

Валерий Яценков

от **Arduino** до **Omega**: платформы для мейкеров шаг за шагом

Санкт-Петербург
«БХВ-Петербург»
2018

УДК 004
ББК 32.973.26
Я92

Яценков В. С.

Я92 От Arduino до Omega: платформы для мейкеров шаг за шагом. — СПб.: БХВ-Петербург, 2018. — 304 с.: ил. — (Электроника)

ISBN 978-5-9775-3863-3

Рассмотрен ряд современных программно-аппаратных платформ для любительского творчества. Отобраны платформы простые для понимания новичками, с низкой ценой стартового комплекта, но в то же время производительные и расширяемые, популярные в среде мейкеров, от школьников и студентов до руководителей кружков и преподавателей. Описаны современные онлайн-сервисы для разработки и макетирования любительских проектов Arduino Create и Autodesk Circuits. Рассказано об обучающей платформе Arduino и среде Arduino IDE, однокристалльной системе ESP8266, платформе для Интернета вещей NodeMCU и языке Lua, микрокомпьютере Omega2 и облачной среде Onion Cloud. Приведены примеры программ и авторских проектов полезных устройств, особое внимание уделено ошибкам и трудностям, с которыми сталкиваются новички. В файловом архиве на сайте издательства предоставлены исходные коды программ, чертежи печатных плат и принципиальных схем.

*Для читателей, интересующихся электроникой,
робототехникой, авиамоделизмом*

УДК 004
ББК 32.973.26

Группа подготовки издания:

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

Подписано в печать 31.08.17.
Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 24,51.
Тираж 1500 экз. Заказ №
"БХВ-Петербург", 191036, Санкт-Петербург, Гончарная ул., 20.

ООО "Печатное дело",
142300, МО, г. Чехов, ул. Полиграфистов, д. 1

ISBN 978-5-9775-3863-3

© ООО "БХВ", 2018
© Оформление. ООО "БХВ-Петербург", 2018

Оглавление

Предисловие	9
Как работать с этой книгой?	10
Глава 1. Платформы для творчества и обучения	13
1.1. Не бойтесь экспериментировать!	14
1.2. Совместимость на уровне периферии и протоколов	16
1.3. Общие средства разработки и языки программирования	16
1.4. Облачные сервисы обмена данными.....	17
1.5. Онлайн-лаборатории и средства разработки.....	18
Глава 2. Советы для начинающих	19
2.1. Универсальные отладочные и макетные платы	19
2.2. Монтажные провода для пайки	23
2.3. Инструменты для подготовки проводов	25
2.4. Источники питания.....	26
2.4.1. Особенности питания от порта USB	27
2.4.2. Сетевые источники питания.....	28
Линейные стабилизаторы напряжения	28
Смещение рабочего напряжения стабилизатора.....	31
Импульсные преобразователи напряжения	31
2.4.3. Химические источники тока	33
Никель-кадмиевые аккумуляторы	34
Литиевые аккумуляторы	35
2.5. Согласование логических уровней.....	39
2.6. Интерфейсы обмена данными	41
2.6.1. Последовательный интерфейс UART	41
2.6.2. Конвертер интерфейсов USB-UART.....	43
2.6.3. Последовательная шина I ² C	45
2.6.4. Последовательный интерфейс SPI	46
2.6.5. Последовательный протокол 1-Wire	47
2.7. Измерительное оборудование.....	48
2.7.1. Цифровой мультиметр.....	49
2.7.2. Цифровой осциллограф.....	49

2.8. Паяльное оборудование	51
2.9. Полезные программы и утилиты.....	53
2.9.1. PuTTY	53
2.9.2. WinSCP	54
2.9.3. Hercules	55
2.9.4. Termite.....	56
2.9.5. Notepad++	57
Глава 3. Онлайн-лаборатория Autodesk Circuits	59
3.1. Регистрация и первый проект.....	60
3.1.1. Создание макета и симуляция.....	60
3.1.2. Принципиальная электрическая схема макета	64
3.1.3. Печатная плата по схеме макета.....	64
3.2. Создание и редактирование компонентов	65
3.2.1. Создание символа компонента	67
3.2.2. Создание монтажного чертежа компонента	69
3.2.3. Работа с чужими компонентами	71
3.2.4. Доступ к своим компонентам	72
3.2.5. Рисование принципиальной схемы	72
3.3. Вывод схемы и чертежа платы на печать	73
3.3.1. Получение рисунка принципиальной схемы	73
3.3.2. Экспорт рисунка печатной платы в формате Eagle.....	73
3.3.3. Экспорт чертежа платы в формате GERBER	73
Глава 4. Среда разработки и макетирования Fritzing	75
4.1. Установка Fritzing.....	75
4.2. Создание макета схемы	76
4.3. Создание принципиальной электрической схемы	77
4.4. Разработка чертежа печатной платы.....	77
4.5. Экспорт чертежа печатной платы	78
4.6. Добавление компонентов в библиотеку.....	79
4.7. Разработка и загрузка программ	79
Глава 5. Обучающая платформа Arduino	80
5.1. Аппаратная база платформы, популярные модели	81
5.1.1. Arduino Nano	81
5.1.2. Arduino Uno	82
5.1.3. Arduino Pro Mini.....	82
5.1.4. Arduino Mega 2560.....	83
5.2. Установка драйверов USB-UART	83
5.3. Система нумерации выводов Arduino	84
5.4. Среда разработки и отладки Arduino IDE.....	85
5.4.1. Установка Arduino IDE.....	86
Установка для ОС Windows.....	86
Установка альтернативных версий IDE.....	86
Установка для ОС Linux.....	86
Установка для Mac OS X.....	87
5.4.2. Подключение платы Arduino и первые программы	88

5.4.3. Установка сторонних библиотек	92
Автоматическая установка библиотеки	93
Установка библиотеки вручную	93
5.4.4. Установка дополнительных описаний плат	94
Автоматическая установка описания	94
Установка описания вручную	95
5.4.5. Сетевой модуль расширения Dragino Yun	96
Почему именно Dragino Yun?	97
Технические характеристики Dragino Yun v2.4	97
Особенности питания шилда Dragino Yun	98
Добавление новых плат в Arduino IDE	99
Подключение к компьютеру для настройки	100
Функции кнопки сброса Dragino Yun	101
Обновление прошивки	102
Базовые настройки	103
Определение типа базовой платы	104
Загрузка скетча через сеть из Arduino IDE	105
Автоматическое обновление скетча	105
Использование консоли Dragino Yun для вывода сообщений	107
Глава 6. Облачная среда разработки Arduino Create	110
6.1. Подготовка среды Arduino Create	111
6.2. Онлайн-редактор Arduino Web Editor	112
Sketchbook	113
Examples	113
Libraries	113
Serial Monitor	114
Help	114
Preferences	114
6.3. Подключение платы Arduino и первая программа	115
6.4. Облачный сервис Arduino Cloud	116
6.5. Библиотека проектов Arduino Project Hub	117
Глава 7. Примеры программ и проектов для Arduino	119
7.1. Использование системного времени Linux	119
7.2. Сохранение данных на карту памяти	122
7.3. Сохранение данных на USB-накопитель	125
7.4. Сохранение данных в таблицу MySQL	125
7.5. Сервис Temboo и передача данных в Google Spreadsheet	128
7.6. Анализатор эфира в диапазоне 2,4 ГГц	137
7.6.1. Модуль радиоприемника	138
7.6.2. Модуль дисплея	139
7.6.3. Модуль Arduino	139
7.6.4. Напряжение питания и согласование логических уровней	139
7.6.5. Схема электрических соединений	140
7.6.6. Алгоритм работы устройства	140
7.7. Миниатюрный монитор силовой литий-полимерной батареи	147
7.7.1. Компоненты монитора	149
7.7.2. Алгоритм работы устройства	149

