
Настройка групп и потоков.....	95
5.2.2. Создание приборной панели.....	96
5.2.3. Установка библиотек.....	100
5.2.4. Подключение к сервису Adafruit IO по Wi-Fi.....	101
5.2.5. Подключение к сервису Adafruit IO по Ethernet.....	104
5.3. Дисплейные модули в проектах Arduino.....	107
5.3.1. Графические библиотеки Arduino.....	108
Графическое ядро Adafruit GFX.....	109
Система координат дисплея.....	109
Цвет пиксела.....	110
Графические примитивы.....	111
Поворот экрана.....	114
5.3.2. Дисплейный модуль OLED 128×64.....	115
Подключение OLED-дисплея к плате Arduino.....	116
Библиотека OLED-дисплея.....	117
5.3.3. Дисплейные модули TFT SPI 128×160 и 240×320.....	119
Подключение к плате Arduino дисплея TFT SPI 128×160.....	121
Назначение выводов дисплея TFT SPI 128×160.....	121
Пример кода с библиотекой <i>Adafruit ST7735</i>	122
Подключение к плате Arduino дисплея TFT SPI 240×320.....	123
Назначение выводов дисплея TFT SPI 240×320.....	124
Пример кода с библиотекой <i>Adafruit ILI9341</i>	124
5.3.4. Универсальный дисплейный шилд.....	125

ЧАСТЬ II. ТЕОРИЯ И ПРАКТИКА..... 127

Глава 6. Аналоговый оптический пульсометр..... 129

6.1. Принцип действия оптического пульсометра..... 129

6.2. Погрешности измерения ЧСС оптическим методом..... 132

6.3. Простой оптический сенсор.....	133
6.4. Пульсометр с OLED-дисплеем	138
6.5. Пульсометр с цветным TFT-дисплеем.....	141
6.6. Задания для самостоятельной работы.....	145
6.7. Опыты с пульсометром	145
6.7.1. Опыт с наложением жгута	145
6.7.2. Опыт с реакцией капилляров на температуру.....	146
6.7.3. Опыт с частотой и глубиной дыхания (рефлекс Геринга).....	146
6.7.4. Рефлекс Ортнера	146
6.7.5. Клиностатический рефлекс Даниелополу	146
6.7.6. Ортостатический рефлекс Превеля.....	146
6.7.7. Опыт с физической нагрузкой.....	147
6.8. Информация для любознательных	147
Глава 7. Шагомер на акселерометре ADXL335.....	148
7.1. Интегральный аналоговый акселерометр ADXL335.....	148
7.2. Подключение ADXL335 к плате Arduino Uno	149
7.3. Подключение шагомера к плоттеру FlexiPlot по Bluetooth.....	152
7.4. Простой шагомер с OLED-дисплеем	157
7.4.1. Определение средней длины шага.....	158
7.4.2. Определение расхода калорий	158
7.5. Задания для самостоятельной работы.....	161
Глава 8. Бесконтактное измерение температуры тела.....	162
8.1. Принцип действия и устройство инфракрасного пирометра.....	163
8.2. Подключение сенсора MLX90615 к плате Arduino	164
8.2.1. Проверка работоспособности сенсора.....	165
8.3. Особенности бесконтактного измерения температуры.....	166
8.4. Пирометр с OLED-дисплеем и настройкой КИ	168
8.5. Задания для самостоятельной работы.....	172
Глава 9. Измеритель интенсивности ультрафиолетового излучения	173
9.1. Диапазоны и свойства ультрафиолетового излучения	173
9.2. Устройство и принцип работы сенсора VEMML6075	175
9.3. Индекс УФ-излучения.....	177
9.4. Измеритель интенсивности УФ-излучения с OLED-дисплеем.....	178
9.4.1. Задания для самостоятельной работы	182
9.5. Онлайн-мониторинг солнечной активности.....	182
9.6. Опыты с измерителем УФ-излучения.....	188
9.6.1. Опыт с измерением УФ-проницаемости оконных стекол	189
9.6.2. Опыт с измерением УФ-проницаемости пластиков.....	190
9.6.3. Опыт с отражением УФ-излучения от песка и воды.....	190
9.6.4. Опыт с проверкой качества солнцезащитных очков.....	190
9.6.5. Опыт с проверкой качества солнцезащитных кремов.....	191

Глава 10. Измерение электрической активности кожи.....	192
10.1. Что такое электрическая активность кожи?	192
10.2. Модули измерителя ЭАК.....	193
10.3. Самодельные контакты сенсора ЭАК.....	194
10.4. Подключение модуля измерителя ЭАК к плате Arduino Uno R3	196
10.4.1. Электрическая безопасность — это важно!	196
10.4.2. Подключение к аналоговому порту	196
Пример чтения графика ЭАК.....	199
10.4.3. Подключение к шине SPI.....	201
Нужен ли TFT-дисплей для измерителя ЭАК?.....	204
10.5. Опыты с измерителем ЭАК	205
10.5.1. Опыт с глубоким дыханием.....	205
10.5.2. Опыт с мускульным напряжением.....	205
10.5.3. Опыт с воздействием температуры.....	205
10.5.4. Опыт с расположением электродов	206
10.5.5. Опыт с эффектом расслабления.....	206
10.5.6. Опыт с воздействием боли	206
10.5.7. Опыт с воздействием страха.....	206
10.5.8. Опыт с адаптацией и восстановлением	206
10.5.9. Опыт со словами и ассоциациями.....	207
10.5.10. Опыт с тревожным ожиданием	207
10.5.11. Опыт с тревожной реакцией.....	207
10.5.12. Опыт с мыслительным усилием.....	208
10.5.13. Опыт с загадыванием числа	208
10.5.14. Опыт с обусловленным откликом.....	208
Глава 11. Цифровой пульсоксиметр.....	209
11.1. Принцип работы и устройство сенсора MAX30102	210
11.1.1. Устройство микросхемы MAX30102	212
11.1.2. Основные технические характеристики пульсоксиметра MAX30102.....	213
11.2. Подключение MAX30102 к плате Arduino Uno	214
11.2.1. Установка библиотеки пульсоксиметра	215
11.2.2. Проверка и настройка пульсоксиметра	215
11.2.3. Встроенный датчик приближения.....	218
11.3. Измеритель частоты пульса и сатурации крови.....	220
Глава 12. Датчик общего качества воздуха	224
12.1. Устройство и характеристики сенсора CCS811	224
12.1.1. Основные технические характеристики датчика качества воздуха CCS811	226
12.1.2. Внимание: особенности эксплуатации сенсора	227
12.2. Модуль расширения CCS811	228
12.2.1. Назначение выводов модуля	228
12.2.2. Установка библиотеки Arduino	229
12.3. Подключение и проверка модуля	229

