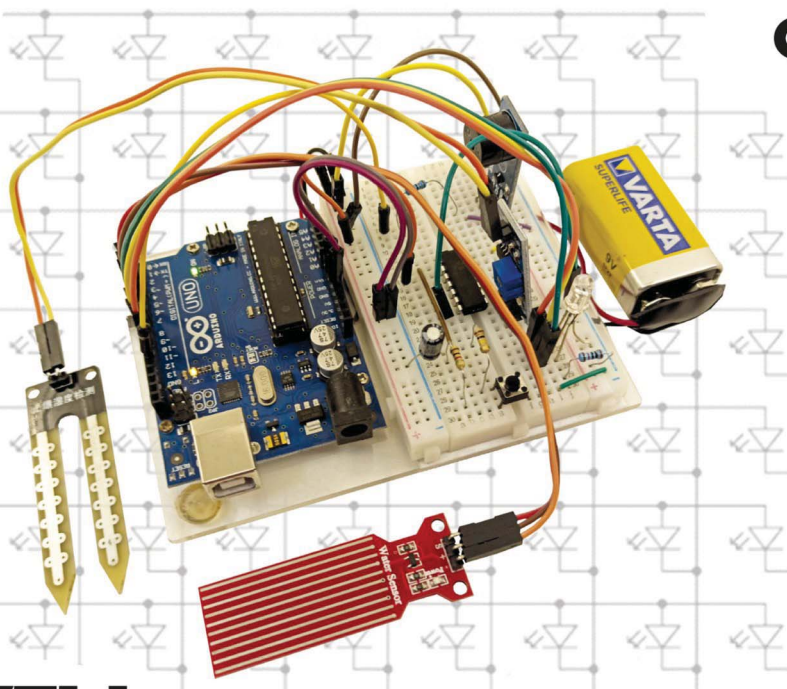


Электроника

Виктор Петин



R2
R3
R4
R5
R6
R7



Проекты с использованием контроллера **Arduino**

4-е издание



Материалы
на www.bhv.ru

- исходные коды программ проектов из книги
- исходные коды библиотек
- описания и спецификация электронных компонентов

Arduino — проекты любой сложности легко и быстро!

УДК 004
ББК 32.973.26
П29

Петин В. А.

П29 Проекты с использованием контроллера Arduino. — 4-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2021. — 560 с.: ил. — (Электроника)

ISBN 978-5-9775-6711-4

Рассмотрены основные платы Arduino и платы расширения (шилды), добавляющие функциональность основной плате. Подробно описан язык и среда программирования Arduino IDE. Приведены практические проекты с использованием контроллеров семейства Arduino в области робототехники, погодных метеостанций, "умного дома", вендинга, телевидения, беспроводной связи (bluetooth, радиопульт, связь с устройствами Android) и др. Все проекты сопровождаются схемами и листингами. На сайте издательства размещен архив с исходными кодами программ и библиотек, описаниями и спецификациями электронных компонентов.

В четвертом издании рассмотрены новые платы Arduino MKR и Nano 33, новые платы расширения, светодиодные матрицы, протокол MQTT, RFID-идентификация, GPS-трекер, Яндекс.Карты и проекты в области нейронных сетей.

Для читателей, интересующихся современной электроникой

УДК 004
ББК 32.973.26

Группа подготовки издания:

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

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

ISBN 978-5-9775-6711-4

© ООО "БХВ", 2021
© Оформление. ООО "БХВ-Петербург", 2021

Оглавление

Предисловие	13
Для кого и о чем эта книга?	13
Структура книги.....	13
Благодарности.....	14
ЧАСТЬ I. ARDUINO — ОБЩИЙ ОБЗОР.....	15
Глава 1. Введение в Arduino	17
1.1. Arduino — что это?.....	17
1.2. В чем преимущество Arduino?	18
1.3. Новые тенденции и перспективы развития Arduino	18
Глава 2. Платы семейства Arduino и платы расширения для них	20
2.1. Обзор плат семейства Arduino.....	20
2.2. Arduino Uno	21
2.3. Arduino Nano.....	22
2.4. Arduino Pro Mini	23
2.5. Arduino LilyPad.....	24
2.6. Arduino Mega2560	25
2.7. Arduino Leonardo	26
2.8. Arduino Due.....	27
2.9. Arduino Yún.....	28
2.10. Arduino MKR WiFi 1010	29
2.11. Arduino Nano 33.....	30
2.12. Платы расширения для Arduino	31
ЧАСТЬ II. СРЕДЫ РАЗРАБОТКИ И ЯЗЫК ПРОГРАММИРОВАНИЯ ПЛАТ ARDUINO	33
Глава 3. Среда разработки Arduino IDE.....	35
3.1. Установка Arduino IDE в Windows.....	35
3.2. Установка Arduino IDE в Linux	36
3.3. Настройка среды Arduino IDE	37

Глава 4. Облачная среда разработки Arduino Create	41
4.1. Начало работы в среде Arduino Create	41
4.2. Загрузка и выполнение пробного скетча	43
Глава 5. Программирование плат 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. <i>;</i> (semicolon, точка с запятой)	53
5.3.2.2. <i>{}</i> (curly braces, фигурные скобки)	53
5.3.2.3. Комментарии: <i>//</i> (single line comment, однострочный), <i>/* */</i> (multi-line comment, многострочный)	54
5.3.3. Арифметические операторы	54
5.3.3.1. <i>=</i> (assignment, оператор присваивания)	54
5.3.3.2. <i>+</i> (сложение), <i>-</i> (вычитание), <i>*</i> (умножение), <i>/</i> (деление)	55
5.3.3.3. <i>%</i> (modulo)	55
5.3.4. Операторы сравнения	55
5.3.5. Логические операторы	55
5.3.5.1. <i>&&</i> (логическое И)	55
5.3.5.2. <i> </i> (логическое ИЛИ)	55
5.3.5.3. <i>!</i> (логическое отрицание)	56
5.3.6. Унарные операторы	56
5.3.6.1. <i>++</i> (увеличение значения), <i>--</i> (уменьшение значения)	56
5.3.6.2. <i>+=</i> , <i>-=</i> , <i>*=</i> , <i>/=</i>	56
5.4. Данные	56
5.4.1. Типы данных	56
5.4.1.1. <i>boolean</i>	56
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>	63
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	68
5.5.3.1. Функция <i>tone()</i>	68
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.....	74
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	77
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>	78
5.5.8. Операции с битами и байтами	79
5.5.8.1. Функция <i>lowByte()</i>	79
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>	80
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>	81
5.5.9.2. Функция <i>detachInterrupt</i>	82
5.6. Управление портами через регистры ATmega	83