7.8. Установка библиотеки ATTiny	154
7.8.1. Подключение программатора	154
7.8.2. Установка фюзов микроконтроллера	156
7.8.3. Запись прошивки	157
7.8.4. Калибровка порога срабатывания	157
Глава 8. Однокристалльная система ESP8266	159
8.1. Ученик обогнал учителя: феномен успеха ESP8266	159
8.1.1. Технические характеристики	160
8.1.2. Особенности эксплуатации	160
8.1.3. Модули на основе ESP8266	161
8.2. Расширение Arduino IDE для работы с ESP8266	163
8.2.1. Установка расширения	163
8.2.2. Особенности программирования ESP8266	163
Порты и прерывания	164
Функции задержки	164
Работа с EEPROM	165
Поддержка интерфейсов I ² C и SPI	165
Специальные функции API ESP8266	165
Специальные функции библиотеки ESP8266WiFi	166
Обращение к функциям SDK ESP8266 из скетча Arduino	166
Глава 9. Примеры программ и проектов для ESP8266	168
9.1. Получение точного времени от сервера NTP	168
9.2. Получение уведомлений от устройств на Android	173
9.2.1. Скетч для принимающего устройства	174
9.2.2. Установка и настройка приложения Android	175
9.2.3. Настройка расширенных уведомлений с приложением Tasker	176
Настройка события Tasker — новое сообщение Viber	177
9.3. Модуль управления экшн-камерой Xiaomi Yui	179
9.3.1. Аппаратная часть модуля	180
9.3.2. Прошивка модуля	181
Алгоритм работы устройства	190
Измерение длительности импульсов	191
Совместимость программы модуля с разными версиями Arduino IDE и камеры	192
9.4. Адаптация взаимодействия с сервисом Temboo	193
Глава 10. Платформа NodeMCU для Интернета вещей	198
10.1. Подготовка к использованию NodeMCU	199
10.1.1. Рекомендованное оборудование	199
10.1.2. Подключение отладочной платы к компьютеру	201
10.1.3. Обновление прошивки NodeMCU	201
Конструктор прошивок	202
Возможная проблема: сбой обновления прошивки	204
10.2. Среда разработки ESPlorer IDE	207
10.3. Пакет разработки Lua for Windows	214
10.4. Язык программирования Lua — освоим за один вечер	215
10.4.1. Типы данных	217
10.4.2. Комментарии	218

10.4.3. Переменные и преобразование типов	218
10.4.4. Работа с таблицами и массивами.....	221
10.4.5. Условный оператор <i>if</i>	222
10.4.6. Цикл с предусловием <i>while</i>	223
10.4.7. Цикл с постусловием <i>repeat</i>	223
10.4.8. Цикл с оператором <i>for</i>	223
10.4.9. Операторы <i>break</i> и <i>return</i>	224
10.4.10. Функции.....	225
10.4.11. Функции обратного вызова.....	226
Глава 11. Примеры программ и проектов для NodeMCU	228
11.1. Использование графического OLED-дисплея	229
11.1.1. Подключение дисплея	229
11.1.2. Настройка модуля U8G	230
11.1.3. Пример программы.....	230
11.1.4. Монитор курса электронной валюты биткоин	232
Загрузка программы в отладочную плату.....	235
Алгоритм работы программы	236
11.1.5. Вывод на OLED-дисплей битовых изображений	237
Создание файла битового изображения.....	237
Пример программы.....	237
11.2. Использование графического TFT-дисплея	239
11.2.1. Подключение дисплея к плате NodeMCU	240
11.2.2. Пример использования графической библиотеки.....	240
Глава 12. Микрокомпьютер Omega2.....	245
12.1. Аппаратный состав платформы.....	246
12.2. Подготовка к работе	250
12.2.1. Настройка при помощи мастера	250
12.2.2. Настройка при помощи командной строки	253
12.3. Браузерное приложение Onion Console	255
12.4. Облачный сервис Onion Cloud.....	257
12.5. Python 2.7 и дополнительные модули	259
12.5.1. Управление портами GPIO	260
12.5.2. Модуль Python SPI	263
12.5.3. Модуль Python I ² C.....	264
12.6. Файловый менеджер Midnight Commander.....	265
12.7. Расширение пространства памяти.....	266
12.7.1. Использование карты MicroSD и USB-накопителя	266
Размонтирование накопителя	267
Форматирование внешних накопителей	267
Изменение точки монтирования по умолчанию	268
12.7.2. Загрузка с внешней карты памяти	269
12.7.3. Свop-файл на внешнем носителе.....	271
12.7.4. Автоматическое включение свop-файла после перезагрузки	272
12.8. Особенности использования Omega2.....	274
12.8.1. Необходимость стабильного питания	274
12.8.2. Необходимость буферизации выводов	274

Глава 13. Примеры программ и проектов для Omega2.....	276
13.1. Подключение OLED-дисплея	276
13.2. Подключение модуля PWM Servo.....	281
13.3. Подключение модуля расширителя портов.....	286
13.4. Модуль светодиодной матрицы 8×8	289
13.5. Модуль семисегментных светодиодных индикаторов	292
13.6. Автономный клиент BitTorrent.....	294
Приложение. Содержание электронного архива.....	299
Предметный указатель	301

Предисловие

Не надо самому выпиливать кубики Лего. Следует знать, какие они бывают, где их купить, и уметь быстро собирать из них. Это совершенно иные навыки, чем разработка игрушек с нуля.

А. Себрант, директор «Яндекса» по маркетингу

Как вы думаете, почему обучающий проект Arduino приобрел феноменальную известность? С технической точки зрения, это всего лишь заурядный микроконтроллер Atmel, смонтированный на несложной плате с разъемами в сопровождении простой среды разработки программ Arduino IDE. Каждый производитель микросхем выпускает отладочные платы и средства разработки приложений, но обычно ими пользуется лишь узкий круг профессионалов.

Причина успеха Arduino в том, что из обучающего проекта он превратился в *платформу* для любительского творчества. Сложно провести четкую границу, за которой средство разработки и отладки становится платформой. Сформулируем главные требования к платформе для обучения и любительского творчества:

- ◆ простота для понимания новичками, старт с нуля;
- ◆ низкая цена стартового комплекта;
- ◆ расширяемость и дополняемость;
- ◆ достаточный запас производительности;
- ◆ вовлеченность сторонних производителей и энтузиастов;
- ◆ обширное сообщество единомышленников.

Простота освоения и гибкость аппаратно-программного комплекса Arduino позволяют начать работу с простыми проектами даже в младшем школьном возрасте. Затем любитель электроники может впасть в растерянность — куда двигаться дальше? Мигающие светодиоды, датчики влажности и выключатели лампочек давно освоены. Хочется чего-то нового и необычного. Рано или поздно любитель встает перед выбором из двух следующих альтернатив:

- ◆ углублять свои навыки в узкой области, разрабатывая специфические проекты и осваивая особые приемы программирования;

- ◆ осваивать новые платформы с более широкими возможностями и на другом оборудовании.

На самом деле, *можно и нужно сочетать оба подхода*. Далее в книге мы покажем, как это сделать.

Успех Arduino не дает покоя разработчикам новых творческих проектов. В этой книге мы изучим несколько программно-аппаратных платформ для любительского творчества. Они существенно различаются между собой — как технически, так и концептуально. Тем не менее, они отлично уживаются вместе, потому что связаны общей идеей: легкий старт с нуля и быстрый рост полезных навыков. Воспринимайте эти платформы, как ступени лестницы, ведущей вверх.

Отдельного упоминания заслуживают современные виртуальные средства для любительских разработок. Существуют даже онлайн-лаборатории, позволяющие макетировать устройства в окне браузера, подключать к ним измерительные приборы и проводить виртуальные эксперименты. Такие сервисы удачно дополняют аппаратную часть платформ и позволяют осваивать азы электроники без лишних затрат времени и денег. В этой книге мы также расскажем о средствах виртуального макетирования и разработки для любителей электроники.

С хорошей обучающей платформой пользователь выходит далеко за рамки стандартных проектов, предложенных разработчиками платформы. Например, на базе Arduino Mega был создан проект полетного контроллера для квадрокоптеров. К нему добавили акселерометры и гироскопы от игровой приставки Nintendo Wii, и получился проект универсального контроллера MultiWii для летающих моделей. На основе платы Arduino Nano был разработан модуль дисплейной телеметрии для авиамodelей MinimOSD. Оба проекта давно живут самостоятельной жизнью, но для работы с программами по-прежнему используется среда Arduino IDE. Поэтому любой пользователь, немного знакомый с Arduino, без труда может обновить и даже изменить прошивки этих устройств.