12.4. Монитор качества воздуха с OLED-дисплеем	231
12.4.1. Опыты с измерителем качества воздуха	234
12.4.2. Задание для самостоятельной работы над проектом.....	234
12.5. Онлайн-мониторинг качества воздуха.....	235
12.5.1. Проблема совместимости ESP8266 и CCS811.....	237
12.5.2. Задание для самостоятельной работы	241
Глава 13. Измерение пыльности и дымности воздуха.....	242
13.1. Датчик пыли SHARP GP2Y1014AU0F	243
13.1.1. Основные технические характеристики датчика пыли GP2Y1014AU0F	244
13.1.2. Воздушный поток через датчик	245
13.1.3. Управление подсветкой.....	245
13.1.4. Назначение выводов датчика пыли.....	246
13.1.5. Монтаж датчика пыли.....	246
13.2. Подключение датчика пыли к плате Arduino	247
13.3. Демонстрационный скетч и калибровка датчика.....	248
13.3.1. Начальное напряжение и калибровка	251
13.4. Определение наличия дыма в воздухе	251
13.4.1. Построение диаграммы в таблице Excel	254
13.4.2. Задания для самостоятельной работы	260
Глава 14. Шагомер на цифровом акселерометре LIS2DS12	261
14.1. Назначение и функции акселерометра LIS2DS12.....	262
14.1.1. Основные технические характеристики акселерометра LIS2DS12	262
14.1.2. Встроенные функции акселерометра LIS2DS12.....	262
14.2. Модуль акселерометра LIS2DS12	263
14.2.1. Назначение выводов модуля	264
14.3. Подключение и проверка модуля.....	264
14.4. Шагомер с OLED-дисплеем.....	268
14.5. Задания для самостоятельной работы.....	272
Глава 15. Трехточечный электрокардиограф	273
15.1. Фазы кардиограммы.....	274
15.2. Назначение и расположение электродов	275
15.3. Модуль электрокардиографа AD8232.....	276
15.3.1. Назначение выводов модуля ЭКГ.....	277
15.3.2. Электрическая безопасность — это важно!	279
15.3.3. Подключение кардиографа к Arduino Uno.....	279
15.4. Кардиомонитор с цветным TFT-дисплеем	283
15.5. Задания для самостоятельной работы.....	287
Глава 16. Измеритель скорости пульсовой волны.....	288
16.1. Схема и макет измерительного устройства	290
16.2. Измерение скорости пульсовой волны	296
16.3. Задания для самостоятельной работы.....	297

Глава 17. Измерение биопотенциала мышц	298
17.1. Изготовление и подключение электродов для миографии	299
17.2. Измерение биопотенциала бицепса	300
17.3. Миография в прикладных проектах	303
17.3.1. Переменный волнообразный сигнал	304
17.3.2. Нестабильный сигнал с помехами	304
17.3.3. Необходимость значительных аппаратных и вычислительных ресурсов	304
ПРИЛОЖЕНИЯ	307
Приложение 1. Содержимое электронного архива	309
Приложение 2. Источники питания для проектов Arduino.....	312
П2.1. Автономное питание макетов и устройств	313
П2.1.1. Источник питания на одной батарее AAA	314
П2.1.2. Заряжаемый источник питания на литий-полимерной батарее	315
Модуль повышающего преобразователя	316
Модуль зарядного устройства Li-Po.....	316
Литий-полимерный аккумулятор.....	317
Схема подключения и монтаж	318
Приложение 3. Коэффициенты излучения поверхности различных материалов	320
Предметный указатель	323

ГЛАВА 1



Ознакомительная

Уважаемые читатели!

Эта книга посвящена, пожалуй, самым ценным для нас объектам исследований — человеческому организму и окружающей среде, в которой он обитает.

Сегодня можно без труда и по доступной цене приобрести современные интегральные сенсоры, которые совсем недавно применялись только в сложном профессиональном оборудовании. Достаточно подключить эти сенсоры к плате Arduino или ESP8266 — и можно проводить увлекательные и безопасные опыты с измерением параметров человеческого организма и окружающей среды!

Мы постарались подготовить для вас книгу, которая станет полным руководством по разработке любительских проектов: начиная от выбора подходящей платы Arduino и заканчивая средствами визуализации результатов измерений. Если вы начинающий любитель или разработчик, то материал глав *первой части книги* послужит вам настольным справочником при работе над различными проектами. Опытные же разработчики найдут во *второй части книги* полезные примеры использования современных сенсоров и советы по решению проблем.

При подготовке описаний проектов автор намеренно оставлял читателям пространство для самостоятельного творчества и размышлений. Большинство описаний дополнено заданиями для самостоятельной работы и перенем опытов.

1.1. О чем расскажет эта книга?

Первая часть книги — подготовительная. Опытные пользователи могут пропустить отдельные главы этой части и обращаться к ним по мере необходимости.

- ◆ Во *второй главе* мы знакомим читателя с наиболее популярными платами платформ Arduino и ESP8266 и рассказываем, как установить среду разработки Arduino IDE и добавить в нее различные библиотеки и расширения.
- ◆ В *третьей главе* мы рассказываем о различных протоколах и шинах обмена данными. Вы будете подключать к плате контроллера сенсоры и модули расширения, которые используют разные шины данных: I²C, SPI, 1-Wire. Следует

знать, как работают эти шины и как реализовать обмен данными в коде программы. Отдельно говорится о настройке беспроводного соединения с компьютером по Bluetooth.

- ◆ В *четвертой главе* мы говорим о различных способах подключения плат Arduino и ESP8266 к сети Интернет для передачи данных на облачный сервер или в другие онлайн-сервисы.
- ◆ В *пятой главе* рассказано о том, как простыми средствами визуализировать данные. Это важный этап работы над проектом! На самом деле исследования и опыты не заканчиваются после получения данных от сенсора. Сами по себе эти данные ничего не значат до тех пор, пока они не обработаны и не представлены в доступном для человека виде. Зачастую получателем данных является неподготовленный пользователь, и ему нужно показать понятные графики, шкалы и числа.

Вторая часть книги содержит описание теоретических основ предметов измерений, практических проектов и опытов с макетами измерительных устройств.

- ◆ В *главе 6* рассказано о физических принципах измерения частоты пульса оптическим методом и приведено описание простого пульсометра на основе аналогового сенсора. Читатель изготовит макет портативного пульсометра с OLED-дисплеем, а также более продвинутый пульсометр с цветным дисплеем.
- ◆ В *главе 7* на примере простого аналогового шагомера рассказано о том, как разрабатываются устройства мониторинга. Читатель научится анализировать «сырые» результаты измерений и на их основе создавать программу для обработки полученных данных. В результате получится простой и надежный шагомер с OLED-дисплеем.
- ◆ В *главе 8* мы изучаем физические принципы бесконтактного измерения температуры и создаем простой бесконтактный термометр-пирометр для дистанционного измерения температуры тела и окружающих предметов.
- ◆ В *главе 9* мы продолжаем заниматься измерением интенсивности оптических излучений. Мы обсудим, какое влияние на человеческий организм оказывает ультрафиолетовое излучение с различной длиной волны, и построим измеритель интенсивности излучения в диапазонах А и В. Затем мы подключим измеритель к сети Интернет и создадим онлайн-приборную панель для дистанционного мониторинга интенсивности излучения.
- ◆ В *главе 10* мы изучаем измерение электрической активности кожи и учимся расшифровывать результаты. Это очень увлекательная тема — ведь именно измеритель электрической активности кожи лежит в основе знаменитого и слегка таинственного «детектора лжи». Описание проекта дополнено большим перечнем опытов — включая эксперименты, которые придуманы профессиональными психологами. При наличии определенного навыка исследователь-любитель сможет по форме графика на дисплее компьютера угадывать слова, предметы и события, о которых думает подопытный.
- ◆ В *главе 11* мы возвращаемся к теме оптической пульсометрии, но теперь будем измерять уровень насыщения крови кислородом. Как обычно, перед началом

работы с макетом познакомимся с физическими принципами исследуемого явления. В отличие от проекта из *главы 6*, в этом проекте использован современный интегральный сенсор для профессионального оборудования.

