

ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИЙ ИМЕНИ МУХАММАДА АЛ-ХОРАЗМИЙ

Зайнидинов Х.Н.,  
Махмуджанов С.У.,  
Маъмиров Х.Х.,  
Гафуржонов М.Р.

# IoT ТЕХНОЛОГИИ

Учебное пособие для  
практических занятий



Ташкент

**МИНИСТЕРСТВО ЦИФРОВЫХ ТЕХНОЛОГИЙ  
РЕСПУБЛИКИ УЗБЕКИСТАН**

**ТАШКЕНТСКИЙ УНИВЕРСИТЕТ  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ИМЕНИ  
МУХАММАДА АЛЬ-ХОРАЗМИЙ**

Факультет «Компьютерная инженерия».

Кафедра «Искусственного интеллекта».

**ТЕХНОЛОГИИ IoT (Интернет вещей)  
Учебное пособие для практических заданий**

**Ташкент**

Методические указания основаны на типовой учебной программе, утвержденной Министерством высшего образования, науки и инноваций Республики Узбекистан.

*Пособие направлено на обучение студентов с систематическими профессиональными знаниями и эффективными практическими навыками в технологии IoT.*

*Пособие дает студентам навыки проектирования сенсорных сетей и планирования сетей для IoT. Темы включают в себя: введение в IoT. Управление данными и знаниями, а также использование устройств в технологии IoT. Понимание архитектуры IoT и применение IoT в различных областях.*

*Пособие предназначено для профессорско-преподавательского состава и студентов высших, средне-специальных и профессиональных учебных заведений.*

#### **Авторы:**

**Зайнидинов Х.Н.** - Заведующий кафедрой «Искусственный интеллект» ТАТУ, профессор, д.т.н.

**Махмуджанов С.У.** - доцент кафедры «Искусственный интеллект» ТАТУ, к.т.н.

**Мамиров Х.Х.** - Ассистент кафедры «Искусственный интеллект».

**Гофурджанов М.Р.** - Ассистент кафедры «Искусственный интеллект».

#### **Рецензенты:**

**Хамдамов У.Р.** – ректор университета «University of Management and Future Technologies», д.т.н., профессор.

**Назирова Э.Ш.** – заведующая кафедрой ТУИТ «Мультимедийные технологии», д.т.н., профессор.

*Пособие издано на основе постановления учебно-методического совета Ташкентского университета информационных технологий имени Мухаммада ал-Хоразми протокол № 6(172) от 10 февраля 2024 г.*

## Оглавление

|  |    |
|--|----|
| ВВЕДЕНИЕ.....  | 5  |
| 1- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ.....   | 9  |
| Тема: Установка и настройка операционной системы на мини-компьютере Raspberry Pi. ....   | 9  |
| 2- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ.....   | 18 |
| Тема: Изучение команд Linux и использование этих команд для выполнения операций на мини-компьютере Raspberry Pi. ....                        | 18 |
| 3- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ.....   | 23 |
| Тема: Работа с языком программирования Python на мини-компьютере Raspberry Pi. ....  | 23 |
| 4- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ.....   | 29 |
| Тема: Использование и управление светодиодом (LED) с помощью мини-компьютера Raspberry Pi 4. ....  | 29 |
| 5- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ.....   | 35 |
| Тема: Использование датчика DHT11 с помощью мини-компьютера Raspberry Pi 4. ....   | 35 |
| 6- ПРАКТИЧЕСКАЯ РАБОТА. ....   | 40 |
| Тема: Использование PIR датчика движения с помощью мини-компьютера Raspberry Pi. ....  | 40 |
| 7- ПРАКТИЧЕСКАЯ РАБОТА. ....   | 42 |
| Тема: Сборка и управление светофором с помощью миникомпьютера Raspberry Pi 4. ....   | 42 |
| 8- ПРАКТИЧЕСКАЯ РАБОТА. ....   | 46 |
| Тема: Использование ультразвукового датчика HC-SR04 на мини-компьютере Raspberry Pi. ....  | 46 |
| 9- ПРАКТИЧЕСКАЯ РАБОТА. ....   | 50 |
| Тема: Определение количества газа в помещении с помощью датчика газа MQ-5 с использованием микроконтроллера Arduino Uno. ....                | 50 |
| 10- ПРАКТИЧЕСКАЯ РАБОТА. ....  | 54 |
| Тема: Определение частоты сердечных сокращений с помощью датчика пульсоксиметра Max30102 с использованием микроконтроллера Arduino Uno. .... | 54 |

## ВВЕДЕНИЕ

Совершенствование системы подготовки кадров в сфере информационных технологий является одним из важных условий успешной реализации стратегии «Цифровой Узбекистан – 2030», развития цифровых технологий и широкого внедрения их в повседневную жизнь населения. Принимаемые меры по повышению эффективности системы профессиональной подготовки и переподготовки в сфере информационных технологий создают прочную основу для обеспечения государственных органов и сетевых организаций квалифицированными ИТ-специалистами.

В 1957 г. Джо Болен, Джордж Бил и Эверетт Роджерс из Государственного университета штата Айова ввели понятие кривой внедрения технологий. Они выдвинули идею о том, что любой новый продукт или решение движется по относительно предсказуемой траектории, похожей на гауссову кривую. Первые разработчики технологии называются новаторами, на втором этапе появляются первопроходцы, за ними следуют массы, а уже за массами — поздние последователи. Такая схема работает до сих пор, хотя в последние пару десятилетий этот жизненный цикл ускорился до сверхсветовой скорости. В некоторых случаях эта схема вместо нескольких лет или десятилетий укладывается в несколько месяцев.

В эпицентре этой ударной волны находится Интернет вещей — причем он только начинается. Когда-нибудь он станет практической основой жизни и бизнеса, пока же этот тренд затрагивает в основном новаторов и первопроходцев. Подключенные друг к другу устройства существовали со времен появления первых компьютерных сетей и бытовой электроники. Однако пока не появился интернет, никому не приходило в голову, что связь может быть глобальной. В 1990-х гг. исследователи выстраивали теоретические построения о том, как сочетание человека и машины даст совершенно новую форму коммуникации и взаимодействия.

А теперь эта реальность разворачивается у нас на глазах. Хотя и не известно, что было той искрой, из которой разгорелось пламя этой революции, можно с уверенностью сказать, что решающим событием стал выпуск компанией Apple в 2007 г. устройства под названием iPhone. Смартфоны попали в руки масс. Благодаря этому стала возможна связь в реальном времени между двумя точками на карте посредством устройства, которое можно держать в руке.

Интернет вещей предлагает нам одновременно телескоп и микроскоп для того, чтобы взглянуть на когда-то незримый мир между людьми, машинами и физическими объектами. Присваивая объектам ярлыки и подключая их к Интернету, мы вдруг получаем возможность не только следить за ними и собирать новые типы данных, но и, комбинируя всевозможные

данные, постигать новые глубины информации и знаний. Знаний, о которых еще несколько лет назад никто и думать не смел.

Идея Интернета вещей сама по себе очень проста. Представим, что все окружающие нас предметы и устройства (домашние приборы и утварь, одежда, продукты, автомобили, промышленное оборудование и др.) снабжены миниатюрными идентификационными и сенсорными (чувствительными) устройствами. Тогда при наличии необходимых каналов связи с ними можно не только отслеживать эти объекты и их параметры в пространстве и во времени, но и управлять ими, а также включать информацию о них в общую «умную планету». В самом общем виде с инфокоммуникационной точки зрения Интернет вещей можно записать в виде следующей символической формулы:

$$\text{IoT} = \text{Сенсоры (датчики)} + \text{Данные} + \text{Сети} + \text{Услуги}$$

Проще говоря, Интернет вещей – это глобальная сеть компьютеров, датчиков (сенсоров) и исполнительных устройств (актуаторов), связывающихся между собой с использованием интернет протокола IP (Internet Protocol). Например, для решения определенной задачи компьютер связывается через публичный интернет с небольшим устройством, к которому подключен соответствующий датчик (например, температуры), как это показано на рисунке.



Очевидно, что при внедрении Интернета вещей вся наша повседневная жизнь кардинально изменится. Уйдут в прошлое поиски нужных вещей, дефициты товаров или их перепроизводство, кражи автомобилей и мобильных телефонов, поскольку будет точно известно, что, в каком месте и в каком количестве находится, производится и потребляется.

Если все объекты (вещи) будут снабжены миниатюрными радиометками, то их можно будет дистанционно идентифицировать, а при наличии определенного «интеллекта» – и управлять ими. По оценкам экспертов компании Cisco количество объектов, которые Интернет вещей

сможет соединить между собой, будет сравнимо с количеством атомов на поверхности Земли.

Концепция IoT играет определяющую роль в дальнейшем развитии инфокоммуникационной отрасли. Это подтверждается как позицией Международного союза электросвязи (МСЭ) и Европейского Союза в данном вопросе, так и включением Интернета вещей в перечень прорывных технологий в США, Китае и других странах. И хотя на международном уровне данная концепция уже обретает черты сформировавшейся технологии, для нее ведутся активные работы в области стандартизации архитектуры, технических компонентов, приложений, но одновременно столь же велико количество мнений о том, как именно будет построен Интернет вещей.

Интернет вещей соединяет искусственный интеллект и разум человека новыми, совершенно удивительными и подчас пугающими способами. Он способен осмыслить движения между предметами и среди предметов, включая людей, животных, транспортные средства, воздушные потоки, вирусы и многое другое. Он распознает взаимосвязи и предсказывает алгоритмы, которые слишком сложны для разума и чувств человека, — например, состояние моста или дороги; тенденции физических процессов, происходящих в атмосфере. Интернет вещей обеспечивает поддержку систем, работающих без наблюдения за ними человеком, и, во что уже трудно поверить, становится со временем умнее, изменяя свой базовый алгоритм.

Интернет вещей (IoT) создает новый мир — измеримый мир, — где люди могут лучше управлять своей жизнью, города могут лучше управлять своей инфраструктурой, а компании могут лучше управлять своим бизнесом. Этот новый интеллектуальный подключенный мир предложит фундаментальные изменения для потребителей и общества и преобразует целые отрасли. Развитие Интернета вещей принесет множество важных улучшений в наш мир и нашу повседневную жизнь, помогая нам принимать более обоснованные решения на основе своевременных и качественных данных.

Интернет вещей (IoT) — это новая технология, которая позволяет пользователям удаленно взаимодействовать с другими устройствами по беспроводной сети. Использование технологии IoT с мини-компьютером Raspberry Pi делает удаленный доступ к другим устройствам простым и экономичным, что очень выгодно. Raspberry Pi управляется своими контактами GPIO, и если вы хотите создать проект IoT, вам нужно будет настроить эти контакты, чтобы разрешить параметры ввода и вывода для Raspberry Pi. Эти контакты оказались полезными для подключения к устройствам Интернета вещей (включая такие датчики, как датчик освещенности, датчик движения, датчик звука и т. д.).

В этой работе вы можете узнать о возможностях мини-компьютера Raspberry Pi, построить архитектуру, датчики и создать систему Интернета вещей.



# 1- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ

## Тема: Установка и настройка операционной системы на мини-компьютере Raspberry Pi.

**Цель работы:** установить и настроить операционную систему Raspberian на мини-компьютере Raspberry Pi.

### Необходимое оборудование :

Используются необходимые компоненты и предметы, перечисленные ниже.

1. <https://www.raspberrypi.org/>
2. Мини-компьютер Raspberry Pi
3. Монитор
4. microSD карта ( SD адаптер с ) — 32 ГБ использовать возможно \_\_ пока более программное обеспечение поставлять \_ \_ скачать давать возможность дам .
5. Кабель Micro HDMI-HDMI.
6. 5 В постоянного тока через разъем USB-C предназначено для питания.

### Теоретическая часть

Raspberry Pi OS — это бесплатная операционная система на базе Linux, оптимизированная для оборудования Raspberry Pi и рекомендованная для эффективного использования на Raspberry Pi. Операционная система Raspbian была создана специально для этого компьютера на базе операционной системы Debian Linux. В его состав входит веб-браузер Midori, офисный редактор Open Office и множество других необходимых программ. Операционная система поставляется с более чем 35 000 пакетов: предварительно скомпилированное программное обеспечение, упакованное в удобный формат для легкой установки на Raspberry Pi. Эта операционная система и все входящие в нее программы имеют открытый исходный код и абсолютно бесплатны.

Важно регулярно обновлять Raspberry Pi. Первая важная причина — безопасность. Устройство с операционной системой Raspberry Pi содержит миллионы строк кода, которому можно доверять.

### Особенности мини-компьютера Raspberry Pi

Raspberry Pi — это недорогой небольшой компьютер, который подключается к монитору или телевизору и использует стандартную клавиатуру и мышь. Это изящное маленькое устройство, которое позволяет людям всех возрастов учиться считать и программировать на таких языках, как Scratch и Python. Он способен делать все, что вы ожидаете от настольного компьютера, включая просмотр веб-страниц и видео высокого качества, создание электронных таблиц, обработку текста и игры.

Кроме того, Raspberry Pi обладает способностью взаимодействовать с внешним миром и использовался в широком спектре проектов цифрового производства: от музыкальных машин и родительских детекторов до метеостанций и скворечников с инфракрасными камерами. Мы видим, как люди по всему миру используют Raspberry Pi, чтобы научиться программировать и понять, как работают компьютеры. Целью создания Raspberry Pi является содействие образованию взрослых и детей, особенно в области компьютеров, информатики и смежных предметов.



Рисунок 1.1. Внешний вид мини-компьютера Raspberry Pi.

### Порядок выполнения практической работы:



Рисунок 1.2. Компоненты, необходимые для установки операционной системы

**Шаг 1** — загрузить Raspberry Pi Imager с официального сайта Raspberry Pi (<https://www.raspberrypi.org/>) . Этот инструмент позволяет вам выбрать операционную систему, автоматически загрузить ее и записать на SD-карту по вашему выбору.

## Downloads

**Raspberry Pi OS** (previously called Raspbian) is our official operating system for **all** models of the Raspberry Pi.

Use **Raspberry Pi Imager** for an easy way to install Raspberry Pi OS and other operating systems to an SD card ready to use with your Raspberry Pi:

- [Raspberry Pi Imager for Windows](#)
- [Raspberry Pi Imager for macOS](#)
- [Raspberry Pi Imager for Ubuntu](#)

Рисунок 1.3. Программное обеспечение Raspberry Pi Imager будет запущено.



Рисунок 1.4. Состояние Raspberry Pi Imager при запуске

**Шаг 2.** Выберите операционную систему. В Raspberry Pi Imager можно выбрать несколько операционных систем, но мы сосредоточимся на операционной системе Raspberry Pi. Существует 3 версии операционной системы Raspberry Pi. Мы кратко рассмотрим каждый из них, но для этого руководства мы будем использовать операционную систему Raspberry Pi (32-разрядную версию).

Выберите операционную систему Raspberry Pi (32-разрядную версию).

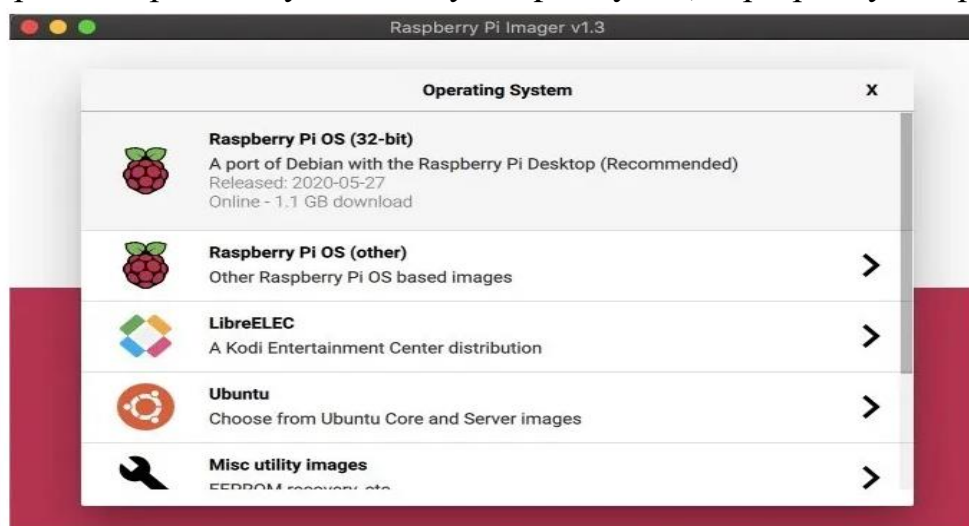


Рисунок 1.5. Окно выбора операционной системы

Raspberry Pi OS (32-разрядная версия) Lite: эта версия предоставляет вам все необходимое для запуска операционной системы. Эта версия не имеет графического пользовательского интерфейса (GUI) и имеет размер около 0,4 ГБ.

