

Виктор Петин

Проекты с использованием контроллера Arduino

2-е издание

Санкт-Петербург
«БХВ-Петербург»
2015

УДК 004.4
ББК 32.973.26-018.2
П29

Петин В. А.

П29 Проекты с использованием контроллера Arduino. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2015. — 448 с.: ил. — (Электроника)

ISBN 978-5-9775-3550-2

Рассмотрены основные платы Arduino и платы расширения (шилды), добавляющие функциональность основной плате. Подробно описан язык и среда программирования Arduino IDE. Тщательно разобраны проекты с использованием контроллеров семейства Arduino. Это проекты в области робототехники, создания погодных метеостанций, "умного дома", вендинга, телевидения, Интернета, беспроводной связи (bluetooth, радиуправление). Для всех проектов представлены схемы и исходный код. Также представлен исходный код для устройств Android, используемых в проектах для связи с контроллерами Arduino. На сайте издательства размещен архив с исходными кодами программ и библиотек, описаниями и спецификациями электронных компонентов и др. Во втором издании добавлены проекты голосового управления с помощью Arduino, работа с адресуемыми RGB-лентами, управление iRobot Create на Arduino. Рассмотрены проекты с использованием платы Arduino Leonardo. Приведены пошаговые уроки для начинающих разработчиков.

Для читателей, интересующихся современной электроникой

УДК 004.4
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Екатерина Капалыгина</i>
Редактор	<i>Григорий Добин</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Марины Дамбиевой</i>

Подписано в печать 28.11.14.
Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 36,12.
Тираж 1500 экз. Заказ №
"БХВ-Петербург", 191036, Санкт-Петербург, Гончарная ул., 20.
Первая Академическая типография "Наука"
199034, Санкт-Петербург, 9 линия, 12/28

ISBN 978-5-9775-3550-2

© Петин В. А., 2015
© Оформление, издательство "БХВ-Петербург", 2015

Оглавление

Предисловие	15
Для кого и о чем эта книга?	15
Структура книги.....	15
Благодарности.....	16
 ЧАСТЬ I. ARDUINO — ОБЩИЙ ОБЗОР.....	17
 Глава 1. Введение в Arduino	19
1.1. Arduino — что это?	19
1.2. В чем преимущество Arduino?	20
1.3. История создания Arduino	20
 Глава 2. Обзор контроллеров семейства Arduino.....	22
2.1. Arduino Pro Mini.....	23
2.2. Arduino Duemilanove.....	23
2.3. Arduino Nano	25
2.4. Arduino LilyPad	26
2.5. Arduino Uno	27
2.6. Arduino Mega2560.....	28
2.7. Arduino Leonardo.....	29
2.8. Arduino Due	30
 Глава 3. Платы расширения Arduino.....	31
 ЧАСТЬ II. СРЕДА РАЗРАБОТКИ И ЯЗЫК ПРОГРАММИРОВАНИЯ КОНТРОЛЛЕРОВ ARDUINO	37
 Глава 4. Среда программирования Arduino IDE	39
4.1. Установка Arduino IDE в Windows.....	39
4.2. Установка Arduino IDE в Linux	41
4.3. Настройка среды Arduino IDE	41

Глава 5. Программирование в Arduino	45
5.1. Базовые знания.....	45
5.1.1. Цифровые выходы.....	45
5.1.2. Аналоговые входы.....	46
5.1.3. Широтно-импульсная модуляция	46
5.1.4. Память в Arduino	46
5.2. Структура программы	48
5.2.1. Функции <i>setup()</i> и <i>loop()</i>	48
5.3. Синтаксис и операторы	49
5.3.1. Управляющие операторы.....	49
5.3.1.1. Оператор <i>if</i> (условие) и операторы сравнения <i>==</i> , <i>!=</i> , <i><</i> , <i>></i>	49
5.3.1.2. Оператор <i>if..else</i>	49
5.3.1.3. Оператор <i>for</i>	50
5.3.1.4. Оператор <i>switch</i>	51
5.3.1.5. Оператор <i>while</i>	51
5.3.1.6. Оператор <i>do ... while</i>	52
5.3.1.7. Оператор <i>break</i>	52
5.3.1.8. Оператор <i>continue</i>	52
5.3.1.9. Оператор <i>return</i>	53
5.3.2. Синтаксис.....	53
5.3.2.1. ; (точка с запятой) ; (semicolon).....	53
5.3.2.2. {} (фигурные скобки) {} (curly braces).....	53
5.3.2.3. Комментарии // (single line comment), /* */ (multi-line comment).....	54
5.3.3. Арифметические операторы.....	54
5.3.3.1. = (assignment) = оператор присваивания	54
5.3.3.2. + (сложение), - (вычитание), * (умножение) , / (деление)	55
5.3.3.3. % (modulo)	55
5.3.4. Операторы сравнения.....	55
5.3.5. Логические операторы.....	55
5.3.5.1. && (логическое И)	55
5.3.5.2. (логическое ИЛИ)	55
5.3.5.3. ! (логическое отрицание).....	56
5.3.6. Унарные операторы	56
5.3.6.1. ++ (увеличение значения) / -- (уменьшение значения)	56
5.3.6.2. += , -= , *= , /=.....	56
5.4. Данные.....	56
5.4.1. Типы данных.....	56
5.4.1.1. <i>boolean</i>	57
5.4.1.2. <i>char</i>	57
5.4.1.3. <i>byte</i>	57
5.4.1.4. <i>int</i>	57
5.4.1.5. <i>unsigned int</i>	58
5.4.1.6. <i>long</i>	58
5.4.1.7. <i>unsigned long</i>	58
5.4.1.8. <i>float</i>	59
5.4.1.9. <i>double</i>	59
5.4.1.10. <i>string</i> — текстовые строки.....	59
5.4.1.11. Массивы	60
5.4.1.12. <i>void</i>	61