- ◆ В *главе 12* рассказано об измерении общего качества воздуха и углеродного эквивалента. Это универсальные параметры, которые характеризуют общий уровень загрязненности воздуха органическими примесями и степень вреда окружающей среде из-за парникового эффекта. Прибор, макет которого мы построим, незаменим для контроля безопасности вдыхаемого воздуха, особенно в производственных помещениях и вблизи химических предприятий. Для этого проекта также предусмотрено подключение к сети Интернет и удаленный мониторинг через онлайн-панель.
- ◆ В *главе 13* мы продолжаем исследовать качество воздуха и приступаем к работе над проектом измерителя концентрации частиц пыли и дыма в окружающем нас пространстве. В описании этого проекта вновь приведен пример разработки прошивки устройства на основе анализа «сырых» данных.
- ◆ В *главе 14* мы возвращаемся к проекту шагомера, но за основу берем современный интегральный акселерометр со встроенным сигнальным процессором. Этот сенсор оснащен встроенными функциями цифровой обработки сигнала и подсчета количества шагов. Внешнему контроллеру достаточно лишь прочесть готовое значение из регистра счетчика и отобразить его на дисплее. Проект наглядно демонстрирует возможности современных интегральных сенсоров.
- ◆ В *главе 15* мы изучаем электрокардиографию. Благодаря наличию специальной микросхемы электрокардиографа, теперь можно даже в домашних условиях изготовить простой кардиомонитор на основе контроллера Arduino и недорогого модуля расширения.
- ◆ В *главе 16* мы объединяем измеритель частоты пульса из *главы 6* с кардиомонитором из *главы 15* и получаем измеритель скорости пульсовой волны. Это важный параметр, характеризующий состояние сердечно-сосудистой системы человека. Обычно для измерений применяется сложное лабораторное оборудование, но мы наглядно продемонстрируем, что базовые измерения можно выполнить даже в домашних условиях при помощи платформы Arduino и модулей расширения.
- ◆ В *главе 17* мы изучаем технологию измерения биологических потенциалов, возникающих при работе мышц человеческого тела. В качестве измерительного устройства мы используем модуль электрокардиографа, с которым работали в *главе 15*. Несмотря на кажущуюся простоту проекта, вычислительные требования к проекту лежат на пределе возможностей платформы Arduino, поэтому мы можем выполнить только простые измерения, а для дальнейшего развития проекта необходимо перейти на более мощную платформу.

Исходные коды программ (листинги), приведенные в книге, можно найти в сопровождающем ее электронном архиве (см. *приложение 1*). Электронный архив выложен на FTP-сервер издательства «БХВ-Петербург» по адресу: <ftp://ftp.bhv.ru/9785977540681.zip>. Ссылка на архив доступна и со страницы книги на сайте www.bhv.ru.

В *приложении 2* приведены данные об источниках питания для проектов Arduino, а в *приложении 3* — таблица коэффициентов излучения поверхности различных материалов.

1.2. Электрическая безопасность — это важно!

Все устройства и опыты, о которых рассказано в книге, абсолютно безопасны для здоровья. Напряжение питания всех устройств не превышает 5 вольт. Поэтому работать с макетами устройств могут даже школьники младшего возраста.

Тем не менее при работе с устройствами и макетами, подключенными к сетевому источнику питания, в том числе при подключении к стационарному компьютеру по USB, существует некоторая опасность поражения электрическим током. Это может случиться при определенном стечении обстоятельств, связанных с неисправностью заземления в здании или повреждением изоляции в источнике питания.

Еще опаснее ситуация, когда из-за ошибок монтажа электропроводки линия «фазы» в розетке оказывается подключенной к заземляющему выводу розетки, а устройство защитного отключения в домашнем электрощитке отсутствует. В этом случае на корпусе компьютера и на общем проводе макета может присутствовать полное напряжение 220 вольт относительно заземленных предметов.

Работая над проектами из этой книги, вы будете подключать к своему телу измерительные электроды, которые обеспечивают очень хороший контакт с кожей. Особенно важно, что, экспериментируя с электрокардиографом, вы будете подключать электроды непосредственно в области сердца. В такой ситуации нельзя полагаться на исправность заземления и правильность монтажа электросети в квартире или лаборатории! Необходимо обеспечить надежную гальваническую развязку между макетом устройства и компьютером. Разумеется, если вы подключаете макет устройства к ноутбуку с батарейным питанием, дополнительная развязка не нужна. Но если вы работаете на настольном компьютере с сетевым питанием, то для обмена данными с компьютером используйте беспроводное соединение по Bluetooth, а устройство питайте от батарейного источника питания.

Итак, вот основные правила электробезопасности при работе с устройствами, электроды которых подключаются непосредственно к телу человека:

- 1. Организуйте беспроводное соединение по Bluetooth.**
- 2. Используйте автономное питание от батарейного источника.**
- 3. Перед подключением провода USB для записи прошивки в контроллер Arduino отключите электроды от тела!**

1.3. Обратная связь

Автор и редакторы издательства будут признательны за ваши отзывы, советы и замечания по содержанию этой книги. Мы обязательно учтем их при подготовке следующих изданий. Пишите по адресу mail@bhv.ru или непосредственно автору по адресу valeriy.yatsenkov@gmail.com.



ЧАСТЬ I

Необходимое оборудование и программы

- Глава 2. Платформы Arduino и ESP8266
- Глава 3. Интерфейсы обмена данными
- Глава 4. Подключение Arduino к сети Интернет
- Глава 5. Визуализация данных

ГЛАВА 2



Платформы Arduino и ESP8266

Как вы думаете, почему относительно несложные и маломощные микроконтроллерные платы Arduino и ESP8266 приобрели невероятную всемирную популярность?

Разумеется, низкая цена имеет свое значение. Но все-таки основная причина популярности состоит в том, что центральная плата — мы будем называть ее «*контроллер*» — является универсальным вычислителем, к которому можно без труда подключить *модули расширения*, которые выполняют различную работу.

Некоторые модули расширения устроены сложнее, чем контроллер. Но даже в этом случае начинающему разработчику достаточно потратить несколько минут на соединение модуля с контроллером, чтобы получить работающее устройство. Затем можно приступать к увлекательным экспериментам, попутно изучая принцип работы различных сенсоров и прочих периферийных устройств.

Если собрать вместе платы, модули, описания, исходные коды, язык и среду программирования, то получится *техническая платформа*. На основе платформы разработчики создают свои проекты и могут быть уверены в совместимости модулей и программ внутри одной платформы.