OS Raspberry Pi (32-разрядная версия): эта версия включает графический интерфейс пользователя и основное программное обеспечение. Размер этого программного обеспечения составляет около 1,1 ГБ.

Полная версия Raspberry Pi OS (32-разрядная версия): эта версия имеет графический интерфейс пользователя и больше установленного программного обеспечения, чем другие версии. Размер этой системы около 2,5 ГБ.

### Шаг 3: Выберите SD-карту.

Теперь вам нужна SD-карта, подключенная к вашему компьютеру, для копирования через выбранную нами операционную систему. Выберите SD-карту, подключенную к вашему компьютеру.



Рисунок 1.6. Начать запись операционной системы на SD-карту



Рисунок 1.7. Запись на SD-карту.

**Шаг 4.** На этом этапе выбранная операционная система будет записана на SD-карту и проверено успешность копирования.



Рисунок 1.8. Процесс загрузки операционной системы на SD-карту  
Процесс записи различается в зависимости от выбранной операционной системы. Обычно это занимает несколько минут.



Рисунок 1.9. Процесс проверки.

Процесс проверки занимает всего минуту.

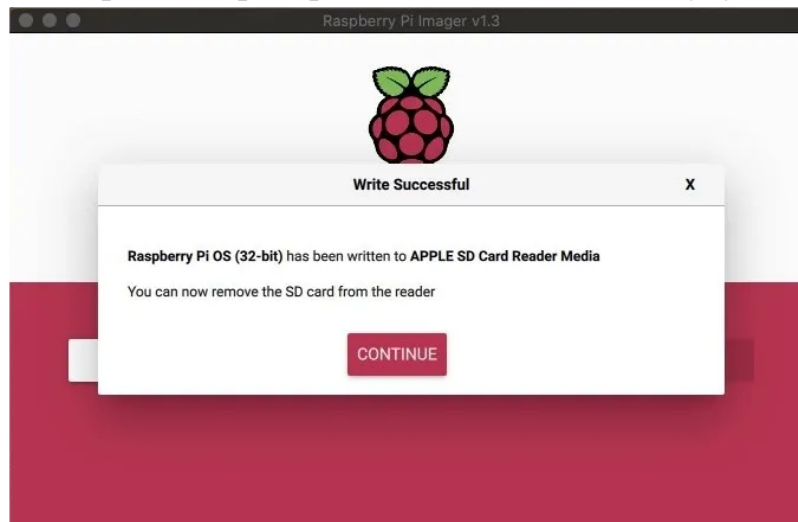


Рисунок 1.10. Статус загрузки операционной системы на SD-карту.

Операционная система скопирована на SD-карту. Теперь мы можем запустить операционную систему Raspberry Pi. После этого нажимаем кнопку продолжить.

**Шаг 5.** Загрузите операционную систему Raspberry Pi.

Карта microSDHC вставляется в мини-компьютер Raspberry Pi. Затем к мини-компьютеру Raspberry Pi подключаются питание, клавиатура, мышь и монитор.

Благодаря диалоговому окну «Добро пожаловать в Raspberry Pi» на экране мы можем сразу перейти в операционную систему Raspberry Pi.

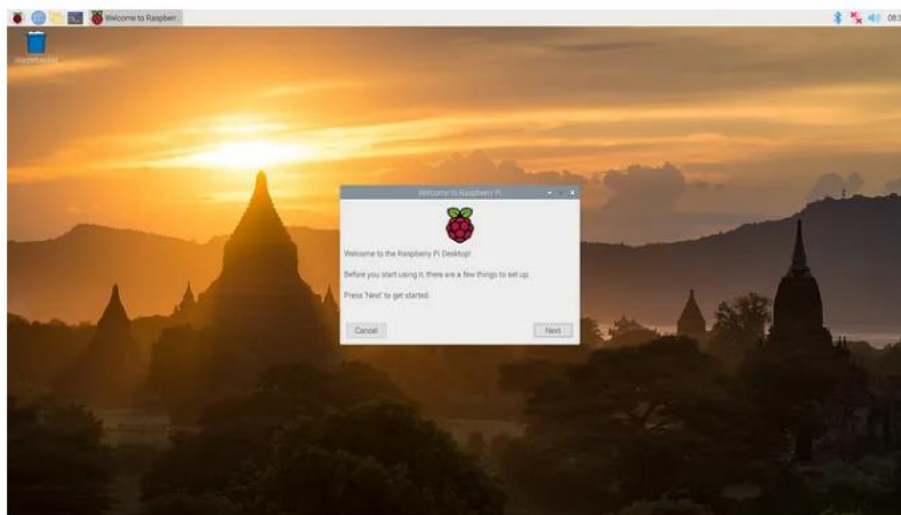


Рисунок 1.11. Диалоговое окно операционной системы  
Нажимается кнопка «Далее».

Следующим экраном будет выбор вашей страны.

Выбор страны

Выбор языка

Выбор часового пояса

Если используется английский язык и английская клавиатура, флажки установлены.

Нажимается следующий.

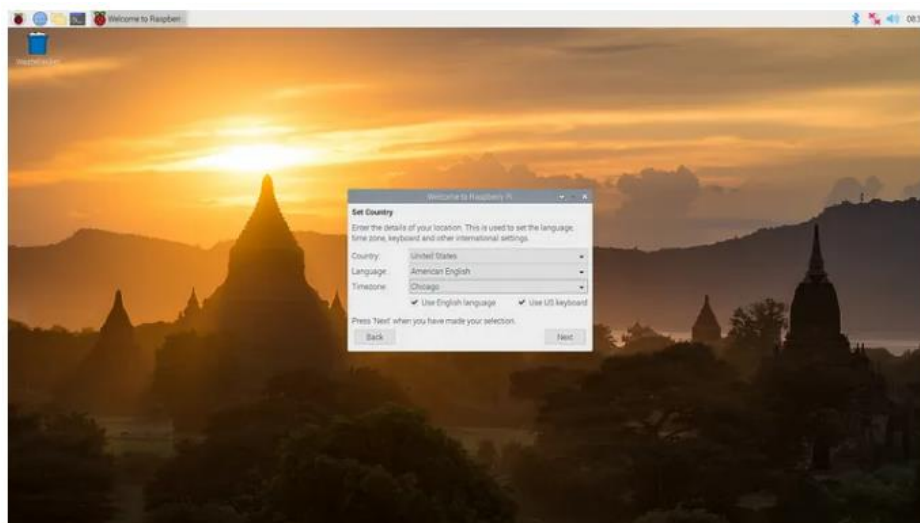


Рисунок 1.12. Окно выбора языка и региона операционной системы

На следующем экране вам будет предложено изменить пароль. Имя пользователя по умолчанию — «pi», а пароль по умолчанию — «raspberrypi».

Введите новый пароль

Подтвердите новый пароль

Нажмите кнопку

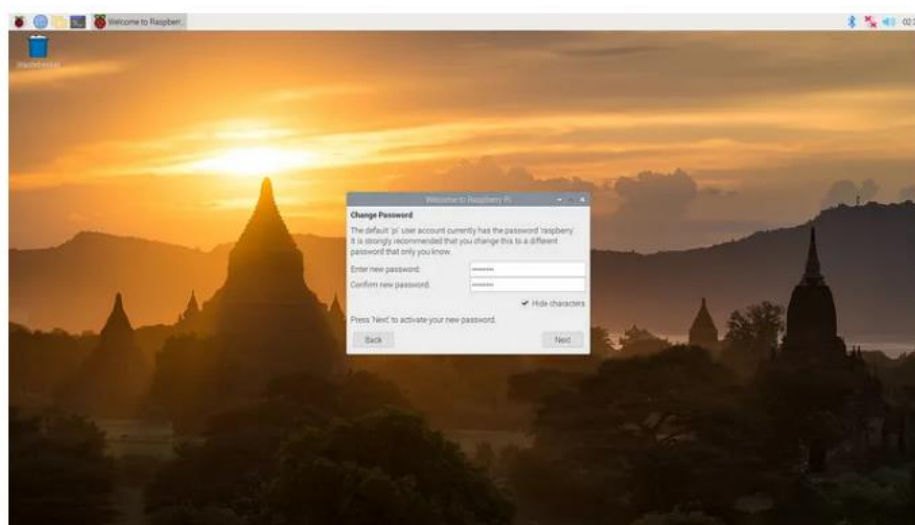


Рисунок 1.13. Окно логина и пароля

На следующем экране спрашивается, есть ли черная рамка вокруг рабочего стола. Рабочий стол должен занимать весь экран. Если нет, операционная система Raspberry Pi может внести изменения, чтобы заполнить черное пространство. Это изменение вступит в силу после перезапуска операционной системы Raspberry Pi.

Если на экране отображается черная рамка вокруг рабочего стола, установите флажок.

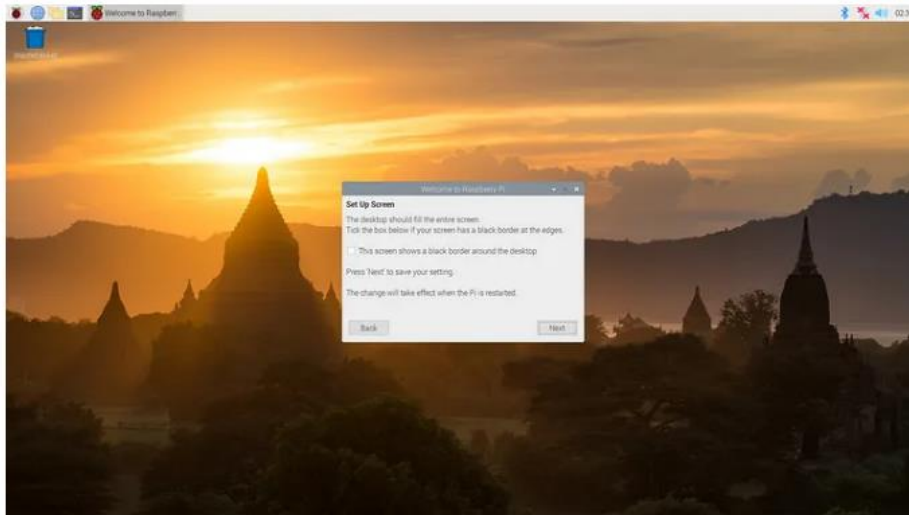


Рисунок 1.14. Окно настроек графики.

Нажмите кнопку «Далее».

На следующем экране вам будет предложено подключить Raspberry Pi к беспроводной сети. Если подключение к Интернету отсутствует, нажмите кнопку «Пропустить», чтобы пропустить этот шаг. Если у вас проводное соединение, этого шага не произойдет.

Выберите беспроводную сеть

Нажмите кнопку

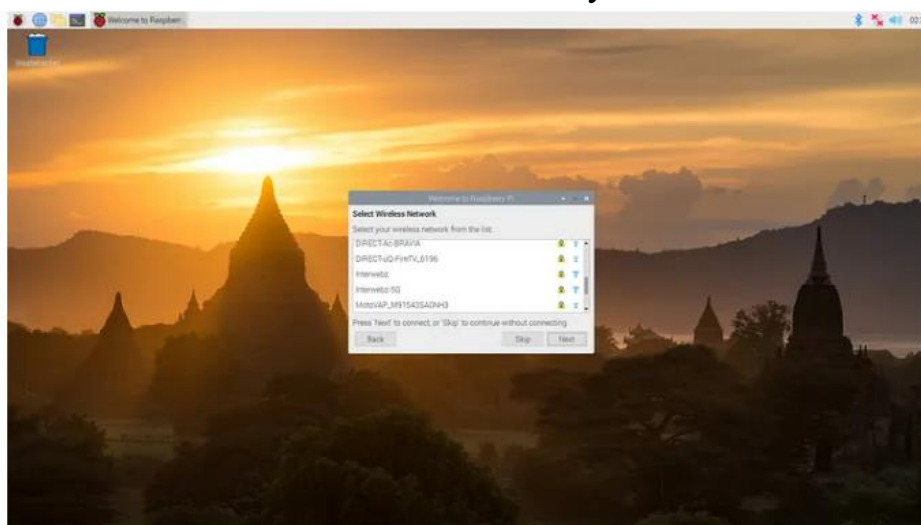


Рисунок 1.15. Окно подключения к Интернету в операционной системе.

На следующем экране вам будет предложено ввести пароль беспроводной сети. Вы можете снять флажок «Скрывать символы», чтобы видеть вводимые символы.

Введите пароль беспроводной сети

Далее нажата кнопка



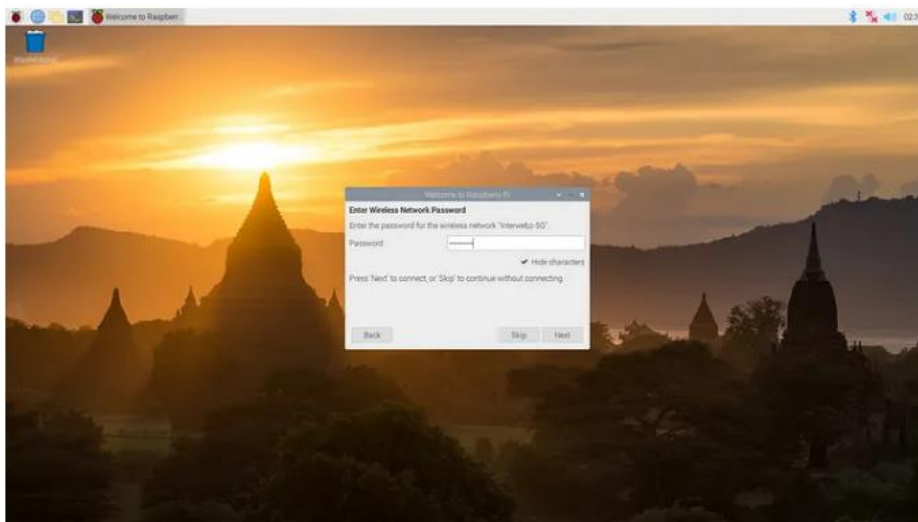


Рисунок 1.16. Окно ввода пароля беспроводной сети

На следующем экране вас спросят, хотите ли вы проверить операционную систему и приложения и обновить их при необходимости. Для выполнения этого шага требуется подключение к Интернету. Если вы хотите продолжить без проверки, нажмите «Пропустить».

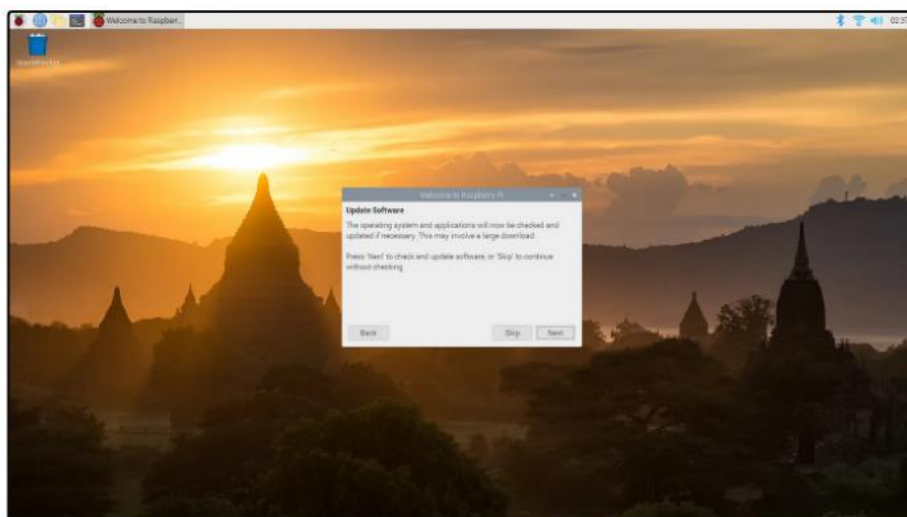


Рисунок 1.17. Проверьте окно операционной системы и приложений.

Последний экран показывает, что настройка завершена и Raspberry Pi готов к запуску операционной системы. Операционная система компьютера Raspberry Pi будет перезагружена, чтобы любые системные изменения вступили в силу.