Соотношение цены и качества электронных компонентов имеет значение для любителей электроники, особенно если это школьники и студенты. Поэтому в описаниях прикладных проектов вы найдете рекомендации по выбору оптимального набора элементов. Имейте также в виду, что издательство «БХВ-Петербург» регулярно выпускает новые наборы электронных компонентов для технического творчества¹.

Как работать с этой книгой?

В книге использован пошаговый переход «от простого к сложному» как в целом, так и внутри тематических глав, посвященных различным платформам. Однако если полностью раскрывать все тонкости настройки и использования их компонентов, то каждая глава превратится в самостоятельную книгу. Поэтому глава начина-

¹ См. http://www.bhv.ru/books/list_covers.php?get=rubrics&id=233.

ется с минимального руководства по старту с нуля и сопровождается несколькими проектами разной сложности, иллюстрирующими возможности платформы. Попутно раскрываются некоторые особенности, редко упоминаемые в руководствах для начинающих.

Благодаря такой структуре, вы можете начинать чтение книги с любого интересующего вас раздела либо изучать их параллельно, работая над комплексным проектом. В любом случае, прогресс в мире электроники стремителен, и даже за время подготовки книги к печати могут произойти большие изменения. Рассматривайте эту книгу, как путеводитель по миру технического творчества, и будьте готовы к потрясающим воображение достижениям современной электроники. Я буду благодарен за отзывы и пожелания, которые обязательно учту при подготовке следующих изданий. Пишите мне по адресу: valeriy.yatsenkov@gmail.com.

* * *

Завершая предисловие, хочу отметить, что в русском языке нет полноценного аналога английскому слову *maker*. У этого слова много переводов: творец, создатель, творческая личность, поэт. В техническом контексте оно обозначает людей, которые в удовольствие себе и на радость окружающим разрабатывают и воплощают различные проекты. Иными словами, *maker* — это креативный любитель-самодельщик. Возможно, когда-то в русском языке приживется фонетическая калька «мейкер», но в этой книге я использую слово *любитель*, подразумевая любителя технического творчества, радиолюбителя или новичка в мире электроники.

ГЛАВА 1



Платформы для творчества и обучения

В далеком детстве я начинал изучение электроники с конструктором «Электронные кубики». Он состоял из оснащенных контактами пластмассовых кубиков, внутри которых находились транзисторы, резисторы, конденсаторы и прочие электронные компоненты (рис. 1.1). Размещая кубики по схеме, можно было собрать много разных устройств, вплоть до радиоприемника. Такой конструктор можно считать предком модульной обучающей платформы. Альтернативой «электронным кубикам» были скромные наборы радиодеталей, из которых мы собирали мелодичные



Рис. 1.1. Конструктор «Электронные кубики» (<http://www.yaplakal.com/forum2/st/550/topic983074.html>)

дверные звонки, датчики влажности и прочие несложные устройства. Для разработки и отладки собственных проектов следовало обладать изрядными знаниями и навыками.

Все изменилось после прихода на рынок дешевых и мощных микроконтроллеров. Раньше мы собирали готовые схемы из множества различных радиодеталей. Теперь акцент творчества сместился на программирование функций универсального процессорного блока, к которому подключаются те или иные внешние устройства. В большинстве случаев пайка не требуется — модули соединяются между собой при помощи стандартных разъемов или монтируются на отладочной плате. Не надо беспокоиться и о функциональной и электрической совместимости модулей, об этом позаботились разработчики и производители оборудования.

Модульный подход резко снизил входной порог знаний для новичков. Представители традиционной инженерной школы иногда считают учебные платформы профанацией и пренебрежительно относятся к «ардуинщикам» за их поверхностные знания. Вероятно, некоторые профессионалы испытывают досаду от того, что электроника и программирование теряют мистический ореол знаний для избранных.

Я не вижу проблемы в том, что работа с любительскими проектами дает поверхностные знания. Следует правильно определить область интересов. Для многих читателей электроника и программирование так и останутся увлекательным развлечением по вечерам. Получение *системных* фундаментальных знаний — это задача специального высшего образования, а не хобби. Разумеется, если речь идет о профессиональной деятельности, то применять в ней любительский подход действительно будет дурным тоном. Я имею в виду так называемое «грязное проектирование» без учета промышленных стандартов, помехоустойчивости и прочих требований к профессиональному продукту. Но если вы хотите интересно и с пользой провести свободное время или обучить школьников основам будущей профессии, то нет ничего лучше, чем обучающие платформы, в которых сочетаются электронное оборудование и программирование. Именно таким платформам посвящена эта книга.

1.1. Не бойтесь экспериментировать!

Выбирая платформу для обучения или хобби, прежде всего, опирайтесь на свои сегодняшние потребности и базовые навыки. Не бойтесь, что слишком простое оборудование быстро исчерпает себя и будет напрасно лежать на полке. Благодаря модульной структуре, даже базовое оборудование можно долго дополнять и изучать. Приобретенные ранее навыки и оборудование сполна пригодятся вам при работе с другими платформами.

Платформы и модули расширения совместимы между собой намного лучше, чем это может показаться неопытному пользователю. Поэтому купленные для Arduino датчики, индикаторы или исполнительные механизмы вы сможете легко применить для новой платформы, будь то Raspberry Pi или Omega2. В равной мере это относится к полученным ранее знаниям и навыкам.

По критерию процессорной части платформы можно условно разделить на два больших семейства:

1. Базовые устройства на основе микроконтроллеров малой и средней мощности. Программируются на уровне прошивки микроконтроллера. Взаимодействуют с простыми периферийными модулями (сенсоры, индикаторы) и решают относительно несложные задачи.
2. Базовые устройства на основе более мощных процессоров. Работают под управлением традиционных операционных систем (Linux, Android, Windows IoT Core) и представляют собой компактный компьютер. Взаимодействуют как с внешними модулями, так и с другими компьютерами. Позволяют разрабатывать и выполнять сложные программы на языках высокого уровня. Часто имеют видеоподсистему, могут подключаться к телевизору или компьютерному дисплею.

Характерным примером платформы первого типа является Arduino, работающая на базе микроконтроллеров компании Atmel. Для этой платформы выпускается широчайший ассортимент модулей расширения, допускающих сборку и отладку любительских конструкций без пайки. В исходном виде оборудование этой платформы мало подходит для встраиваемых применений наподобие центра управления «умным домом» — слишком ненадежна и неуклюжа конструкция, состоящая только из макетных плат. Отладочные платы Arduino можно успешно использовать для тестирования программного обеспечения и конструкции компонентов «умного дома» с последующим переносом прошивок в специально изготовленные устройства.

В качестве примера платформы второго типа можно привести Raspberry Pi. Это полноценный микрокомпьютер с операционной и файловой системами, но оснащенный аппаратными портами ввода/вывода и видеоподсистемой. К нему можно подключать исполнительные устройства и внешние модули.

Некоторые платформы занимают промежуточное положение. Например, это беспроводная платформа для «умного дома» NodeMCU. Она оснащена встроенным интерпретатором языка Lua и файловой системой. В то же время, для устройств этой платформы можно создавать программы в среде Arduino IDE, а многие модули предназначены для макетного монтажа.

Платформы также можно условно различать по степени модульности. Модульный подход в большей или меньшей степени присущ практически всем любительским платформам. Благодаря модульной конструкции обеспечивается расширяемость и низкая стоимость начального набора компонентов. Нужные модули можно приобретать по мере необходимости. Эти модули принято также называть *шилдами* (от англ. *shield* — навесная монтажная плата).

Лидером по количеству расширяющих модулей является платформа Arduino. Для нее выпускаются все мыслимые варианты внешних устройств — от одиночной лампочки или кнопки до цветного графического дисплея и приемника GPS. Можно сделать вывод, что чем проще «ядро» платформы, тем больше функциональности возлагается на подключаемую периферию. Это спорное обобщение, но в целом оно верно отражает положение дел.

1.2. Совместимость на уровне периферии и протоколов

В большинстве модулей используются стандартные протоколы и каналы обмена данными (UART, SPI, I²C, 1-Wire, Wi-Fi, Bluetooth). Поэтому одни и те же периферийные устройства могут использоваться в проектах на основе разных платформ. Трудности могут возникнуть из-за отсутствия на некоторых платформах готовых библиотек устройств. В этом случае вам придется потрудиться немного больше, чтобы понять, как устроена и работает библиотека этого устройства на другой платформе, а затем перенести программу на новое оборудование. Это один из признаков вашего творческого роста.