Среди начинающих разработчиков особенно популярны две платформы: Arduino на основе микроконтроллеров Atmel и ESP8266 на основе системы на чипе ESP8266 со встроенным блоком Wi-Fi. Эти платформы во многом взаимно совместимы на уровне исходного кода и модулей расширения.

О работе с современными модулями расширения рассказано в следующих главах книги. А сейчас приглашаем новых пользователей Arduino ближе познакомиться со средой разработки Arduino IDE и аппаратной частью платформ Arduino и ESP8266.

2.1. Платы контроллеров Arduino

Для большинства любительских проектов начального и среднего уровня, включая проекты из этой книги, достаточно приобрести одну из трех недорогих плат: Arduino Nano, Arduino Uno или Arduino Mini. Все три платы реализованы на микроконтроллере ATmega328. При использовании любой из этих плат нет необходимо-

сти вносить изменения в исходные коды программ или в схему устройства — достаточно подключить соединительные провода к выводам контроллера с нужными наименованиями в соответствии со схемой проекта.

Если ресурсов микроконтроллера ATmega328 не хватает для вашего проекта, воспользуйтесь платой Arduino Mega 2560 R3, созданной на основе более мощного микроконтроллера ATmega2560. В этом случае могут потребоваться незначительные изменения исходного кода скетчей.

Схемы расположения выводов всех упомянутых в книге плат несложно найти в Интернете, выполнив в справочной системе соответствующий запрос.

2.1.1. Arduino Nano

Контроллер Arduino Nano (рис. 2.1) — это «рабочая лошадка» любителей электроники. С обратной стороны платы установлены линейный стабилизатор +5 В и конвертер USB-UART. Микросхема конвертера содержит встроенный стабилизатор +3,3 В. Этим напряжением можно питать внешние трехвольтовые модули с током потребления до 50 мА. Преимуществами этой версии являются компактность и возможность установки непосредственно в безопасную макетную плату.

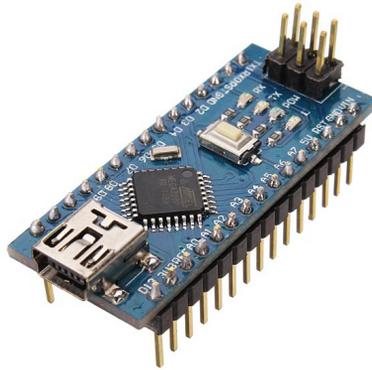


Рис. 2.1. Плата Arduino Nano

Основные технические характеристики платы Arduino Nano

- ◆ Контроллер: ATmega328P
- ◆ Рабочее напряжение: 5 В
- ◆ Внешнее питание: 7...12 В
- ◆ Тактовая частота: 16 МГц
- ◆ Цифровые порты: 14 (из них 6 ШИМ)
- ◆ Аналоговые входы: 8 (10 битов АЦП)
- ◆ Предельный ток порта: 40 мА
- ◆ Память программ: 32 Кбайт (бутлодер 0,5 Кбайт)

- ◆ Память SRAM (ОЗУ): 2 Кбайт
- ◆ Память EEPROM: 1 Кбайт
- ◆ Размеры: 45×18 мм
- ◆ Вес: 5 г



ВНИМАНИЕ!

Внешнее нерегулируемое напряжение питания в диапазоне +7...+12 В следует подавать только на вход **VIN!** При этом на выводах **5V** и **3.3V** появятся соответствующие стабилизированные напряжения, которые можно использовать для питания внешних элементов схемы с суммарным потребляемым током до 50 мА. Стабилизированное внешнее напряжение +5 В можно подавать напрямую на вывод **5V**.

2.1.2. Arduino Mini

Arduino Mini (рис. 2.2) — это вариант платы с минимальным количеством внешних компонентов. Она не содержит конвертер USB-UART, поэтому для соединения с компьютером и программирования понадобится качественный внешний конвертер (см. *разд. 3.2.1* и *4.1.1*). Официальная плата выпускается в двух вариантах рабочего напряжения и тактовой частоты: 5 В (16 МГц) и 3,3 В (8 МГц). От рабочего напряжения зависит микросхема стабилизатора, установленная на плате. Китайские производители выпускают несколько вариантов плат, которые различаются расположением выводов.

Плату Arduino Mini удобно монтировать непосредственно на поверхность печатной платы готового устройства. В сочетании с низкой ценой это делает плату незаменимым компонентом многих любительских и мелкосерийных проектов.

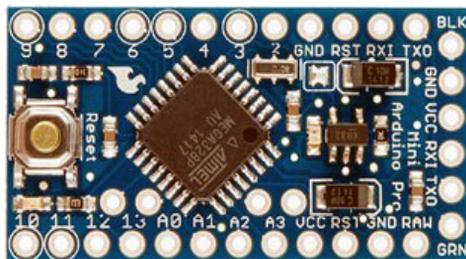


Рис. 2.2. Плата Arduino Pro Mini

Основные технические характеристики платы Arduino Mini

- ◆ Контроллер: ATmega328P
- ◆ Рабочее напряжение: 5 В / 3,3 В
- ◆ Внешнее питание: 7...12 В (RAW)
- ◆ Тактовая частота: 16 МГц / 8 МГц
- ◆ Цифровые порты: 14 (из них 6 ШИМ)

- ◆ Аналоговые входы: 6 (10 битов АЦП)
- ◆ Предельный ток порта: 40 мА
- ◆ Память программ: 32 Кбайт (бутлоадер 0,5 Кбайт)
- ◆ Память SRAM (ОЗУ): 2 Кбайт
- ◆ Память EEPROM: 1 Кбайт
- ◆ Размеры: 33,6×17,8 мм
- ◆ Вес: 3,8 г



ВНИМАНИЕ!

Внешнее напряжение питания в диапазоне +7...+12 В следует подавать только на вход RAW! Вывод VCC подключен непосредственно к линии питания микроконтроллера. Допускается подавать на него только стабилизированное напряжение +5 В или +3,3 В — в зависимости от модификации платы. Будьте внимательны при покупке платы и подключении внешнего конвертера USB-UART.

2.1.3. Arduino Uno R3

Плата Arduino Uno R3 (рис. 2.3) является третьей версией (Revision 3) самой популярной платы Arduino Uno. Основное отличие платы Arduino Uno от Nano заключается в монтажном факторе. Если Nano вставляют в макетную плату, то Uno сама является макетной платой. Для нее выпускают множество расширений (шилдов), которые совпадают по форме и вставляются в разъемы поверх основной платы. Кроме того, на плате Uno есть дополнительный разъем для внешнего источника питания напряжением 7...12 В, электролитические конденсаторы в цепи питания и более сложная схема коммутации питания с автоматическим переключением между источниками. На плате установлен самовосстанавливающийся полимерный предохранитель, дополнительно защищающий порт USB компьютера от перегрузки.

