

Оглавление

Предисловие от издательства	11
Об авторах.....	12
О рецензентах.....	13
Предисловие	14
Часть I. Введение в Pico	19
Глава 1. Начало работы с Raspberry Pi Pico	20
Технические требования.....	20
Представление Raspberry Pi Pico и RP2040	21
Микроконтроллер RP2040	22
Обзор вариантов платы Pico.....	23
Пайка соединительного разъема Pico	27
Пайка разъема	27
Реализация примера «Hello World!»	29
Кнопка сброса для Pico (дополнительно)	29
MicroPython	29
Прошивка двоичного файла MicroPython	30
Написание первой программы	31
Реализация примера мигания светодиода	33
Описание примера кода	35
Пример CircuitPython.....	36
Второй пример мигания светодиода.....	39
CircuitPython или MicroPython?	40
Подключение полезного дополнительного оборудования для Pico	41
Pico Breadboard Kit (набор Pico с макетной платой)	42
Pico GPIO Expansion Board (плата расширения выводов GPIO Pico)	42
Pico HAT Expansion (расширение Pico HAT)	42
Grove Shield for Pi Pico (плата расширения Grove для Pi Pico)	43
Pimoroni Pico Decker (четырёхкратный расширитель).....	44
Итоги	45
Глава 2. Последовательные интерфейсы и их приложения.....	46
Технические требования.....	46
Установка необходимых библиотек	47
Датчик температуры HTU21D-F.....	47
Датчик температуры и влажности DHT20.....	48
OLED-дисплей (контроллер SSD1306).....	48
Беспроводной модуль	49
Использование интерфейса UART для связи между двумя платами Pico	49
Интерфейс UART	50
Настройка Pico.....	51
Программирование UART на платах Pico	52
Тестирование кода	54
Приложения интерфейса UART	57

Подключение датчиков через интерфейс I2C	58
Введение в интерфейс I2C	58
Подтягивающие резисторы	60
Тестирование датчика температуры HTU21D-F	61
Код датчика температуры HTU21D-F.....	61
Тестирование датчика температуры DHT20	62
Код для датчика температуры/влажности АНТ20.....	64
Плата Feather RP2040	65
Устранение неполадок	66
Отображение данных о температуре на дисплее с SPI-интерфейсом.....	68
Последовательный периферийный интерфейс (SPI)	68
Подключение дисплея	69
Отображение данных о температуре и влажности.....	69
Плата LILYGO RP2040	71
Настройка беспроводного модуля ESP32	72
Итоги	74
Глава 3. Проекты домашней автоматизации	75
Технические требования.....	75
Установка необходимых библиотек	76
NeoPixel LED	76
Беспроводной модуль	76
Подключение контактных датчиков	77
Управление приборами.....	82
Размещение состояний датчиков в облаке.....	84
Настройка Adafruit IO	84
Порядок размещения.....	86
Следующие шаги	88
Управление светодиодными лентами.....	88
Введение в Arduino Nano RP2040 Connect	90
Установка CircuitPython на RP2040 Connect	91
Подключение RP2040 к интернету.....	93
Итоги	94
Глава 4. Весело проводите время в саду!.....	95
Технические требования.....	96
Почему садоводство?.....	96
Установка необходимых библиотек	97
Датчик содержания влаги в почве	98
Беспроводной модуль	99
NeoPixel LED	99
Настройка датчика влажности почвы.....	99
Настройка беспроводного модуля.....	103
Настройка светодиода NeoPixel.....	104
Размещение данных в ThingSpeak	107
Собираем все вместе	111
Итоги	112

Часть II. Обучение через соиздание	113
Глава 5. Строим метеостанцию	114
Технические требования.....	115
Проведение гражданских научных экспериментов.....	116
Установка необходимых библиотек	116
Датчик AM2315	116
Датчик BME280	117
Датчик ультрафиолетового излучения VEML6075.....	117
Тестирование датчиков.....	117
Тестирование датчика BME280	117
Тестирование датчика температуры/влажности AM2315	120
Тестирование датчика VEML6075	122
Тестирование датчиков измерителя погоды.....	123
Тестирование датчика осадков	124
Проверка анемометра и флюгера	126
Проверка датчика направления.....	128
Тестирование беспроводного модуля.....	130
Сборка и тестирование метеостанции.....	130
Следующие шаги.....	131
Итоги	132
Глава 6. Проектируем настенный семисегментный дисплей	133
Технические требования.....	133
О мотивации проекта.....	134
Возможные варианты использования.....	136
Установка необходимых библиотек	136
Беспроводной модуль	136
Выбор семисегментных индикаторов	137
Подключение настенного семисегментного дисплея.....	139
Создание драйверов для семисегментного дисплея.....	141
Использование дисплея	143
Простой веб-сервер.....	143
Пример управления через последовательный порт.....	146
Отслеживание физической активности	147
Собираем все вместе	147
Итоги	149
Глава 7. Разрабатываем устройство слежения за качеством воздуха	150
Технические требования.....	151
Мотивация проекта	151
Установка необходимых библиотек	152
Шаговый двигатель	152
Беспроводной модуль	153
Датчик CO ₂ SCD30.....	153
Использование общедоступных источников для получения данных о качестве воздуха.....	153
Выполнение запроса на Pico	157