5.4.2. Константы	61
5.4.3. Переменные	62
5.4.3.1. Объявление переменных	62
5.4.3.2. Границы переменных	62
5.4.4. Преобразование типов данных	63
5.4.4.1. <i>char()</i>	63
5.4.4.2. <i>byte()</i>	63
5.4.4.3. <i>int()</i>	63
5.4.4.4. <i>long()</i>	63
5.4.4.5. <i>float()</i>	64
5.5. Функции	64
5.5.1. Цифровой ввод/вывод	64
5.5.1.1. Функция <i>pinMode</i>	64
5.5.1.2. Функция <i>digitalWrite()</i>	64
5.5.1.3. Функция <i>digitalRead()</i>	65
5.5.2. Аналоговый ввод/вывод	66
5.5.2.1. Функция <i>analogRead()</i>	66
5.5.2.2. Функция <i>analogReference()</i>	67
5.5.2.3. Функция <i>analogWrite()</i>	67
5.5.3. Дополнительные функции ввода/вывода	69
5.5.3.1. Функция <i>tone()</i>	69
5.5.3.2. Функция <i>noTone()</i>	69
5.5.3.3. Функция <i>shiftOut()</i>	69
5.5.3.4. Функция <i>pulseIn()</i>	71
5.5.4. Работа со временем	72
5.5.4.1. Функция <i>millis()</i>	72
5.5.4.2. Функция <i>micros()</i>	72
5.5.4.3. Функция <i>delay()</i>	73
5.5.4.4. Функция <i>delayMicroseconds()</i>	74
5.5.5. Математические функции	75
5.5.5.1. Функция <i>min(x,y)</i>	75
5.5.5.2. Функция <i>max(x, y)</i>	75
5.5.5.3. Функция <i>abs()</i>	75
5.5.5.4. Функция <i>constrain(x, a, b)</i>	76
5.5.5.5. Функция <i>map(value, fromLow, fromHigh, toLow, toHigh)</i>	76
5.5.5.6. Функция <i>pow(base, exponent)</i>	77
5.5.5.7. Функция <i>sq(x)</i>	77
5.5.5.8. Функция <i>sqrt(x)</i>	77
5.5.6. Тригонометрические функции	78
5.5.6.1. Функция <i>sin(rad)</i>	78
5.5.6.2. Функция <i>cos(rad)</i>	78
5.5.6.3. Функция <i>tan(rad)</i>	78
5.5.7. Генераторы случайных значений	78
5.5.7.1. Функция <i>randomSeed(seed)</i>	78
5.5.7.2. Функция <i>random()</i>	79
5.5.8. Операции с битами и байтами	79
5.5.8.1. Функция <i>lowByte()</i>	80
5.5.8.2. Функция <i>highByte()</i>	80

5.5.8.3. Функция <i>bitRead()</i>	80
5.5.8.4. Функция <i>bitWrite()</i>	80
5.5.8.5. Функция <i>bitSet()</i>	81
5.5.8.6. Функция <i>bitClear()</i>	81
5.5.8.7. Функция <i>bit()</i>	81
5.5.9. Внешние прерывания	81
5.5.9.1. Функция <i>attachInterrupt</i>	82
5.5.9.2. Функция <i>detachInterrupt</i>	82
5.6. Управление портами через регистры ATmega	83

ЧАСТЬ III. ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ ARDUINO 87

Глава 6. Arduino и набор функций *Serial*..... 89

6.1. Функции библиотеки <i>Serial</i>	90
6.1.1. Функция <i>Serial.begin()</i>	90
6.1.2. Функция <i>Serial.end()</i>	90
6.1.3. Функция <i>Serial.available()</i>	90
6.1.4. Функция <i>Serial.read()</i>	91
6.1.5. Функция <i>Serial.flush()</i>	92
6.1.6. Функция <i>Serial.print()</i>	92
6.1.7. Функция <i>Serial.println()</i>	93
6.1.8. Функция <i>Serial.write()</i>	93
6.1.9. Функция <i>Serial.peek()</i>	93
6.2. Библиотека <i>SoftwareSerial</i>	94
6.2.1. Функция <i>SoftwareSerial()</i>	94
6.2.2. Функция <i>SoftwareSerial.listen()</i>	94
6.2.3. Функция <i>SoftwareSerial.isListening()</i>	95
6.2.4. Функция <i>SoftwareSerial.overflow()</i>	96