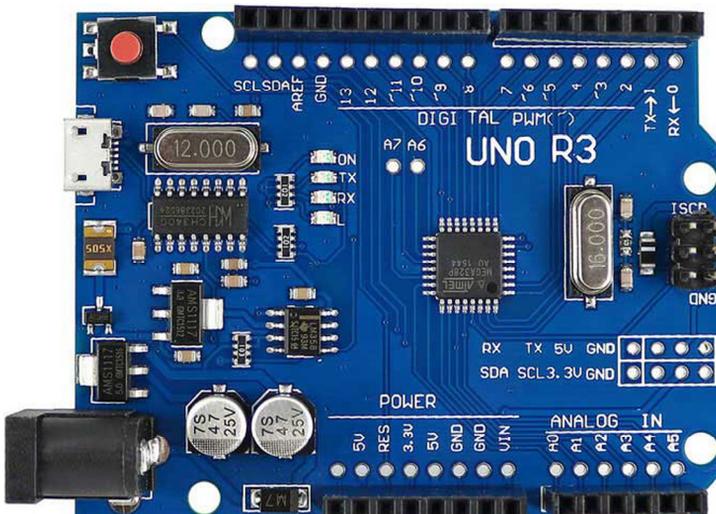


Рис. 2.3. Плата Arduino Uno R3

Основные технические характеристики платы Arduino Uno

- ◆ Контроллер: ATmega328P
- ◆ Рабочее напряжение: 5 В
- ◆ Внешнее питание: 7...12 В
- ◆ Тактовая частота: 16 МГц
- ◆ Цифровые порты: 14 (из них 6 ШИМ)
- ◆ Аналоговые входы: 6 (10 битов АЦП)
- ◆ Предельный ток порта: 40 мА
- ◆ Память программ: 32 Кбайт (бутлоадер 0,5 Кбайт)
- ◆ Память SRAM (ОЗУ): 2 Кбайт
- ◆ Память EEPROM: 1 Кбайт
- ◆ Размеры: 68,6×53,4 мм
- ◆ Вес: 25 г

2.1.4. Arduino Mega 2560 R3

Плата Arduino Mega 2560 R3 (рис. 2.4) реализована на микроконтроллере ATmega2560. По сравнению с предыдущими платами, Arduino Mega 2560 имеет больше цифровых и аналоговых портов, два аппаратных последовательных порта, увеличенный объем памяти. На плате установлен самовосстанавливающийся полимерный предохранитель, дополнительно защищающий порт USB компьютера от перегрузки. Благодаря большому количеству портов, к базовой плате можно подключать быстродействующие цветные дисплеи с параллельной восьмибитной шиной, при этом остается достаточно выводов для работы с другими периферийными устройствами.

Разводка выводов платы выполнена так, чтобы к ней можно было подключить большинство модулей расширения, предназначенных для Arduino Uno R3.

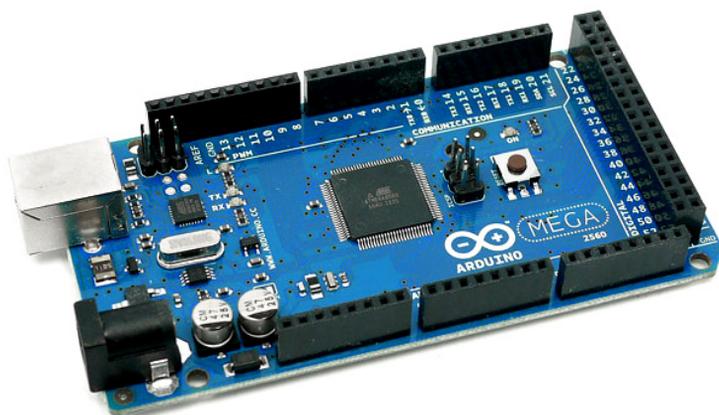


Рис. 2.4. Плата Arduino Mega 2560

Основные технические характеристики платы Arduino Mega 2560

- ◆ Контроллер: ATmega2560
- ◆ Рабочее напряжение: 5 В
- ◆ Внешнее питание: 7...12 В
- ◆ Тактовая частота: 16 МГц
- ◆ Цифровые порты: 54 (из них 15 ШИМ)
- ◆ Аналоговые входы: 16
- ◆ Предельный ток порта: 40 мА
- ◆ Память программ: 256 Кбайт (бутлоадер 8 Кбайт)
- ◆ Память SRAM (ОЗУ): 8 Кбайт
- ◆ Память EEPROM: 4 Кбайт
- ◆ Размеры: 101,5×53,3 мм
- ◆ Вес: 37 г

2.2. Однокристалльная система ESP8266

Однокристалльная система, или система на кристалле (SoC, System-on-Chip), — это микросхема, объединяющая на одном кристалле несколько модулей с принципиально различными функциями и выполняющая задачи целого устройства. Как правило, такая система содержит микропроцессорное ядро, встроенную память, внутренние шины данных и специфические функциональные модули.

Микросхема ESP8266 является примером SoC, потому что состоит из микропроцессора и функционального модуля Wi-Fi, выполненных на одном кристалле. Таким образом, на одной микросхеме можно выполнить законченное устройство, подключаемое к беспроводной сети Интернет.

Изначально SoC ESP8266 разрабатывалась как микросхема для модулей Wi-Fi, подключаемых к другим микроконтроллерам. На практике оказалось, что возможности встроенного процессора настолько велики, что в большинстве случаев позволяют обойтись вообще без внешнего микроконтроллера. Кроме разработки программ в привычной среде Arduino IDE, для ESP8266 можно писать программы на языках Lua, MicroPython и Smart-JS и даже запускать их в среде простой операционной системы.

2.2.1. Основные технические характеристики микросхемы ESP8266

- ◆ Процессор: однопядерный Tensilica L106 с тактовой частотой до 160 МГц
- ◆ Поддерживаемые стандарты WI-FI: 802.11 b / g / n
- ◆ Поддерживаемы типы шифрования: WEP, WPA, WPA2

- ◆ Поддерживаемые режимы работы: Клиент (STA), Точка доступа (AP), Клиент+Точка доступа (STA+AP)
- ◆ Количество одновременных соединений TCP: 5
- ◆ Напряжение питания: 1,7...3,6 В
- ◆ Потребляемый ток: до 215 мА (в зависимости от режима работы)
- ◆ Количество GPIO: 17 (количество доступных зависит от модификации модуля)
- ◆ Интерфейсы: ADC 10 битов, I²C, UART, SPI, PWM
- ◆ Внешняя Flash-память: от 512 Кбайт до 16 Мбайт
- ◆ Объем памяти данных (EEPROM): 80 Кбайт
- ◆ Объем памяти команд (RAM): 64 Кбайт

2.2.2. Модули и платы ESP8266

Микросхема ESP8266 выполнена в малогабаритном корпусе и в любительской практике используется только в виде готовых модулей с необходимыми внешними компонентами схемы (рис. 2.5). Но даже в таком виде модули редко применяются в любительских проектах, потому что их неудобно подключать к макетной плате.