ЧАСТЬ III. ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ ARDUINO 85

Глава 6. Проекты для изучения выводов Arduino 87

6.1. Цифровые выходы — «бегущий огонь» на светодиодах	87
6.1.1. Подключение светодиода к выводу Arduino	88
6.1.2. Подключение к плате Arduino 8 светодиодов	92
6.2. Цифровые входы — управляем светодиодами с помощью кнопок	97
6.2.1. Подключение кнопки к плате Arduino	97
6.2.2. Управление кнопками количеством горящих светодиодов	104
6.3. Аналоговые входы — светодиодный индикатор аналоговых значений	107
6.3.1. Подключение потенциометра к плате Arduino	109
6.3.2. Вывод показаний потенциометра на светодиодную шкалу	112
6.4. ШИМ — радуга на RGB-светодиоде	114
6.4.1. Подключение к плате Arduino RGB-светодиода	115
6.5. Светодиодные индикаторы	118
6.5.1. Подключение к плате Arduino семисегментного индикатора	118
6.6. Расширение цифровых выходов — микросхема 74HC595	122
6.6.1. Подключение к плате Arduino сдвигового регистра 74HC595	123
6.7. Расширение цифровых входов и выходов — микросхема MCP23017	128
6.8. Расширение аналоговых входов — мультиплексор CD4051	129

Глава 7. Использование библиотек в проектах Arduino 132

7.1. Установка библиотек	132
7.1.1. Установка библиотеки через Менеджер библиотек	133
7.1.2. Установка библиотеки из ZIP-архива	133
7.1.3. Установка библиотеки вручную	135
7.2. Подключение библиотеки	135
7.3. Создание собственной библиотеки	136
7.3.1. Создание заголовочного файла <i>D5651.h</i>	136
7.3.2. Создание файла реализации <i>D5651.cpp</i>	137
7.3.3. Создание файла <i>keywords.txt</i>	138

Глава 8. Arduino и последовательный порт UART	140
8.1. Библиотека Serial	140
8.1.1. Функция <i>Serial.begin</i>	140
8.1.2. Функция <i>Serial.print</i>	141
8.1.3. Функция <i>Serial.println</i>	141
8.1.4. Функция <i>Serial.write</i>	141
8.1.5. Функция <i>Serial.available</i>	142
8.1.6. Функция <i>Serial.read</i>	142
8.2. Использование UART для отладки программ.....	142
8.2.1. Подключение к плате Arduino нескольких кнопок.....	142
8.3. Использование UART для установки параметров	145
8.4. Библиотека SoftwareSerial.....	149
8.5. Соединение по UART двух плат Arduino	150
Глава 9. Подключение датчиков к плате Arduino.....	153
9.1. Подключение аналоговых датчиков.....	153
9.1.1. Подключение к плате Arduino аналогового датчика температуры LM335	154
9.2. Подключение датчиков по протоколу 1-Wire	156
9.2.1. Подключение к плате Arduino цифрового датчика температуры DS18B20.....	157
9.3. Подключение датчиков влажности и температуры DHT	163
9.4. Подключение датчиков по протоколу I ² C	167
9.4.1. Подключение к плате Arduino датчика интенсивности света BH1750	169
Глава 10. Использование дисплеев в проектах Arduino.....	173
10.1. Символьные дисплеи на микроконтроллере HD44780.....	173
10.1.1. Функция <i>begin()</i>	176
10.1.2. Функция <i>clear()</i>	176
10.1.3. Функция <i>home()</i>	177
10.1.4. Функция <i>setCursor()</i>	177
10.1.5. Функция <i>write()</i>	177
10.1.6. Функция <i>print()</i>	177
10.1.7. Функция <i>cursor()</i>	178
10.1.8. Функция <i>noCursor()</i>	178
10.1.9. Функция <i>blink()</i>	178
10.1.10. Функция <i>noBlink()</i>	178
10.1.11. Функция <i>display()</i>	179
10.1.12. Функция <i>noDisplay()</i>	179
10.1.13. Функция <i>scrollDisplayLeft()</i>	179
10.1.14. Функция <i>scrollDisplayRight()</i>	179
10.1.15. Функция <i>autoscroll()</i>	179
10.1.16. Функция <i>noAutoscroll()</i>	180
10.1.17. Функция <i>leftToRight()</i>	180
10.1.18. Функция <i>rightToLeft()</i>	180
10.1.19. Функция <i>createChar()</i>	180
10.2. Подключение дисплеев на контроллере HD44780 по протоколу I ² C.....	181
10.2.1. Вывод на ЖК-дисплей данных с датчика, работающего по протоколу I ² C.....	183
10.3. Графический дисплей Nokia	186
10.4. OLED-дисплеи	192
10.4.1. Электронные часы на OLED-дисплее	193

10.5. Дисплеи Nextion.....	197
10.5.1. Создание нового проекта для дисплея Nextion	198
10.5.2. Прошивка дисплея через UART	202
10.5.3. Прошивка дисплея с помощью карты microSD	203
10.5.4. Подключение дисплея Nextion к плате Arduino	203
10.6. Светодиодные матрицы	207
10.6.1. Четырехразрядная светодиодная матрица.....	207
10.6.2. Вывод на четырехразрядную светодиодную матрицу спрайтов и символов	210
10.6.3. Бегущая строка на четырехразрядной светодиодной матрице	213
10.6.4. Русификация «бегущей строки» на четырехразрядной светодиодной матрице	215
10.6.5. Матрица 16×16 на светодиодах WS2812.....	218
10.6.6. Arduino-библиотека Adafruit Neopixel	220
10.6.7. Графический аудиоспектроанализатор на матрице 16×16 светодиодов WS2812.....	221
Глава 11. Подключение к Arduino исполнительных устройств	225
11.1. Подключение к плате Arduino электромагнитного или твердотельного реле.....	225
11.2. Подключение к плате Arduino электродвигателя постоянного тока	229
11.2.1. Управление двигателем с помощью транзистора.....	229
11.3. Управление двигателями с помощью драйвера	231
11.4. Подключение к плате Arduino сервопривода	234
11.4.1. Использование сервопривода в проекте звуковой сигнализации.....	237
11.5. Подключение к плате Arduino шагового двигателя.....	239
11.5.1. Управление дроблением шага и направлением вращения шагового двигателя с платы Arduino	242
11.6. Подключение к плате Arduino бесколлекторного двигателя	245
Глава 12. Arduino и беспроводная связь.....	248
12.1. ИК-управление	248
12.1.1. Управление сервоприводом с помощью ИК-связи.....	251
12.2. Радиомодули для частоты 433 МГц	254
12.2.1. Управление светодиодом платы Arduino с другой такой же платы по радиоканалу 433 МГц	254
12.3. Радиомодули NRF24L01	257
12.3.1. Организация связи между двумя платами Arduino с использованием модулей NRF24L01	259
12.4. Использование Arduino с аппаратурой радиуправления	264
12.4.1. Принципы формирования радиосигнала	266
12.4.2. Организация связи приемника с передатчиком	267
12.4.3. Разработка скетча для приема платой Arduino команд передатчика	268
12.5. Arduino и Bluetooth	271
Глава 13. Arduino и Интернет вещей	279
13.1. Подключение к Интернету с помощью платы расширения Ethernet shield	279
13.1.1. Получение IP-адреса по DHCP.....	280
13.1.2. Отправка данных на сайт «Народный мониторинг» через Ethernet shield.....	282
13.2. Подключение к Интернету с помощью платы расширения GSM/GPRS shield	289
13.2.1. Отправка и получение SMS-сообщений с помощью GSM/GPRS shield.....	291
13.2.2. Отправка данных на сайт «Народный мониторинг» через GSM/GPRS shield.....	295