Подключение датчика CO ₂ к Pico.....	161
Подключение шагового двигателя	169
Устройство дисплея	173
Создание интерактивного дисплея.....	173
Итоги	174
Часть III. Темы повышенной сложности.....	175
Глава 8. Беспроводная связь.....	176
Технические требования	176
Установка необходимых библиотек	177
Adafruit Bluefruit LE SPI Friend	177
Дополнительно: модуль LoRa	177
Дополнительно: датчик CO ₂	179
Подключение модуля Bluetooth с низким энергопотреблением.....	179
Размещение показаний датчика через модуль Bluetooth.....	186
Подключение модуля Sigfox.....	192
Что такое Sigfox?.....	192
Модуль Sigfox.....	193
Настройка модуля Sigfox.....	194
Пример кода	196
Подключение модулей LoRa	198
Что такое LoRa?	198
Примерный сценарий.....	199
Итоги	203
Глава 9. Строим робота!.....	204
Технические требования	205
Установка необходимых компонентов	205
Установка батарей.....	207
Управление светодиодами.....	208
Выбор двигателя и способы управления различными типами.....	211
Двигатели постоянного тока	211
Шаговые двигатели	211
Серводвигатели	212
Управление двигателем постоянного тока	213
Управление серводвигателем.....	215
Тестирование датчиков.....	217
Ультразвуковой датчик.....	217
Датчики отслеживания линии.....	220
Тестирование робота.....	222
Соревнования по робототехнике	223
Итоги	223
Глава 10. Знакомство с приложениями TinyML	224
Технические требования.....	224
Дополнительное оборудование.....	224
Введение в TinyML.....	225

Представляем Arducam Pico4ML.....	229
Распознавание ключевых слов в звуковых образцах.....	230
О платформе Edge Impulse	230
Классификация изображений.....	243
Разработка фронтальных устройств	244
Итоги	244
Глава 11. Создаем готовый продукт	245
Технические требования.....	245
Истоки Pico-телефона	246
Определение требований.....	247
Выбор компонентов	247
Построение макета	249
Установка необходимых библиотек.....	250
Тестирование модуля Notecard	250
Тестирование клавиатуры	253
Проектирование печатной платы	255
Подготовка принципиальной схемы	257
Раскладка печатной платы	257
Выбор корпуса	258
Генерация файлов Gerber.....	259
Изготовление плат	259
Монтаж печатной платы.....	260
Подготовка платы к эксплуатации.....	261
Окончательная сборка и тестирование	261
Развитие проекта.....	262
Замена Pico	262
Pimoroni PGA2040.....	262
RP2040 Stamp	263
Итоги	264
Глава 12. Дополнительные возможности работы с Pico.....	265
Технические требования.....	265
Обновление прошивки Pico	265
Программирование Pico с помощью Arduino IDE	268
Загрузка и установка Arduino IDE	268
Установка пакета для платы Pico	269
Программирование на C/C++ с использованием Pico SDK	274
Отладчики для Raspberry Pi Pico	274
Инструменты для создания прототипов и разработки продукта	274
Макетная плата с указанием разводки выводов Pico	274
Получение профиля потребления вашего продукта	275
Nordic Power Profiler Kit	276
Joulescope	276
Программирование PIO	277
Итоги	278
Предметный указатель	280

Об авторах

Сай Яманур – старший инженер по приложениям интернета вещей (IoT) в компании по производству промышленных газов в Буффало, штат Нью-Йорк. Имеет более чем 10-летний опыт работы в качестве эксперта по встраиваемым системам, работая как над разработкой, так и над внедрением аппаратного и программного обеспечения. Соавтор двух книг об использовании Raspberry Pi для выполнения проектов «сделай сам». Представил персональную панель мониторинга здоровья на общенациональной выставке Maker Fair. В настоящее время Сай работает над проектами, направленными на улучшение качества жизни (QoL) людей с хроническими заболеваниями.

«Я хочу поблагодарить своих родителей и моего брата и соавтора Шри за всю помощь и поддержку; наших технических рецензентов Салмана Фариса и Джонатана Виттса за их пронцательные комментарии и за тщательный анализ нашей работы; Рахула Наира за предоставленную нам возможность работать с издательством Packt. Я также хотел бы поблагодарить Роми Диас и Вайдехи Савант за их терпение и поддержку нашей работы».

Шрихари Яманур – инженер-механик с опытом работы в области проектирования медицинских устройств, CAD/CAM, механотроники и надежной аппаратуры. В сотрудничестве со своим братом разрабатывает аппаратные продукты с открытым исходным кодом, направленные на повышение популярности в любительских кругах. Имеет множество сертификатов в области обеспечения качества, САПР и ВЭД. Помимо дизайна, производства и качества электронных изделий, его текущие интересы включают изменение поведения и активное самосовершенствование в борьбе с диабетом, инновационные парадигмы и методологии в области здоровья, а также влияние искусственного интеллекта на здравоохранение. Он является соавтором двух книг по приложениям Raspberry Pi и пишет блоги на различные темы.