Глава 7. Arduino и знакосинтезирующие жидкокристаллические индикаторы 97

7.1. Принцип работы модулей ЖКИ WINSTAR WH1604	98
7.2. Библиотека <i>LiquidCrystal</i>	103
7.2.1. Функция <i>LiquidCrystal()</i>	104
7.2.2. Функция <i>begin()</i>	105
7.2.3. Функция <i>clear()</i>	105
7.2.4. Функция <i>home()</i>	105
7.2.5. Функция <i>setCursor()</i>	105
7.2.6. Функция <i>write()</i>	106
7.2.7. Функция <i>print()</i>	106
7.2.8. Функция <i>cursor()</i>	107
7.2.9. Функция <i>noCursor()</i>	107
7.2.10. Функция <i>blink()</i>	107
7.2.11. Функция <i>noBlink()</i>	107
7.2.12. Функция <i>display()</i>	107
7.2.13. Функция <i>noDisplay()</i>	108
7.2.14. Функция <i>scrollDisplayLeft()</i>	108
7.2.15. Функция <i>scrollDisplayRight()</i>	108

7.2.16. Функция <i>autoscroll()</i>	108
7.2.17. Функция <i>noAutoscroll()</i>	108
7.2.18. Функция <i>leftToRight()</i>	109
7.2.19. Функция <i>rightToLeft()</i>	109
7.2.20. Функция <i>createChar()</i>	109
7.3. Библиотека <i>LiquidCrystalRus</i>	111
Глава 8. Библиотека <i>EEPROM</i>	116
8.1. Функции библиотек <i>EEPROM</i>	116
8.1.1. Функция чтения <i>EEPROM.read</i>	116
8.1.2. Функция записи <i>EEPROM.write</i>	117
8.2. Примеры использования памяти <i>EEPROM</i>	118
8.2.1. Воспроизведение звука	118
8.2.2. Звонок с мелодиями	121
Глава 9. Использование Arduino Leonardo в качестве USB-устройства	126
9.1. Имитация клавиатуры на Arduino Leonardo	127
9.2. Имитация компьютерной мыши на Arduino Leonardo	129
Глава 10. Arduino и 1-Wire	133
10.1. Что такое 1-Wire?	133
10.2. Применение 1-Wire	136
10.3. Протокол 1-Wire	137
10.3.1. Описание интерфейса 1-Wire	137
10.3.2. Обмен информацией по 1-Wire	138
10.3.3. Протокол обмена информацией 1-Wire	141
10.4. Библиотека <i>OneWire</i>	143
10.4.1. Функция <i>begin()</i>	144
10.4.2. Функция <i>requestFrom()</i>	144
10.4.3. Функция <i>beginTransmission()</i>	144
10.4.4. Функция <i>endTransmission()</i>	145
10.4.5. Функция <i>write()</i>	145
10.4.6. Функция <i>available()</i>	145
10.4.7. Функция <i>read()</i>	146
10.4.8. Функция <i>onReceive()</i>	146
10.4.9. Функция <i>onRequest()</i>	146
10.5. Устройство iButton и программирование электронного замка	146
10.5.1. Поиск устройств 1-Wire и получение уникального кода	148
10.5.2. Режимы работы	150
10.5.3. Режим пользователя	151
10.5.4. Первоначальный ввод ключа в пустую базу	151
10.5.5. Просмотр, запись и удаление кодов ключей в режиме администратора	153
10.5.6. Блок-реле. Открывание замка	156
10.5.7. Проигрывание мелодий	157
Глава 11. Arduino и цифровой датчик температуры DS18B20	159
11.1. Описание датчика DS18B20	159
11.2. Использование библиотеки <i>OneWire</i>	162
11.3. Библиотека <i>DallasTemperature</i>	163

Глава 12. Arduino и датчики температуры и влажности DHT	165
12.1. Характеристики датчиков DH11, DH22.....	165
12.2. Подключение к Arduino	166
12.3. Библиотека <i>DHT</i>	167
Глава 13. Сетевой обмен с помощью Arduino	169
13.1. Устройство Arduino Ethernet shield	169
13.2. Библиотека <i>Ethernet library</i>	171
13.2.1. Класс Ethernet (<i>Ethernet class</i>)	171
Функция <i>Ethernet.begin()</i>	172
Функция <i>Ethernet.localIP()</i>	172
13.2.2. Класс IPAddress (<i>IPAddress class</i>)	172
Функция <i>IPAddress()</i>	172
13.2.3. Класс Server (<i>Server class</i>).....	173
Функция <i>ethernetServer()</i>	173
Функция <i>begin()</i>	173
Функция <i>available()</i>	173
Функция <i>write()</i>	174
Функция <i>print()</i>	175
Функция <i>println()</i>	175
13.2.4. Класс Client (<i>Client class</i>).....	175
Функция <i>client()</i>	176
Функция <i>EthernetClient()</i>	176
Функция <i>connected()</i>	176
Функция <i>connect()</i>	177
Функция <i>write()</i>	177
Функция <i>print()</i>	177
Функция <i>println()</i>	177
Функция <i>available()</i>	178
Функция <i>read()</i>	178
Функция <i>flush()</i>	178
Функция <i>stop()</i>	179
13.2.5. Класс EthernetUDP (<i>EthernetUDP class</i>)	179
Функция <i>begin()</i>	179
Функция <i>read()</i>	179
Функция <i>write()</i>	180
Функция <i>beginPacket()</i>	180
Функция <i>endPacket()</i>	181
Функция <i>parsePacket()</i>	181
Функция <i>available()</i>	182
Функция <i>remoteIP()</i>	182
Функция <i>remotePort()</i>	182
13.3. Домашняя метеостанция с доступом через Интернет	182
13.3.1. Устройство, настройка и отладка метеостанции	182
13.3.2. Создание виджета для планшетов с ОС Android	187
13.3.3. Размещение данных метеостанции на сайте	188
Глава 14. Arduino и карта памяти SD	190
14.1. Arduino-библиотека <i>SD</i>	190
14.1.1. Класс <i>SD</i>	190
Функция <i>begin()</i>	191