13.3. Протокол MQTT	299
13.3.1. Отправка данных по протоколу MQTT	300
13.3.2. Получение данных по протоколу MQTT	304
13.3.3. Android-приложение IoT MQTT Dashboard	307
13.4. Плата Arduino MKR WiFi 1010 для проектов IoT	308
13.5. Отправка данных в облако Arduino IoT Cloud и получение их оттуда	312
Глава 14. RFID-идентификация	318
14.1. Считыватель RFID RC522	318
14.2. Организация контроля доступа по RFID-меткам	321
14.3. Запись информации на RFID-метку	323
14.4. Проект «Говорящая фотография»	330
Глава 15. Специальные возможности отдельных плат Arduino	334
15.1. Использование Arduino Leonardo в качестве USB-устройства	334
15.1.1. Arduino Leonardo: имитация клавиатуры	335
15.1.2. Блокируем клавиатуру с наступлением темноты	337
15.1.3. Arduino Leonardo: имитация компьютерной мыши	337
15.2. Плата Arduino Esplora	340
15.2.1. Arduino Esplora: установка цветов RGB-светодиода	343
15.2.2. Arduino Esplora: создание игры	344
15.3. Плата Arduino LilyPad	362
15.4. Плата Arduino Yún	367
15.4.1. Плата расширения Arduino Yún shield	367
15.4.2. Arduino Yún shield: управляем веб-камерой	369
Глава 16. Взаимодействие Arduino с другими программируемыми системами	372
16.1. Использование Arduino в проектах LEGO	372
16.1.1. Получение микрокомпьютером LEGO данных с датчика влажности и температуры DHT11, подключенного к плате Arduino	373
16.2. Arduino в проектах ROS	376
16.2.1. Установка ROS	377
16.2.2. Узлы и темы в ROS	377
16.2.3. Пакет <i>rosserial</i>	378
16.2.4. Подготовка сообщения (publisher) на Arduino	379
16.2.5. Создание подписки (subscriber) на Arduino	383
16.2.6. Связь через ROS двух плат Arduino	385
16.3. Arduino и Raspberry pi	386
16.3.1. Установка WebIOPi на Raspberry Pi	387
16.3.2. Обмен данными по последовательному порту	389
16.3.3. Управление движущейся платформой на базе Arduino по web-интерфейсу Raspberry Pi	390
Глава 17. Программирование в среде Arduino IDE других плат	398
17.1. ESP8266 — микроконтроллер с интерфейсом Wi-Fi	398
17.1.1. Установка Arduino IDE для работы с ESP8266	399
17.1.2. Печать курса валют на термопринтере в проекте Интернета вещей	402

17.2. Z-Uno — плата для прототипирования устройств Z-Wave	409
17.2.1. Установка Arduino IDE для Z-Uno	411
17.2.2. Подключение к плате Z-Uno датчика влажности DHT11	414
ЧАСТЬ IV. ИНТЕРЕСНЫЕ ПРОЕКТЫ НА ARDUINO	417
Глава 18. Умная теплица.....	419
18.1. Мониторинг климатических параметров умной теплицы.....	420
18.2. Индикация показаний умной теплицы.....	425
18.3. Организация полива, обдува и освещения в умной теплице.....	431
18.4. Переносим функции мониторинга и управления теплицей на устройство с ОС Android.....	440
18.5. Создаем собственное мобильное приложение для управления умной теплицей.....	447
18.6. Превращаем нашу умную теплицу в объект Интернета вещей	451
Глава 19. GPS-трекер и онлайн-сервис поиска стоянок	461
19.1. Подключение GPS-модуля к плате Arduino	461
19.2. Отправка данных по GPRS на сервер	464
19.3. Создание веб-страницы с использованием API Яндекс.Карт	472
Глава 20. Проекты для вендинга: купюроприемник, монетоприемник, разменный автомат.....	476
20.1. Купюроприемник ICT серий A7 и V7.....	476
20.1.1. Подключение купюроприемника ICT V7 к плате Arduino.....	480
20.1.2. Скетч для получения номинала принимаемой купюры	482
20.2. Монетоприемник СН-926	483
20.2.1. Настройка монетоприемника	484
20.2.2. Калибровка монетоприемника	485
20.3. Разменный автомат (хоппер) Cube Hopper MK II.....	485
20.3.1. Подключение хоппера к плате Arduino	487
20.3.2. Программирование хоппера	487
Глава 21. Создание управляющей платы для автомойки самообслуживания	492
21.1. Блок приема денежных средств и блок индикации	492
21.2. Выбор программ работы мойки	496
21.3. Отсчет времени выполнения программы. Реализация паузы	499
21.4. Режим администратора. Установка параметров	501
Глава 22. Arduino и интерфейс USB: управление роботами.....	503
22.1. Интерфейс USB.....	503
22.2. Плата расширения USB Host Shield	504
22.3. HID-устройства USB	505
22.4. Подключение HID-мыши USB	508
22.5. Управление роботом с помощью руля Defender.....	508
22.6. Управление роботом с помощью геймпада Defender.....	518

Глава 23. Камера PiXu: реализация компьютерного зрения	525
23.1. Настройка камеры	526
23.2. Подключение камеры PiXu к плате Arduino	527
23.3. Организация слежения камерой за объектом.....	529
Глава 24. Проекты на плате Nano 33 BLE Sense	532
24.1. Начало работы с платой Nano 33 BLE Sense.....	533
24.2. Bluetooth Low Energy (BLE).....	536
24.3. Отправка данных с датчиков платы Nano 33 BLE Sense по BLE	537
24.4. Библиотека TensorFlow Lite	540
24.5. Пример создания классификатора объектов с обучением	541
24.5.1. Сбор данных.....	542
24.5.2. Обучение модели.....	544
24.5.3. Скетч классификатора для запуска нейронной сети на плате Nano 33 BLE Sense	549
ПРИЛОЖЕНИЯ	553
Приложение 1. Перечень использованных источников	555
Приложение 2. Описание электронного архива.....	556
Предметный указатель	557



ГЛАВА 1

Введение в Arduino

1.1. Arduino — что это?

Появление первых микроконтроллеров ознаменовало начало новой эры в развитии микропроцессорной техники. Сосредоточение в одном корпусе большинства системных устройств сделало микроконтроллер подобным обычному компьютеру. В отечественной литературе они даже назывались однокристалльными микроЭВМ. Соответственно и желание использовать микроконтроллеры как обычные компьютеры возникло практически одновременно 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. Потребуется 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. При этом 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 имеющейся платы, ну и, конечно, фантазией разработчика.