Устройство на основе Arduino само по себе может стать внешним модулем для более мощной платформы. Например, на основе платы Arduino можно изготовить многофункциональный выносной модуль метеостанции, который будет измерять температуру, влажность, давление, освещенность и по Bluetooth или Wi-Fi передавать данные центральному узлу на основе Raspberry Pi. В этом случае найдется применение не только оборудованию, но и навыкам программирования в среде Arduino IDE.

1.3. Общие средства разработки и языки программирования

Среда разработки программ Arduino IDE и язык Wiring/Processing, на котором пишут программы для Arduino, столь же популярны и просты, как и оборудование, на котором они работают. Поэтому реализацию среды разработки можно найти в самых неожиданных местах. Например, поддержка языка Processing в стиле Arduino IDE есть в специальном расширении среды Visual Studio 2015 для Windows. Вы можете написать программу на привычном для вас простом языке Processing и выгрузить ее в микрокомпьютер Raspberry Pi 3 с операционной системой Windows 10 IoT Core. После этого вам останется один шаг до программирования на языке C++. Совсем недавно считалось, что трудно придумать более непохожие среды программирования и устройства. Но спрос рождает предложение.

Можно поступить проще и установить Arduino IDE для ОС Linux на микрокомпьютер Raspberry Pi, чтобы использовать его, как универсальное средство разработки в домашней лаборатории.

Общие языки программирования также облегчают переход между платформами. Кроме уже упомянутого языка Processing, являющегося вариацией C++, вы можете писать программы на скриптовом языке Lua, который применяется даже при разработке приложений для ОС Android. Можно сменить прошивку NodeMCU на MicroPython, чтобы освоить язык Python, а затем программировать на нем для Omega2 и других устройств. Подробнее об этом будет сказано в следующих главах.

1.4. Облачные сервисы обмена данными

Кроме прямого физического обмена данными между платформами вы можете использовать *облачные сервисы*. Что это такое? Облачный сервис представляет собой сервер в Интернете, на котором работают специальные программы, а также имеется пространство для хранения пользовательских данных. Обычный пользователь не знает, где физически находятся серверы и устройства хранения данных. С его точки зрения, обмен данными происходит с неким виртуальным «информационным облаком» Интернета.

Ваше устройство может соединиться через Интернет с облачным сервисом, чтобы передать серверу данные для обработки и хранения, либо получить команды и данные с вашего компьютера или планшета (рис. 1.2). Каждое устройство, подключенное к облаку, получает уникальное имя или номер и пароль. Образно говоря, облачный сервис играет две роли: посредника в обмене данными и нотариуса, который заверяет подлинность соединений.

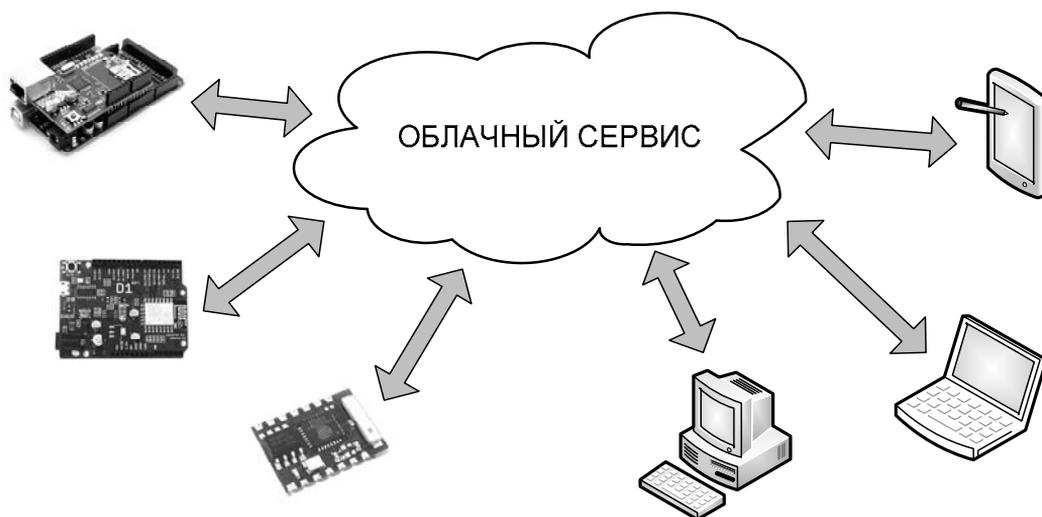


Рис. 1.2. Взаимодействие устройств с облачным сервисом

Устройства, взаимодействующие с облачным сервисом, могут быть разбросаны по всему земному шару. Необходимо иметь лишь доступ в Интернет в нужное время. Иногда, в особых случаях, данные передают при помощи SMS или в сообщениях электронной почты. Функциональность облачного сервиса не зависит от того, компонент какой из платформ к нему обращается. Устройства пользователей могут быть самыми разными, но должны передавать и получать данные в заранее оговоренном формате сервиса (API, Application Programming Interface, формат обмена между приложениями).

Сервер обрабатывает и компонует данные для представления в удобном виде. Например, зайдя на страницу облачного сервиса со стационарного компьютера или смартфона, вы можете посмотреть статистику потребления электроэнергии «умно-

го дома» или диаграмму скорости бега в колесе вашего любимого хомяка. При этом подсчет пробега хомяка будет выполнять устройство на основе платы Arduino и оптического датчика вращения, а рассматривать отчет сервиса можно на дисплее, подключенном к Raspberry Pi.

1.5. Онлайн-лаборатории и средства разработки

Облачные сервисы используются не только для управления Интернетом вещей (IoT, Internet of Things) и сбора данных с датчиков. Набирают популярность онлайн-лаборатории для разработки и макетирования любительских проектов. Такие сервисы сочетают в себе возможности облачной технологии и компьютерных симуляторов электронных схем. Если раньше для разработки схемы и печатной платы устройства нужно было устанавливать приложение на стационарный компьютер, то теперь можно вести разработку и виртуальное макетирование, а также запускать симуляцию схемы и даже прошивки контроллера в окне браузера. В некоторых онлайн-лабораториях есть виртуальные измерительные приборы, которые можно «подключать» к схеме.

Разумеется, в первую очередь создаются облачные среды разработки для самых популярных платформ, таких как Arduino и Raspberry Pi.

Проекты пользователей сохраняются в облаке и доступны из любого места, где есть Интернет. В большинстве случаев обеспечивается независимость от операционной системы компьютера. Можно организовать совместную работу в команде, члены которой разбросаны по всему миру, или опубликовать проект на своей странице в соцсети.

* * *

В этой главе мы схематически обрисовали некоторые варианты взаимодействия и совместного использования устройств, чтобы показать: чем больше платформ вы освоили и чем сильнее они различаются, тем богаче возможности для творчества. В следующих главах вы изучите приемы работы с популярными платформами и сервисами.

ГЛАВА 2



Советы для начинающих

Вы можете начать изучение любительских проектов с готового набора деталей, но рано или поздно столкнетесь с необходимостью приобретать дополнительные модули, макетировать собственные разработки и сопрягать их между собой. В этой главе мы рассмотрим общие вопросы макетирования, электропитания и сопряжения различных элементов любительских конструкций.

2.1. Универсальные отладочные и макетные платы

Наиболее популярны два типа плат для отладки и макетирования электронных устройств: *беспаячные (ламельные)* и *печатные*. Примеры макетных плат ламельного типа приведены на рис. 2.1, а плат печатного типа — на рис. 2.4.