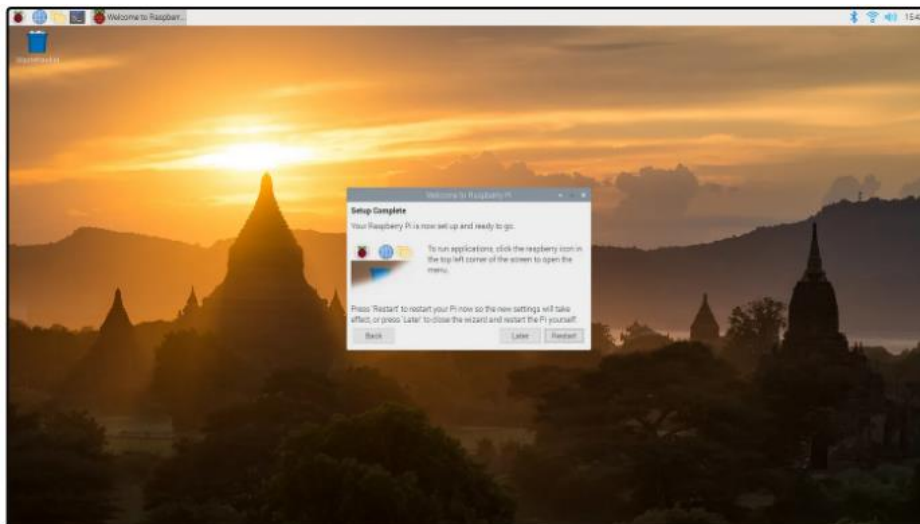


Рисунок 1.18. Перезапустить окно

Наличие этой операционной системы является основой для перехода к другим интересным и техническим проектам на мини-компьютере Raspberry Pi.

### ЗАДАНИЯ

1. Настройте установленную операционную систему.
2. Установите необходимое программное обеспечение (Arduino)
3. Учимся заходить из другой Операционной системы через программу VNC.
4. <https://www.raspberrypi.org>.
5. Установка и анализ 3-х разных операционных систем для Raspberry Pi.

## 2- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ

**Тема: Изучение команд Linux и использование этих команд для выполнения операций на мини-компьютере Raspberry Pi.**

**Цель работы:** изучить функции команд Linux на мини-компьютере Raspberry Pi и выполнять действия на основе этих команд.

**Необходимое оборудование:**

Используются необходимые компоненты и элементы, перечисленные ниже.

1. Мини-компьютер Raspberry Pi с установленной операционной системой.
2. Монитор
2. Ethernet или Wi-Fi
3. Клавиатура и мышь

### Теоретическая часть

Команды Raspberry Pi очень полезны для управления компьютером Raspberry Pi, который можно назвать одним из самых популярных компьютеров в мире. Это позволяет нам, особенно студентам и разработчикам,

работать над широким спектром приложений. Но когда вы начнете использовать черный или зеленый экран вместо цветного графического интерфейса на Windows или Mac, это станет для многих сюрпризом. Однако существует множество команд Raspberry Pi, которые можно использовать для управления и запуска приложений на Pi с панели задач. Хотя это неудобная система, она дает вам больше контроля над вашей системой и средой Linux.

### Полезные команды для Raspberry Pi

Raspberry Pi можно подключить ко многим электронным устройствам для выполнения вычислений. Если вы хотите изучить Интернет вещей, вам понадобится помощь со стороны входных и выходных контактов общего назначения, которые поставляются с Raspberry Pi. Ниже мы рассмотрим все важные и наиболее часто используемые команды Raspberry Pi, которые помогут вам устанавливать и запускать приложения, контролировать производительность системы и изменять файлы.

#### Порядок прохождения практики:

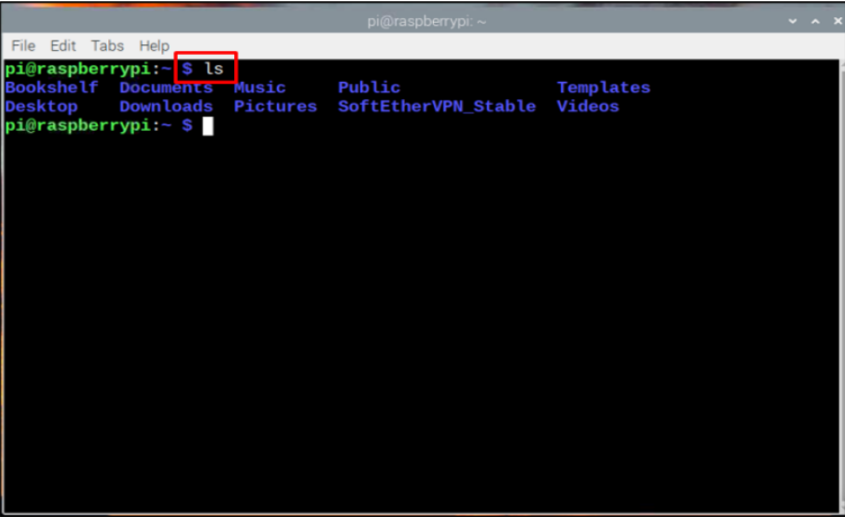
1) **ls**: посмотреть список файлов на компьютере

-a: показать все файлы (включая скрытые файлы)

-t: вывод в порядке последних созданных файлов

-R: просмотреть все подкаталоги

**Например: ls -a** для вывода списка всех файлов.



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~$ ls  
Bookshelf Documents Music Public Templates  
Desktop Downloads Pictures SoftEtherVPN_Stable Videos  
pi@raspberrypi:~$
```

Рисунок 2.1. Просмотр файлов на компьютере с помощью команды **ls** .

## 2) **pwd** : местоположение текущего каталога (папки)

```
pi@raspberrypi:~ $ pwd
/home/pi
pi@raspberrypi:~ $ ls
pi@raspberrypi:~ $ mkdir NewFolder
pi@raspberrypi:~ $ ls
NewFolder
pi@raspberrypi:~ $ cd NewFolder
pi@raspberrypi:~/NewFolder $ cd ..
pi@raspberrypi:~ $ ls
NewFolder
pi@raspberrypi:~ $ cd NewFolder
pi@raspberrypi:~/NewFolder $ touch NewFile.txt
pi@raspberrypi:~/NewFolder $ ls
NewFile.txt
pi@raspberrypi:~/NewFolder $ cp NewFile.txt OtherFile.txt
pi@raspberrypi:~/NewFolder $ ls
NewFile.txt OtherFile.txt
pi@raspberrypi:~/NewFolder $ rm NewFile.txt
pi@raspberrypi:~/NewFolder $ ls
OtherFile.txt
pi@raspberrypi:~/NewFolder $ mv OtherFile.txt /home/pi/
pi@raspberrypi:~/NewFolder $ cd ..
pi@raspberrypi:~ $ ls
NewFolder OtherFile.txt
pi@raspberrypi:~ $
```

Рисунок 2.2. Создание каталога с помощью команд Linux.

## 3) **cd**: изменить каталог (папку).

cd.

cd..

cd / home

## 4) **mkdir**: создать каталог (папку).

mkdir TEST

## 5) **rmdir**: удалить каталог (папку).

rmdir TEST

## 6) **touch**: создать пустой файл

touch aaa

## 7) **cat**: показать содержимое файла

cat aaa

## 8) **cat >**: Сохраняет данные, полученные с помощью клавиатуры и сохраненные с помощью клавиш Ctrl-D.

cat> aaa.py

## 9) **cp**: скопировать файл

cp aaa bbb

## 10) **rm**: удалить файлы

rm aaa

**11) mv:** перемещать и переименовывать файлы

`mv aaa ссс:` переименовать `aaa` в `ссс`

**12) man:** призыв о помощи

- `man ls`
- `man man`
- `man date`
- `man cal`

**13) clear:** очистить экран.

**14) Пользователи и разрешения**

Классификация пользователей: Корень, Группа, Пользователь

Разрешения могут быть установлены для каждого пользователя.

- `drwxrwxrwx` означает:

`d`: каталог, `r`: чтение, `w`: запись, `x`: выполнение

Корень (`гwx`), группа (`гwx`), пользователь (`гwx`) доступны.

Формат: имя файла `chmod xxx`

**15) ln:** файл ссылки

`ln -s aaa.py ссс.py`

**16) Разделить файлы**

`rm (-f) ссс.py`

**17) find:** найти файл

найти `a*`: находит файлы, начинающиеся с `a`, в текущем каталоге.

**18) grep:** находит шаблон в заданном файле и отображает его

`grep "a" aaa.py`

**19) gzip:** Архивирование

`gzip ссс.py`

\* Создайте файл архива как `ссс.py.gz`.

**20) Gunzip:** Распаковать архив

`Gunzip ссс.py.gz`

Тип архива: `tar`, `gzip`, `bzip2`, `xz` и т. д.

**21) sudo:** выполнить команду от имени администратора

sudo ls

**22) apt-get:** команда для установки пакета с удаленного сервера

обновление sudo apt-get

обновление sudo apt-get

В основном используется с командой sudo для установки пакетов с удаленного сервера.

**23) nano editor:** используется для обозначения редактора в Linux.

nano aaa.py

Сохраните и выйдите, используя ctrl-x.

Запустите сохраненный файл

Измените номер доступа, чтобы повторить попытку.

## ЗАДАНИЯ

**1. Создайте каталог с именем linux и в нем создайте файлы с именами test1.py и test2.py.**

Тест1 должен содержать следующие слова: Я студент ГАТУ.

Тест2 должен содержать следующие слова: Узбекистан – лучшая страна.

**2. Создайте каталог с именем user.**

Создайте файл с именем Assessment.py внутри каталога.

Введите информацию в файл

Создайте ссылку под названием Evaluation1.

Проверьте, содержит ли файл определенный символ.

Проверьте, есть ли в каталоге файл, начинающийся с буквы p.

**3. Удалите все файлы и каталоги.**

Формат: имя файла chmod xxx

Создайте еще один файл(ы) с тем же содержимым.

**4. Создайте файл Python с помощью наноредактора для экрана ниже.**

Имя файла должно быть «Final Test».

Проверьте содержимое с помощью команды cat.

Создайте папку под названием «IoT». Создайте несколько файлов внутри этой папки. Заархивируйте папку.

## 5. Проверьте содержимое с помощью редактора nano.

Запустите файл Python.

Измените значение x и проверьте результат.

Создайте папку под названием «Raspberry Pi». Создайте несколько файлов внутри этой папки. Заархивируйте папку. Разархивируйте архив.

## 3- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ

### Тема: Работа с языком программирования Python на мини-компьютере Raspberry Pi.

**Цель работы:** изучить язык программирования Python на мини-компьютере Raspberry Pi и установить необходимые библиотеки для объектов Интернет.

#### Необходимое оборудование :

Используются необходимые компоненты и предметы, перечисленные ниже.

1. <https://www.raspberrypi.org/>
2. Мини-компьютер Raspberry Pi
3. Интернет
4. Клавиатура
5. Мышь

#### Теоретическая часть

Raspberry Pi — небольшой, доступный и мощный мини-компьютер, который становится популярным во всем мире. Его можно использовать для самых разных проектов: от простых проектов в области электроники до сложных проектов в области искусственного интеллекта и машинного обучения. Одним из самых популярных языков программирования, используемых с Raspberry Pi, является Python, поэтому в этом уроке мы рассмотрим, как использовать оба.

Python является неотъемлемой частью Raspberry Pi. Использование Python на Raspberry Pi имеет несколько преимуществ. Во-первых, Python очень легко изучить, особенно для новичков. Простой синтаксис и читабельность делают его отличным выбором для тех, кто только начинает программировать. Во-вторых, Python — очень универсальный язык, который можно использовать для широкого спектра приложений: от веб-разработки до анализа данных и машинного обучения. Наконец, Python имеет большое сообщество пользователей и разработчиков, а это значит, что вы можете легко найти помощь и ресурсы, когда они вам понадобятся.

## Преимущества Python на мини-компьютере Raspberry Pi

Одним из основных преимуществ использования Python на миникомпьютере Raspberry Pi является его способность взаимодействовать с аппаратным обеспечением устройства. Это открывает мир возможностей для создания проектов, взаимодействующих с датчиками, светодиодами, двигателями и многим другим. Вы можете использовать библиотеки Python, такие как RPi.GPIO, для управления контактами GPIO (ввод-вывод общего назначения) на Raspberry Pi. Еще одним преимуществом использования Python на мини-компьютере Raspberry Pi является его способность работать с внешними библиотеками и API. Например, вы можете использовать библиотеку Python Requests для отправки HTTP-запросов к API или использовать библиотеку Python OpenCV для выполнения задач по обработке изображений и компьютерному зрению.

Python — это язык высокого уровня для всех видов программирования. Установите вашу операционную систему. Если вы еще этого не сделали, сначала необходимо установить операционную систему на мини-компьютер Raspberry Pi.

При установке операционной системы на мини-компьютер Raspberry Pi вместе с ним устанавливается и программное обеспечение Thonny Python. Ниже находится окно программы Thonny Python.

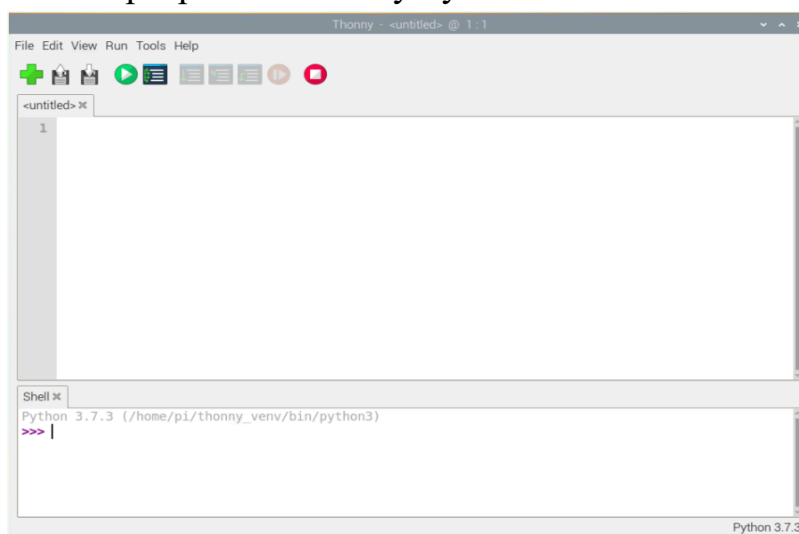


Рисунок 3.1. Это базовый интерфейс для написания кода, его сохранения и запуска, а также просмотра результатов при его запуске.



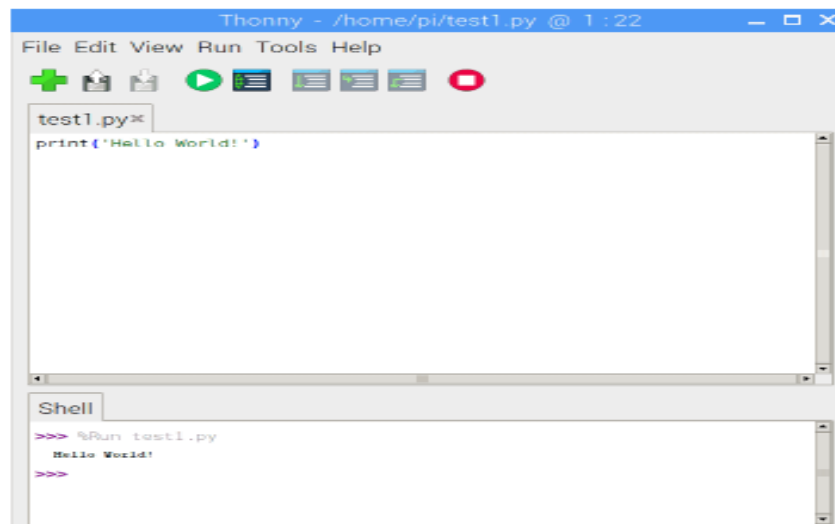


Рисунок 3.2. Статус публикации результата.

## Порядок прохождения практики:

### Импорт модуля в Python

Ключевое слово или команда `import` обеспечивает доступ к другим модулям языка программирования Python. Python имеет множество модулей с мощными функциями. Например, модуль `math` обеспечивает доступ к математическим функциям, используемым в вашем коде:

`Import math`

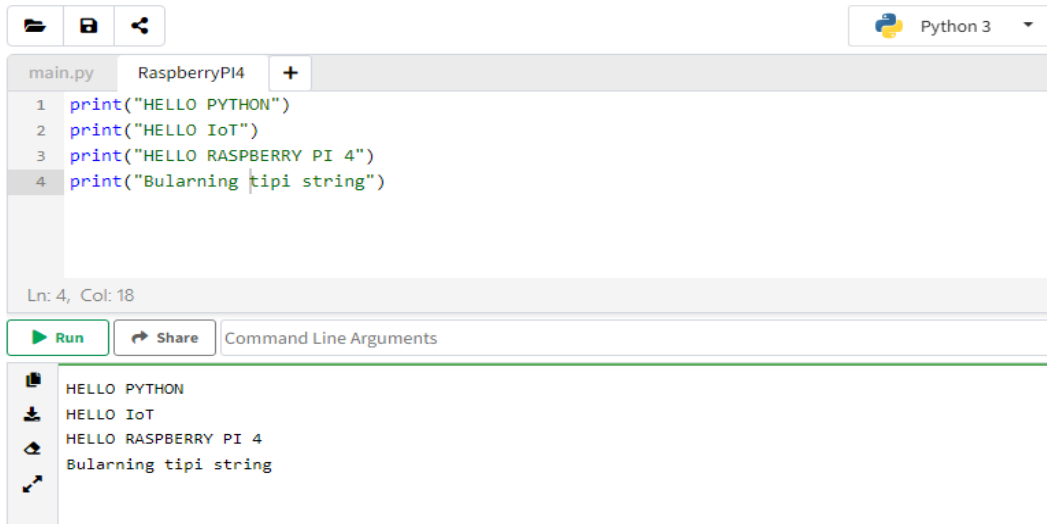
### Использование команды печати с типами данных

#### Числовые типы данных:

Мощной особенностью программирования является возможность манипулировать переменными. Переменные можно рассматривать как контейнеры, в которых хранится значение. Другими словами, имя, которое ссылается на значение. Например, здесь встречается целочисленный тип данных, где  $a = 27$ . Здесь переменная `a` объявлена с целочисленным значением 27.

`a = 27`

Оператор присваивания присваивает значение переменной `a` со значением целочисленного типа данных.