1.3. Новые тенденции и перспективы развития Arduino

Классический форм-фактор Arduino является знаковым, но он постепенно отмирает. Упо-подобные платы, которые стали стандартом де-факто в мире любителей радиоэлектроники, медленно уходят на пенсию.

В настоящее время внимание Arduino в большей степени направлено на Интернет вещей (IoT). Сначала появилась серия плат MKR, предоставляющая различные варианты подключения к сети и управления питанием, что побуждает людей использовать их в качестве единого стандартного формата для проектов IoT. Платы MKR повысили уровень стандартизации процесса проектирования, облегчая жизнь разработчика, пытающегося продать свой конечный продукт на рынке.

Затем появилась серия Nano 33. Ее платы имеют практически те же размеры, что и оригинал Nano, но несут на борту новые процессоры и отличаются низким энергопотреблением. Это и плата Nano Every для обычных проектов на базе мощного микроконтроллера Microchip ATmega4809 с микросхемой ATSAMD11 ARM Cortex-M0 и процессором для USB-to-serial связи, и Nano IoT на микросхеме ATSAMD21 ARM Cortex-M0 с поддержкой Wi-Fi и Bluetooth LE и с крипточипом для безопасного хранения сертификатов и общих ключей, совместимая с новым облаком Arduino IoT, и Nano 33 BLE — энергоэффективные модули для работы по технологии Bluetooth BLE.

Эти новые Nano-платы могут быть совместимы с макетом, но больше предназначены для поверхностного монтажа в виде модулей. Они на самом деле нацелены на мелких производителей, которые изготовили прототипы на классических или МКR-платах и хотят довести свой продукт до полномасштабного производства.

На выставке CES 2020 компания Arduino представила новое семейство мощных устройств Arduino Portenta, разработанное для требовательных промышленных приложений, обработки задач искусственного интеллекта и для робототехники. Оно оборудовано широким спектром поддерживаемых соединений для подключения периферийных устройств.



ГЛАВА 2

Платы семейства Arduino и платы расширения для них

2.1. Обзор плат семейства Arduino

Основные версии плат Arduino представлены следующими моделями:

- ❑ **Due** — плата на базе 32-битного ARM микропроцессора Atmel SAM3X8E ARM Cortex-M3;
- ❑ **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 плата, используемая как макет. Она подключается к компьютеру при помощи кабеля USB Mini-B;
- ❑ **Mega** — версия серии плат Mega на базе микроконтроллера ATmega1280;
- ❑ **Mega2560** — плата на базе микроконтроллера ATmega2560 с использованием микроконтроллера ATmega8U2 для последовательного соединения по USB-порту;
- ❑ **Mega ADK** — версия платы Mega2560 с поддержкой интерфейса USB-host для связи со смартфонами на Android и другими устройствами с интерфейсом USB;
- ❑ **Arduino BT** — плата с модулем Bluetooth для беспроводной связи и программирования;
- ❑ **LilyPad** — плата, разработанная для носимых устройств (может зашиваться в ткань);
- ❑ **Fio** — плата, разработанная для беспроводных применений. Содержит разъем для радио XBee, разъем для батареи Li-Po и встроенную схему подзарядки;
- ❑ **Mini** — самая маленькая плата Arduino;
- ❑ **Pro** — плата, разработанная для опытных пользователей (может являться частью большего проекта);

- ❑ **Pro Mini** — как и плата Pro, разработана для опытных пользователей, которым требуется низкая цена, меньшие размеры и дополнительная функциональность;
- ❑ **Arduino Yún** — плата на основе Arduino Leonardo, объединяющая в себе достоинства двух платформ, поддерживаемых Свободным сообществом: Arduino и Linux. Получившийся симбиоз предоставляет огромные возможности для использования Интернета в своих проектах;
- ❑ **Arduino MKR** — серия плат, разработанная компанией Arduino, предоставляет различные варианты подключения к сети и управления питанием, в большой степени направлена на Интернет вещей (IoT);
- ❑ **Arduino Nano 33** — серия плат Nano нового поколения. Платы имеют практически те же размеры, что и оригинал, но отличаются новыми процессорами, низким энергопотреблением и наличием беспроводных интерфейсов.

Рассмотрим более подробно некоторые из этих плат.

2.2. Arduino Uno

Плата Arduino Uno (рис. 2.1) построена на микроконтроллере ATmega328. В отличие от всех предыдущих ее версий, использовавших для связи по USB микроконтроллер FTDI USB, новая Arduino Uno использует с этой целью микроконтроллер ATmega8U2.

Характеристики платы Arduino Uno представлены в табл. 2.1.



Рис. 2.1. Плата Arduino Uno

Таблица 2.1. Характеристики платы Arduino Uno

Микроконтроллер	ATmega328
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	32 Кбайт, при этом 0,5 Кбайт используются для загрузчика
ОЗУ	2 Кбайт
EEPROM	1 Кбайт
Тактовая частота	16 МГц

2.3. Arduino Nano

Плата Nano (рис. 2.2), построенная на микроконтроллере ATmega328 (Arduino Nano 3.0) или ATmega168 (Arduino Nano 2.x), имеет небольшие размеры и может использоваться в лабораторных работах.



Рис. 2.2. Плата Arduino Nano

Arduino Nano способна получать питание через подключение USB Mini-B или от внешнего источника питания: нерегулируемого 6–20 В (вывод 30) или регулируемого 5 В (вывод 27). Источник с самым высоким напряжением выбирается автоматически.

Характеристики платы Arduino Nano представлены в табл. 2.2.

Таблица 2.2. Характеристики платы Arduino Nano

Микроконтроллер	ATmega168 или ATmega328
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 (ATmega168) или 32 Кбайт (ATmega328), при этом 2 Кбайт используются для загрузчика
ОЗУ	1 Кбайт (ATmega168) или 2 Кбайт (ATmega328)
EEPROM	512 байтов (ATmega168) или 1 Кбайт (ATmega328)
Тактовая частота	16 МГц

2.4. Arduino Pro Mini

Плата Arduino Pro Mini (рис. 2.3) построена на микроконтроллере ATmega168.



