

Содержание

Введение	5
1. Установка драйвера микросхемы Ch343	7
2. Проверка подключения драйвера Ch343	9
3. Установка среды программирования ESP 32	10
4. Установка Python	11
5. Прошивка firmware в ESP 32	12
6. Установка интерпретатора MicroPython	14
7. Краткие сведения о программировании на языке Python	17
7.1. Алфавит и словарь языка Python	17
7.2. Операторы	18
7.3. Вывод данных	19
7.4. Ввод данных с клавиатуры	20
7.5. Отступы в скриптах программ	21
7.6. Циклы в Python	21
7.7. Списки	23
7.8. Стандартные функции языка Python	24
8. On-line запуск программ на ESP 32	25
9. Off-line запуск и исполнение программ на ESP 32	27
9.1. Запуск программ без использования boot-файлов	27
9.2. Запуск программы с использованием boot-файлов	28
10. Электрические схемы, закон Ома	30
10.1. Резисторы	32
10.2. Потенциометры	34
10.3. Закон Ома	35
10.4. Светодиоды	36
11. Описание выводов ESP 32	38
11.1. Входы-выходы общего назначения	39
11.2. Логические ноль (лог.0) и единица (лог.1)	40
11.3. Импульсные цифровые сигналы	41
11.4. Ограничения на использование портов ESP 32	43
11.5. Как устроены универсальные порты	43
12. Краткие сведения о беспроводной связи ESP 32	46
13. Макетная плата и источники питания	47
14. Предостережения	50

15. Проекты.....	51
15.1. Мигаем встроенным светодиодом ESP 32	51
15.2. Мигаем внешним светодиодом	53
15.3. Включение исполнительного устройства кнопкой	54
15.4. Настольная лампа (устраняемдребезг контактов).....	57
15.5. Бегущая точка (индикатор уровня сигналов).....	60
15.6. Проект “Дышащий светодиод”	63
15.7. Проект “Комета”	65
15.8. Управление RGB-светодиодом	69
15.9. Зуммеры, NPN- и PNP-транзисторы	72
15.10. Последовательный обмен данными	77
15.11. Измерение аналоговых напряжений	84
15.12. Ручное регулирование яркости светодиода	87
15.13. Ночная лампа-автомат.....	88
15.14. Джойстик	90
15.15. Семисегментный индикатор	92
15.16. Управление электродвигателями.....	95
15.17. Управление серводвигателем	100
15.18. Использование дисплея LCD1602.....	104
15.18.1. Дисплей-счетчик	106
15.18.2. Позиционирование символов на дисплее	108
15.18.3. Создание специальных символов.....	109
15.19. Ультразвуковой дальномер	111
15.20. Передача данных через Bluetooth.....	114
15.21. Управление ESP 32 со смартфона через Bluetooth.....	121
15.22. Автоматическая передача данных через Bluetooth	123
15.23. Wi-Fi сети для ESP 32	126
15.24. Трансляция видео с камеры ESP 32 в локальной сети	140
15.25. Делаем фотографию	149
15.26. Видеотрансляция с камеры ESP 32 в сети Интернет.....	150

Введение

Почему “самоучитель”?

К сожалению, качественные кружки и курсы робототехники недоступны школьникам сел и небольших городов. Даже тем, кто живет в крупных городах, такие кружки не очень-то и доступны: они платные, требуют значительных затрат времени на поездки туда и обратно, то есть, гораздо дороже хорошей книги. On-line лекции имеют свои неудобства. В книге вы мгновенно можете вернуться выше всего на две строки или сразу на несколько страниц. Можно в любое время приостановить чтение, чтобы обдумать то, что прочитали. В on-line лекциях поиск нужного вам места в лекции отвлекает, сбивает ход мыслей... Книга предлагает выверенную систему подачи большого по объему материала, от простого к сложному, в которой читатель полностью контролирует скорость изучения. С этих точек зрения, самоучитель — лучший вариант. Тем более, что в 21-м веке умение учиться самостоятельно — важнейшее конкурентное преимущество.