```
main.py  RaspberryPI4  +
1 print("HELLO PYTHON")
2 print("HELLO IoT")
3 print("HELLO RASPBERRY PI 4")
4 print("Bularning tipi string")

Ln: 4, Col: 18

Run Share Command Line Arguments

HELLO PYTHON
HELLO IoT
HELLO RASPBERRY PI 4
Bularning tipi string
```

Рисунок 3.3. Thonny — программа, написанная на Python.

### Логические типы данных:

Другим типом данных, используемым в языке программирования Python, является логическое значение, которое используется для представления истинного значения выражения. Эти значения могут быть истинными или ложными, см. следующий пример:



```
main.py  RaspberryPI4  +
1 a = 27
2 b = 27
3 c = 25
4 print(a == b > c)
5

Ln: 5, Col: 1

Run Share Command Line Arguments

True

** Process exited - Return Code: 0 **
```

Рисунок 3.4. Объявление переменной в Thonny Python.

Здесь переменная **a** сравнивается с переменной **b**; Поскольку оба имеют одинаковое значение, это приведет к значению **True**. Его полезность можно увидеть при проверке строк с использованием логических методов. То есть вы можете использовать логическую проверку для управления строками в Python.

## Список типов данных:



```
main.py  RaspberryPI4  +
1  raspberry = ['Python', 'Java', 'C++', 'Android', 'IoT', 'Super']
2  print(raspberry[0])
3  print(raspberry[4])
4  print(raspberry[5])

Ln: 4, Col: 20

Run  Share  Command Line Arguments

Python
IoT
Super
```

Рисунок 3.5 Работа со списками в Thonny Python

Списки представляют собой коллекции значений, а не отдельные значения, и полезны, когда вам нужно сохранить данные для последующих вычислений. В Python указание списка можно выполнить, присвоив несколько объектов имени переменной с помощью оператора =.

Список значений должен быть заключен в ['and'].

Чтобы напечатать значение (например) с индексом 0, мы используем следующую команду:

```
print(raspberry[0])
```

Чтобы напечатать значение по индексу 2, мы используем следующую команду:

```
print(raspberry[1])
```

## Условная логика: оператор If-Else.



```
main.py  RaspberryPi4  +  
1 a = 20  
2 if a >17:  
3     print("bu son 17 dan katta")  
4 else:  
5     print("bu son 17 dan kichik ")  
6  
7  
Ln: 1, Col: 7  
  
Run Share Command Line Arguments  
  
bu son 17 dan katta
```

Рисунок 3.6.Использование оператор If...else в Python Thonny

Условная логика необходима для написания сложных программ. Одну из самых простых форм можно найти в операторе **IF**. Прежде чем рассматривать условные выражения, полезно рассмотреть отступы более подробно.

### ЗАДАНИЯ

1. Изучите 60 операторских функций языка программирования Python.
2. Создание и использование 10 программ, связанных с объектами Интернета с помощью этих операторов.
3. Создание 10 классов на языке программирования Python.
4. Установите необходимые библиотеки с помощью команд Linux, используемых в операционной системе Raspberry Pi.
5. Установка 10 различных сенсорных модулей, используемых в Интернете вещей.

## 4- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ

### Тема: Использование и управление светодиодом (LED) с помощью мини-компьютера Raspberry Pi 4.

**Цель:** в этом практическом занятии мы научимся включать и выключать светодиод, используя контакты GPIO мини-компьютера Raspberry Pi для управления электронными компонентами.

#### Необходимое оборудование:

Для выполнения этого практического задания вам понадобится следующее оборудование.

1. Мини-компьютер Raspberry Pi
2. Его карта MicroSD
3. Базовый электронный комплект
4. Световые диоды
5. Резисторы и печатная плата

#### Теоретическая часть

Мини-компьютер Raspberry Pi имеет контакты GPIO (входной выход общего назначения), которые позволяют управлять рядом электронных компонентов.

В этом уроке мы также рассмотрим, как использовать эти контакты GPIO для управления светодиодом с помощью Raspberry Pi.



Рисунок 4.1. Мини-компьютер Raspberry Pi 4.

#### Понимание GPIO

Порты GPIO обычно представляют собой физические порты в виде квадратных металлических контактов, которые позволяют передавать электрический сигнал.

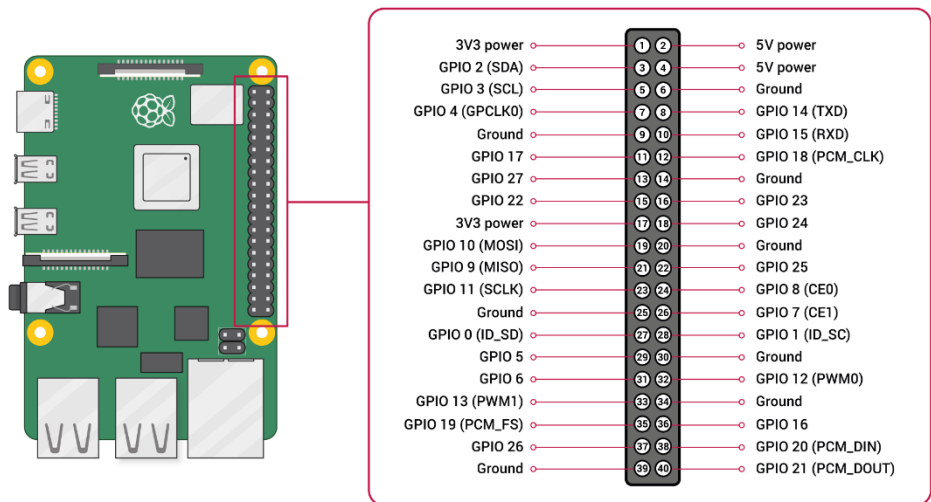


Рисунок 4.2. Контакты GPIO для подключения необходимого оборудования

Порт GPIO передает относительно двоичный сигнал. В случае Raspberry Pi порты GPIO работают при напряжении около 3,3 В и 20 мА.

Таким образом, порты GPIO — это простой способ связи или управления физическими элементами.

Последние модели Raspberry Pi имеют 40 разъемов GPIO, которые разделены на разные категории в зависимости от конкретного использования.

Питание: Источник бесперебойного питания

Земля: Земля

I2C: синхронный последовательный протокол

SPI: синхронный последовательный протокол

UART: асинхронный протокол + последовательная передача параллельной связи.

GPIO: порт общего назначения, который при необходимости можно изменить на I2C или SPI.

Порты GPIO пронумерованы от 1 до 40, начиная с верхнего левого угла. Это называется режимом нумерации «табличек». Существует еще один режим нумерации, основанный на адресе ЦП, который называется режимом «BCM».

Первое, что нужно знать, это то, что в светодиоде всегда должен использоваться резистор. Это связано с тем, что светодиоды имеют ограничение по напряжению. Проще говоря, если будет небольшое превышение напряжения и светодиод сразу сгорит, значит резистор - это защита.

Итак, нам нужно выбрать, какое сопротивление использовать. Существует физическая формула расчета номинала используемого для этого резистора.

$$R_{\text{мин}} = (U - U_{\text{светодиод}}) / I_{\text{макс.}}$$

Здесь

$R_{\text{мин}}$  : минимальное сопротивление при использовании, выраженное в Омах (Ом).

$U$ : напряжение источника питания, выраженное в вольтах (В).

$U_{\text{светодиод}}$  : пороговое напряжение светодиода, выраженное в вольтах (В).

$I_{\text{макс}}$  : Максимальная интенсивность светодиода, выраженная в амперах (А).

Это дает следующую формулу:

$$R_{\text{мин}} = (3,3 - 1,5) / 0,020$$

Следовательно, получаем минимальное сопротивление 90 Ом, преобразуем его в 100 Ом. Вместо этого мы выбираем резистор сопротивлением 270 Ом (или 330 Ом, нормированные значения), чтобы защитить нас от возможного высокого напряжения на GPIO.

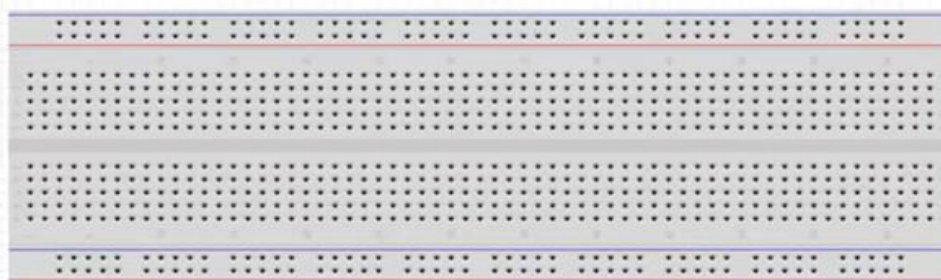
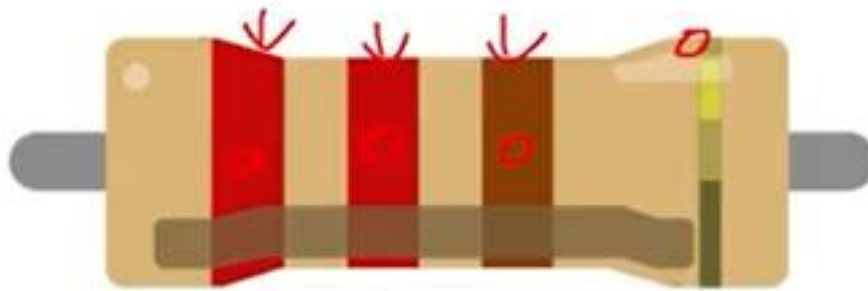


Рисунок 4.3 Схематический вид карты сборки

Эта плата — лучший вариант для начинающих, чтобы научиться создавать схемы простым способом, поскольку она является незаменимым продуктом для создания прототипов любого проекта. Печатная плата изготовлена из пластика и имеет небольшие отверстия, которые электронно соединены друг с другом по определенному шаблону. Печатная плата поставляется в нескольких размерах, и размер зависит от специфики проекта, который необходимо реализовать с использованием печатной платы. Если в проектах задействовано несколько устройств, рекомендуется использовать большой размер, а для проектов, требующих меньшего количества оборудования, рекомендуется использовать малый размер. Печатная плата состоит из металлических полосок, заключённых в пластиковый корпус.

Другими словами, на плате устройства просто соединяются с помощью перемычек.



#### 4.4 Внешний вид резистора

Как следует из названия, резисторы сопротивляются потоку электричества, и чем выше значение сопротивления, тем большее сопротивление он оказывает и тем меньше тока через него протекает. Мы используем это, чтобы контролировать, какой ток протекает через светодиод и, следовательно, насколько ярко он светится.

Информация о резисторах:

Единица сопротивления называется Ом, что обычно является аббревиатурой греческого слова Омега. Поскольку Ом – это малая величина сопротивления (он вообще не имеет сопротивления), то приведем номиналы резисторов в кОм (1000 Ом) и МОм (1000 000 Ом). Их называют килоом и мегаом.

В этом уроке мы будем использовать четыре разных значения сопротивления: 270 Ом, 470 Ом, 2,2 Ом и 10 Ом. Все эти резисторы выглядят одинаково, за исключением того, что у них есть линии разного цвета. Эти линии сообщают вам номинал резистора.

Цветовая маркировка резисторов работает следующим образом: для таких резисторов три цветные линии, затем золотая линия на одном конце.

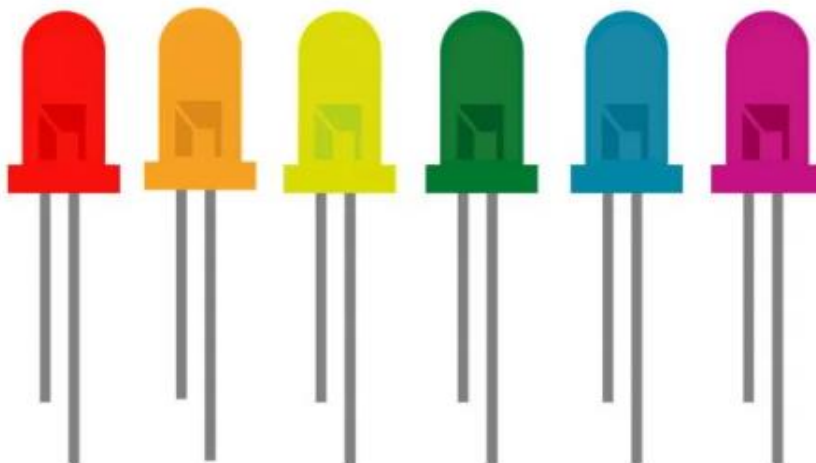


Рисунок 4.5. Световые диоды



Светодиод — это особый тип диода, который может излучать свет. Хотя все диоды излучают свет, в большинстве случаев этот свет недостаточно яркий. Светодиоды специально сконструированы так, чтобы излучаемый свет мог рассеиваться, поэтому мы можем видеть его, а не поглощать полупроводником. Светодиоды бывают нескольких цветов, включая белый. Существуют даже светодиоды, которые сочетают в себе три основных цвета: красный, зеленый и синий, создавая тысячи цветов.

Вот так выглядит светодиодное устройство. Светодиоды имеют короткие и длинные ножки. Короткая ножка называется «катодом», часто обозначается буквой «к», и ее следует подключать к отрицательному напряжению («-»). Более длинная ножка называется «анодом», часто обозначается буквой «а», и ее следует подключать к положительному («+») напряжению.

Прежде чем приступить к выполнению этого руководства, нам необходимо установить операционную систему Raspberry Pi на базе мини-компьютера Raspberry pi 4 на нашу SD-карту, и мы сможем напрямую управлять компьютером Raspberry Pi с помощью клавиатуры, мыши и экрана.

#### **Порядок прохождения практики:**

1. Поместите светодиод на печатную плату.
2. Также размещаем резистор на плате. Здесь один конец резистора соединен с короткими ножками диода.
3. Другой конец резистора подключен к выводу 6-GND GPIO.
4. От вывода 7 GPIO длинные ножки диода подключаются проводами.
5. Программа управления светодиодами написана на Thonny Python.
6. Программа запустится.

Примечание. Короткая ножка диода называется катодом, а длинная ножка — анодом.

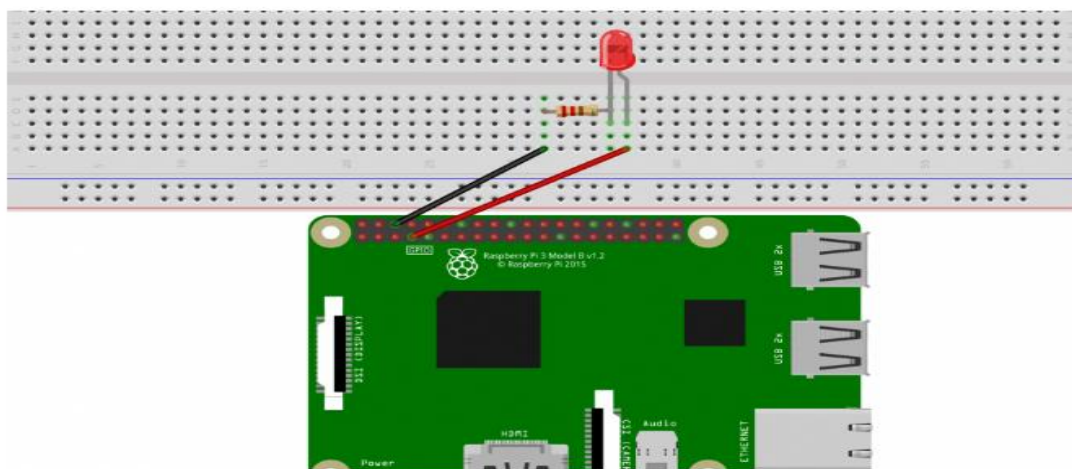


Рисунок 4.6. Схема подключения диода к контактам GPIO через плату

## Программная часть:

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(4, GPIO.OUT)
while True:
    GPIO.output(4, True)
    time.sleep(1)
    GPIO.output(4, False)
    time.sleep(1)
```