Рис. 2.3. Плата Arduino Pro Mini

Характеристики платы Arduino Pro Mini представлены в табл. 2.3.

Таблица 2.3. Характеристики платы Arduino Pro Mini

Микроконтроллер	ATmega168
Рабочее напряжение	3,3 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

Таблица 2.3 (окончание)

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 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 или 5 В;
- GND — выводы заземления.

2.5. Arduino LilyPad

Плата Arduino LilyPad (рис. 2.4) разработана так, чтобы ее можно было зашить в ткань одежды вместе со встроенными источниками питания, датчиками, проводами и приводами.

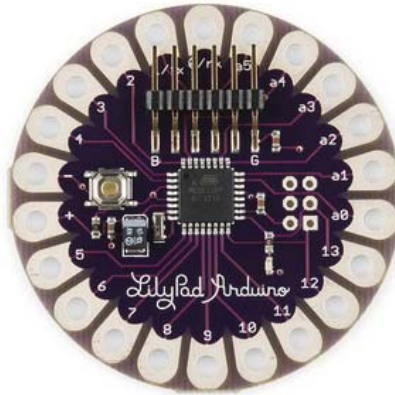


Рис. 2.4. Плата Arduino LilyPad

Характеристики платы Arduino LilyPad представлены в табл. 2.4.

Таблица 2.4. Характеристики платы Arduino LilyPad

Микроконтроллер	ATmega168 или ATmega328
Рабочее напряжение	2,7–5,5 В
Входное напряжение	2,7–5,5 В

Таблица 2.4. Характеристики платы Arduino LilyPad

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 (ATmega168) или 32 Кбайт (ATmega328), при этом 2 Кбайт используются для загрузчика
ОЗУ	1 Кбайт (ATmega168) или 2 Кбайт (ATmega328)
EEPROM	512 байтов (ATmega168) или 1 Кбайт (ATmega328)
Тактовая частота	16 МГц

2.6. Arduino Mega2560

Плата Arduino Mega2560 (рис. 2.5) построена на микроконтроллере ATmega2560.



Рис. 2.5. Плата Arduino Mega2560

Характеристики платы Arduino Mega2560 представлены в табл. 2.5.

Таблица 2.5. Характеристики платы Arduino Mega2560

Микроконтроллер	ATmega2560
Рабочее напряжение	5 В
Входное напряжение (рекомендуемое)	7–12 В

Таблица 2.5 (окончание)

Входное напряжение (предельное)	6–20 В
Цифровые входы/выходы	54 (14 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	16
Постоянный ток через вход/выход	40 мА
Постоянный ток для вывода 3,3 В	50 мА
Флеш-память	256 Кбайт, из которых 8 Кбайт используются для загрузчика
ОЗУ	8 Кбайт
EEPROM	4 Кбайт
Тактовая частота	16 МГц

2.7. Arduino Leonardo

Плата Arduino Leonardo (рис. 2.6) построена на базе микроконтроллера ATmega32U4, имеющего, в отличие от всех других микропроцессоров, встроенную поддержку для USB-соединения.

Характеристики платы Arduino Leonardo представлены в табл. 2.6.

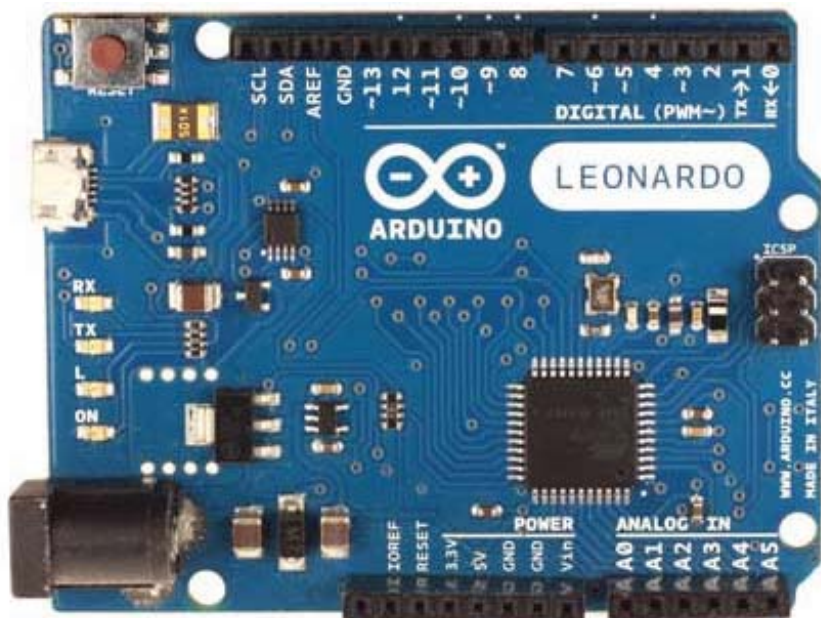


Рис. 2.6. Плата Arduino Leonardo

Таблица 2.6. Характеристики платы Arduino Leonardo

Микроконтроллер	ATmega32U4
Рабочее напряжение	5 В
Входное напряжение (рекомендуемое)	7–12 В
Входное напряжение (предельное)	6–20 В
Цифровые входы/выходы	20 (7 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	12
Постоянный ток через вход/выход	40 мА
Постоянный ток для вывода 3,3 В	50 мА
Флеш-память	32 Кбайт, из которых 4 Кбайт используются для загрузчика
ОЗУ	2 Кбайт
EEPROM	1 Кбайт
Тактовая частота	16 МГц

2.8. Arduino Due

Arduino Due (рис. 2.7) представляет собой первую плату Arduino на основе 32-битного микроконтроллера с ARM-ядром — процессора Atmel SAM3X8E ARM Cortex-M3.

В отличие от других плат Arduino, максимальное рабочее напряжение, которое выдерживают входы/выходы Arduino Due, составляет 3,3 В.