Почему ESP-32? Контроллеры являются ключевыми элементами практически всех современных устройств промышленного и бытового назначения, включая роботов. Контроллеры обеспечивают как управление работой отдельных компонентов устройств, так и координацию работы всех компонентов устройств. Центральный контроллер, получая и обрабатывая информацию от подчиненных контроллеров и(или) сенсоров и исполнительных устройств, обеспечивает выполнение всех функций, для которых создано устройство. Подчиненные центральному контроллеры:

- обрабатывают данные от сенсоров (обнаруживают препятствия, определяют скорость, направление и траекторию движения робота и другие параметры);
- управляют исполнительными устройствами, обеспечивающими перемещение робота в пространстве и его манипуляторами;
- взаимодействуют с оператором с помощью мониторов, клавиатур и др..

ESP-32 — один из самых современных, универсальный и доступный по цене микроконтроллер, пригодный для широкого круга применений. Его двухъядерный процессор работает на частоте до 240 МГц, обеспечивая отличную вычислительную мощность. Он имеет большой объем памяти, встроенные Wi-Fi и Bluetooth. Широкий набор интерфейсов (ADC, UART, I2C, I2S, SPI) позволяет подключать к нему различные сенсоры, мониторы и исполнительные устройства.

Пригодятся маленькие кусачки и пинцет: они облегчат монтаж схем проектов на макетной плате.

1. Установка драйвера микросхемы Ch343

USB (*Universal Serial Bus*) — универсальная последовательная шина приема и передачи данных. **USB-разъемы (порты)** — это привычные всем пользователям ПК **интерфейсы**, которые обеспечивают подключение к ПК клавиатур, мышек, флэш-памяти и др.. Они не только позволяют ПК обмениваться данными с периферийными устройствами, но и обеспечивают эти устройства питающим напряжением 5В. Они также обеспечивают возможность горячей замены, что позволяет, не выключая ПК, (без сбоя) подключить или отключать периферийные устройства.

До появления **USB-интерфейса** подключение периферийных устройств к ПК для последовательного обмена данными осуществлялось с использованием **COM-портов** (communication port), которые работали на основе несколько других правил.



Рис. 1.1 Розетка 9-контактного разъема COM-порта

Многие промышленные устройства, контроллеры, медицинское оборудование, приборы для диагностики автомобилей и другое оборудование продолжают использовать **COM-порты** для связи с ПК. С их помощью можно подключать оборудование на расстояниях до десятков метров, а USB-порты работают без сбоев на расстоянии только до 3-х метров. Сигналы в **COM-портах**, в отличие от **USB**,

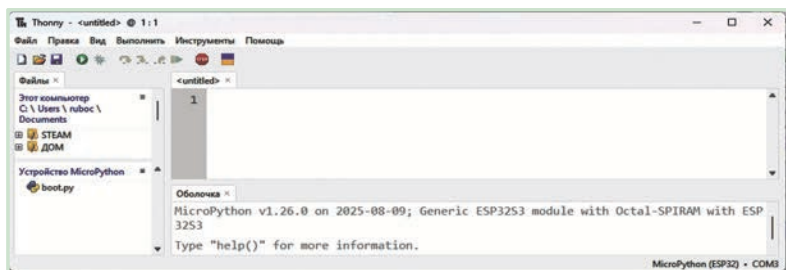


Рис.6.6 Интерфейс Thonny с проводником

7. Краткие сведения о программировании на языке Python

7.1. Алфавит и словарь языка Python

Основой языка программирования Python, как и любого другого языка, является **алфавит** — набор допустимых символов, которые можно использовать для записи программы. **В алфавит Python входят:**

- латинские строчные и прописные буквы (a, b, c, ..., x, y, z, A, B, C, ..., X, Y, Z);
- арабские цифры (0, 1, 2, ..., 7, 8, 9);
- специальные символы (*знак подчеркивания, знаки препинания, круглые, квадратные и фигурные скобки, знаки арифметических операций, знак “#” начала однострочного комментария и др.*).