Внутри у беспаячных макетных плат находятся упругие пластинчатые контакты — *ламели*, изготовленные из фосфористой бронзы, покрытой никелем. Сверху ламели

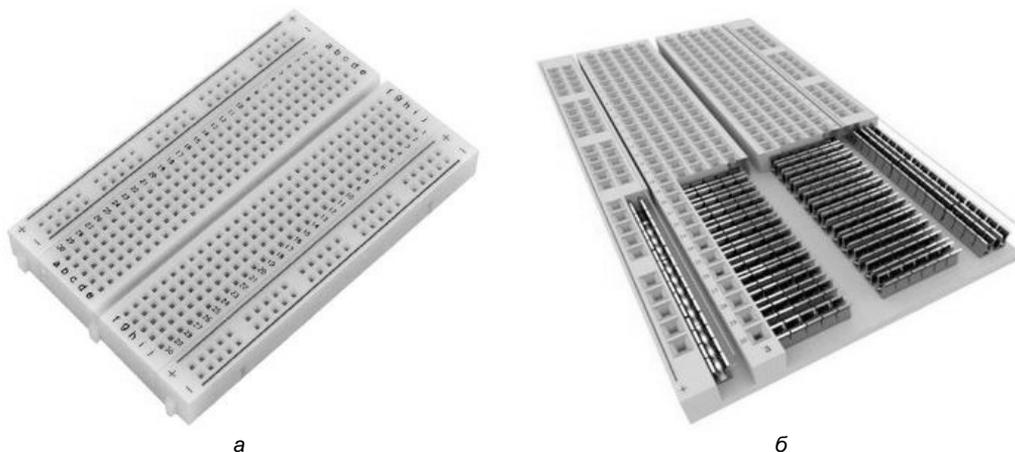


Рис. 2.1. Макетная плата ламельного типа (а) и ее внутреннее устройство (б)

накрыты трафаретом с отверстиями. Отверстия расположены на стандартном расстоянии 2,54 мм по горизонтали и вертикали. В них можно вставлять как гребенчатые разъемы модулей, так и выводы микросхем в корпусах типа DIP. Как видно на разрезе платы, по две шины питания проходят вдоль длинных сторон платы, а на поле для монтажа деталей короткие шины расположены поперек и соединяют группы по пять отверстий.

В англоязычном мире такие платы называют *breadboard* (доска для резки хлеба). У этого термина занятная история. Дело в том, что для моделирования первых радиолюбительских конструкций действительно применяли деревянные разделочные доски, на которые шурупами прикручивали проводники и радиодетали (рис. 2.2).

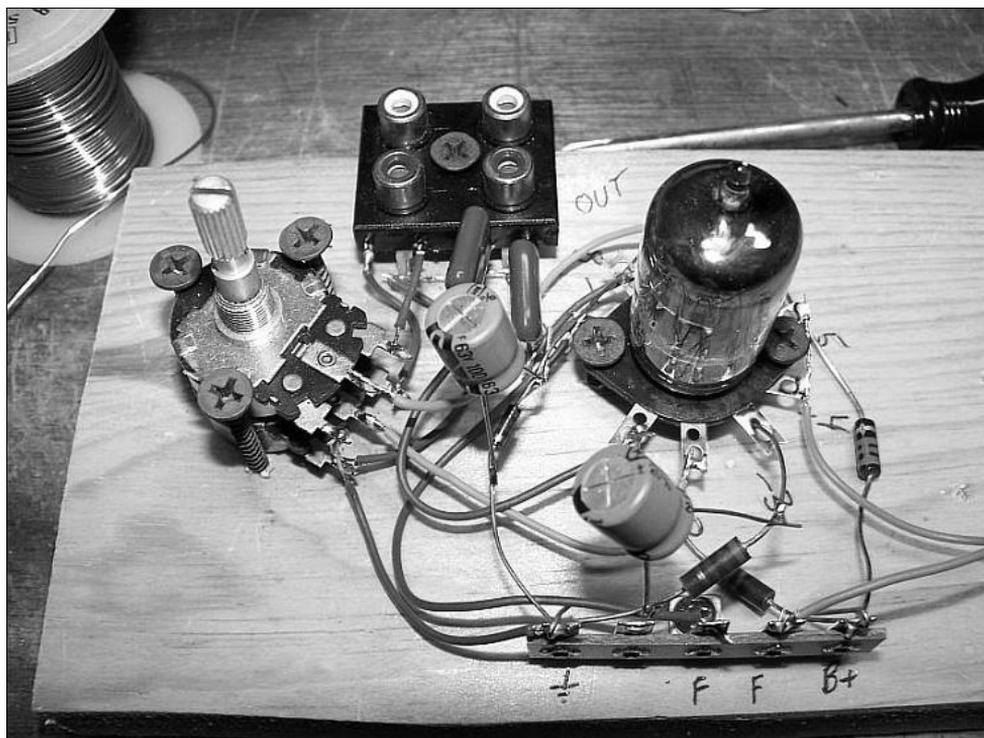


Рис. 2.2. Так выглядели макетные платы на кухонной разделочной доске (<http://electronics.stackexchange.com/questions/48516/why-are-they-called-breadboards>)

Достоинства беспаячной платы:

- ◆ простота и наглядность при макетировании несложных схем;
- ◆ универсальность;
- ◆ возможность многократного макетирования без потери свойств или разрушения;
- ◆ отсутствие необходимости в пайке;
- ◆ возможность быстро изменить схему соединений, добавить или удалить компоненты.

Недостатки беспаячной платы:

- ◆ большое количество ненадежных контактов в схеме, «шумы» в контактах;
- ◆ сильные взаимные помехи в соединительных проводах;
- ◆ сложность использования компонентов для поверхностного монтажа и малогабаритных компонентов;
- ◆ путаница и большая вероятность ошибки в соединениях при макетировании сложных схем.

Беспаячные платы идеально подходят для быстрого и наглядного макетирования несложных устройств, но неприменимы для окончательной сборки законченных изделий.

Для соединений между модулями и компонентами на ламельной плате вам потребуется набор специальных соединительных проводов. Это гибкие разноцветные провода с разъемной частью на обоих концах (рис. 2.3). Наверняка вы уже видели такие или подобные соединители.

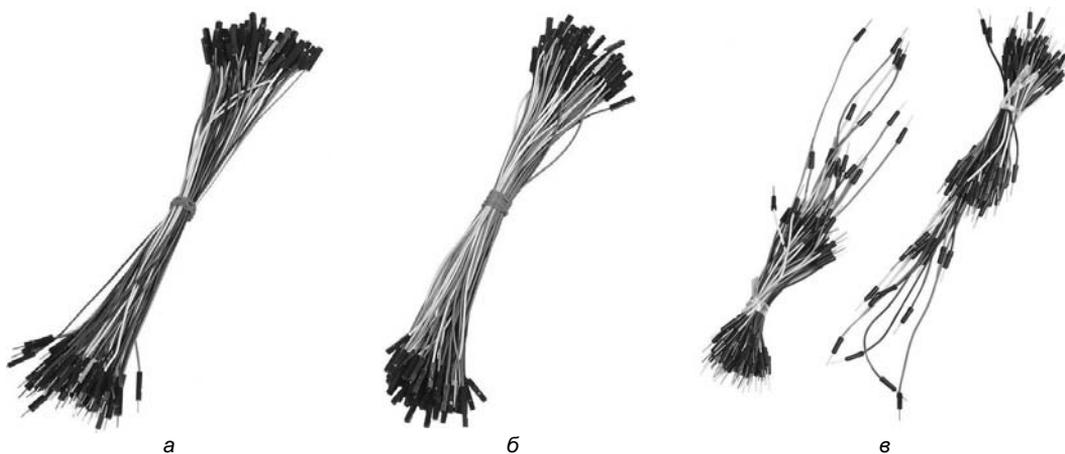


Рис. 2.3. Различные варианты соединительных проводов для макетирования

Провода с разъемами типа штекер-гнездо (рис. 2.3, *а*) применяются для подключения выносных модулей или компонентов, расположенных за пределами макетной платы. Штекеры вставляются в гнезда макетной платы, а гнезда надеваются на разъем модуля или выводы радиоэлемента. Провода с разъемами типа гнездо-гнездо (рис. 2.3, *б*) могут использоваться для соединения модулей между собой напрямую, вообще без макетной платы. Наиболее часто применяют провода типа штекер-штекер (рис. 2.3, *в*) — с их помощью выполняют большинство соединений на макетной плате. Таких проводов разной длины и цвета в наборе должно быть больше всего. В крайнем случае можно использовать самодельные провода с зачищенными на 4–5 мм концами. Хорошо подходят отрезки провода КСВВ 4х0.4, который используется при монтаже пожарно-охранной сигнализации. Но все-таки лучше сразу приобрести большой набор готовых соединительных проводов. Они прослужат вам много лет.