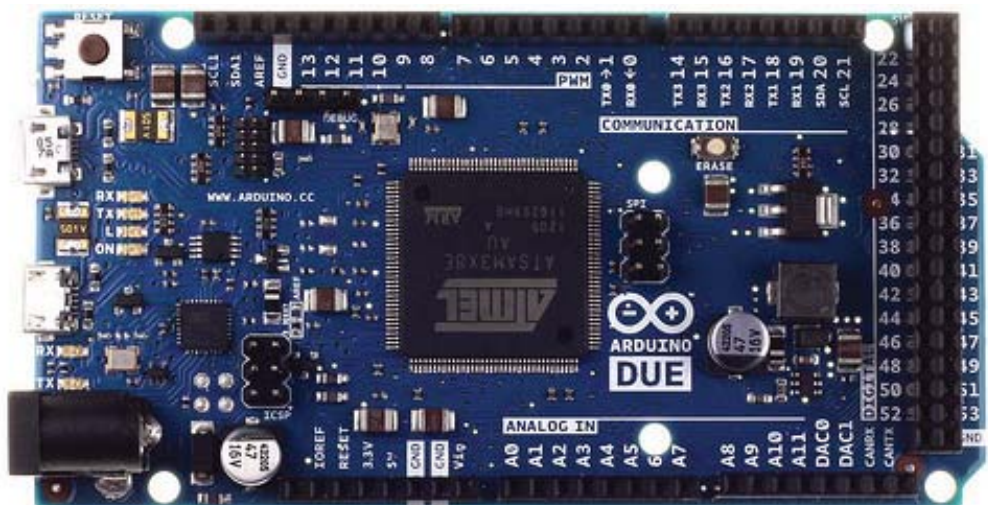


Рис. 2.7. Плата Arduino Due

Характеристики платы Arduino Due представлены в табл. 2.7.

Таблица 2.7. Характеристики платы Arduino Due

Микроконтроллер	AT91SAM3X8E
Рабочее напряжение	3,3 В
Входное напряжение (рекомендуемое)	7–12 В
Входное напряжение (предельное)	6–20 В
Цифровые входы/выходы	54 (12 из которых могут использоваться как выходы ШИМ)
Аналоговые входы	12
Аналоговые выходы	2 (ЦАП)
Постоянный ток через вход/выход	50 мА
Постоянный ток для вывода 3,3 В	800 мА
Постоянный ток для вывода 5 В	800 мА
Флеш-память	512 Кбайт
ОЗУ	96 Кбайт (два банка: 64 и 32 Кбайт)
Тактовая частота	84 МГц

2.9. Arduino Yún

Arduino Yún (рис. 2.8) — это плата на основе Arduino Leonardo, объединяющая в себе, как уже упоминалось ранее, достоинства двух платформ, поддерживаемых Свободным сообществом: Arduino и Linux. Получившийся симбиоз предоставляет огромные возможности для использования Интернета в своих проектах.

Arduino-часть платы содержит микроконтроллер ATmega32U4, работающий на частоте 16 МГц. «Распиновка» Arduino Yún аналогична Arduino Leonardo, поэтому вместе с Arduino Yún вы можете использовать большинство плат расширения Arduino.

Linux-часть платы Arduino Yún использует микрокомпьютер Atheros AR9331, работающий под управлением операционной системы Linino — специально подготовленной версии OpenWRT (популярного дистрибутива Linux для встраиваемых систем).

Микрокомпьютер Atheros AR9331 работает на частоте 400 МГц, имеет 64 Мбайт оперативной и 16 Мбайт флеш-памяти, встроенные интерфейсы Wi-Fi и Ethernet, USB-host и слот для карты microSD. Linino содержит в себе пакетный менеджер opkg, который позволяет устанавливать большое количество Linux-приложений, а также интерпретатор языка Python 2.7, с помощью которого вы можете писать для Linino свои приложения.



Рис. 2.8. Плата Arduino Yún

Память для Linux-приложений может быть расширена с помощью съемного носителя (карты microSD или USB-флешки).

2.10. Arduino MKR WiFi 1010

Arduino MKR WiFi 1010 (рис. 2.9) — это платформа из семейства MKR для разработчиков с минимальным опытом работы в сети, желающих создавать проекты IoT. Плата состоит из трех основных блоков:

- микроконтроллера ATSAM21G18 с тактовой частотой 48 МГц и вычислительным ядром Cortex-M0+. Благодаря своей 32-битной архитектуре контроллер справляется с ресурсоемкими математическими вычислениями, обрабатывает аналоговые сигналы с большой точностью и воспроизводит музыку через встроенный ЦАП;
- модуля беспроводной связи NINA-W10 от компании u-blox со встроенным чипом ESP32. Модуль обеспечивает беспроводной обмен данными в диапазоне 2,4 ГГц по протоколам Wi-Fi и Bluetooth. Регулировка выходной мощности позволяет достичь оптимального соотношения между дальностью связи, скоростью передачи данных и энергопотреблением;
- крипточипа ECC508 для защиты передаваемых данных с использованием шифрования SHA-256. Крипточип дает возможность выполнять HTTPS-запросы без нагрузки на основной процессор.

Поскольку платформа выполнена на архитектуре ARM Cortex-M0+, «родное» напряжение ее выводов — 3,3 В. То есть выходы для логической единицы выдают

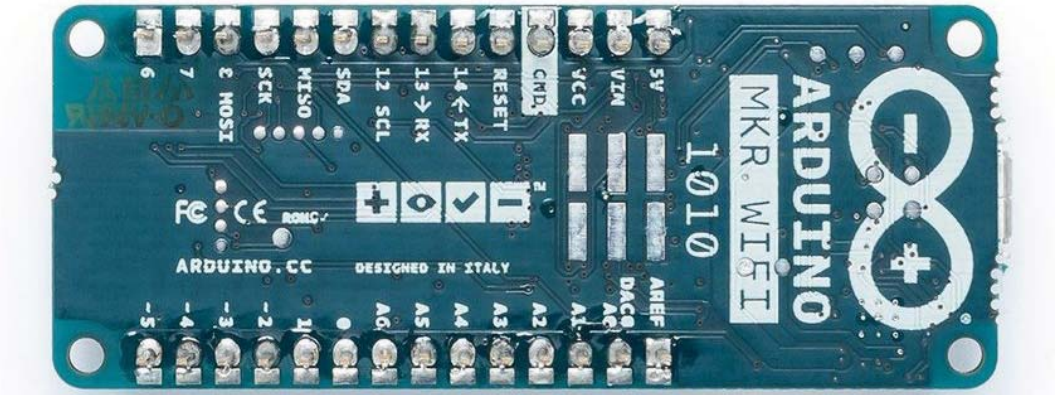


Рис. 2.9. Плата Arduino MKR WiFi 1010

3,3 В и рассчитаны принимать напряжение не более этого. Большее напряжение повредит микроконтроллер.

2.11. Arduino Nano 33

Arduino Nano 33 — серия плат Nano нового поколения. Платы, как уже отмечалось ранее, имеют практически те же размеры, что и оригинал, но отличаются новыми процессорами, низким энергопотреблением и наличием беспроводных интерфейсов. К настоящему моменту выпущено четыре платы этой серии (рис. 2.10): Arduino Nano Every, Nano 33 IoT, Nano 33 BLE и Nano 33 BLE Sense.