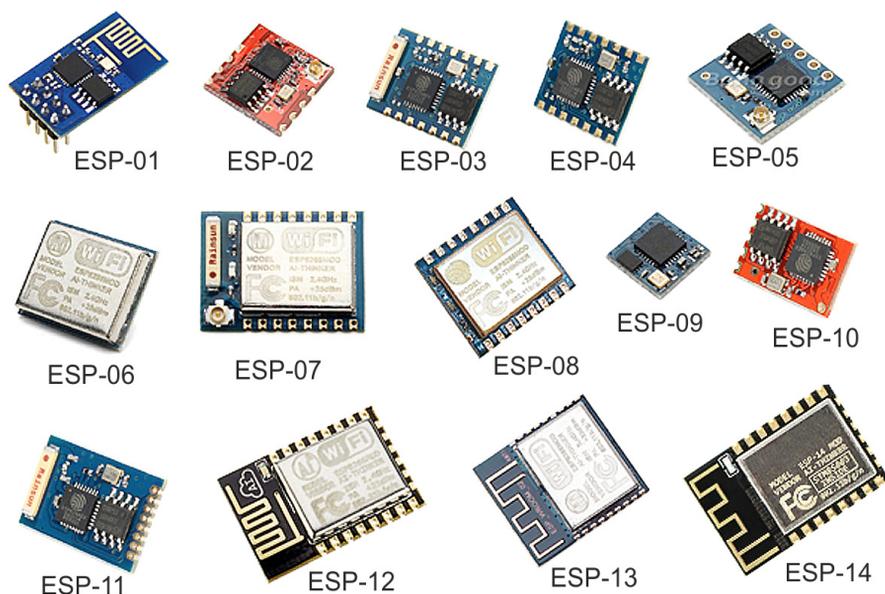


Рис. 2.5. Стандартные модули на основе ESP8266

Производители оборудования для разработчиков-любителей обычно устанавливают стандартный модуль ESP-12 на материнскую плату, оснащенную разъемами, стабилизатором питания, конвертером USB-UART и прочими вспомогательными компонентами. Сейчас на рынке представлено бесчисленное множество конструк-

ций плат ESP8266, и начинающему разработчику сложно выбрать оптимальный вариант.

Мы рекомендуем приобрести одну из двух наиболее удобных и популярных плат: WeMos D1 (рис. 2.6) или NodeMCU (рис. 2.7).

Плата WeMos D1 по форме и расположению выводов максимально совместима с платой Arduino Uno R3, поэтому к ней можно подключить большинство модулей расширения Arduino.

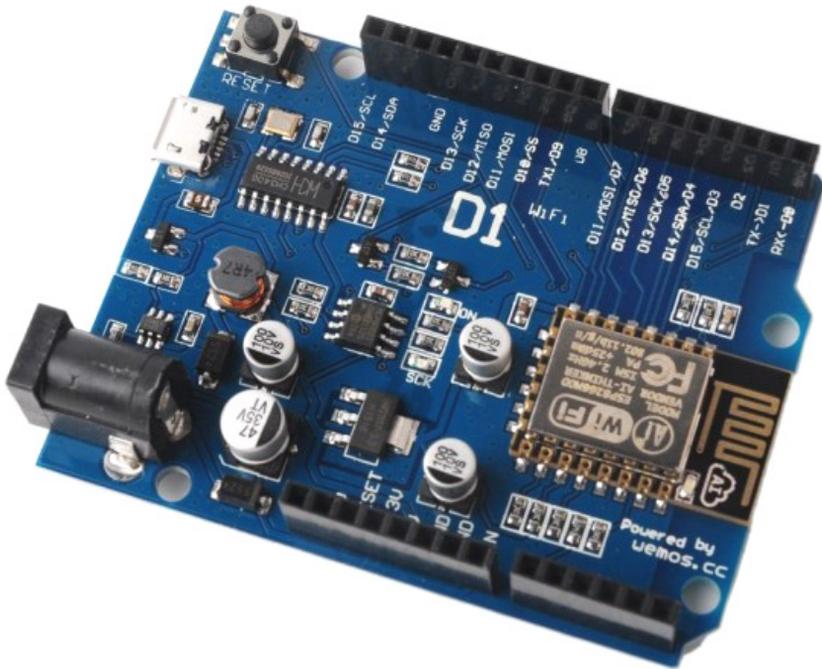


Рис. 2.6. Плата WeMos D1

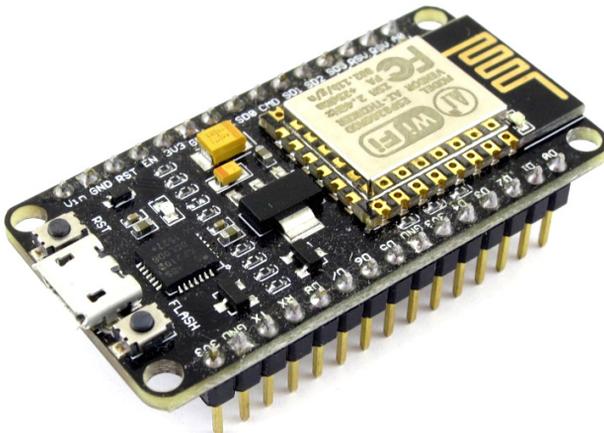


Рис. 2.7. Плата NodeMCU

Плата NodeMCU изначально была разработана для открытого проекта NodeMCU, но, благодаря удачной конструкции, стала очень популярной среди любителей. Эта плата предназначена для установки в безопасную макетную плату и тоже содержит все необходимое для автономной работы.

2.2.3. Особенности эксплуатации ESP8266

При разработке устройств на основе ESP8266 — особенно при переносе готовых проектов с платформы Arduino — необходимо учитывать некоторые специфические особенности микросхемы.

- ◆ **Максимальная нагрузка на порт.** В отличие от плат Arduino, допускающих нагрузку на порт до 40 мА, что позволяет не задумываясь подключать к ним малогабаритные сверхъяркие светодиоды и прочую подобную нагрузку, порты ESP8266 выдерживают ток не более 14 мА, поэтому их можно перегрузить даже обычным светодиодом. Следует тщательно ограничивать нагрузку на порты плат ESP8266 или применять ключевые транзисторы.
- ◆ **Согласование логических уровней.** Согласно спецификации производителя, только линии UART (Tx, Rx) допускают работу с напряжениями пятивольтовой логики. Подача на остальные выводы логических уровней, превышающих напряжение питания микросхемы, с большой вероятностью выведет ее из строя. При совместном использовании модулей с разным напряжением питания необходимо применять согласование логических уровней (см. *разд. 4.1*).
- ◆ **Режим программирования.** Для перевода микросхемы в режим записи прошивки необходимо установить низкий уровень на входе **GPIO0** и кратковременно подать низкий уровень на вход сброса **RESET** (либо держать вывод **GPIO0** соединенным с общим проводом в момент включения питания). Микросхема перейдет в режим программирования и коротко мигнет встроенным светодиодом, подключенным к выводу **GPIO2**. После этого можно вернуть высокий уровень на вход **GPIO0**. На большинстве плат, включая WeMos D1 и NodeMCU, смонтирована схема из двух транзисторов для автоматического перехода в режим программирования по сигналам DTR и RTS от конвертера USB-UART. Если на вашей плате такой схемы нет, или вы используете стандартный модуль ESP-xx (см. рис. 2.5), то вам придется вручную переводить микросхему в режим записи.
- ◆ **Температурный режим.** В режиме максимальной мощности пиковый потребляемый ток может достигать 240 мА, а корпус микросхемы в условиях недостаточного отведения тепла — нагреваться до 80 °С. Несмотря на то, что микросхема может работать при температуре до +125 °С, перегрев модуля может привести к росту погрешности синтезатора частоты трансмиттера Wi-Fi, а циклические тепловые перегрузки ускоряют выход модуля из строя из-за нарушения пайки.
- ◆ **Требования к монтажу.** Микросхема чувствительна к наводкам сигнала на линии портов от собственной антенны Wi-Fi. Длинные соединительные проводники между платой на основе ESP8266 и внешними устройствами могут привести

к нестабильной работе, что выражается в зависании и самопроизвольной перезагрузке микросхемы.