Функция <i>exists()</i>	191
Функция <i>mkdir()</i>	191
Функция <i>rmdir()</i>	191
Функция <i>open()</i>	191
Функция <i>remove()</i>	192
14.1.2. Класс <i>File</i>	192
Функция <i>available()</i>	192
Функция <i>close()</i>	193
Функция <i>flush()</i>	193
Функция <i>peek()</i>	193
Функция <i>position()</i>	194
Функция <i>print()</i>	194
Функция <i>println()</i>	194
Функция <i>seek()</i>	195
Функция <i>size()</i>	195
Функция <i>read()</i>	195
Функция <i>write()</i>	195
Функция <i>isDirectory()</i>	196
Функция <i>openNextFile()</i>	196
Функция <i>rewindDirectory()</i>	196
14.2. Запись показаний датчиков на SD-карту	196
Глава 15. Arduino и светодиодные матрицы	202
15.1. Светодиоды и светодиодные матрицы	202
15.2. Светодиодная матрица FYM-23881BUG-11	203
15.3. SPI-расширитель выходов 74HC595	204
15.4. Игра "Тетрис" на светодиодных матрицах FYM-23881BUG-11	206
15.4.1. Управление изображением на светодиодных матрицах	208
15.4.2. Фигуры игры "Тетрис"	209
15.4.3. Управление фигурами игры "Тетрис"	211
15.4.4. Проверка столкновения фигур	213
15.5. Светодиодная матрица RGB	215
15.6. RGB-ночник, управляемый с помощью движения рук	216
Глава 16. Arduino и управляемые светодиодные ленты RGB	221
16.1. RGB-светодиодная лента WS2812	221
16.2. Arduino-библиотека Adafruit_NeoPixel	223
16.3. Создание табло результатов ЧМ-2018 по футболу на RGB-ленте WS2812	224
Глава 17. Работа Arduino с вендинговыми аппаратами	230
17.1. Купюроприемник ICT серий A7 и V7	230
17.2. Подключение купюроприемника ICT V7 к Arduino	234
17.3. Скetch для получения номинала принимаемой купюры	236
17.4. Монетоприемник CH-926	237
Настройка монетоприемника	238
Калибровка монетоприемника	239
17.5. Подключение монетоприемника к Arduino	239
17.6. Написание скетча	240

Глава 18. Arduino и радиочастотная идентификация (RFID).....	242
18.1. Радиочастотная идентификация	242
18.2. Датчик считывания RFID-карт	245
18.3. Скетч для считывания RFID-карт.....	247
Глава 19. Arduino и датчики расстояния	250
19.1. Ультразвуковые дальномеры HC-SR04	250
19.2. Принцип работы ультразвукового дальномера HC-SR04	251
19.3. Библиотека <i>Ultrasonic</i>	251
19.4. Инфракрасные датчики расстояния Sharp	252
19.5. Подключение датчиков Sharp к Arduino	255
Глава 20. Arduino и передача данных в инфракрасном диапазоне.....	256
20.1. Обмен данными в инфракрасном диапазоне	256
20.2. Протоколы для ИК-пультов	258
20.2.1. Протокол RC5	258
20.2.2. Протокол NEC	259
20.2.3. Протокол JVC	261
20.2.4. Протокол Sony	262
20.3. Подключение ИК-приемника	263
20.4. Библиотека <i>IRremote</i>	263
20.5. Скетч для получения кодов ИК-пульта.....	265
Глава 21. Создаем робота.....	271
21.1. Ходовая часть.....	271
21.2. Драйвер двигателей L293D	274
21.3. Массив возможных состояний моторов	277
21.4. Разработка скетча движений робота	279
21.5. Движение робота по линии в автономном режиме.....	282
Глава 22. Arduino и шаговые двигатели	291
22.1. Управление шаговым двигателем	292
22.2. Arduino-библиотека <i>Stepper</i>	293
22.2.1. Функция <i>Stepper()</i>	293
22.2.2. Функция <i>setSpeed(rpm)</i>	293
22.2.3. Функция <i>step(steps)</i>	294
22.3. Пример использования библиотеки <i>Stepper</i>	294
22.4. Arduino-библиотека <i>AccelStepper</i>	296
Глава 23. Arduino и сервоприводы	297
23.1. Сервоприводы	297
23.2. Arduino-библиотека <i>Servo</i> для управления сервоприводом	300
23.2.1. Функция <i>attach()</i>	301
23.2.2. Функция <i>detach()</i>	301
23.2.3. Функция <i>write(int)</i>	302
23.2.4. Функция <i>writeMicroseconds(int)</i>	302
23.2.5. Функция <i>read()</i>	302
23.2.6. Функция <i>attached()</i>	302
23.3. Робот-паук на сервоприводах	303
23.4. Скетч для управления роботом-пауком	307