«Я хочу поблагодарить своих родителей, наставников, друзей, кошек, а также моего брата и соавтора Саи за всю помощь и поддержку. Я хотел бы выразить благодарность своим наставникам Анну Тамуру и доктору Судхи Гаутаме, а также моему другу Сатьяканту Тьягарадже за то, что он поддержал меня в трудные времена, и команде издательства Packt за их поддержку этой книги и других наших усилий на протяжении многих лет».

О рецензентах

Салман Фарис – энтузиаст разработки и быстрого создания прототипов цифровых продуктов из Индии. Имеет степень бакалавра в области компьютерных наук и диплом цифровой разработки Академии Fab. В настоящее время работает инженером технической поддержки в английской компании Nebra и является ключевым участником сообщества разработчиков MakerGram, где занимается разработкой электронных и аппаратных продуктов.

Салман также является частью экспертной группы Edge Impulse, сообществ Qubitro, RAK и Seeed Studio, а также послом и основным участником крупнейшего в Индии собрания разработчиков Maker Faire (Хайдарабад) и соорганизатором фестиваля Maker Fest в Керале.

«Сначала я хотел бы поблагодарить Аллаха за Его всемогущее руководство в любых решениях, которые я принимаю. Я также хотел бы поблагодарить издательство Packt Publishing за возможность ознакомиться с этой замечательной книгой, особенно Шагуна и Эшвина, которые руководили рецензированием и помогли мне советом и поддержкой на протяжении всего процесса. Спасибо моим родителям, братьям и сестрам, родственникам, друзьям, наставникам и команде Nebra».

Джон Виттс работает в области информационных технологий в сфере образования уже более 17 лет. Имеет ученую степень по изобразительному искусству, а также дизайну и разработке электронного обучения. В своей нынешней роли директора по цифровой стратегии Джон руководит всеми технологическими решениями в своей школе, а также преподает информатику учащимся в возрасте 11–16 лет. Джон также проводит мероприятия Hull Raspberry Jam в своем родном городе: бесплатные семинары по программированию для молодых людей, использующих компьютер Raspberry Pi. Джон рецензировал ряд изданий для Packt и написал свою собственную книгу «Wearable-Tech Projects with the Raspberry Pi Zero», опубликованную Packt. В свободное время Джон с удовольствием занимается объединением компьютерных технологий с искусством (generative art), используя библиотеки JavaScript и элементы физических вычислений на Raspberry Pi.

«Я хотел бы поблагодарить мою жену Салли и наших трех дочерей, Мейбл, Эмбер и Аду, за всю их поддержку, позволившую мне работать над этой книгой, а также авторов и всю команду издательства Packt за то, что они позволили мне принять участие в процессе создания этой замечательной публикации».

Предисловие

Когда в январе 2021 года компанией Raspberry Pi Foundation был сделан анонс Raspberry Pi Pico, мы были поражены новыми возможностями, которые в плате за 4 доллара США открылись для любителей, специалистов широкого профиля, гражданских¹ и профессиональных ученых, преподавателей и студентов по всему миру. Доступный в различных формах, мощный, но недорогой микроконтроллер действительно может работать сам по себе и с другими инструментами, помогая людям разрабатывать очень мощные и элегантные решения. Мы ожидаем, что, подобно предыдущим поколениям продуктов от Raspberry Pi Foundation, Raspberry Pi Pico совершит еще одну революцию в области технологий, образования, развлечений и других массовых начинаний.

Основываясь на нашем опыте создания публикаций об **одноплатных компьютерах** (Single-Board Computers, SBC) Raspberry Pi, мы написали эту книгу, чтобы познакомить читателя с новыми и старыми проектами, для удовлетворения различных потребностей целевой аудитории: студентов, преподавателей, инженеров, ученых, художников и технических энтузиастов, которые хотят разрабатывать встроенные системы, предназначенные для экономической автоматизации, устройств IoT², робототехники, медицинских устройств и художественных проектов.

Мы постарались сохранить разнообразие в проектах, а также представили различные датчики, способы программирования, описания интерфейсов и другие подробности, достаточные для того, чтобы как новички, так и продвинутые читатели могли создавать и реализовывать свои задумки на основе Raspberry Pi Pico.

Для кого предназначена эта книга

Как уже говорилось, мы разработали материалы к проектам, рассчитанным на широкий круг читателей. Возможно, вы опытный любитель или профессионал, заинтересованный в понимании того, как Pico может помочь вам в ваших проектах. У вас может быть маленький или большой опыт работы с электроникой, одноплатными компьютерами, микроконтроллерами или программированием. Вы можете обладать всеми необходимыми навыками и быть в поиске новых проектов, чтобы развлечь себя или преподавать своим ученикам. Эта книга рассчитана на людей с самым разнообразным опытом и направлением работы.

Тем не менее некоторый базовый опыт в программировании, электронике и смежных областях будет очень полезен при ознакомлении с материалами и проектами в книге. Если вы хотите начать свой опыт программирования на

¹ Гражданская наука (citizen science) – направление исследований с привлечением добровольцев из числа непрофессионалов. Термин «гражданский ученый» (citizen scientist) распространился на Западе в последнее десятилетие. Как правило, этим термином называют волонтеров, участвующих на добровольных началах в каком-либо профессиональном проекте, а не просто ученых-любителей. – Прим. перев.