2.3. Среда разработки Arduino IDE

Важную роль в успехе платформы Arduino сыграла интегрированная среда разработки Arduino IDE (Integrated Development Environment). По аналогии с аппаратной частью, среда Arduino IDE тоже устроена по модульному принципу — в среду разработки легко добавляются описания новых контроллеров и библиотеки сторонних разработчиков. Основные библиотеки встроены в среду по умолчанию. Для программирования применяется адаптированный вариант языка Processing.

Среда состоит из редактора исходного кода с подсветкой синтаксиса, компилятора AVR-GCC, средства загрузки прошивок AVRdude, монитора последовательного порта и плоттера для рисования графиков. Пользователь может оставаться в среде Arduino IDE в течение всего цикла разработки — от первой буквы кода до отладки готового устройства.

2.3.1. Установка Arduino IDE

Скачайте свежую версию Arduino IDE для своей операционной системы с сайта разработчика <http://www.arduino.cc/en/Main/Software>.

Установка для ОС Windows

Пользователям Windows для первой установки на компьютер рекомендуется выбрать вариант Windows Installer. Кроме автоматической распаковки, он создаст нужные записи в реестре, ассоциирует файлы, имеющие расширение `ino`, с редактором Arduino IDE, а также создаст специальный рабочий каталог для хранения файла конфигурации и служебных файлов.

Установка альтернативных версий IDE

Установщик Windows Installer проверяет наличие других версий Arduino IDE и предлагает их удалить. Поэтому для установки старой версии (или нескольких разных версий) необходимо скачать ZIP-архив, а не установщик, и развернуть его в отдельный каталог. Затем можно создать на рабочем столе ярлыки для запуска с указанием номера версии.

При совместной работе над сложными проектами или тестировании чужих проектов рекомендуется использовать одинаковые версии IDE. При использовании различающихся версий IDE возможна ситуация, когда скомпилированные прошивки будут отличаться по размеру, быстрдействию или наличию скрытых ошибок.

Установка для ОС Linux

Раньше способ установки Arduino IDE зависел от версии Linux, а для пользователей Raspberry Pi вообще существовала отдельная инструкция. Новые сборки IDE со-

держат все необходимые скрипты для автоматической установки среды независимо от версии ОС. Вам нужно лишь правильно определить тип сборки ОС Linux, которую вы используете: 32bit, 64bit или ARM — и скачать подходящую версию установочного пакета по адресу: <http://www.arduino.cc/en/Main/Software>. Версия указана в названии файла.

Распакуйте архив в любой удобный вам каталог. Программа будет запускаться на выполнение из этого каталога. В процессе распаковки будет создан каталог с именем `arduino-1.8.x`. Найдите в программе файл `install.sh` и щелкните на нем правой кнопкой мыши. В контекстном меню выберите пункт **Run in Terminal**. После короткого процесса установки на рабочем столе появится новый значок.

Если в контекстном меню нет опции запуска скрипта, откройте терминал и перейдите в каталог `arduino-1.8.x` (в нашем примере — версия IDE 1.8.7):

```
cd arduino-1.8.7
./install.sh
```

Теперь среда Arduino IDE готова к запуску. Может случиться так, что при попытке записи прошивки в плату вы получаете сообщение об ошибке: **Error opening serial port...** Это означает, что нужно задать разрешения для работы с портом. Подключите свою плату к компьютеру. Введите в терминале команду:

```
ls -l /dev/tty*
```

В ответ вы получите строку наподобие такой:

```
crw-rw---- 1 root dialout 188, Feb 16 13:22 ttyUSB1
```

Строк может быть несколько, но нас интересуют только те, в которых содержится слово `dialout`. Они могут содержать как текст `ttyUSBx`, так и `ttyACMx`, где `x` — это номер порта, к которому подключена плата. Дата и время в строке указывают на время, когда вы подключили плату. Запомните наименование и номер порта и укажите его в настройках Arduino IDE. Теперь надо включить нового пользователя (себя) в группу `dialout`:

```
sudo usermod -a -G dialout <ваше имя пользователя>
```

Изменения начнут действовать только после выхода из системы и повторного входа. Далее вы сможете загружать прошивки как обычно.

Разумеется, для большинства версий Linux среду Arduino IDE можно установить из репозитория. Но при этом будет установлена далеко не самая последняя версия программы. Кроме того, вы будете лишены возможности выбора версии в случае особых требований к проекту.

Установка для macOS

Скачайте архив нужной версии Arduino IDE со страницы загрузки. Если вы используете браузер Safari, то архив будет развернут автоматически. При использовании иного браузера архив придется развернуть самостоятельно. Скопируйте приложение Arduino в каталог Applications (Программы). Значок Arduino появится в панели запуска программ LaunchPad. Вы можете также скопировать приложение в любой другой каталог и запускать оттуда.

2.3.2. Установка библиотек Arduino

Библиотека Arduino — это набор стандартных функций для решения определенных прикладных задач. Как правило, это массово востребованные задачи: работа с датчиками, дисплеями, микросхемами внешней памяти, поддержка протоколов обмена. Чтобы не тратить зря время и силы на повторное написание уже существующих программ, имеет смысл воспользоваться готовыми решениями. Обычно они многократно испытаны и отлажены членами творческого сообщества Arduino. Библиотека может содержать примеры использования.

Многие библиотеки написаны специалистами компаний, выпускающих электронные модули и компоненты. Наиболее универсальные и востребованные библиотеки входят в состав среды Arduino IDE. Но рано или поздно вы столкнетесь с необходимостью добавить библиотеки, которых нет в стандартном наборе.

Автоматическая установка библиотеки

Библиотека может присутствовать во встроенном списке среды Arduino IDE, но требовать установку. Поэтому, прежде всего, обратимся к встроенному списку библиотек. Перейдите в пункт меню **Скетч | Подключить библиотеку | Управлять библиотеками** и внимательно изучите список или воспользуйтесь поиском. Поиск работает по фрагменту слова из названия или описания библиотеки. Найдя нужную библиотеку, выделите ее в списке и нажмите кнопку **Установка**. Установленные библиотеки помечаются словом **INSTALLED** возле заголовка.