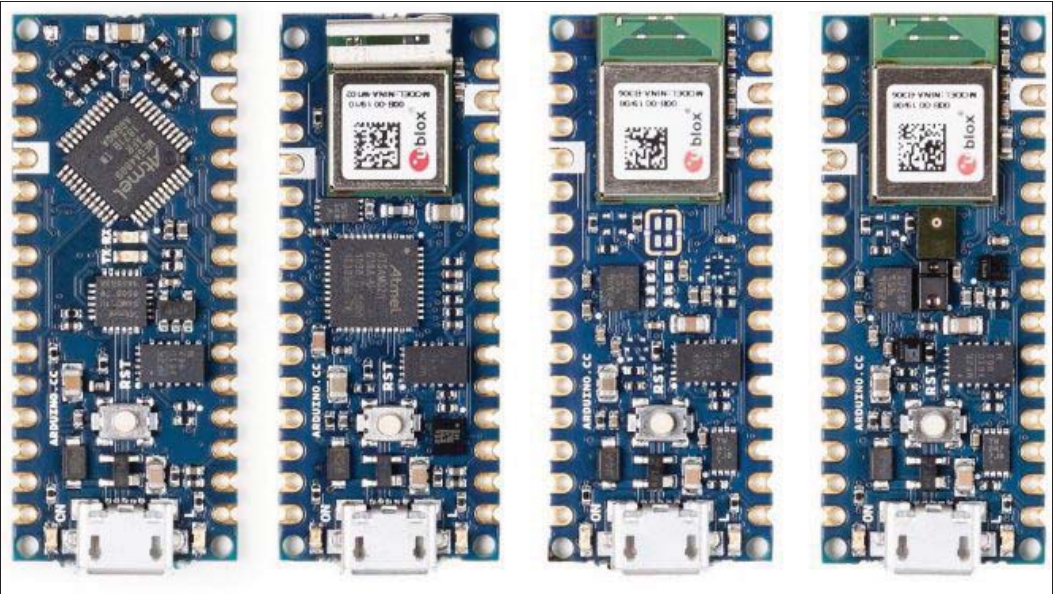


Рис. 2.10. Семейство Arduino Nano 33:
Nano Every, Nano 33 IoT, Nano 33 BLE, Nano 33 BLE Sense (слева направо)

- В отличие от оригинальной платы Arduino Nano, несущей на борту 8-битный микроконтроллер Microchip ATmega328P, плата Arduino Nano Every построена на основе гораздо более мощного микроконтроллера Microchip ATmega4809 с микросхемой ATSAM11 ARM Cortex M0 и процессором для USB-to-serial связи.
- Плата Arduino Nano 33 IoT основана на микросхеме ATSAM21 ARM Cortex-M0 с поддержкой Wi-Fi и Bluetooth LE, предоставляемой встроенным микроконтроллером Espressif ESP32 в форме модуля NINA-W102 u-blox. Плата также оснащена 9-осевым инерциальным измерительным устройством (Inertial measurement unit, IMU), крипточипом для безопасного хранения сертификатов и общих ключей и совместима с новым облаком Arduino IoT.
- В отличие от Nano Every и Nano 33 IoT, плата Arduino Nano 33 BLE не основана на процессоре с микрочипом. Вместо этого она оснащена модулем u-blox NINA-B306, который, в свою очередь, построен на базе платформы Nordic nRF52840 архитектуры ARM Cortex-M4F. Плата Nano 33 BLE также несет на борту 9-осевой IMU.
- Плата Nano 33 BLE Sense построена на том же модуле u-blox NINA-B306, но — в дополнение к 9-осевой IMU — поставляется с гораздо большим набором датчиков: давления, влажности, температуры и света, а также датчиком жестов и встроенным микрофоном.

2.12. Платы расширения для Arduino

Большую популярность плата Arduino приобрела не только из-за низкой стоимости, легкости разработки и программирования, но главным образом благодаря наличию плат расширения (так называемых *шилдов*), добавляющих Arduino дополнительную функциональность. Шилды (кроме маленьких модулей и платы LilyPad) подключаются к Arduino с помощью имеющихся на них штыревых разъемов (рис. 2.11).

Существует множество различных по функциональности шилдов — от простейших, предназначенных для макетирования, до сложных, представляющих собой отдельные многофункциональные устройства.

Далее приведено краткое описание некоторых шилдов:

- **Ethernet Shield** — обеспечивает подключение к Интернету;
- **XBee Shield** — обеспечивает при помощи модуля Maxstream Xbee Zigbee беспроводную связь нескольких устройств Arduino;
- **MicroSD Shield** — обеспечивает запись данных на карты microSD;
- **MP3 Shield** — плата для воспроизведения звука в форматах Ogg Vorbis, MP3, AAC, WMA и MIDI и записи звука в формате Ogg Vorbis;
- **Motor Shield** — обеспечивает управление двигателями постоянного тока;
- **GSM/GPRS Shield** — позволяет отправлять SMS-сообщения, делать телефонные звонки, обмениваться данными по GPRS;

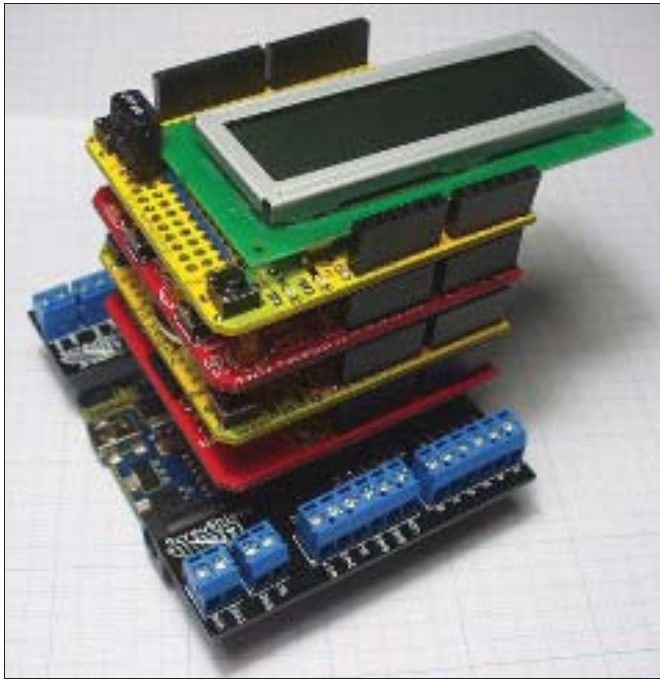


Рис. 2.11. Модульная структура установки плат расширения для Arduino

□ **Cosmo WiFi Connect** — обеспечивает организацию беспроводной сети стандарта IEEE 802.11b/g.