Глава 24. Arduino и Bluetooth	312
24.1. "Голубой зуб"	312
24.2. Модуль Bluetooth HC-05	312
24.3. Управление роботом с Android-устройства по Bluetooth	318
Глава 25. TV-выход на Arduino	327
25.1. Схема подключения.....	327
25.2. Arduino-библиотека <i>TVOut</i>	327
25.2.1. Функция установки режима <i>begin()</i>	328
25.2.2. Функции задержки.....	328
Функция <i>delay()</i>	328
Функция <i>delay_frame()</i>	328
25.2.3. Функции получения параметров	329
Функция <i>hres()</i>	329
Функция <i>vres()</i>	329
Функция <i>char_line()</i>	329
25.2.4. Основные графические функции.....	329
Функция <i>set_pixel()</i>	329
Функция <i>get_pixel()</i>	330
Функция <i>fill()</i>	330
Функция <i>clear_screen()</i>	330
Функция <i>invert()</i>	331
Функция <i>shift_direction()</i>	331
Функция <i>draw_line()</i>	331
Функция <i>draw_row()</i>	331
Функция <i>draw_column()</i>	332
Функция <i>draw_rect()</i>	332
Функция <i>draw_circle()</i>	333
Функция <i>bitmap()</i>	333
25.2.5. Функции вывода текстовой информации	333
Функция <i>select_font()</i>	334
Функция <i>print_char()</i>	334
Функция <i>set_cursor()</i>	334
Функция <i>print()</i>	334
Функция <i>println()</i>	335
25.2.6. Функции вывода аудио	335
Функция <i>tone()</i>	335
Функция <i>noTone()</i>	335
25.3. Создание пользовательских шрифтов.....	336
25.4. Создание графических примитивов	337
25.5. Создание простейшей игровой консоли	340
25.6. Разработка игры для игровой консоли.....	342
25.6.1. Создание переменных игры.....	342
25.6.2. Управление положением "игрока" с помощью джойстика.....	343
25.6.3. Генерация и перемещение объектов-цифр	345
25.6.4. Проверка столкновения "игрока" и объектов-цифр	346
25.6.5. Счетчик баллов "игрока"	347
25.6.6. Переход на новый уровень	348

25.6.7. Отображение данных игры на табло.....	348
25.6.8. Звуковое сопровождение игры.....	349
25.6.9. Основной цикл игры.....	349
25.6.10. Добавляем меню для выбора игр	350
Глава 26. Arduino и радиуправление	352
26.1. Принципы формирования радиосигнала	353
26.2. Установка связи приемника с передатчиком	355
26.3. Разработка скетча приема команд для Arduino	356
Глава 27. Arduino и беспроводной радиомодуль NRF24L01.....	359
27.1. Радиомодуль NRF24L01.....	359
27.2. Библиотека для работы с модулем NRF24L01	360
27.2.1. Параметры библиотеки <i>Mirf</i>	361
27.2.2. Функции библиотеки <i>Mirf</i>	361
Функция <i>init()</i>	361
Функция <i>setRADDR()</i>	361
Функция <i>setTADDR()</i>	361
Функция <i>config()</i>	361
Функция <i>dataReady()</i>	362
Функция <i>getData()</i>	362
Функция <i>send()</i>	362
Функция <i>isSending()</i>	362
27.3. Пример соединения двух плат Arduino с помощью модуля NRF24L01	363
Глава 28. Работа Arduino с USB-устройствами	366
28.1. Интерфейс USB.....	366
28.2. USB Host Shield.....	367
28.3. HID-устройства USB	368
28.4. Подключение HID-мыши USB	371
28.5. Использование HID-устройства (руль Defender) для управления роботом	371
28.6. Управление iRobot Create с помощью беспроводного геймпада через Arduino	381
Глава 29. Arduino и ROS.....	388
29.1. Что такое ROS?	388
29.2. Дистрибутивы ROS.....	389
29.3. Установка ROS.....	389
29.4. Узлы и темы в ROS.....	390
29.5. Пакет <i>rosterial</i>	391
29.6. Подготовка сообщения (publisher) на Arduino	392
29.7. Создание подписки (subscriber) на Arduino	395
29.8. Связь через ROS двух плат Arduino	397
Глава 30. Голосовое управление	400
30.1. Управление электроприборами с помощью радиорозеток UNIEL	400
30.2. Радиомодули FS1000A	402
30.3. Модуль распознавания голоса Voice Recognition Module V2	405
30.3.1. Обучение модуля	405
30.4. Система голосового управления.....	408

ПРИЛОЖЕНИЯ	415
Приложение 1. Список использованных источников.....	417
Приложение 2. Начальная школа.....	418
Урок 1. Управление светодиодом. Включение/выключение. Плавное регулирование яркости.....	418
Урок 2. Подключение кнопки.....	421
Урок 3. Работа с монитором порта	425
Урок 4. Подключение датчика температуры и влажности, работа с монитором порта	428
Урок 5. Подключение модуля часов	431
Урок 6. Работа с жидкокристаллическим экраном	433
Урок 7. Подключение RGB-светодиода, использование широтно-импульсной модуляции.....	435
Урок 8. Подключение фоторезистора.....	438
Урок 9. Подключение фоторезистора и 8-ми светодиодов	440
Приложение 3. Описание электронного архива.....	443

Предисловие

Для кого и о чем эта книга?