Если вы увлеклись техническим хобби, рано или поздно вам придется научиться паять. Но осваивать основы электроники лучше всего с беспаячными макетными платами. Это экономит много времени и сил. Даже при наличии большого опыта и хорошей домашней мастерской беспаячные платы остаются удобным средством макетирования. Я радиоинженер, занимаюсь радиолюбительством более тридцати лет и могу изготавливать дома печатные платы высокого качества. Но все равно использую беспаячные платы, когда нужно быстро макетировать устройство, состоящее из готовых модулей и небольшого количества навесных компонентов. Это особенно удобно в ситуациях, когда основное внимание следует уделить разработке и тестированию прошивки. Иногда оказывается, что идея не оправдала себя, или устройство лучше выполнить на другой элементной базе. В таком случае пайка макетной платы и, тем более, изготовление пробной печатной платы были бы напрасной тратой времени.

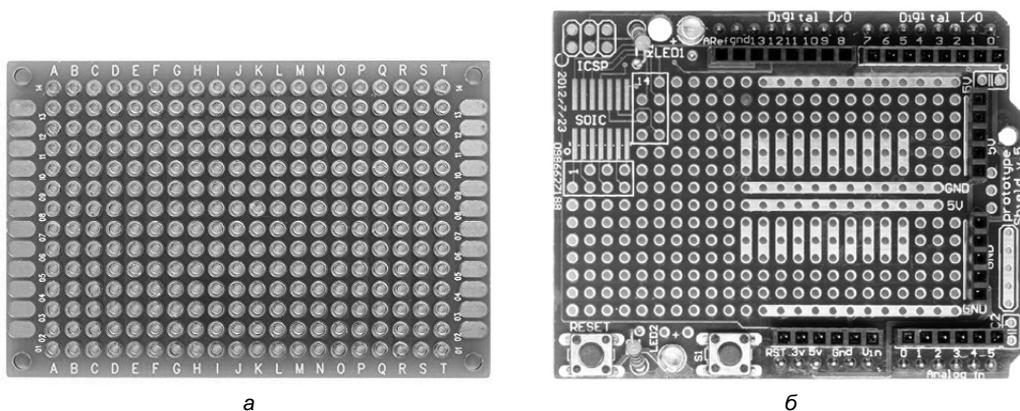


Рис. 2.4. Печатные макетные платы

Макетные печатные платы изготавливают по той же технологии, что и обычные печатные платы, с готовыми отверстиями под выводы элементов и контактными площадками (рис. 2.4, а). Шаг отверстий, как правило, стандартный, 2,54 мм (0,1 дюйма). Такое же межвыводное расстояние применяется в разъемах плат и модулей Arduino и многих других модулей. Некоторые макетные платы оснащены монтажными площадками под микросхемы и дискретные элементы для поверхностного монтажа. Для желающих разрабатывать собственные расширения в стиле Arduino выпускают комбинированные макетные платы (рис. 2.4, б). Кроме пространства для макетирования они имеют установленные гребенчатые разъемы и по габаритам совместимы с разъемами популярных плат Arduino или Raspberry Pi. Для покупки в зарубежных интернет-магазинах ищите печатные макетные платы по ключевым словам *Prototype PCB* (*Printed Circuit Board* — печатная плата). Беспаячные платы продаются под названием *Solderless Breadboard*.

Печатные макетные платы тоже имеют достоинства и недостатки. Достоинства печатных плат:

- ◆ можно выпилить из большой заготовки макетную плату нужного размера и формы;

- ◆ печатные макетные платы выпускают в широчайшем ассортименте, часто с готовыми установочными местами под микросхемы и прочие компоненты;
- ◆ паяные соединения на них компактны, аккуратны и надежны;
- ◆ в некоторых случаях можно обойтись без изготовления постоянной платы и поместить в корпус любительского устройства схему на макетной плате.

Недостатки печатных плат имеют значение, прежде всего, для новичка:

- ◆ для монтажа деталей требуются навыки пайки и паяльное оборудование;
- ◆ скорость монтажа пайкой у новичков обычно низкая;
- ◆ припаянные детали сложно использовать повторно, а испорченные детали трудно заменять.

Для соединений на печатной макетной плате используют разные способы. Некоторые компоненты удастся соединить между собой непосредственно при помощи их выводов, пропуская их через отверстия платы и отгибая в стороны. Если соединения не пересекаются, применяются одножильные луженые проводники. В остальных случаях обычно используют гибкие изолированные провода.

2.2. Монтажные провода для пайки

У мягких монтажных проводов в виниловой изоляции, которые обычно применяют радиолюбители, есть серьезный недостаток. При попытке залудить и припаять зачищенный кончик провода изоляция сокращается по длине и плавится от нагрева, оголяя излишний участок провода. Это досадное явление очень мешает работе и грозит замыканиями. Проблему можно решить двумя способами: опытным путем подобрать провода, изоляция которых плохо плавится и не стягивается при нагреве, либо использовать провода в специальной термостойкой изоляции из фторопласта (МГТФ) или силикона. Провода в изоляции из фторопласта редко встречаются в продаже и дорого стоят. Они также редко бывают окрашены в разные цвета.

Тонкие провода в силиконовой изоляции идеально подходят для макетирования и монтажа любительских конструкций. Они легко зачищаются и лудятся, их силиконовая изоляция не плавится даже при сильном нагреве. Несколько лет назад тонкие силиконовые провода стоили весьма дорого и редко встречались в продаже. Сейчас такие провода можно найти в китайских и российских интернет-магазинах. Причем, у российских продавцов цены зачастую ниже (но ассортимент меньше). Чтобы не ошибиться в выборе проводов, следует разбираться в их маркировке.



Диаметр провода обозначается аббревиатурой AWG (American Wire Gauge). Иногда числовое значение AWG называют *калибром провода*. Исторически значение AWG происходит от технологии производства проволоки волочением. Заготовку, которая обозначается как 0AWG, последовательно протягивают через ряд уменьшающихся калиброванных отверстий (волоков, фильер). Соответственно, числовое значение AWG обозначает количество протягиваний. Поэтому, чем больше значение AWG, тем тоньше провод. Например, обозначение 24AWG (или AWG24) соответствует диаметру проводника, который протянули через стандартные дюймовые фильеры 24 раза.

Многожильные монтажные провода состоят из большого количества скрученных тонких проводников (рабочих жил), поэтому калибр AWG относится к общему диаметру проводящей жилы. Отдельно в характеристиках провода указывают количество рабочих жил и их диаметр. Чем тоньше рабочие жилы, тем мягче и гибче монтажный провод в целом.

Усредненные характеристики многожильных проводов по шкале AWG приведены в табл. 2.1. Данные из этой таблицы можно использовать при выборе мягких монтажных проводов, как в силиконовой, так и в виниловой изоляции. В зависимости от производителя, диаметр рабочих жил, их количество и толщина изоляции могут незначительно различаться.

В радиолюбительской практике для монтажа и макетирования удобнее всего использовать провода 30AWG, 28AWG и 26AWG. Более толстые провода используют для питания силовых цепей. Например, в электрических авиамоделях для подключения силовых литий-полимерных батарей обычно применяют провода калибром от 18 до 12AWG.

Таблица 2.1. Размерность электрических проводов по шкале AWG
(одиночная рабочая жила 0,08 мм)

Калибр по AWG	Кол-во жил	Диаметр проводника, мм	Внешний диаметр, мм	Толщина изоляции, мм	Предельный рабочий ток, А
30AWG	11	0,30	0,8	0,05	0,8
28AWG	16	0,32	1,2	0,08	1,2
26AWG	30	0,44	1,5	0,15	3,5
24AWG	40	0,58	1,6	0,20	5,0
22AWG	60	0,72	1,7	0,30	8,7
20AWG	100	0,92	1,8	0,50	13
18AWG	150	1,20	2,3	0,55	22
16AWG	252	1,53	3,0	0,80	35
14AWG	400	1,75	3,5	0,90	55
12AWG	680	2,50	4,5	1,00	88
10AWG	1050	3,60	5,8	1,20	140
8AWG	1650	4,40	6,3	1,20	190
6AWG	3200	5,20	8,5	1,65	230

Если нужно быстро найти подручные монтажные провода для макетирования, а покупать их некогда, попробуйте очистить от внешней изоляции тонкий ненужный USB-кабель, которыми обычно комплектуют различные китайские гаджеты. Как правило, таких кабелей со временем накапливается дома довольно много, и они лежат без дела. Хорошо подходят жилы кабеля от неисправной компьютерной

мыши — они тонкие и мягкие. К сожалению, такие провода часто бывают рассчитаны на монтаж обжимкой в разъемах и плохо переносят пайку.