² Internet of Things, «интернет вещей». – Прим. перев.

Python, то можете обратиться к другой нашей публикации *Python Programming with Raspberry Pi* («Программирование на Python с помощью Raspberry Pi»), также выпущенной издательством Packt³.

О чем рассказывает эта книга

В **главе 1** «Начало работы с Raspberry Pi Pico» излагаются основы Raspberry Pi Pico, его разновидности, аксессуары и способы программирования. Мы также покажем, как создать классический пример «Hello World» и заставить мигать светодиод.

В **главе 2** «Последовательные интерфейсы и их приложения» мы исследуем, как использовать последовательные интерфейсы Raspberry Pi Pico для взаимодействия с датчиками, дисплеями и другим оборудованием. Мы также продемонстрируем, как заставить работать модуль Wi-Fi и подключить Raspberry Pi Pico к интернету.

В **главе 3** «Проекты домашней автоматизации» рассматриваются простые проекты домашней автоматизации, которые можно выполнить за выходные, а также приложения с последовательными интерфейсами. Мы также представляем Arduino RP2040 Connect и то, как его можно использовать вместо Pico.

Глава 4 «Весело проводите время в саду!» позволяет нам глубже разобраться в реализации проектов с помощью Pico. Мы подключаем датчик почвы к живому растению, измеряем температуру и влажность почвы, загружаем данные на аналитическую платформу IoT и визуализируем собранные данные.

Глава 5 «Строим метеостанцию» – это особое удовольствие для любителей погоды и гражданских ученых. Мы построим метеостанцию с различными датчиками и вариантами интерфейса с Raspberry Pi Pico.

Глава 6 «Проектируем настенный семисегментный дисплей» посвящена созданию средств отображения. Мы обсуждаем управление дисплеем через последовательный порт или из локальной сети.

В **главе 7** «Разрабатываем устройство слежения за качеством воздуха», продолжая предыдущую главу, мы демонстрируем другое применение средств наглядного отображения, на этот раз используя два разных подхода: один с использованием существующих источников данных, а другой – с использованием датчика двуокиси углерода для определения качества воздуха.

В **главе 8** «Беспроводная связь» мы выходим за рамки Wi-Fi и исследуем иные способы сбора и передачи данных по беспроводной сети, применяя LoRa, Sigfox и Bluetooth. Это позволит вам свободно разрабатывать беспроводные приложения с помощью Pico.

³ На русском языке можно рекомендовать онлайн-курс «Программирование на Python на Raspberry Pi» (<https://myraspberrypi.ru/programmirovanie-python-na-raspberry-pi.html>). Для более подробного ознакомления с языком Python рекомендуется книга «Основы программирования на языке Python» (М.: ДМК Пресс, 2017), для общего введения в одноплатный компьютер Raspberry Pi – книга «Raspberry Pi. Руководство по настройке и применению» (М.: ДМК Пресс, 2014). – Прим. перев.

Глава 9 «*Строим робота!*» предназначена для энтузиастов робототехники. Мы демонстрируем, как можно управлять роботом с помощью PiCo. В этой главе мы также представим MicroPython для тех, кто планирует использовать легкий код.

Глава 10 «*Знакомство с приложениями TinyML*» – это руководство для тех из вас, кто хочет разрабатывать приложения искусственного интеллекта (ИИ) с помощью PiCo. Мы познакомим вас с TinyML – фреймворком⁴, специально ориентированным на облегченные приложения ИИ. Покажем вам пример распознавания ключевых слов, который поможет подготовить почву для дальнейшего изучения этого направления.

Глава 11 «*Создаем готовый продукт*» демонстрирует, как еще больше ускорить процесс и создать готовый продукт. Мы демонстрируем метод создания несущей печатной платы для PiCo, а также разъясняем, как использовать готовый модуль для подключения.

В **главе 12** «*Дополнительные возможности работы с PiCo*» мы завершаем книгу советами и рекомендациями, которые помогут продвинуть ваши проекты с помощью PiCo дальше. Мы обсуждаем, как можно обновить прошивку PiCo, как Arduino IDE можно использовать для программирования PiCo, как провести измерение потребления PiCo и программирование контактов ввода/вывода (PIO).

Мы надеемся, что продемонстрированные в книге проекты подготовят вас к будущим приключениям с Raspberry Pi PiCo.

Как извлечь максимум пользы из этой книги

Программное/аппаратное обеспечение, описанное в книге	Требования к операционной системе
CircuitPython	Windows, macOS или Linux
MicroPython	Windows, macOS или Linux
Arduino IDE (C/C++)	Windows, macOS или Linux

Проекты, обсуждаемые в этой книге, требуют большого количества разнообразного аппаратного обеспечения. Чтобы сохранить единство повествования, мы использовали Raspberry Pi PiCo во всех главах. Вам также понадобится беспроводной модуль ESP32 для подключения к сети. В каждой главе приведен список рекомендуемого оборудования, и мы по возможности перечислили альтернативные варианты. Мы оставляем за вами право заменять компоненты по своему усмотрению.