Для записи операций используют следующие неделимые символы:

Символ	Значение
==	знак равно
!=	знак не равно
>= и <=	больше или равно, меньше или равно

Некоторые английские слова используются в качестве служебных:

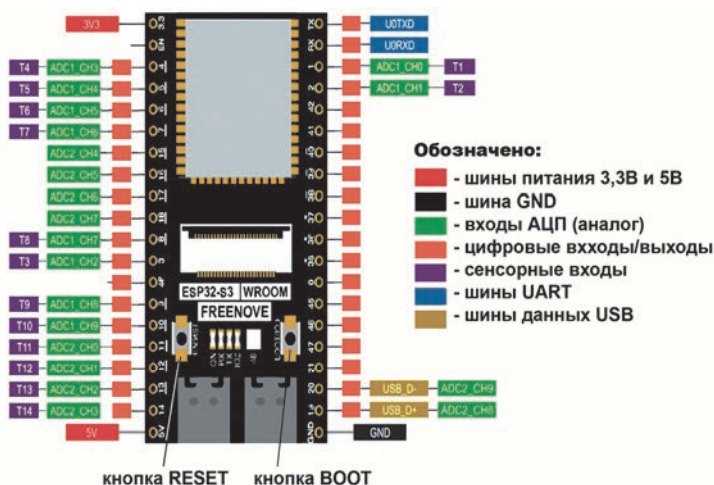
Служебные слова	Значение	Служебные слова	Значение
and	логическое И	input	ввод
break	прервать	int	целый

самым, можно менять величину тока через СИД и яркость его свечения. Резистор R2 необходим, чтобы ограничить ток через СИД, когда поворотом ручки потенциометра R1 его сопротивление станет равным нулю. То есть, максимальный рабочий ток СИД определяется величиной резистора R2 (именно он является ограничительным). Минимальный ток в этой схеме определяется суммой сопротивлений резистора R2 и потенциометра R1.

Вопросы по пройденному материалу:

1. Как рассчитать сопротивление резисторов, включенных последовательно?
2. Как рассчитать сопротивление резисторов, включенных параллельно?
3. Как рассчитать ток, протекающий в электрической цепи?
4. Как рассчитать сопротивление ограничительного резистора светодиода?

11. Описание выводов ESP 32



Temperature monitoring started...
Current temperature: 24.8 °C
Current temperature: 27.2 °C
Current temperature: 27.3 °C
Client connected from: ('192.168.0.103', 52590)
Response sent to client
Current temperature: 24.6 °C

Набери в строке поиска браузера **192.168.0.104**, браузер откроет веб-страницу (см. рис.15.52):

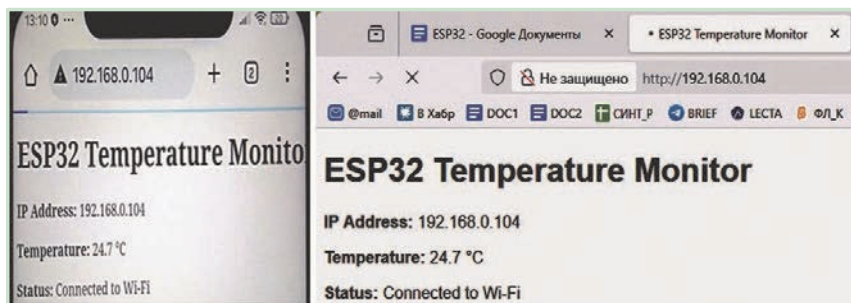


Рис.15.52 вид веб-страницы на смартфоне и ПК

Примечание. При отладке работы проекта отключай USB-кабель ESP 32 от ПК и нажимай красную кнопку **Stop Thonny** перед каждым новым запуском программы, иначе программа может иногда “подвисать”.

Для точного измерения температуры важны реальные технические характеристики как термистора, так и резистора, включенного последовательно с термистором. Замерь их электрические сопротивления мультиметром и вставь реальные значения сопротивлений в формулу программы.

В комнате температурный датчик будет показывать практически одну и ту же температуру. Это затрудняет оценку того, насколько правильно работает проект. Чтобы обеспечить быстрое и заметное изменение показаний датчика, рекомендую поставить последовательно с терморезистором не один, а два резистора (второй - на 2,2 кОм). А параллельно второму резистору поставить кнопку, закорачивающую резистор при нажатии (см.рис.14.49). Теперь, нажав кнопку ты должен видеть скачкообразное изменение температуры (см. выше сообщения в окне **Оболочка**: нажатие кнопки скачком “повысило” температуру с 24,8 до 27,2 градусов).