2.3. Инструменты для подготовки проводов

При макетировании пайкой на печатной плате кончики соединительных проводов приходится зачищать десятки и сотни раз. Именно эта операция занимает больше всего времени. Для быстрой и аккуратной зачистки монтажных проводов применяют инструменты различной конструкции под общим названием *stripper* (stripper). Пример такого устройства показан на рис. 2.5.



Рис. 2.5. Стриппер для зачистки проводов в виниловой и силиконовой изоляции

К сожалению, профессиональные стрипперы слишком дорогие, громоздкие, занимают много места в рабочем столе и редко подходят для зачистки тонких монтажных проводов калибра 28...30AWG. Я давно и успешно применяю самодельное приспособление (рис. 2.6). Это очень простое и компактное устройство можно сделать за один вечер из подручных материалов. Для его изготовления вам потребуются:

- ◆ старый металлический пинцет;
- ◆ обломки закаленного полотна от ножовки по металлу;
- ◆ алмазный надфиль или ручная бормашинка с алмазным диском;
- ◆ винт М2...М3 для регулировки;
- ◆ паяльник мощностью 60...100 ватт, припой, кислотный флюс.

Изготовьте рабочие режущие пластины из ножовочного полотна. Постарайтесь сразу как можно аккуратнее заточить режущие кромки под углом около 30 градусов. Лучше всего делать это при помощи алмазного надфиля.

Укоротите пинцет примерно наполовину. С одной стороны просверлите отверстие под регулировочный винт. Если нет возможности нарезать в отверстии резьбу, припаяйте гайку нужного размера. Припаяйте режущие пластины к остаткам пинцета. Обратите внимание на правильную ориентацию пластин — они должны работать по принципу ножниц, то есть при сближении соприкасаться плоскими сторонами выреза (рис. 2.6, а). Чтобы регулировочный винт не проворачивался самопроизвольно, используйте пружину или стопорную гайку (рис. 2.6, б).

На этом изготовление самодельного стриппера окончено. Остается отрегулировать смыкание пластин таким образом, чтобы они надрезали изоляцию, но не повредили центральную жилу провода.

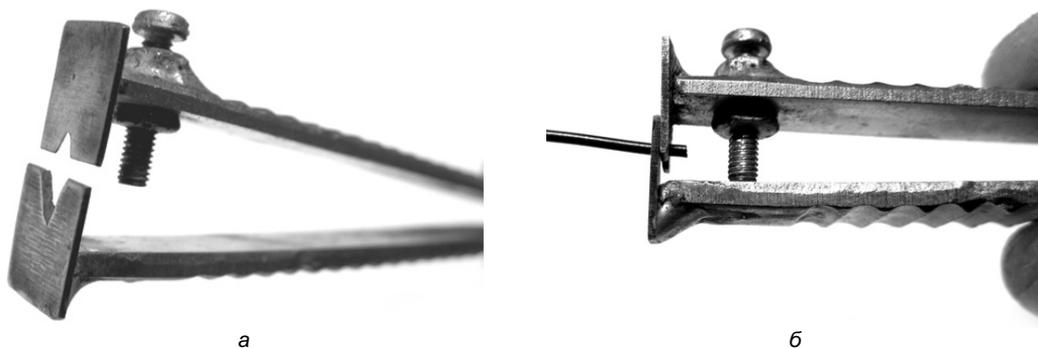


Рис. 2.6. Самодельный стриппер для зачистки проводов

2.4. Источники питания

К организации электропитания своих устройств нельзя относиться легкомысленно. Многие проблемы в работе электронного оборудования возникают именно по причине некачественного источника питания:

- ◆ недостаточной его мощности;
- ◆ заниженном или завышенном выходном напряжении;
- ◆ пульсации питающего напряжения;
- ◆ помех в цепях питания.

Попытка сэкономить, используя кустарные или некачественные источники питания, обычно заканчивается испорченным оборудованием, которое стоит значительно дороже источника, а на ремонт уходит больше времени, чем на подготовку правильного электропитания.



В любительской электронике обычно применяется питающее напряжение 5 В, но ему на смену в новых устройствах приходит напряжение 3,3 В. Это стандартное напряжение питания большинства современных микросхем. К сожалению, сейчас для питания любительских проектов часто требуются оба напряжения, потому что в одной конструкции объединяются старые и новые компоненты, но не все новые моду-

ли снабжены встроенным источником +3,3 В. Попытка запитать такое устройство напряжением 5 В, равно как и переплюсовка, приводит к мгновенному выходу низковольтных микросхем из строя. Будьте предельно внимательны при подключении линий питания!

2.4.1. Особенности питания от порта USB

Вариант питания от компьютерного порта USB наиболее часто встречается в любительской практике, особенно во время разработки и отладки устройства. Он удобен, когда вы подключаете устройство через встроенный или внешний конвертер USB-UART для программирования и настройки. По одному кабелю происходит обмен данными с компьютером и поступает питание на устройство.

Но порты USB имеют ограничения по предельному току в нагрузке. За потребляемым током следит хост-контроллер. Это специальная микросхема на материнской плате компьютера, которая обеспечивает работу портов. Стандартом USB предусмотрены предельные токи, потребляемые от порта: USB 2.0 — до 500 мА, USB 3.0 — до 950 мА. Для большинства любительских устройств этого вполне достаточно. Если же в устройстве применяется мощный процессор, светодиоды, электромоторы, дисплей, то предельный ток порта может быть превышен.

Порты USB снабжены защитой от замыканий в цепи питания. Обычно это *самовосстанавливающийся предохранитель*, установленный на материнской плате. Он восстанавливает свои проводящие свойства через некоторое время после срабатывания. Если после замыкания в нагрузке и повторного включения компьютера на разьеме порта не появилось питающее напряжение, не пугайтесь. Отключите компьютер и дайте предохранителю 20–30 минут на восстановление.

При экспериментах с самодельными устройствами, имеющими комбинированное питание USB/батареи, велика вероятность случайно подать на порт USB так называемое *встречное* напряжение питания. Это может повредить материнскую плату, особенно в ноутбуках! В качественных модулях заводского производства устанавливают защитно-развязывающие диоды, подключенные по схеме, приведенной на рис. 2.7.

Батарею подключают к клемме +VBAT, а плюсовую линию питания от разъема USB — к клемме +5V USB. В зависимости от того, на какой из клемм присутствует

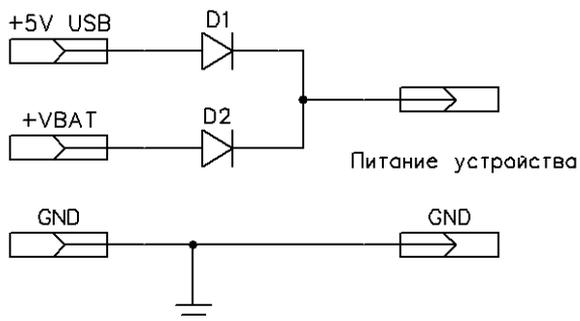


Рис. 2.7. Защитно-развязывающие диоды в цепи питания устройства

напряжение питания, ток протекает либо через диод D1, либо через D2. Если подключить одновременно два источника, устройство будет питаться от того из них, чье напряжение выше.

Относительно полярности батареи диод D1 включен встречно, поэтому не может проводить ток от батареи в порт USB. Аналогично, напряжение порта USB не может повредить батарею благодаря диоду D2. Кроме того, эти диоды защищают устройство от переплюсовки питания — самой частой причины необратимой поломки любительских устройств. Если конструкция исключает подсоединение двух источников одновременно, то устанавливается один диод. Он защищает устройство от переплюсовки, а порт USB — от случайной подачи встречного напряжения.



Несмотря на то, что в защитно-развязывающих цепях применяют специальные диоды с пониженным падением прямого напряжения (Low-Drop diode), напряжение порядка 0,2–0,3 В все же падает на полупроводниковом переходе диода. Кроме того, на USB-портах некоторых компьютеров присутствует слегка сниженное напряжение питания, около 4,7...4,8 В. В таком случае оставшееся напряжение порядка 4,5 В может оказаться слишком низким для питания устройства. Например, производители авиамодельной бортовой электроники для питания полетных контроллеров часто рекомендуют использовать источники напряжением 5,25...5,35 В, чтобы компенсировать падение напряжения на диодах. Заниженное напряжение питания приводит к нестабильной работе устройства и выражается в самопроизвольной перезагрузке или «зависании» микроконтроллеров и хаотичных показаниях датчиков. На поиск причины сбоев может уйти много времени. Если устройство без видимых причин ведет себя непредсказуемо, начните с проверки качества питания.