Предлагаемая книга ориентирована на читателей, желающих быстро войти в темы программирования микроконтроллеров и использования микроконтроллеров для связи с внешними системами в проектах автоматизации и робототехники.

Книга содержит описание языка программирования плат Arduino в среде Arduino IDE и предлагает изучение предмета на реальных проектах, имеющих практическое значение. В ней вы найдете множество примеров и проектов использования Arduino, представляющих собой законченные решения, пригодные для использования в ваших проектах.

Для тех, у кого нет опыта разработки проектов Arduino, а также написания программного кода для электронных устройств, к книге предлагается специальный набор деталей и компонентов, с помощью которого вы сможете быстро адаптироваться в мире Arduino и получить удовольствие, видя, как оживают ваши творения! Уроки учебного набора (*приложение 2*) тематически привязаны к соответствующим главам книги, и вы найдете в этих главах необходимые ссылки. Используя детали и компоненты набора, вы также сможете реализовать более сложные проекты, описание которых содержится в прилагаемом к набору руководстве.

Книга сопровождается электронным архивом, содержащим исходный код всех рассмотренных примеров и проектов, а также используемые в проектах необходимые библиотеки (*см. приложение 3*). Этот электронный архив можно скачать с FTP-сервера издательства "БХВ-Петербург" по ссылке <ftp://ftp.bhv.ru/9785977535502.zip>, а также со страницы книги на сайте www.bhv.ru.

Структура книги

Книга состоит из трех частей и включает предисловие, тридцать глав и три приложения.

Часть I содержит описание Arduino, обзор контроллеров семейства Arduino и плат расширения для Arduino.

В *части II* книги рассмотрены среда разработки и язык программирования для контроллеров Arduino.

Часть III посвящена созданию конкретных устройств на основе контроллера Arduino. Проекты содержат электрические схемы и листинги программ. Рассмотрено использование плат расширения (шилдов). В книге широко используются библиотеки Arduino.

В *приложениях* приведены перечень использованной литературы и интернет-ресурсов, уроки учебного набора и описание электронного архива, сопровождающего книгу.

Благодарности

Хочу поблагодарить родных и близких, которые с пониманием относились к потраченному на книгу (за счет общения с ними) времени.

Большая благодарность издательству "БХВ-Петербург", где поверили в необходимость этой книги, и всем сотрудникам издательства, которые помогали мне в ее создании.

Благодарю также всех читателей, купивших эту книгу, — надеюсь, она поможет вам в разработке собственных проектов на основе Arduino.

ЧАСТЬ I



Arduino — общий обзор

Глава 1. Введение в Arduino

Глава 2. Обзор контроллеров семейства Arduino

Глава 3. Платы расширения Arduino



ГЛАВА 1

Введение в Arduino

1.1. Arduino — что это?

Появление первых микроконтроллеров ознаменовало начало новой эры в развитии микропроцессорной техники. Наличие в одном корпусе большинства системных устройств сделало микроконтроллер подобным обычному компьютеру. В отечественной литературе они даже назывались однокристальными микроЭВМ. Соответственно и желание использовать микроконтроллеры как обычные компьютеры появилось практически с их появлением. Но желание это сдерживалось многими факторами. Например, чтобы собрать устройство на микроконтроллере, необходимо знать основы схемотехники, устройство и работу конкретного процессора, уметь программировать на ассемблере и изготавливать электронную технику. Потребуются также программаторы, отладчики и другие вспомогательные устройства. В итоге без огромного объема знаний и дорогостоящего оборудования не обойтись. Такая ситуация долго не позволяла многим любителям использовать микроконтроллеры в своих проектах. Сейчас, с появлением устройств, дающих возможность работать с микроконтроллерами без наличия серьезной материальной базы и знания многих предметов, все изменилось. Примером такого устройства может служить проект Arduino итальянских разработчиков.

Arduino и его клоны представляют собой наборы, состоящие из готового электронного блока и программного обеспечения. Электронный блок здесь — это печатная плата с установленным микроконтроллером и минимумом элементов, необходимых для его работы. Фактически электронный блок Arduino является аналогом материнской платы современного компьютера. На нем имеются разъемы для подключения внешних устройств, а также разъем для связи с компьютером, по которому и осуществляется программирование микроконтроллера. Особенности используемых микроконтроллеров ATmega фирмы Atmel позволяют производить программирование без применения специальных программаторов. Все, что нужно для создания нового электронного устройства, — это плата Arduino, кабель связи и компьютер. Второй частью проекта Arduino является программное обеспечение для создания управляющих программ. Оно объединило в себе простейшую среду разработки и язык программирования, представляющий собой вариант языка C/C++ для микро-

контроллеров. В него добавлены элементы, позволяющие создавать программы без изучения аппаратной части. Так что для работы с Arduino практически достаточно знания только основ программирования на C/C++. Создано для Arduino и множество библиотек, содержащих код, работающий с различными устройствами.

1.2. В чем преимущество Arduino?