⁴ Фреймворк (framework) – программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. – *Прим. перев.*

Если вы используете цифровую версию этой книги, мы советуем вам вводить код самостоятельно или получить доступ к коду из репозитория книги на GitHub (ссылка доступна в следующем разделе). Это поможет вам избежать любых потенциальных ошибок, связанных с копированием и вставкой кода.

Если вы обнаружите какие-либо конкретные проблемы, связанные с оборудованием или с примерами кода, размещенными в репозитории, пожалуйста, не стесняйтесь поднимать проблему на GitHub.

Загрузка файлов с примерами кода

Вы можете загрузить файлы с примерами кода для этой книги с GitHub по адресу <https://github.com/PacktPublishing/Raspberry-Pi-Pico-DIY-Workshop>. Если код будет обновлен, он будет обновлен в репозитории GitHub.

У нас также есть другие пакеты кода из нашего богатого каталога книг и видео, доступного по адресу <https://github.com/PacktPublishing/>. Загляните туда!

Код в действии

Видеоролики с кодом в процессе работы для этой книги можно посмотреть по адресу <https://bit.ly/3OZJb5Z>.

Загрузите цветные изображения

Мы также предоставляем PDF-файл, содержащий цветные изображения скриншотов и рисунков, используемых в этой книге. Вы можете скачать его здесь: https://static.packt-cdn.com/downloads/9781801814812_ColorImages.pdf.

Текстовые соглашения

В этой книге используется ряд текстовых соглашений.

Код в тексте: указывает кодовые слова и имена переменных в тексте, имена таблиц базы данных, названия программных библиотек. Пример: «Модуль board содержит определения контактов и периферийных устройств, специфичных для платы».

Имена файлов, имена папок, расширения файлов, пути файловой системы. Пример: «Файл *rp2040-datasheet.pdf* содержит документацию микроконтроллера RP2040».

Доменные имена и адреса в интернете, если они не представляют собой законченного URL, представляются полужирным курсивом: *bit.ly*.

Блок кода задается следующим образом:

```
from machine import Pin
import utime
led = Pin(25, Pin.OUT)
while True:
```

Когда мы хотим привлечь ваше внимание к определенной части блока кода, соответствующие строки или элементы выделяются жирным шрифтом:

```
while True:  
    led.toggle()  
    utime.sleep(1)
```

Любой ввод или вывод из командной строки записывается следующим образом:

```
>>> print("Hello World")
```

Жирный шрифт: обозначает новый термин или важное слово. Названия в меню или диалоговых окнах также выделены жирным шрифтом. Пример: «Импульс можно создать на вкладке **Create impulse** слева».

Советы или важные примечания

Обозначаются вот так.

Связаться с нами

Обратная связь от наших читателей всегда приветствуется.

Несмотря на то что мы приложили все усилия для обеспечения точности нашего контента, ошибки случаются. Если вы нашли ошибку в этой книге, мы были бы признательны, если бы вы сообщили нам об этом. Если у вас есть вопросы по какому-либо аспекту этой книги, напишите нам по адресу dmkpress@gmail.com и укажите название книги в теме вашего сообщения.

Поделитесь своим впечатлением

Когда вы читаете «*Raspberry Pi Pico в любительских проектах*», мы будем рады узнать ваше мнение о книге! Пожалуйста, разыщите книгу в каталоге сайта издательства «ДМК Пресс» (<https://dmkpress.com>) и поделитесь своими впечатлениями.

Ваш отзыв важен для нас!

Часть I

Введение в Pico

Цель этой части – представить Raspberry Pi Pico, его варианты и периферийные устройства, доступные на Raspberry Pi Pico. Часть I начинается с проекта мигающего светодиода и описания последовательных интерфейсов, доступных на микроконтроллере RP2040. Затем мы будем продвигаться вперед, работая над простыми проектами домашней автоматизации и садоводства.

Эта часть содержит следующие главы:

- глава 1 «Начало работы с Raspberry Pi Pico»;
- глава 2 «Последовательные интерфейсы и их приложения»;
- глава 3 «Проекты домашней автоматизации»;
- глава 4 «Весело проводите время в саду!».

Глава 1

Начало работы с Raspberry Pi Pico

В этой главе мы хотели бы углубиться в краткое введение в Raspberry Pi Pico и микроконтроллер RP2040. Мы рассмотрим функции Raspberry Pi Pico, периферийные устройства RP2040, дополнительное оборудование для Pico и платы на основе RP2040, разработанные другими производителями. Мы также обсудим варианты языка программирования, доступные для Pico, и дополним главу обсуждением простых примеров «Hello World!», где мы печатаем что-то на экране и мигаем светодиодом.

К концу этой первой главы вы освоитесь с Pico, будете готовы приступить к программированию микроконтроллера RP2040 и начать планировать реализацию проектов из последующих глав этой книги, а также продумывать, как вы можете создавать свои собственные проекты с помощью Raspberry Pi Pico!

Мы собираемся осветить следующие основные темы:

- представление Raspberry Pi Pico и RP2040;
- обсуждение вариантов платы Pico;
- пайка разъемов на плате Pico;
- реализация примера «Hello World!»;
- реализация примера мигания светодиода;
- определение полезного дополнительного оборудования для Pico.