2.4.2. Сетевые источники питания

Стандартный сетевой адаптер (Wall Adaptor) от любого гаджета — это второй по популярности источник питания любительских устройств. Он полностью готов к использованию и имеет защиту от перегрузки. Следует лишь удостовериться, что он обеспечивает на выходе именно 5 В и достаточный ток. Желательно использовать адаптер с гнездом USB, так как это позволит применять стандартные провода для питания многих популярных отладочных модулей.



Не используйте дешевые подделки под сетевые адаптеры Apple — внутри у них находятся примитивные схемы без какой-либо защиты. Обратите внимание, что сетевые адаптеры большинства планшетов выдают напряжение от 7 до 9 В. В этом случае вам понадобится дополнительный регулятор напряжения.

В случаях, когда требуется несколько питающих напряжений, либо выходное напряжение источника не соответствует нужному значению, применяются различные *преобразователи напряжения*. Они могут понижать или повышать исходное напряжение. Далее мы рассмотрим различные варианты преобразователей.

Линейные стабилизаторы напряжения

Линейные стабилизаторы напряжения, по сути, представляют собой регулируемое сопротивление с источником образцового напряжения и цепью обратной связи. Если входное напряжение растет, то ровно в такой же степени (линейно) возрастает

внутреннее сопротивление регулятора. Благодаря этому выходное напряжение не меняется. Аналогичным образом регулятор поддерживает напряжение на выходе при колебаниях мощности нагрузки. Регулятор реагирует на изменение напряжения по линейному закону, отсюда происходит название стабилизатора.

Современные линейные стабилизаторы представляют собой микросхему, на вход которой подают исходное напряжение, а на выходе получают стабильное заданное напряжение, которое определяется номиналом микросхемы (рис. 2.8). Обычно в любительской практике востребованы напряжения 5 и 3,3 В. Если напряжение 5 В можно получить от порта USB или сетевого источника в готовом виде, то напряжение 3,3 В в любом случае придется формировать отдельно.

Достоинства линейных стабилизаторов:

- ◆ очень низкая цена и доступность в продаже;
- ◆ маленькие габариты;
- ◆ готовые стандартные напряжения;
- ◆ простота монтажа и ремонта.

Недостатки линейных стабилизаторов:

- ◆ низкий КПД — стабилизатор включается последовательно с нагрузкой, и излишек напряжения падает на стабилизаторе. При этом стабилизатор нагревается, а лишняя энергия напрасно теряется в виде тепла. Чем больше падение напряжения (разность напряжения между входом и выходом) и ток в цепи, тем сильнее нагревается стабилизатор. Терять в виде тепла драгоценную энергию батарей — расточительно;
- ◆ ограниченный диапазон входных напряжений. Например, для обычного стабилизатора на 5 В допустимое входное напряжение редко превышает 15 В;
- ◆ линейные стабилизаторы работают только на понижение входного напряжения.

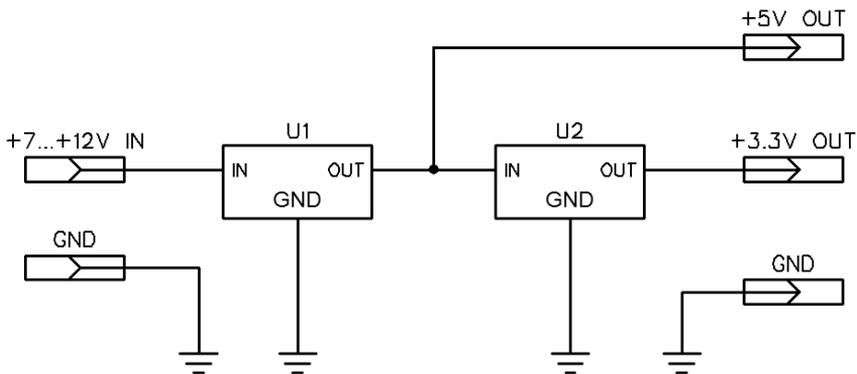


Рис. 2.8. Схема последовательного включения линейных стабилизаторов питания

Перечисленные недостатки линейных стабилизаторов во многих случаях с лихвой перекрываются их дешевизной, распространенностью и схемной простотой. Особенно хорошо они подходят для несложных устройств с низким потреблением тока.

К этой категории относятся и любительские схемы. Рекомендую приобрести линейный стабилизатор, оформленный в виде модуля, совместимого по установочным размерам со стандартной беспаячной макетной платой. Один из вариантов такого модуля изображен на рис. 2.9. Он формирует стандартные напряжения +5 и +3,3 В. При помощи перемычек можно отдельно задать напряжения на обеих шинах вдоль макетной платы. Разумеется, для такого стабилизатора нужен внешний источник с выходным напряжением +7...+15 В.

Основные элементы модуля (рис. 2.9):

- ◆ 1, 2 — переключатели напряжения +5/+3,3 на шине питания;
- ◆ 3 — линейный стабилизатор +5 В;
- ◆ 4 — линейный стабилизатор +3,3 В;
- ◆ 5 — диод защиты от переполюсовки входного напряжения;
- ◆ 6 — разъем для подключения внешнего источника питания +7...+15 В;
- ◆ 7 — разъем USB с выходным напряжением +5 В для подключения внешних потребителей.

Микросхемы стабилизатора серии AMS1117 выдерживают ток нагрузки до 1 А (при условии, что внешний источник способен обеспечить такой ток). Но не забывайте, что чем выше входное напряжение, тем сильнее нагрев стабилизатора.

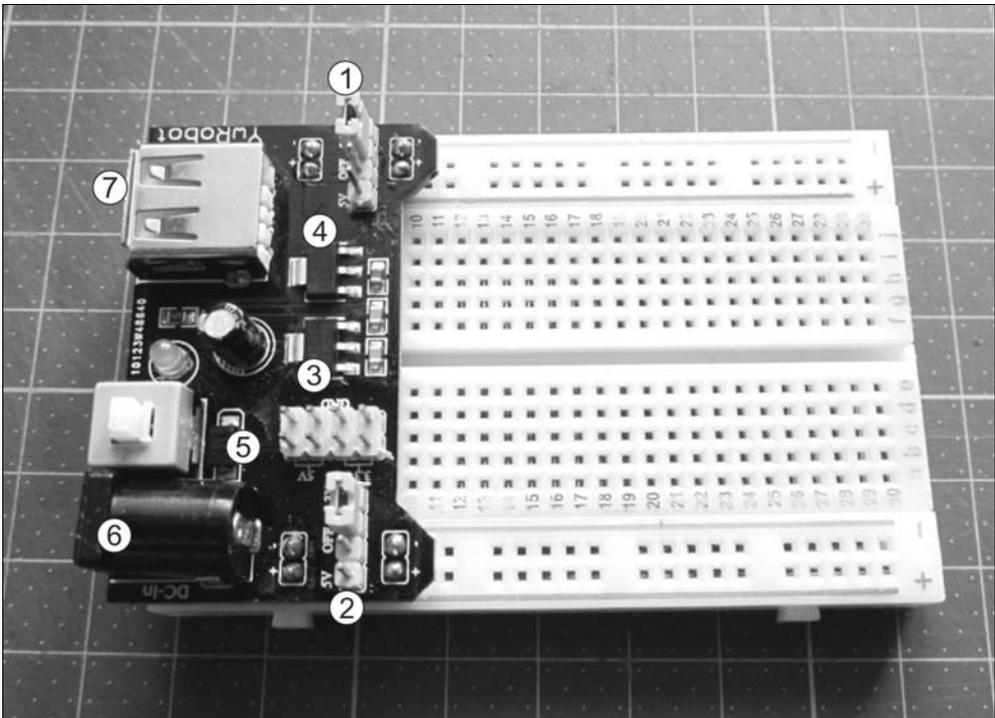


Рис. 2.9. Источник питания для беспаячной макетной платы