Программа для включения и выключения 2-х фотодиодов разного цвета.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(4, GPIO.OUT)
GPIO.setup(5, GPIO.OUT)
while True:
    GPIO.output(4, True)
    time.sleep(1)
    GPIO.output(5, False)
    time.sleep(1)
    GPIO.output(4, False)
    time.sleep(1)
    GPIO.output(5, True)
    time.sleep(1)
```

## ЗАДАНИЯ

1. Используйте 2 фотодиода разного цвета так, чтобы, когда фотодиод 1 включен, фотодиод 2 был выключен, а фотодиод 1 был выключен, когда фотодиод 2 включен. Пусть 2 фотодиода включаются и выключаются одновременно.
2. Включаются и выключаются фотодиоды с помощью специальной кнопки, т.е. загораются при нажатии кнопки, гаснут при ненажатии кнопки.
3. Измените уровень освещенности фотодиодов, перемещая потенциометр.
4. Измените количество горящих светодиодов, перемещая потенциометр.
5. Установите схему RGB-светодиода и создайте программу для его включения и выключения.

## 5- ПРАКТИЧЕСКОЕ ЗАНЯТИЕ

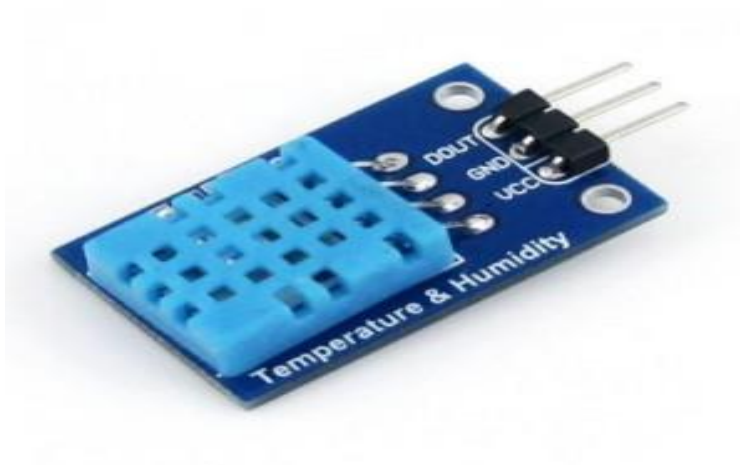
### Тема: Использование датчика DHT11 с помощью мини-компьютера Raspberry Pi 4.

**Цель:** в этом практическом упражнении научитесь непрерывно измерять и контролировать температуру и влажность в помещении в режиме реального времени с помощью датчика температуры и влажности DHT11.

#### Необходимое оборудование:

1. ЖК монитор
2. Датчик DHT11
3. Raspberry Pi
4. Плата платы
5. Проводящие провода

#### Теоретическая часть



**Рисунок 5.1. Датчик ДХТ11**

В этом датчике используются недорогие цифровые датчики для измерения температуры и влажности. Этот датчик и ряд микроконтроллеров доступны для измерения влажности и температуры в реальном времени, включая Arduino, Raspberry Pi и другие.

Доступны датчик и модуль влажности и температуры DHT11. Что отличает этот датчик от модуля, так это использование подтягивающих резисторов и диодов.

1. Датчики DHT можно использовать в качестве компенсатора с ультразвуковыми датчиками для более точного измерения расстояния.

2. Датчик DHT используется в различных приложениях, таких как измерение значений влажности и температуры в системах отопления, вентиляции и кондиционирования воздуха (HVAC).

3. Этот датчик можно использовать на складе, поскольку уровень влажности воздуха влияет на различные физические, химические и биологические процессы.

4. Метеостанции также используют эти датчики для прогнозирования погодных условий.

Цифровые датчики могут контролировать температуру и относительную влажность. Два измерения преобразуются в цифровой сигнал с помощью аналого-цифрового преобразователя. Датчики температуры с долгосрочной стабильностью и отличными характеристиками являются одними из самых популярных на рынке. Вот почему мы сосредоточимся на этом в этом уроке.

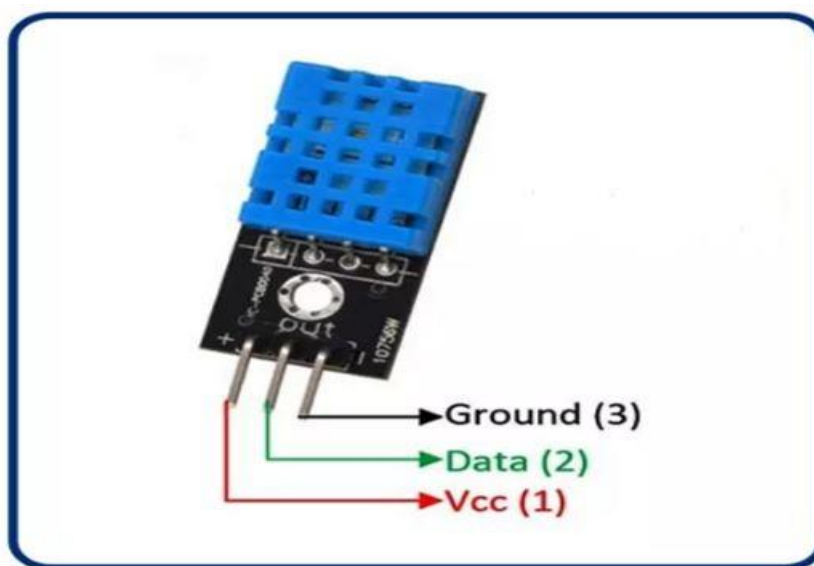


Рисунок 5.2 Распиновка датчика DHT11

**Vcc** — входная мощность блока питания от 3,5 до 5,5 Вольт.

**Данные** — отправка данных на последовательные выходные контакты

Земля – система, подключенная к датчику.

Затем его можно использовать для расчета влажности и температуры.

#### **Порядок прохождения практики:**

1. Подключите контакт заземления датчика DHT11 к контакту заземления контактов GPIO через черный провод.
2. Контакт напряжения (питание) датчика DHT11 подключается к контакту 5 В контакта GPIO с помощью красного провода.
3. Контакт данных датчика DHT11 подключен к контакту GPIO 4 контактов GPIO синим проводом.
4. Программа Thonny Python написана и запущена.

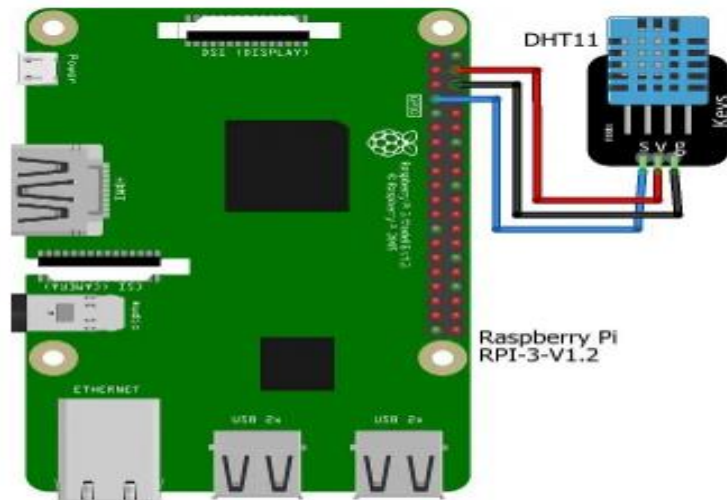


Рисунок 5.3. Схема датчика DHT11, подключенного к контактам GPIO

### Программная часть:

```

https://RandomNerdTutorials.com/raspberry-pi-dht11-dht22-python/
# На основе примера библиотеки Adafruit_CircuitPython_DHT.
import time
import board
import adafruit_dht

# Sensor data pin is connected to GPIO 4
# sensor = adafruit_dht.DHT22(board.D4)
# Uncomment for DHT11
sensor = adafruit_dht.DHT11(board.D4)

while True:
    try:
        # Print the values to the serial port
        temperature_c = sensor.temperature
        temperature_f = temperature_c * (9 / 5) + 32
        humidity = sensor.humidity
        print("Temp={0:0.1f}°C, Temp={1:0.1f}°F,
Humidity={2:0.1f}%".format(temperature_c, temperature_f,
humidity))
    except RuntimeError as error:
        # Ошибки случаются довольно часто, DHT трудно читать,
        продолжайте.
        print(error.args[0])
        time.sleep(2.0)
        continue
    except Exception as error:
        sensor.exit()
        raise error
    time.sleep(3.0)

```

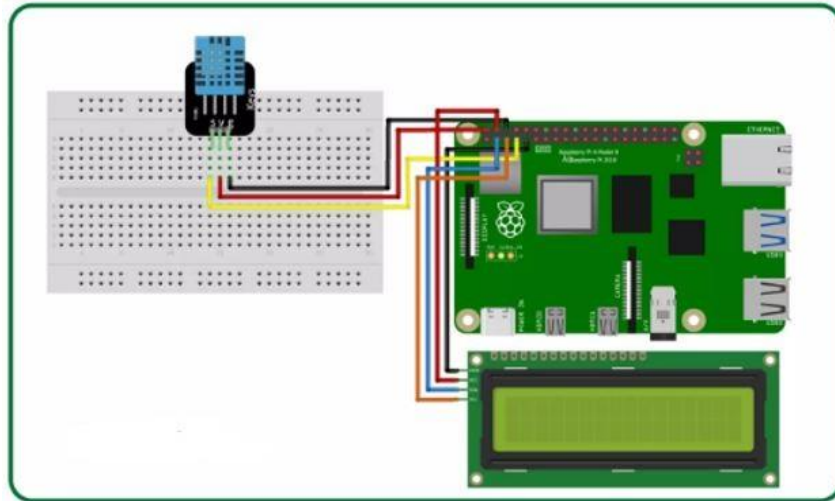


Рисунок 5.4. Принципиальная схема датчика DHT11, подключенного к дисплею.

### Порядок работы:

1. Сигнальный контакт датчика DHT11 подключен к GPIO 4 платы Raspberry Pi 4 через желтый провод.
2. Вывод напряжения датчика (питание) подключается к выводу 5 В платы Raspberry Pi с помощью красного провода.
3. Контакт GND датчика подключен к контакту GND Raspberry Pi черным проводом.

Чтобы подключить датчик к Raspberry Pi, выполните следующие действия. Теперь эту схему следует подключить к LCD-экрану 16x2.

4. GND подключается к контакту GND Raspberry Pi через черный провод на ЖК-дисплее.
5. Подключите контакт напряжения LCD-дисплея к контакту 5 В платы Raspberry Pi с помощью красного провода.
6. Контакт SDA LCD-дисплея подключен к контакту SDA Raspberry Pi синим кабелем.
7. Контакт SCL LCD-дисплея подключен к плате Raspberry Pi с помощью оранжевого провода.

### Программная часть:

Чтобы использовать в нашей программе параметры, специфичные для системы, мы должны сначала импортировать модуль sys. Существует также файл поддержки под названием I2C LCD Driver для платы Raspberry Pi. Модуль Adafruit DHT используется для подключения датчика DHT11 к Raspberry Pi.

```
import sys
import I2C_LCD_driver
import Adafruit_DHT
```

```

import time
mylcd = I2C_LCD_driver.lcd()
mylcd.lcd_clear()
mylcd.lcd_display_string('LearnElectronics' ,1)
mylcd.lcd_display_string('DHT11 with RPi' ,2,1)
try:
    while True:
        humidity, temperature = Adafruit-DHT.read_retry(11,4)
if (temperature != None and humidity != None):
    mylcd.lcd_clear()
    mylcd.lcd_display_string('Temp: {0:0.1f} C ',format(temperature),1,2)
    mylcd.lcd_display_string('Humidity: {0:0.1f} % ',format(humidity),2,1)
time.sleep(1)
except KeyboardInterrupt:
mylcd.lcd_clear()
mylcd.lcd_display_string('Thanks' ,1,5)
mylcd.lcd_display_string('for watching' ,2,2)
time.sleep(5)
mylcd.lcd_clear()

```

Следующий текст появляется на экране перед отображением на экране значений температуры и влажности.

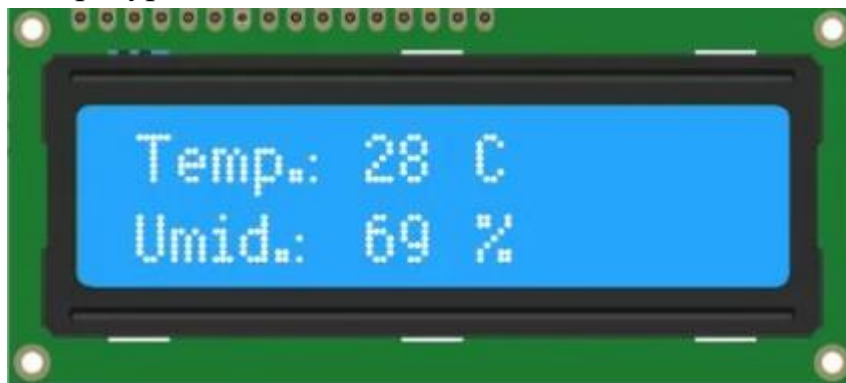


Рисунок 5.5. Статус при отображении результата

### ЗАДАНИЯ

1. В этом практическом упражнении измерьте температуру и влажность с помощью печатной платы.
2. Измерение температуры и влажности с помощью датчиков DHT11/DHT22 и отображение измеренных данных.
3. В этом практическом упражнении мы используем диоды, чтобы красный диод загорался, когда температура и влажность выше нормы.
4. Отправка по электронной почте информации об измеренной температуре и влажности.

5. Отправьте измеренные значения температуры и влажности на другое устройство через Bluetooth.

## 6- ПРАКТИЧЕСКАЯ РАБОТА.

### Тема: Использование PIR датчика движения с помощью мини-компьютера Raspberry Pi.

**Цель работы:** На этом уроке мы будем использовать PIR-датчик движения для обнаружения движения и отображения этого движения с помощью светодиодов.

#### Необходимое оборудование:

1. Световые диоды
2. Резисторы 220 Ом
3. Соединительные провода
4. PIR -датчик движения
5. Карта печатной платы
6. Мини-компьютер Raspberry Pi 4

#### Теоретическая часть

В этом уроке мы научимся обнаруживать простое движение с помощью PIR (пассивного инфракрасного) датчика на Raspberry Pi. Датчик PIR используется для обнаружения движения людей, животных или других объектов. Тела, температура которых выше абсолютного нуля, излучают тепловую энергию в виде излучения. Обычно это излучение невидимо для человеческого глаза, поскольку оно излучается в инфракрасном диапазоне, но его можно обнаружить с помощью электронных устройств, таких как PIR-датчики, предназначенных для этой цели.

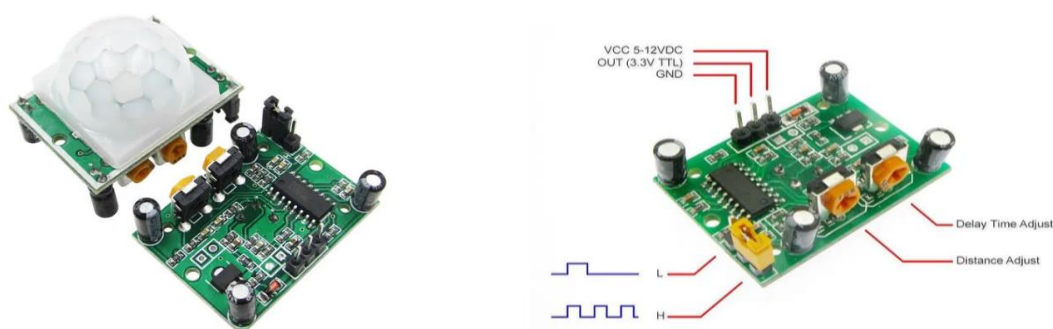


Рисунок 6.1. PIR -датчик движения

Датчик движения PIR: PIR означает пассивный инфракрасный датчик, он состоит из инфракрасного детектора, линзы Френеля и поддерживающей схемы обнаружения. Объектив может обнаруживать все, что излучает инфракрасное тепло, например, людей и животных. Тепло улавливается датчиком движения.



Датчик движения PIR имеет два потенциометра, которые используются для регулировки времени задержки и чувствительности. Настройка времени задержки используется для указания того, как долго выходной сигнал будет высоким после обнаружения движения; это может быть от 5 секунд до 5 минут

Регулировка чувствительности служит для установки дальности обнаружения, которая может составлять от 3 до 7 метров.

Датчик движения PIR имеет три контакта для подключения питания (VCC), сигнала (выход) и заземления.

PIR-датчики движения маленькие, дешевые, с низким энергопотреблением, просты в использовании и не изнашиваются. По этой причине они обычно встречаются в оборудовании и гаджетах, используемых дома или на работе.

### Порядок выполнения практической работы:

1. На плате установить световой диод и резистор.
2. Контакты GND печатной платы и контакты GND GPIO соединить вместе черным проводом.
3. Короткая ножка светодиода и контакт GND на плате соединить с помощью перемычки.
4. Длинную ножку светодиода подключить к одной стороне резистора. Другую сторону резистора подключить к контакту GPIO 2 через перемычку.
5. Контакт GND PIR-датчика и контакт GND на плате соединить черным проводом.
6. Вывод VCC датчика PIR подключить к выводу 5V GPIO.
7. Вывод OUT датчика PIR подключить к выводу 17 GPIO.
8. Код написать на языке Thonny Python и запустить.

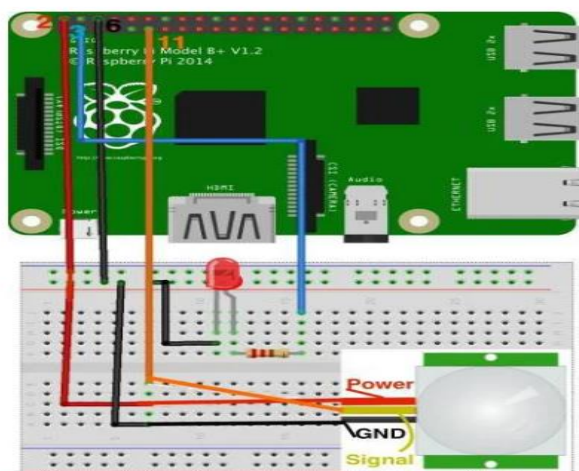


Рисунок 6.2. Принципиальная схема PIR-датчика движения, подключенного к контактам GPIO

## Программная часть:

```
import RPi.GPIO as GPIO
import time
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.IN)
GPIO.setup(3, GPIO.OUT)
while True:
    i=GPIO.input(11)
    if i==0:
        print("Harakat aniqlanmadi",i)
        GPIO.output(3, 0)
        time.sleep(0.1)
    elif i==1:
        print("Harakat aniqlandi",i)
        GPIO.output(3,1)
        time.sleep(0.1)
```

## ЗАДАНИЯ

1. Это упражнение выполнить без использования светодиодов, то есть с выводом результата на последовательный монитор в виде текста.
2. Отобразить результат в практической работе
3. В этой практической работе использовать камеру, то есть заставить камеру делать снимок при обнаружении движения.
4. В практической работе использовать несколько светодиодов. Создать и использовать программу с помощью функций.
5. В практической работе, когда PIR-датчик обнаруживает движение, он должен рассчитывать и отображать расстояние до места движения.

## 7- ПРАКТИЧЕСКАЯ РАБОТА.

### Тема: Сборка и управление светофором с помощью миникомпьютера Raspberry Pi 4.

**Цель работы:** Построить схему светофора и управлять ею с помощью контроллера светофора.

#### Необходимое оборудование:

1. Мини компьютер Raspberry Pi 4.
2. Карта печатной платы.
3. Соединительные провода.
4. Резисторы.
5. 3 светодиода.
6. Цифровой дисплей.

## Теоретическая часть

Светодиод — это особый тип диода, который может излучать свет. Хотя все диоды излучают свет, в большинстве случаев этот свет недостаточно ярк, чтобы мы его не могли видеть. Светоизлучающие диоды специально разработаны для рассеивания излучаемого света.

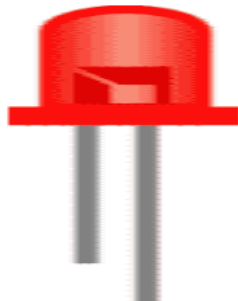


Рисунок 7.1. Светодиодный дисплей

Светодиоды бывают нескольких цветов. Существуют даже светодиоды, которые сочетают в себе три основных цвета: красный, зеленый и синий, создавая тысячи цветов.

Светодиоды имеют короткую и длинную «ножку». Короткая ножка называется «катодом», часто обозначается буквой «к», и ее следует подключать к отрицательному («-») напряжению. Более длинная ножка называется «анодом», часто обозначается буквой «а», и ее следует подключать к положительному («+») напряжению.

В символическом смысле, то есть на схемах, изображающих электронные схемы, светодиоды изображаются следующим образом:

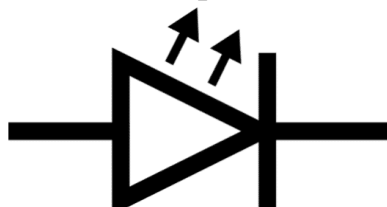


Рисунок 7.2. Изображение светодиода в электронных схемах.

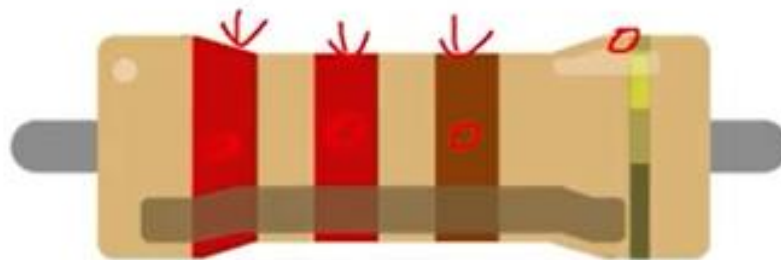


Рисунок 7.3. Электрический резистор

Резистор (также известный как электрический резистор) определяется как пассивный электрический элемент с двумя выводами, который обеспечивает электрическое сопротивление потоку тока. Сопротивление — это мера сопротивления протеканию тока в резисторе. Чем больше сопротивление резистора, тем больше сопротивление протеканию тока. Существуют различные типы резисторов, таких как термистор. Основная функция резистора в электрической и электронной цепи — «сопротивляться» потоку электронов, то есть электрическому току. Именно поэтому его называют «резистором».

### Порядок выполнения практической работы:

1. На плате разместить красные, желтые и зеленые светодиоды.
2. Разместить резисторы на плате.
3. Печатную плату подключить к портам GPIO Raspberry Pi с помощью кабелей.
4. Написать и запустить код на языке Thonny Python

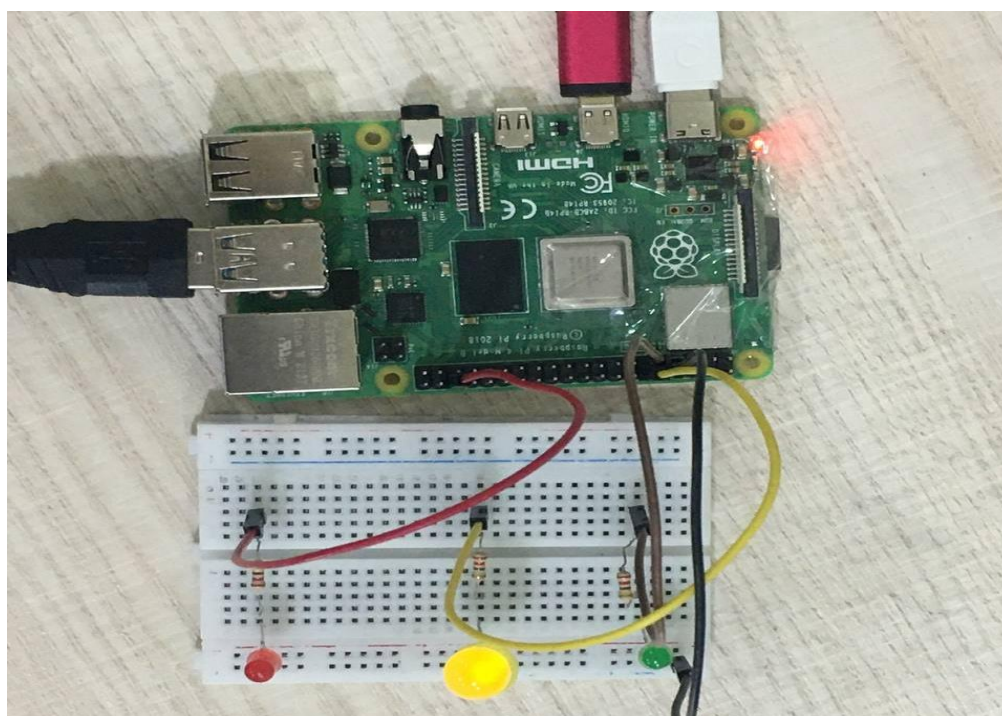


Рисунок 7.4. Схема светофора построена с использованием необходимого оборудования

## Программная часть:

```
From gpiozero import LED
import time
red = LED(16)
yellow = LED(18)
green = LED(17)
while True:
    red.on()
    time.sleep(5)
    red.off()

    yellow.on()
    time.sleep(2)
    yellow.off()

    green.on()
    time.sleep(5)
    green.off()

    yellow.on()
    time.sleep(2)
    yellow.off()
```

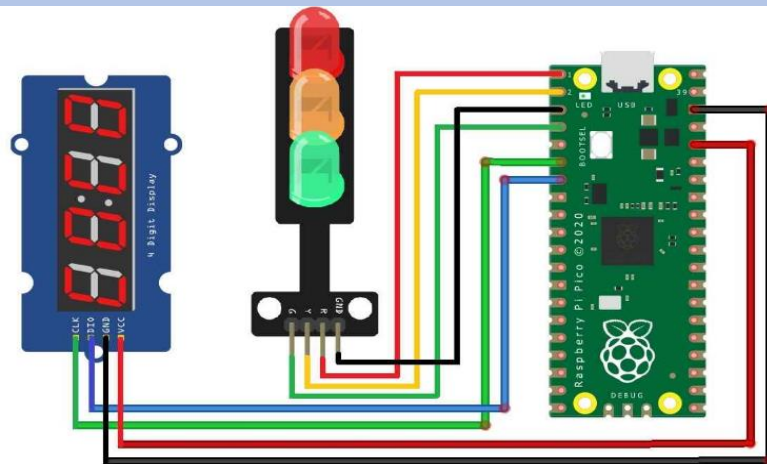


Рисунок 7.5. Схема светофора

Здесь 4 контакта. Контакты R, G, Y и GND модуля управления светофором подключены к Raspberry Pi GP0, GP1, GP2 и GND.

Аналогично, цифровой дисплей имеет 4 контакта: VCC, GND, CLK и DIO. Выводы VCC и GND подключены к 5 В и GND Raspberry Pi. Контакт CLK и DIO подключен к Raspberry Pi GP4 и GP5.

## ЗАДАНИЯ

1. Задание выполнить с помощью онлайн-тренажера Tinkercad.com.
2. Сборку светофора с использованием мини-компьютера Raspberry Pi и необходимого оборудования, осуществить после прохождения программы онлайн-симулятора.
3. Осуществить управление встроенным светофором с помощью цифрового дисплея.
4. Осуществить автоматическое управление светофорами в зависимости от загруженности дорог.
5. Добавить специальную кнопку в светофор, сделанный на мини-компьютере Raspberry Pi и управлять им с помощью этой кнопки.

## 8- ПРАКТИЧЕСКАЯ РАБОТА.

### **Тема: Использование ультразвукового датчика HC-SR04 на мини-компьютере Raspberry Pi.**

**Цель работы:** Научиться измерять расстояние между вашим телом и объектом, отправляя и получая ультразвуковые звуковые волны с помощью ультразвукового датчика.

#### **Необходимое оборудование:**

Для выполнения этого практического задания вам понадобится следующее оборудование:

1. Мини-компьютер Raspberry Pi
2. Ультразвуковой датчик
3. Соединительные провода
4. Карта печатной платы
5. резистор 1 кОм
6. резистор 2 кОм

## Теоретическая часть

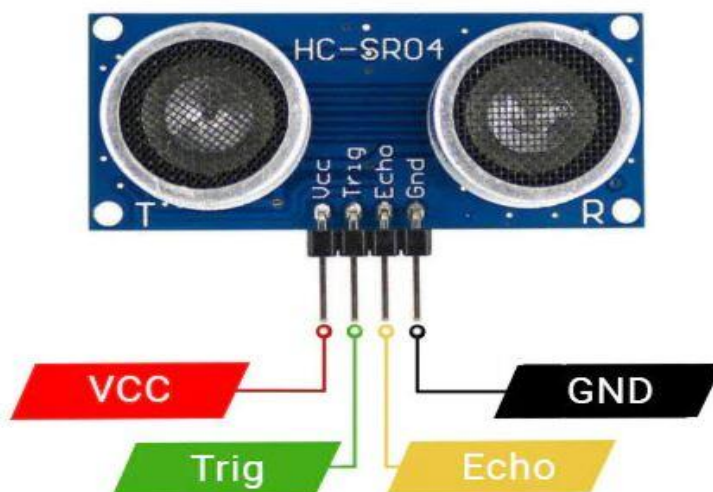


Рисунок 8.1. Ультразвуковой датчик HC-SR04.

Ультразвуковой датчик — это электронное устройство, которое может измерять расстояние до целевого объекта, посылая ультразвуковые звуковые волны и преобразуя отраженный звук в электрический сигнал. По сути, такой датчик использует преобразователь для отправки и получения ультразвуковых импульсов, который, в свою очередь, отправляет информацию о близости объекта. Следует отметить, что эти ультразвуковые волны распространяются быстрее, чем слышимый звук; то есть эти датчики излучают звуковые волны на частотах выше диапазона человеческого слуха.

Ультразвуковые датчики состоят из двух основных компонентов: один представляет собой преобразователь, излучающий звук с помощью пьезоэлектрических кристаллов, а другой — звуковой приемник.

### **Ультразвуковые датчики: преимущества и недостатки**

Ультразвуковые датчики, как и любая другая технология, наиболее широко используются в определенных условиях или приложениях. Среди их многочисленных преимуществ можно отметить:

Ультразвуковые датчики не влияют на цвет обнаруживаемого объекта, даже если это вода или стекло. Большинство ультразвуковых датчиков могут обнаруживать объекты на расстоянии от нескольких сантиметров до пяти метров. Около 20 метров можно измерить с помощью специально разработанных единиц. Эта передовая технология существует уже давно и очень надежна. Ультразвуковые датчики способны обеспечить измерения с высоким уровнем точности. Они могут выполнить множество измерений за короткий промежуток времени, что приводит к высокой скорости обновления.

Поскольку для них не требуются редкие материалы, они обычно относительно недороги.

Ультразвуковые датчики, использующие модули с кодированными чипами, могут противостоять электрическим шумам, а также большинству акустических шумов, поэтому их можно использовать в шумных помещениях. Прежде чем принимать окончательное решение о том, какой датчик использовать, важно учитывать ограничения ультразвуковых датчиков. Температура и влажность влияют на скорость звука, что может повлиять на точность измерений.

Нет возможности определить, где находится ультразвуковой датчик относительно зоны обнаружения объекта, хотя зона обнаружения трехмерная. Он также не может различить, круглый ли объект или круглый по цвету. Возможно, не стоит использовать ультразвуковые датчики в проектах, которые должны быть небольшими и скрытыми, например, внутри автомобиля или промышленного оборудования.

Если вы хотите проявить творческий подход к ультразвуковым датчикам, вы можете использовать одностороннюю функциональность передатчика и приемника. Животные могут слышать ультразвуковые импульсы, которые не слышны человеческому уху. Ультразвуковые передатчики и приемники имеют множество применений, в том числе отпугивание животных, таких как птицы, и обнаружение шума.

### Порядок выполнения практической работы:

1. На плате установить резистор сопротивлением 1 кОм.
2. К плате подключить резистор сопротивлением 2 кОм.
3. Резисторы сопротивлением 1 кОм и 2 кОм подключить к выводу 18 GPIO, как показано ниже.
4. Выводы GPIO, ультразвуковой датчик и монтажная плата соединить перемычками следующим образом.
5. Код написать на языке Thonny Python и запустить.

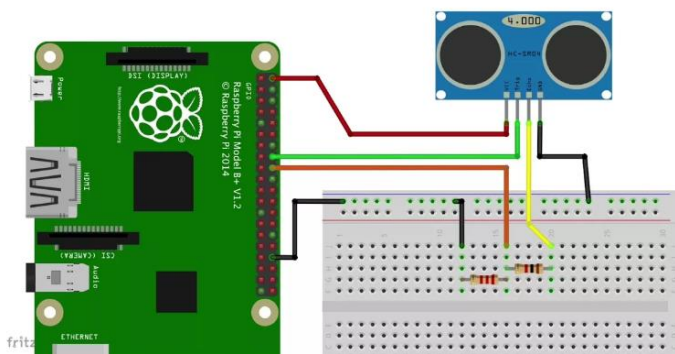


Рисунок 8.2. Принципиальная схема подключения ультразвукового датчика к компьютеру Raspberry Pi.



## Программная часть:

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO_TRIG = 23
GPIO_ECHO = 24
GPIO.setup(GPIO_TRIG, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)
GPIO.output(GPIO_TRIG, GPIO.LOW)
Time.sleep(2)
GPIO.output(GPIO_TRIG, GPIO.HIGH)
Time.sleep(0.00001)
GPIO.output(GPIO_TRIG, GPIO.LOW)
while GPIO.input(GPIO_ECHO) == 0:
    start_time = time.time()
while GPIO.input(GPIO_ECHO) == 1:
    Bounce_back_time = time.time()
    pulse_duration = Bounce_back_time - start_time
    distance = round(pulse_duration * 17150,2)
print(f'Distance: {distance} cm')
GPIO.cleanup()
```