Технические требования

Аппаратное и программное обеспечение, необходимое для этой вводной главы, будет использоваться на протяжении всей книги. В следующих главах мы также представим дополнительные или специфические для главы требования к оборудованию.

Требования к оборудованию для этой главы:

- ноутбук или персональный компьютер с портом **универсальной последовательной шины (USB)**;
- *дополнительно:* паяльное оборудование, включая электропаяльник, припой, защитные очки и разное оборудование;

- *дополнительно*: безопасная макетная плата и комплект соединительных проводов.

Видеоролики с кодом в действии для этой главы можно посмотреть по адресу <https://bit.ly/3MRdYjx>.

Представление Raspberry Pi Pico и RP2040

Raspberry Pi Pico – это новый одноплатный микроконтроллер, представленный компанией Raspberry Pi Foundation. Плата микроконтроллера Pico, несмотря на невысокую цену, обладает неплохими характеристиками. Pico построен на RP2040, двухъядерном микроконтроллере с ядром Cortex-M0+. Плата поставляется в общей сложности с 40 контактами, по 20 контактов с каждой стороны, как показано на рис. 1.1. Pico поставляется с 2 МБ встроенной флеш-памяти и светодиодом на выводе GP25. GP (или полностью **GPIO**) означает **General Purpose Input/Output** – контакт ввода/вывода общего назначения.



Рис. 1.1. Raspberry Pi Pico

Спецификация для Raspberry Pi Pico доступна по адресу: <https://bit.ly/3cww1lc>. В проектах, изложенных в этой книге, мы будем использовать различные периферийные устройства, доступные на Pico. Удобно распечатать разводку выводов, предоставленную Raspberry Pi Foundation (источник: <https://bit.ly/3wa0nwq>). Схема разводки может помочь с выбором выводов при планировании проекта. На рис. 1.2 показана схема разводки, представленная Adafruit Industries.

Плата Pico может использоваться в различных приложениях, связанных с роботами, дистанционным мониторингом, гражданской наукой и т. д. В этой книге мы познакомим вас с различными примерами применения, включающими в том числе и периферийные устройства микроконтроллера RP2040.

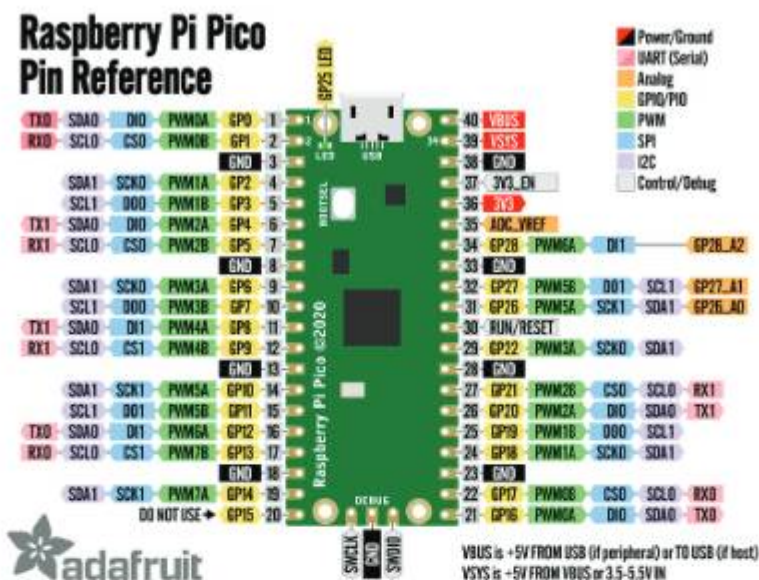


Рисунок 1.2. Разводка выводов Raspberry Pi Pico
(источник: Adafruit Industries)

Микроконтроллер RP2040

RP2040 представляет собой двухъядерный микроконтроллер ARM Cortex-M0+ с 264 килобайтами (КБ) статической оперативной памяти (SRAM), но не имеет встроенной флеш-памяти. RP2040 включает в себя целый ряд периферийных устройств: двухпроводной последовательный интерфейс (I2C), последовательный периферийный интерфейс (SPI), универсальный асинхронный интерфейс приемника-передатчика (UART) и контакты с программируемым вводом/выводом (PIO). Выводы PIO микроконтроллера RP2040 позволяют вам создавать свой собственный интерфейс, такой как дополнительный UART или видеинтерфейс. В главе 12 «Дополнительные возможности работы с Pico» мы обсудим использование PIO для подключения периферийных устройств.

Вот список ресурсов для более подробного изучения RP2040:

- технические характеристики RP2040 доступны по следующей ссылке:
<https://bit.ly/3rw41x5>;
- техническое описание Raspberry Pi Pico доступно по следующей ссылке:
<https://bit.ly/3cww1lc>;
- видео от Raspberry Pi foundation о PIO в микроконтроллере RP2040 можно найти по следующей ссылке:
<https://bit.ly/39ni6Xg>;
- документацию по RP2040 и Raspberry Pi Pico от Raspberry Pi Foundation можно найти по следующей ссылке:
<https://bit.ly/3flFLv9>.