Существуют также шилды: **Video Overlay Shield** — для наложения текста на аналоговое видео, **EasyVR Arduino Shield** — многоцелевой модуль распознавания речи, **Music Shield** — профессиональный аудиокодек и др.

Количество плат расширения (шилдов) постоянно растет. Ознакомьтесь с их списком можно на официальном сайте проекта Arduino по адресу: <http://www.arduino.cc/playground/Main/SimilarBoards#goShie>.

ЧАСТЬ II



Среды разработки и язык программирования плат Arduino

Глава 3. Среда разработки Arduino IDE

Глава 4. Облачная среда разработки Arduino Create

Глава 5. Программирование плат Arduino



ГЛАВА 3

Среда разработки Arduino IDE

Разработка собственных приложений на базе плат, совместимых с архитектурой Arduino, осуществляется в бесплатной среде разработки Arduino IDE. Среда предназначена для написания, компиляции и загрузки собственных программ в память микроконтроллера, установленного на плате Arduino-совместимого устройства. Основу среды разработки составляет язык Processing/Wiring — это фактически обычный язык C++, дополненный простыми и понятными функциями для управления вводом/выводом на контактах. Существуют версии среды для операционных систем Windows, macOS и Linux.

Последнюю версию среды Arduino (1.8.13) и бета-версию Arduino (с экспериментальными возможностями) можно скачать со страницы загрузки официального сайта: <http://arduino.cc/en/Main/Software>.

ПРИМЕЧАНИЕ

В главе 4 вашему вниманию представлена новая облачная среда разработки Arduino Create.

3.1. Установка Arduino IDE в Windows

Отправляемся на страницу <http://arduino.cc/en/Main/Software> (рис. 3.1), выбираем версию для операционной системы Windows и скачиваем архивный файл. Он занимает чуть более 80 Мбайт и содержит все необходимое, в том числе и драйверы. По окончании загрузки распаковываем скачанный файл в удобное для себя место.

Теперь необходимо установить драйверы. Подключаем плату Arduino к компьютеру. На ней должен загореться индикатор питания — зеленый светодиод. Windows начинает попытку установки драйвера, которая заканчивается сообщением: **Программное обеспечение драйвера не было установлено**.

Открываем Диспетчер устройств. В составе устройств находим значок Arduino Uno — устройство отмечено восклицательным знаком. Щелкаем правой кнопкой мыши на значке Arduino Uno и в открывшемся окне выбираем пункт **Обновить драйверы** и далее пункт **Выполнить поиск драйверов на этом компьютере**. Указываем путь к драйверу — ту папку на компьютере, куда распаковывали скачанный

архив. Пусть это будет папка `drivers` каталога установки Arduino. Игнорируем все предупреждения Windows и получаем в результате сообщение: **Обновление программного обеспечения для данного устройства завершено успешно**. В заголовке окна будет указан и COM-порт, на который установлено устройство.

Осталось установить и запустить среду разработки Arduino IDE.

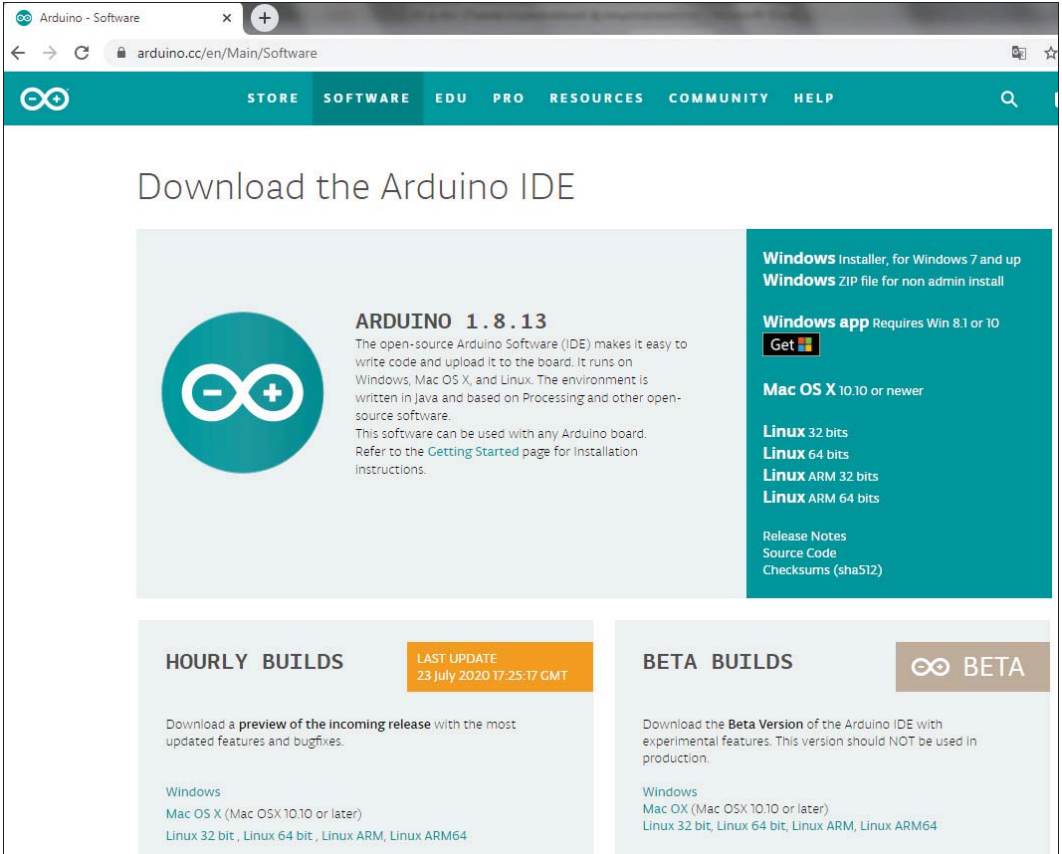


Рис. 3.1. Страница загрузки официального сайта Arduino

3.2. Установка Arduino IDE в Linux

В Linux Ubuntu среда Arduino IDE устанавливается просто — она находится в репозитории стандартных приложений Linux. Выбираем Arduino IDE из списка доступных программ в меню Ubuntu: **Приложения | Центр приложений Ubuntu | Загрузить приложение**. В списке разделов выбираем **Инструменты разработчика**, в списке следующего уровня — **Все приложения** и в следующем открывшемся списке — **Arduino IDE** (рис. 3.2). Щелкаем левой кнопкой мыши на значке этой программы — справа от нее появляется кнопка **Установить**, нажимаем на эту кнопку, и среда устанавливается автоматически.