Пользователь современного компьютера не задумывается о функционировании отдельных частей ПК. Он просто запускает нужные программы и работает с ними. Точно так же и Arduino позволяет пользователю сосредоточиться на разработке проектов, а не на изучении устройства и принципов функционирования отдельных элементов. Нет надобности и в создании законченных плат и модулей. Разработчик может использовать готовые платы расширения или просто напрямую подключить к Arduino необходимые элементы. Все остальные усилия будут направлены на разработку и отладку управляющей программы на языке высокого уровня. В итоге доступ к разработке микропроцессорных устройств получили не только профессионалы, но и просто любители что-то сделать своими руками. Наличие готовых модулей и библиотек программ позволяет непрофессионалам в электронике создавать готовые работающие устройства для решения своих задач. А варианты использования Arduino ограничены только возможностями микроконтроллера и имеющегося варианта платы, ну и, конечно, фантазией разработчика.

1.3. История создания Arduino

В 2002 году программист Массимо Банци (Massimo Banzi) был принят на работу в должности доцента в Институт проектирования взаимодействий города Ивреа (Interaction Design Institute Ivrea, IDII) для продвижения новых способов разработки интерактивных проектов. Однако крошечный бюджет и ограниченное время доступа к лабораторной базе сводили его усилия практически на нет. В проектах Банци использовал устройство BASIC Stamp, разработанное калифорнийской компанией Parallax. Stamp представлял собой небольшую печатную плату с размещенными на ней источником питания, микроконтроллером, памятью и портами ввода/вывода для соединения с различной аппаратурой. Программирование микроконтроллера осуществлялось на языке BASIC. BASIC Stamp имел две проблемы: недостаток вычислительной мощности и достаточно высокую цену — плата с основными компонентами стоила около 100 долларов. И команда Банци решила самостоятельно создать плату, которая удовлетворяла бы всем их потребностям.

Банци и его сотрудники поставили себе целью создать устройство, представляющее собой простую, открытую и легкодоступную платформу для разработки, с ценой — не более 30 долларов — приемлемой для студенческого кармана. Хотели они и выделить чем-то свое устройство на фоне прочих. Поэтому в противовес другим производителям, экономящим на количестве выводов печатной платы, они решили добавить их как можно больше, а также сделали свою плату синей, в отличие от обычных зеленых плат.

Продукт, который создала команда, состоял из дешевых и доступных компонентов — например, базировался он на микроконтроллере ATmega328. Но главная задача состояла в том, чтобы гарантировать работу устройства по принципу *plug-and-play*, — чтобы пользователь, достав плату из коробки и подключив к компьютеру, мог немедленно приступить к работе.

Первый прототип платы был сделан в 2005 году, она имела простейший дизайн и еще не называлась Arduino. Чуть позже Массимо Банци придумал назвать ее так — по имени принадлежащего ему бара, расположенного в городе Ивреа. Бренд "Arduino" без какой-либо рекламы и привлечения средств маркетинга быстро приобрел высокую популярность в Интернете. С начала распространения продано более 250 тыс. комплектов Arduino, и это не учитывая множества клонов. В мире насчитывается более двухсот дистрибьюторов продукции Arduino — от крупных фирм, таких как SparkFun Electronics, до мелких компаний, работающих на местный рынок. На сегодня платформа Arduino представлена не одной платой, а целым их семейством. В дополнение к оригинальному проекту, называемому Arduino Uno, новые модели, имеющие на плате более мощные средства, носят название Arduino Mega, компактные модели — Arduino Nano, платы в водонепроницаемом корпусе — LilyPad Arduino, а новая плата с 32-разрядным процессором Cortex-M3 ARM — Arduino Due.

Своим успехом проект Arduino обязан существовавшему до него языку Processing и платформе Wiring. От этих проектов Arduino унаследовал одну сильную черту — удобную для пользователя среду разработки. До появления Arduino программирование микроконтроллеров требовало сложного и рутинного предварительного обучения. А с Arduino даже те, кто не имеет опыта работы с электронными устройствами, теперь могут приобщиться к миру электроники. Начинающим уже не приходится тратить много времени на изучение сопутствующего материала — они могут быстро разработать прототип, который будет полноценно рабочим.

По словам Массимо Банци, пятьдесят лет назад, чтобы написать программное обеспечение требовалась команда людей в белых халатах, которые знали все об электронных лампах. Теперь же, с появлением Arduino, множество людей получили возможность создавать электронные устройства самостоятельно. Как утверждает один из членов команды Банци, инженер по телекоммуникациям Дэвид Куартиллер, философия Arduino как раз и состоит в том, что желающий разобраться в электронике может сразу же приступить к ее изучению, вместо того чтобы сначала учиться алгебре.



ГЛАВА 2

Обзор контроллеров семейства Arduino

Основные версии плат Arduino представлены следующими моделями:

- ❑ **Due** — плата на базе 32-битного ARM микропроцессора Cortex-M3 ARM SAM3U4E;
- ❑ **Leonardo** — плата на микроконтроллере ATmega32U4;
- ❑ **Uno** — самая популярная версия базовой платформы Arduino;
- ❑ **Duemilanove** — плата на микроконтроллере ATmega168 или ATmega328;
- ❑ **Diecimila** — версия базовой платформы Arduino USB;
- ❑ **Nano** — компактная платформа, используемая как макет. Nano подключается к компьютеру при помощи кабеля USB Mini-B;
- ❑ **Mega ADK** — версия платы Mega 2560 с поддержкой интерфейса USB-host для связи с телефонами на Android и другими устройствами с интерфейсом USB;
- ❑ **Mega2560** — плата на базе микроконтроллера ATmega2560 с использованием чипа ATmega8U2 для последовательного соединения по USB-порту;
- ❑ **Mega** — версия серии Mega на базе микроконтроллера ATmega1280;
- ❑ **Arduino BT** — платформа с модулем Bluetooth для беспроводной связи и программирования;
- ❑ **LilyPad** — платформа, разработанная для переноски, может зашиваться в ткань;
- ❑ **Fio** — платформа разработана для беспроводных применений. Fio содержит разъем для радио XBee, разъем для батареи LiPo и встроенную схему подзарядки;
- ❑ **Mini** — самая маленькая платформа Arduino;
- ❑ **Pro** — платформа, разработанная для опытных пользователей, может являться частью большего проекта;
- ❑ **Pro Mini** — как и платформа Pro, разработана для опытных пользователей, которым требуется низкая цена, меньшие размеры и дополнительная функциональность.

Рассмотрим более подробно некоторые из этих плат.

2.1. Arduino Pro Mini

Arduino Pro Mini (рис. 2.1) построена на микроконтроллере ATmega168.

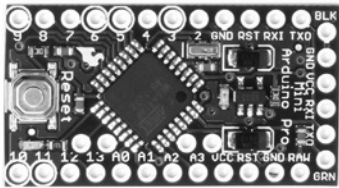


Рис. 2.1. Плата Arduino Pro Mini

Характеристики платы Arduino Pro Mini представлены в табл. 2.1.

Таблица 2.1. Характеристики платы Arduino Pro Mini

Микроконтроллер	ATmega168
Рабочее напряжение	3,3 В или 5 В (в зависимости от модели)
Входное напряжение	3,35–12 В (модель 3,3 В) или 5–12 В (модель 5 В)
Цифровые входы/выходы	14 (6 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	6
Постоянный ток через вход/выход	40 мА
Флеш-память	16 Кбайт (2 — используются для загрузчика)
ОЗУ	1 Кбайт
EEPROM	512 байтов
Тактовая частота	8 МГц (модель 3,3 В) или 16 МГц (модель 5 В)

Arduino Pro Mini может получать питание: через кабель FTDI, или от платы-конвертера, или от регулируемого источника питания 3,3 В или 5 В (зависит от модели платформы) через вывод VCC, или от нерегулируемого источника через вывод RAW.

Выводы питания:

- ❑ RAW — для подключения нерегулируемого напряжения;
- ❑ VCC — для подключения регулируемых 3,3 В или 5 В;
- ❑ GND — выводы заземления.

2.2. Arduino Duemilanove

Arduino Duemilanove (рис. 2.2) построена на одном из микроконтроллеров: ATmega168 или ATmega328.

Характеристики платы Arduino Duemilanove представлены в табл. 2.2.

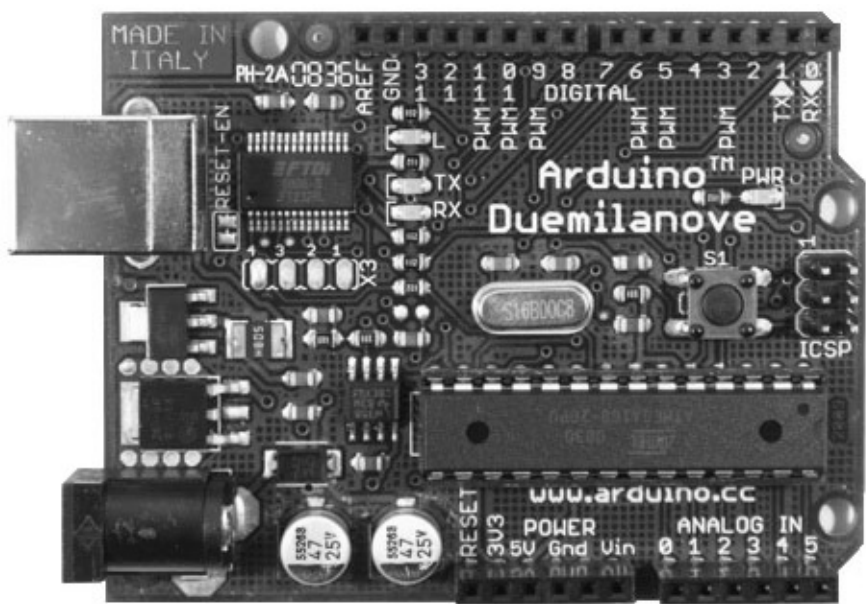


Рис. 2.2. Плата Arduino Duemilanove

Таблица 2.2. Характеристики платы Arduino Duemilanove

Микроконтроллер	АТmega168 или АТmega328
Рабочее напряжение	5 В
Входное напряжение (рекомендуемое)	7–12 В
Входное напряжение (предельное)	6–20 В
Цифровые входы/выходы	14 (6 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	6
Постоянный ток через вход/выход	40 мА
Постоянный ток для вывода 3,3 В	50 мА
Флеш-память	16 Кбайт (АТmega168) или 32 Кбайт (АТmega328), при этом 2 Кбайт используются для загрузчика
ОЗУ	1 Кбайт (АТmega168) или 2 Кбайт (АТmega328)
EEPROM	512 байтов (АТmega168) или 1 Кбайт (АТmega328)
Тактовая частота	16 МГц