Мы рекомендуем вам загрузить технические описания Pico и RP2040. Они пригодятся в качестве справочного материала при разработке, и в определенных местах этой книги мы будем отсылать вас к этим документам для получения дополнительной информации.

О приобретении компонентов в России

Приобрести Raspberry Pi Pico и его варианты, а также комплектующие, необходимые для выполнения проектов из книги, в России можно в магазинах сети «Чип и Дип», в российских интернет-магазинах, торгующих электронными компонентами (например, iarduino.ru; electronshik.ru; amperka.ru и др.), а также на AliExpress. В этой главе далее и некоторых других разделах книги встречаются ссылки на производителей и интернет-магазины, доступные в США и Европе, но не работающие с Россией (например, adafruit.com или digkey.com). В большинстве случаев подобная ссылка сопровождается сноской с указанием российского источника напрямую. Если она отсутствует, то указанные там товары или их аналоги следует самостоятельно поискать в отечественных интернет-магазинах или через специальных посредников по приобретению.

Обзор вариантов платы Pico

С момента запуска Raspberry Pi Pico было выпущено несколько вариантов платы на основе RP2040 от различных компаний, выпускающих свободное аппаратное обеспечение. Опишем более подробно некоторые из них.

- **SparkFun Thing Plus – RP2040:** это плата с открытым исходным кодом от SparkFun (<https://bit.ly/2N55vUn>). Thing Plus выпускается в форм-факторе Feather от Adafruit⁵. Уникальность этой платы заключается в том, что она поставляется с держателем карты microSD и индивидуально адресуемым RGB-светодиодом. На рис. 1.3 представлена плата RP2040 Thing Plus с нижней стороны.



Рисунок 1.3. SparkFun Thing Plus (RP2040)

⁵ Форм-фактор Feather – популярная типовая конфигурация платы размерами 60×23 мм (2,38×0,9 дюйма), разработанная компанией Adafruit. В форм-факторе Feather выпускается множество любительских электронных устройств по всему миру (среди самых популярных, например, ESP 8266 и ESP 32). Подробнее об этом стандарте см. <https://learn.adafruit.com/adafruit-feather/feather-specification>. – Прим. перев.

- Процессор **SparkFun MicroMod RP2040**. Это еще один вариант от SparkFun (<https://bit.ly/3clp0hG>). Он поставляется с 16 МБ встроенной флеш-памяти в форм-факторе MicroMod, использующем стандарт M.2⁶. На рис. 1.4 вы можете увидеть верхнюю сторону платы. Обратите внимание на выемку в форме полумесяца, которая используется для крепления модуля к несущей плате с помощью винта M2.5.



Рисунок 1.4. Процессор MicroMod RP2040

SparkFun также производит различные несущие платы для экосистемы MicroMod. Например, несущая плата (<https://bit.ly/3cnlrHF>), показанная на следующем рисунке, была разработана для управления дисплеем с интерфейсом **HDMI (high-definition multimedia interface, мультимедийный интерфейс высокой четкости)** с использованием RP2040.

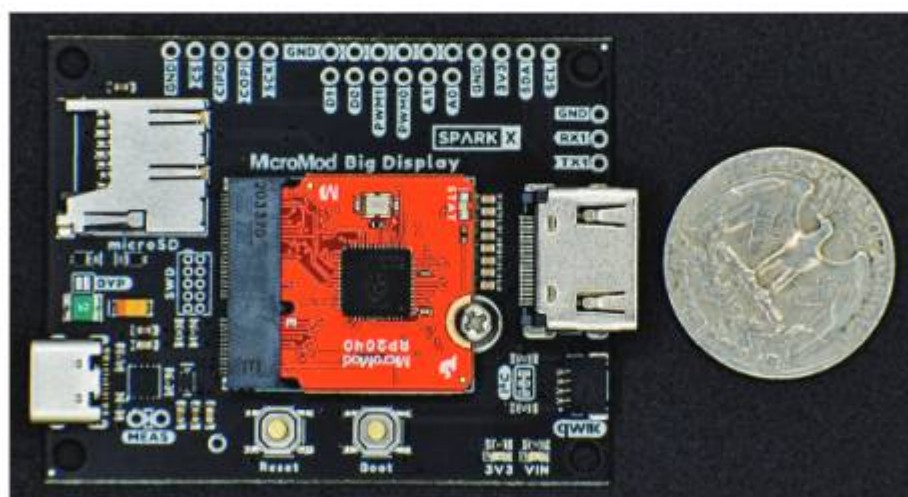


Рисунок 1.5. Большая дисплейная плата MicroMod для процессора RP2040

- **SparkFun Pro Micro – RP2040**. Плата Pro Micro – RP2040 (<https://bit.ly/3cnhVgH>) – это вариант, который принадлежит к относительно небольшой экосистеме семейства плат Pro Micro. На плате размещается 16 МБ

⁶ Формат M.2 – спецификация компьютерных карт расширения и их разъемов, расширяющая возможности PCI Express в сторону большей компактности. Плата стандарта M.2 имеет ножевой разъем, аналогичный PCI Express, но с уменьшенным шагом 0,5 мм (см. рис. 1.4). – Прим. перев.