Если библиотека отсутствует во встроенном списке, то разработчики обычно распространяют ее в виде ZIP-архива. Скачайте архив библиотеки и не распаковывайте его. Если имя архива содержит в названии фрагмент `-master`, удалите этот фрагмент.

Откройте пункт меню **Скетч | Подключить библиотеку | Добавить ZIP-библиотеку**. Выберите архив библиотеки в диалоговом окне. Архив будет автоматически распакован в каталог `libraries` внутри вашего рабочего каталога Arduino IDE. Теперь снова откройте меню **Скетч | Подключить библиотеку**. Название новой библиотеки должно появиться в списке библиотек с пометкой **INSTALLED**. Использование библиотеки можно начинать сразу после установки.

Установка библиотеки вручную

При автоматической установке библиотеки разворачиваются в каталог `My Documents\Arduino\libraries\` (Windows) или `Documents/Arduino/libraries/` (macOS, Linux) и доступны для всех версий Arduino IDE. Но библиотеки, как и обычные программы, иногда страдают несовместимостью с отдельными версиями IDE. Если установлено несколько версий Arduino IDE, а библиотеку надо установить только для одной из них, потребуется ручная установка.

Закройте Arduino IDE. Распакуйте архив с библиотекой. Если каталог библиотеки содержит в названии фрагмент `-master`, удалите этот фрагмент. Откройте каталог, в который вы раньше установили нужную версию IDE. Найдите каталог `libraries` и

откройте его. Внутри должны находиться некоторые каталоги стандартных библиотек. Поместите туда же каталог с вашей библиотекой. Запустите Arduino IDE и убедитесь, что новая библиотека появилась в списке.

2.3.3. Установка расширения ESP8266 для Arduino IDE

Выберите пункт меню **Файл | Настройки** и вставьте в поле **Дополнительные ссылки для Менеджера плат** строку ссылки:

```
http://arduino.esp8266.com/staging/package_esp8266com_index.json
```

Если в поле для дополнительных ссылок уже есть другая запись, отделите ее запятой. Теперь перейдите в меню **Инструменты | Плата | Менеджер плат** и дождитесь, пока менеджер плат соединится с ресурсами по указанной ссылке и обновит список плат. Затем найдите в списке плату **esp8266 by ESP8266 Community** (обычно в конце списка) и нажмите кнопку **Установка**. Теперь вы можете работать с платой на основе ESP8266 точно так же, как с любой платой Arduino. Обратите внимание, что в меню **Файл | Примеры** появился раздел с примерами для ESP8266. Кроме этого, автоматически установлена специальная библиотека **ESP8266WiFi** для поддержки функций беспроводного соединения.

Расширение среды Arduino IDE, которое вы установили, представляет собой *обертку* (wrapper) для набора официальных функций API SDK, предоставляемых производителем микросхемы ESP8266. Когда в вашей программе совершается любое действие, например запись логического уровня в порт ввода/вывода, на самом деле компилятор не генерирует собственный оригинальный код, а обращается к готовой функции API SDK и передает ей нужные параметры. Заслуга разработчиков расширения ESP8266 состоит в том, что они проделали за нас всю самую сложную работу, переведя обращения к API SDK на простой и понятный язык Processing для Arduino IDE.

2.3.4. Особенности программирования ESP8266

Простые программы, наподобие программ управления портами GPIO, будут работать на ESP8266 точно так же, как на оборудовании Arduino. Но если вы не будете учитывать аппаратные особенности ESP8266 при разработке более сложных программ, то обязательно наткнетесь на «подводные камни», которые могут доставить много проблем. Нельзя забывать, что в микросхеме ESP8266, кроме процессорной части, имеется блок Wi-Fi, который «живет своей жизнью», занимает ресурсы процессора и требует отдельного внимания.

В этом разделе мы уточним только те особенности программирования ESP8266, которые имеют прямое отношение к проектам из этой книги.

Порты и прерывания

Формально, вы можете обратиться в программе к любому из портов GPIO0 — GPIO16, но для свободного пользования доступны не все порты, т. к. часть выводов

занята для служебных нужд. Количество линий портов, выведенных на внешние контакты платы, зависит от конструкции модуля, но некоторые порты заняты всегда, либо их не рекомендуется использовать. В частности, порты GPIO1 и GPIO3 заняты под линии порта UART Tx и Rx. По этим линиям производится запись прошивки, а также обмен последовательными данными с внешними устройствами. К выводу GPIO2 по умолчанию подключен встроенный светодиод большинства модулей. Однако в модуле ESP-01 светодиод подключен к порту GPIO1, поэтому невозможно одновременно управлять встроенным светодиодом и использовать этот последовательный порт. Порт GPIO0 служит для управления режимом программирования. Порт GPIO15 определяет режим загрузки с карты памяти SD и по умолчанию в большинстве случаев соединен на плате с общим проводом. Порт GPIO16 (часто обозначаемый как DEEP_SLEEP) используется для вывода микросхемы из режима пониженного энергопотребления. В таком случае он должен быть соединен с выводом RESET микросхемы. Во многих модулях это соединение выполнено на печатной плате, поэтому крайне нежелательно использовать GPIO16 для своих целей. Таким образом, не рекомендуется задействовать порты GPIO0, GPIO1, GPIO2, GPIO3, GPIO15 и GPIO16 для каких-либо иных целей, кроме назначенных по умолчанию.

Прерывания поддерживаются функциями `attachInterrupt` и `detachInterrupt` и могут быть назначены на любой GPIO, кроме GPIO16.

Организация задержек

Функции `delay()` и `delayMicroseconds()` работают аналогично одноименным функциям Arduino, при этом работа модуля Wi-Fi и стека TCP/IP не прерывается. Поэтому функцию `delay()` иногда используют для того, чтобы приостановить выполнение скетча и дать возможность отработать модулю Wi-Fi. Специальные функции блока Wi-Fi и стека TCP/IP обрабатывают события только в конце каждого цикла `loop()` вашего скетча либо во время задержки `delay()`. Если внутри цикла `loop()` есть фрагменты, которые выполняются дольше 50 миллисекунд, необходимо использовать функцию `delay()`, чтобы сохранить работоспособность стека Wi-Fi. Функция `delayMicroseconds()` блокирует выполнение других задач, включая Wi-Fi, поэтому не рекомендуется использовать ее для организации задержек более 20 миллисекунд, — иначе сработает сторожевой таймер и перезагрузит микросхему.

Поддержка интерфейсов I²C и SPI

Для интерфейса I²C (библиотека `wire` — см. *разд. 3.3.1*) реализован только режим ведущего с частотой до 450 кГц. Для всех модулей, кроме ESP-01, по умолчанию используются выводы 4 (SDA) и 5 (SCL). Выводы могут быть переназначены при помощи функции `Wire.pins(int sda, int scl)`. Например: `Wire.pins(3, 7)`.

Библиотека SPI (см. *разд. 3.4.1*) поддерживается почти полностью, за исключением полярности тактовых импульсов Clock Polarity (CPOL). Поэтому режимы `SPI_MODE2` и `SPI_MODE3` не работают. Но практически во всех любительских проектах они и не требуются.