## ЗАДАНИЯ

1. Полностью изучить конструкцию ультразвукового датчика и областей его применения.
2. Осуществить практическое задание без использования карты печатной платы.
3. Использовать дисплей на практических занятиях, т.е. отобразить результат измеренного расстояния на дисплее.
4. В практическом упражнении при расчете расстояния, когда светодиоды достигают небольшого расстояния, они должны выдавать предупреждение, а когда они удаляются, светодиоды должны гаснуть.
5. Выполните это практическое упражнение, используя микроконтроллер Arduino Uno.

## 9- ПРАКТИЧЕСКАЯ РАБОТА.

### Тема: Определение количества газа в помещении с помощью датчика газа MQ-5 с использованием микроконтроллера Arduino Uno.

**Цель работы:** Научиться определять количество газа в помещении с помощью датчика газа MQ-5 и создадим систему оповещения при превышении количества газа над нормой.

#### Необходимое оборудование:

Для выполнения этого практического задания вам понадобится следующее оборудование:

1. Микроконтроллер Arduino Uno
2. Датчик газа MQ-5
3. Соединительные провода
4. Карта печатной платы
5. Резистор 1 кОм
6. Программное обеспечение Arduino IDE

#### Теоретическая часть

Датчик MQ-5 — недорогой, простой в использовании датчик с широким диапазоном обнаружения, поскольку он обнаруживает горючие газы в большинстве проектов Arduino. Сегодня природный газ доступен повсюду. Он обеспечивает энергию для различных задач, как в домашнем хозяйстве, так и в промышленности. Датчик горючих газов MQ-5 может обнаруживать газы LPG, H<sub>2</sub>, LPG, CH<sub>4</sub> и CO. Мы в этом практическом упражнении узнаем о датчике MQ-5 и подключим его к микроконтроллеру Arduino Uno для обнаружения газов.

#### Характеристики датчика газа MQ-5:

1. Широкий диапазон обнаружения.
2. Быстрый отклик и высокая чувствительность.
3. Повышенная чувствительность к сжиженному нефтяному газу, природному газу.
4. Небольшая чувствительность к алкоголю, дыму.
5. Стабильная и долгосрочная работа.
6. Он имеет простую структуру.

Обозначения компонентов датчика MQ-5 приведены ниже.

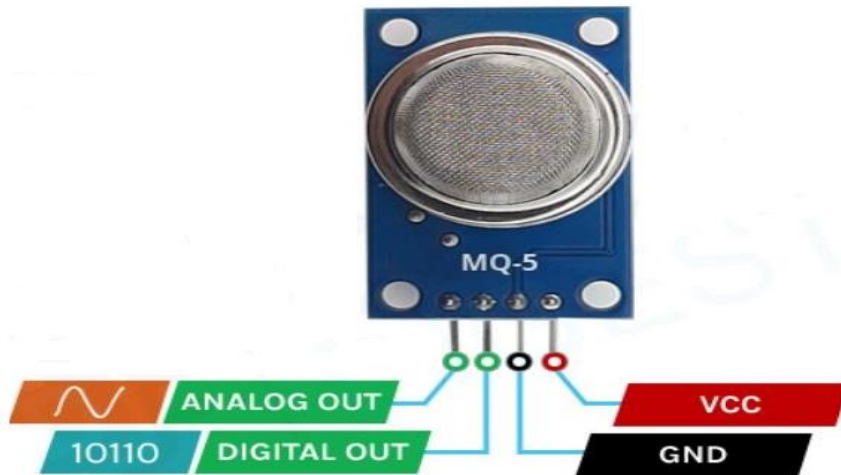


Рисунок 9.1. Датчик газа MQ-5.

Здесь:

**VCC** -Источник питания

**GND** – Комплект питания

**DOUT** - Цифровой вывод данных датчика

**AOUT** – Вывод данных датчика в аналоговой форме



Рисунок 9.2. Устройство газового сенсора MQ-5.

Модуль датчика MQ-5 имеет четыре контакта, два из которых предназначены для VCC и GND. Как и другие основные сенсорные модули, два других могут одновременно выдавать аналоговые и цифровые данные. Поскольку диапазон рабочего напряжения модуля составляет 5В с точностью 0,1%, для питания схемы мы будем использовать вывод Arduino 5В. Как видно на картинке, модуль имеет два встроенных светодиода. Светодиод Power загорается, когда плата включена, а светодиод Dout загорается, когда достигается значение срабатывания потенциометра. Аналоговый сигнал от датчика газа преобразуется в цифровой сигнал компаратором операционного усилителя на этой плате.

### Порядок прохождения практики:

1. Вывести 5V микроконтроллер Arduino Uno и вывести VCC газового датчика MQ-5 соединенных перемычками.
2. Вывести GND микроконтроллер Arduino Uno соединенный с выводом GND датчика газа MQ-5 с помощью перемычек.
3. Осуществить аналоговый вывод A0 микроконтроллера Arduino Uno подключенного к выводу A0 датчика газа MQ-5 через перемычки.
4. Осуществить контакт 2 микроконтроллера Arduino Uno и контакт D0 датчика газа MQ-5 соединенных вместе.
5. Написать и запустить программу.

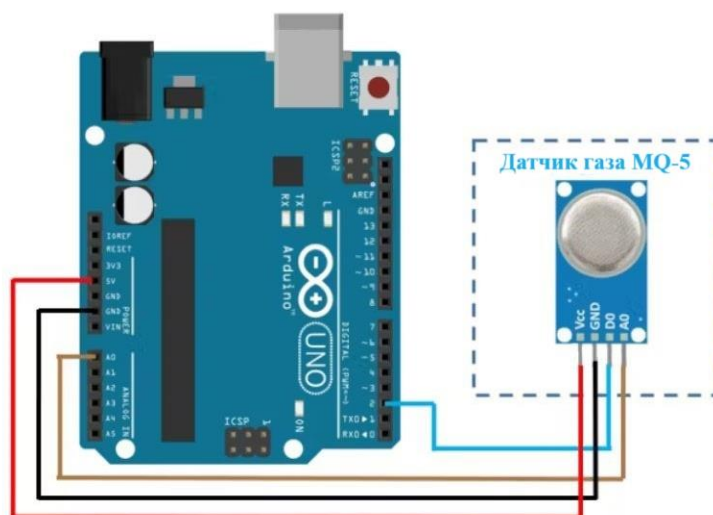


Рисунок 9.3. Схема подключения датчика газа MQ-5 к микроконтроллеру Arduino Uno.

## Программная часть:

```
float sensor=A0;
float gas_value;
void setup(){
  pinMode(sensor,INPUT);
  Serial.begin(9600);
}
void loop(){
  gas_value=analogRead(sensor);
  Serial.println(gas_value);
  delay(5000);
}
```

Когда мы запускаем программу, результат следующий.

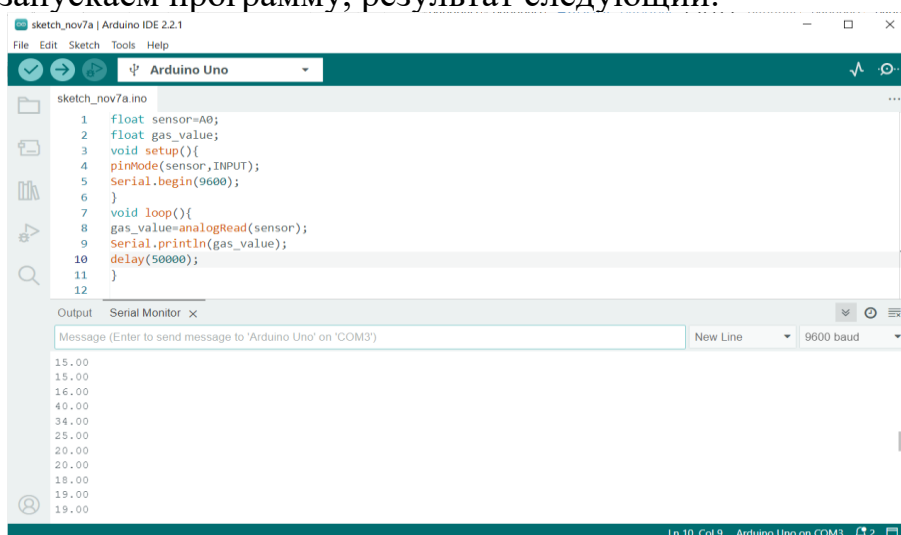


Рисунок 9.4. Отображение результата в последовательном мониторе

## ЗАДАНИЯ

1. Узнайте, как построить диаграмму результата работы этой программы на последовательном плоттере.
2. Осуществите практическое упражнение по сборке и работе с печатной платой и одновременному выводу результатов на последовательный монитор и последовательный плоттер.
3. Используйте дисплей в практическом занятии, то есть выведите на дисплей измеренное количество газа.
4. В этом практическом упражнении заставьте светодиоды загораться, когда количество газа в помещении превышает норму.
5. Выполните это практическое задание, используя мини-компьютер Raspberry Pi.

## 10- ПРАКТИЧЕСКАЯ РАБОТА.

### Тема: Определение частоты сердечных сокращений с помощью датчика пульсоксиметра Max30102 с использованием микроконтроллера Arduino Uno.

**Цель:** датчик пульсоксиметра Max30102 с помощью определения частоты сердечных сокращений человека, измерения уровня насыщения крови кислородом и температуры.

#### Необходимое оборудование:

Для выполнения этого практического задания вам понадобится следующее оборудование:

1. микроконтроллер Arduino Uno
2. Датчик пульсоксиметра Max30102
3. Соединительные провода
4. Карта печатной платы
5. Программное обеспечение Arduino IDE

#### Теоретическая часть

В этом уроке мы будем использовать датчик пульсоксиметра MAX30102 с Arduino для измерения частоты сердечных сокращений, насыщения крови кислородом и температуры. MAX30102 — это датчик, который может измерять частоту сердечных сокращений и уровень кислорода в крови. Мы также используем библиотеку SparkFun MAX3010x. Мы используем OLED-дисплей для просмотра значений SpO2 и BPM.

MAX30102 — это датчик пульсоксиметрии и монитора сердечного ритма. Он сочетает в себе два светодиода, фотодетектор, оптимизированную оптику и малошумящую аналоговую обработку сигналов для обнаружения сигналов пульсоксиметрии и частоты сердечных сокращений. Вы можете использовать этот датчик с любым микроконтроллером, например, Arduino, Raspberry Pi, ESP8266 или ESP32, и легко измерять параметры здоровья пациента.



Рисунок 10.1. Датчик пульсоксиметра MAX30102.

MAX30102 — это небольшой датчик, который можно использовать для измерения частоты сердечных сокращений и уровня кислорода в крови. Он работает, пропуская свет через кожу (обычно кончик пальца), а затем измеряя,

какая часть этого света отражается обратно. Этот отражающий свет передает датчику информацию о частоте сердечных сокращений и уровне кислорода, что может быть важно для мониторинга вашего здоровья.

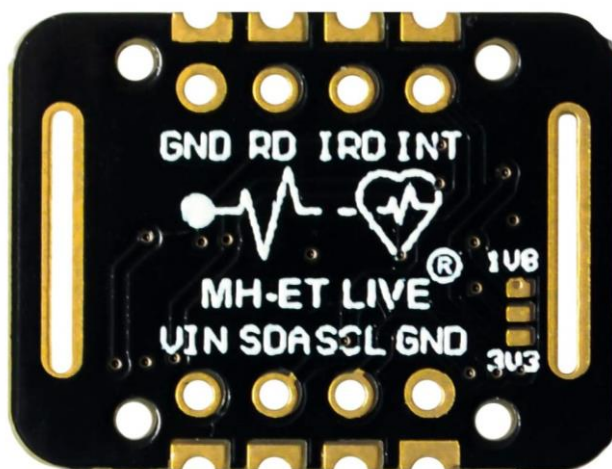


Рисунок 10.2. Названия контактов датчика пульсоксиметра MAX30102

В MAX30102 используются два светодиода: один для красного света (660 нм) и один для инфракрасного света (880 нм), а также фотодетектор. Он измеряет количество света, поглощаемого насыщенным кислородом и дезоксигенированным гемоглобином в крови. Сравнивая поглощение на этих двух длинах волн, он рассчитывает насыщение крови кислородом (SpO<sub>2</sub>) и частоту сердечных сокращений. Его часто используют в фитнес-трекерах, медицинских устройствах и других гаджетах для отслеживания вашего самочувствия.

MAX30102 работает, направляя красный и инфракрасный свет на кожу, обычно на кончик пальца или мочку уха. Затем он измеряет количество отраженного света с помощью фотодетектора.

**Принцип работы:**

Сигнал оптического отражения (PPG)

**Основные размеры:**

Уровень насыщения крови кислородом (SpO<sub>2</sub>)

Частота сердечных сокращений (частота пульса)

**Условия эксплуатации:**

Напряжение источника питания: 3,3 В – 5 В

Рабочая температура: от -40°C до +85°C.

**Интерфейс связи:**

Протокол связи I2C (межинтегральная схема)

Дополнительные возможности:

Низкое энергопотребление

Отмена встроенного внешнего освещения

Возможность высокой частоты дискретизации

### Как работает пульсоксиметр?

Пульсоксиметр работает, пропуская свет (обычно красный и инфракрасный) через кожу, обычно на кончик пальца. Количество поглощенного света указывает на уровень насыщения крови кислородом. Небольшое устройство в форме зажима надевается на палец, ухо или ногу во время измерения пульсоксиметрии.