флеш-памяти, индивидуально адресуемые светодиоды RGB и универсальные зубчатые площадки выводов, которые позволяют либо установить разъемы, либо припаивать модуль непосредственно к другой печатной плате. Универсальные контакты Pro Micro показаны на рис. 1.6.



Рисунок 1.6. SparkFun Pro Micro – RP2040

- **Pimoroni Tiny 2040.** Эта плата от Pimoroni (<https://bit.ly/3d9f7Tf>) размером примерно с четверть дюйма имеет 8 МБ флеш-памяти и RGB-светодиод. Универсальные зубчатые площадки выводов позволяют припаять его либо к разъему, либо непосредственно к пользовательской печатной плате. Мы должны отметить, что вам понадобится вырез в вашей плате, чтобы припаять таким образом модуль. Это связано с тем, что микроконтроллер в плате Tiny 2040 находится на нижней стороне, как показано на рис. 1.7. Мы далее проиллюстрируем установку этой платы на пользовательскую печатную плату.

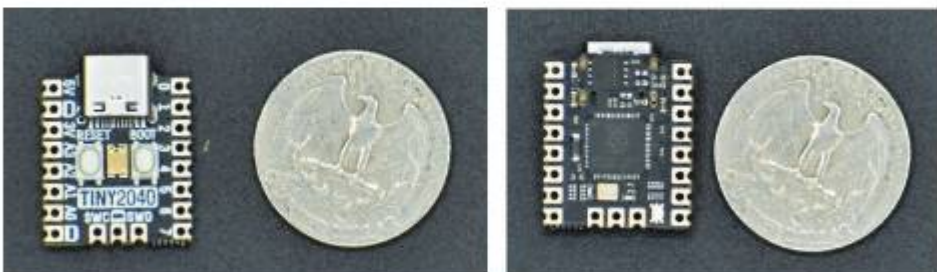


Рисунок 1.7. Pimoroni Tiny 2040

- **Adafruit Feather RP2040.** Как следует из названия, эта плата от Adafruit (<https://bit.ly/3cm3tW0>) представляет собой печатную плату с микроконтроллером RP2040, несущую 8 МБ флеш-памяти. Как и в случае с SparkFun Thing Plus, она отлично сочетается с разъемами Qwiic/STEMMA⁷. Плата показана на рис. 1.8.

⁷ Qwiic и STEMMA – стандарты разъемов для подключения внешних модулей от компаний SparkFun (Qwiic) и Adafruit (STEMMA, см. также далее). Другие известные в нашей стране подобные стандарты – Grove (Seeed Studio), отечественные Trema (arduino.ru) и Тройка (amperka.ru). – Прим. перев.

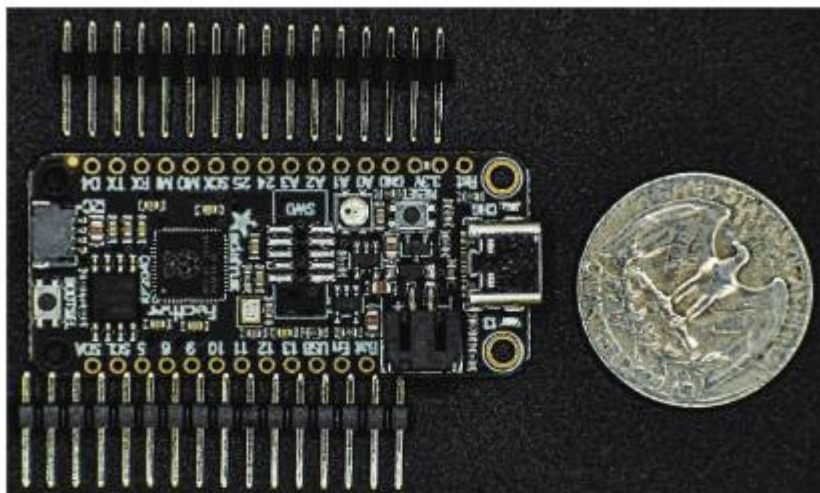


Рисунок 1.8. Adafruit Feather RP2040

- **Adafruit ItsyBitsy RP2040.** Эта плата от Adafruit (<https://bit.ly/3sqdB5R>) является дополнением к их линейке продуктов Itsy Bitsy. С точки зрения разводки выводов она идентична другим продуктам Itsy Bitsy от Adafruit. Эта плата поставляется с 8 МБ встроенной флеш-памяти. Вариант Itsy Bitsy, показанный на следующем рисунке, удобен для макетирования, что позволяет встроить плату в ваш проект.



Рисунок 1.9. Adafruit ItsyBitsy RP2040

- **Adafruit QT Py RP2040.** Эта плата (<https://bit.ly/3IU201q>) является дополнением к семейству продуктов QT Py от Adafruit. Она также поставляется с 8 МБ встроенной флеш-памяти. Зубчатые контакты модуля, показанные на рисунке ниже, позволяют спроектировать несущую плату, с помощью которой плата QT Py RP2040 может быть встроена в ваш дизайн. Поскольку микроконтроллер RP2040 расположен на нижней стороне, вам необходимо убедиться, что в вашей конструкции есть вырез для размещения QT Py.