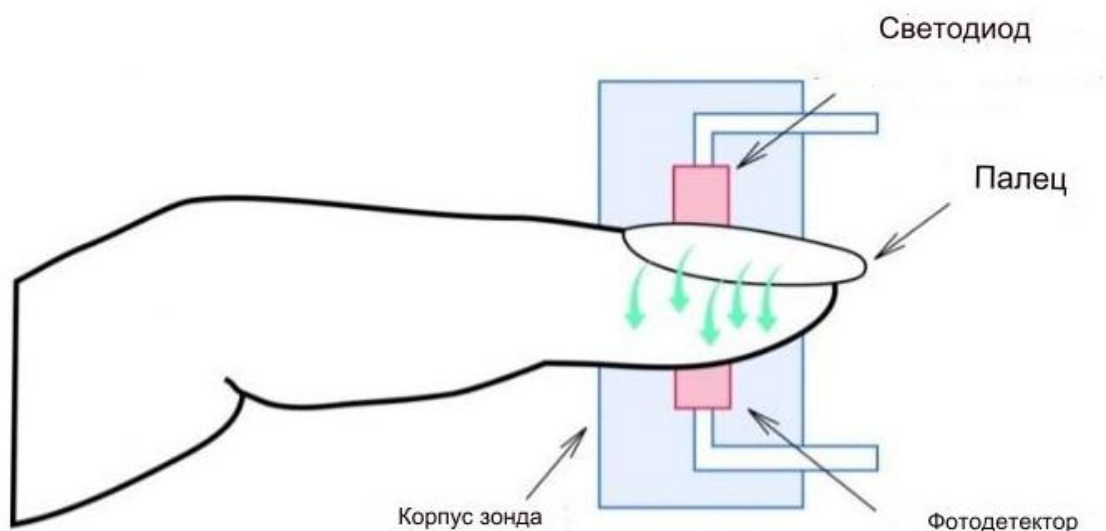


Рисунок 10.3. Процесс работы датчика

### Измерение сердечного ритма

Оксигенированный гемоглобин ( $\text{HbO}_2$ ) артериальной крови имеет коэффициент поглощения ИК. Чем краснее кровь (чем выше гемоглобин), тем больше поглощается ИК-лучей. С каждым ударом сердца, когда кровь пересчитывается через палец, количество отраженного света меняется, создавая изменяющуюся форму волны на выходе фотодетектора.

Давайте посмотрим на взаимодействие датчика сердечного ритма и пульсоксиметра MAX30102 PPG с Arduino.

Подключаем датчик MAX30102 к плате Arduino. Схема подключения Arduino к MAX30102 очень проста.

### Порядок выполнения практической работы:

1. На плате установить датчик пульсоксиметра Max30102.
2. Вывод VIN датчика Max30102 подключить к выводу Arduino Uno 5V.
3. Вывод GND Arduino Uno и вывод GND датчика Max30102 соединить вместе.
4. Вывод SCL датчика Max30102 подключить к выводу A5 Arduino Uno.



5. Вывод SDA датчика Max30102 подключить к выводу A4 Arduino Uno.
6. Скомпилировать и запустить программу.

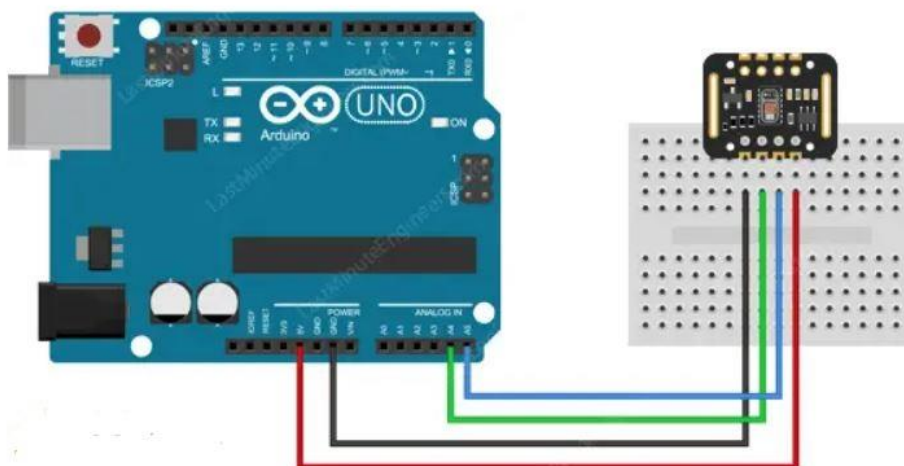


Рисунок 10.4. Схема подключения датчика Max30102 к микроконтроллеру Arduino Uno

### Программная часть:

```
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"
MAX30105 particleSensor;
const byte RATE_SIZE = 4;
byte rates[RATE_SIZE];
byte rateSpot = 0;
long lastBeat = 0;
float beatsPerMinute;
int beatAvg;
void setup()
{
  Serial.begin(115200);
  Serial.println("Initializing...");

  if (!particleSensor.begin(Wire, I2C_SPEED_FAST))
  {
    Serial.println("MAX30105 was not found. Please check wiring/power. ");
    while (1);
  }
  Serial.println("Place your index finger on the sensor with steady pressure.");
```

```

particleSensor.setup();
particleSensor.setPulseAmplitudeRed(0x0A);
particleSensor.setPulseAmplitudeGreen(0);
}
void loop()
{
long irValue = particleSensor.getIR();
if (checkForBeat(irValue) == true)
{
long delta = millis() - lastBeat;
lastBeat = millis();
beatsPerMinute = 60 / (delta / 1000.0);
if (beatsPerMinute < 255 && beatsPerMinute > 20)
{
rates[rateSpot++] = (byte)beatsPerMinute;
rateSpot %= RATE_SIZE;
beatAvg = 0;
for (byte x = 0 ; x < RATE_SIZE ; x++)
beatAvg += rates[x];
beatAvg /= RATE_SIZE;
}
}
Serial.print("IR=");
Serial.print(irValue);
Serial.print(", BPM=");
Serial.print(beatsPerMinute);
Serial.print(", Avg BPM=");
Serial.print(beatAvg);
if (irValue < 50000)
Serial.print(" No finger?");
Serial.println();
}

```

После завершения загрузки кода откройте последовательный монитор.

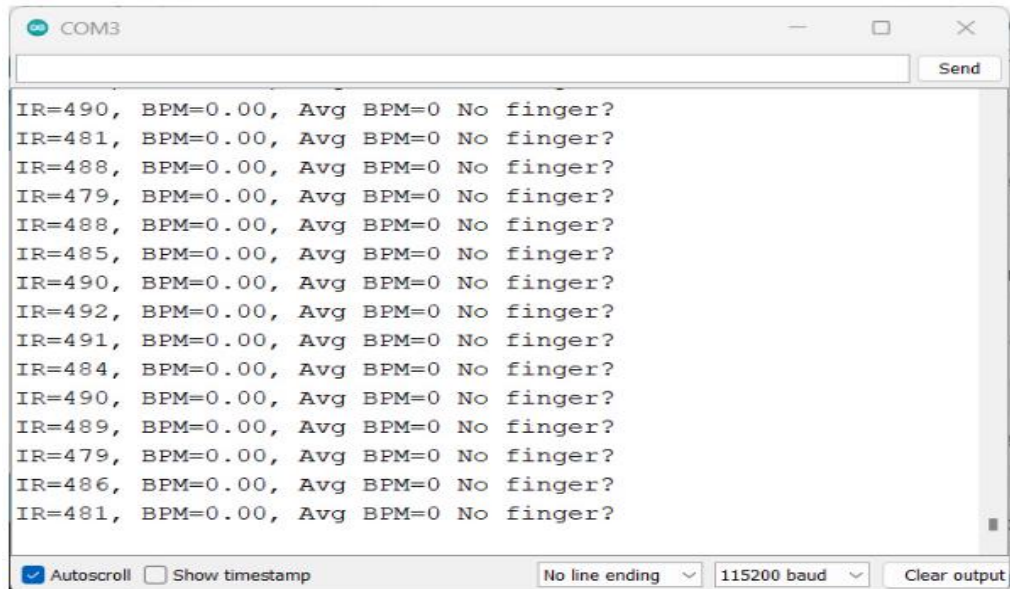
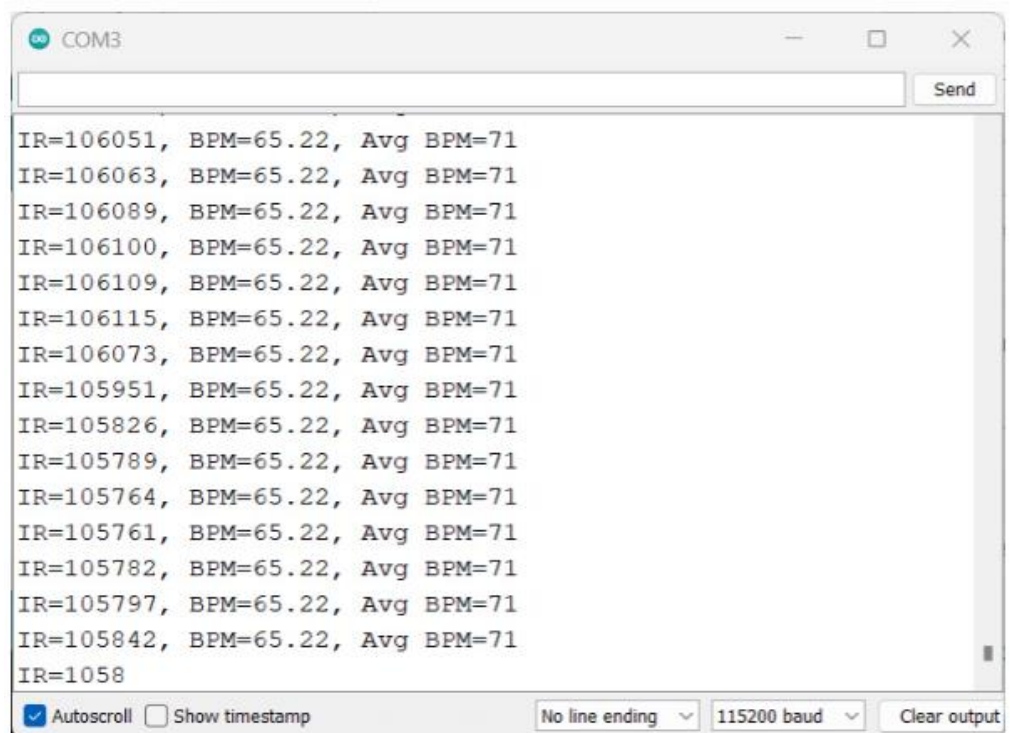


Рисунок 10.5. Состояние запуска программы (т.е. состояние, когда палец не приложен к сенсору)

При первом запуске программы Serial Monitor покажет значение IR как 480-90 и значение BPM как 0,00. Это связано с тем, что датчик изначально не обнаруживает палец, что приводит к нулю BPM. Однако, как только палец прикладывается к датчику, значения отображаются четко.



10.6- Картина. Состояние получения результата (т. е. состояние, когда палец приложен к датчику)

## ЗАДАНИЯ

1. Обнаружить и отобразить сердечный ритм с помощью датчика пульсоксиметра Max30102 с использованием микроконтроллера Arduino Uno.
2. Отобразить измерения температуры тела с помощью датчика пульсоксиметра Max30102 с использованием микроконтроллера Arduino Uno..
3. Вывести диаграммы пульса с помощью Serial Plotter.
4. Определить и отобразить уровень насыщения крови кислородом.
5. В этом практическом упражнении с помощью светодиодов зажгите желтый светодиод, если частота пульса ниже нормы, и красный светодиод, если она высокая.

## ЗАКЛЮЧЕНИЕ

Интернет вещей — это вторая волна мощной цифровой революции, которая началась с повсеместным распространением компьютеров в 1970–1980-х гг. И как все революции, она обещает, что будут и победители, и побежденные. Интернет вещей дает потребителю новые товары и услуги, и многие из существующих товаров и услуг оказываются безнадежно устаревшими. Технологии делают ненужными старые должности, но при этом приводят к появлению новых направлений деятельности. Взаимосвязанные системы влияют на сферу образования, государственное управление и бизнес, вносят существенные изменения в наши действия, поведение и социальные нормы. Новые технологии влияют абсолютно на все, начиная от процедур голосования на выборах до посещения ресторанов и способов проведения отпуска.

Тем не менее потенциальная польза не обходится без больших проблем и множества непредусмотренных последствий. В будущем могут появиться новые виды преступлений, оружия и методов ведения войны. Также, вследствие того, что люди становятся все дальше друг от друга — а этому в числе прочего способствуют новые технологии, — могут возникнуть серьезные политические и социальные проблемы. Обществу определенно придется пересматривать взгляды на представления о конфиденциальности и безопасности.

В последующие годы Интернет вещей затронет практически каждый аспект нашей жизни. Но, поскольку рассмотреть каждый в настоящей книге не представляется возможным, мы лишь одним глазком заглянем в мир, который обещает изменить нашу жизнь быстрее и серьезнее, чем все прочие технические изобретения в истории человечества. Вопрос состоит не в том, наступит ли эпоха Интернета вещей, а в том, как именно это произойдет и насколько сильно изменит мир.

## ЛИТЕРАТУРА

1. Богородицкая, И.А. М2М – новые возможности для развития сотового бизнеса [текст] /И.А. Богородицкая // Электросвязь. – 2012. – №1. – С. 38-39.
2. А. В. Росляков, С. В. Ваняшин, А. Ю. Гребешков - Интернет вещей /учебное пособие/Самара – 2015.
3. Васильков, А. Микрокомпьютеры для интернета вещей: от умного дома к поумневшему окружению [текст] / А. Васильков // Компьютерра, 14 июня 2013г.
4. Гиббс, М. Интернет вещей – не только для «умных» [текст] / М. Гиббс // Сети/network world. – 2013. – №3
5. Гольшко, А. Строим «интеллектуальный городок» [текст] / А. Гольшко // Мобильные телекоммуникации. – 2013. - №10. – С. 46-51
6. Х.Н. Зайнидинов, С.У. Махмуджанов, Р.Д. Аллабергенев, Д.С. Яхшибаев Интернет вещей (учебное пособие). Т., «Аloqachi», 2019, 206 стр.
7. Х.Н. Зайнидинов, С.У. Махмуджанов, И.Юсупов, Н.С. Маматов Интернет вещей (учебное).Т., «Аloqachi», 2023, 206 стр.
8. Коржов, В. Опасный Интернет вещей. / В. Коржов // Открытые системы. СУБД. – 2013. – №4. – С. 29-30.
9. Круз, Л. Сотовые телефоны станут датчиками? [текст] / Л. Круз // Мобильные телекоммуникации. – 2013. – №4-5. – С. 36-38
10. Кучерявый, Е.А. Интернет нановещей и наносети [текст] / Е.А. Кучерявый, С. Баласубраманиям // Электросвязь. – 2014. – №4. – С. 24-26.
11. Кучерявый, Е. А. Принципы построения сенсоров и сенсорных сетей [текст] / Е. А. Кучерявый, С. А. Молчан, В. В. Кондратьев // Электросвязь. – 2006. – №6. – С.10–15.
- 12.Самсонов, М. Интернет вещей в умном городе [текст] / М. Самсонов, А. Гребешков, А. Росляков, С. Ваняшин // ИнформКурьер-Связь. – 2013. – №10. – С. 58-61.
- 13.Черняк, Л. Интернет вещей: новые вызовы и новые технологии [текст] / Л. Черняк // Открытые системы. СУБД. – 2013. – №4. – С. 14-18
- 14.М. Kranz, Building the Internet of Things, 2017, Hoboken, New Jersey: JohnWiley & Sons, Inc.
15. J. Perkins, “Internet of Things Improving Goldcorp Operations,” Mining Innovation News, 2015, <https://republicofmining.com/2015/03/13/internet-of-things-improving-goldcorp-operations-by-james-perkins-mining-innovation-news-march-12-2015/>
- 16.Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012, December). Future internet: the internet of things architecture, possible applications and key challenges. In Frontiers of Information Technology (FIT), 2012 10th International Conference on (pp. 257-260). IEEE.

17. James Kirkland , “Internet of Things: insights from Red Hat” , Website: <https://developers.redhat.com/blog/2015/03/31/internet-of-things-insights-from-red-hat/> , Accessed : 2nd February 2018
18. Raj Kamal, “**Internet of Things**” **Architecture and Design Principles**, <<McGraw Hill Education (India) Private Limited >> 2017, 551 pages. ISBN-13: 978-93-5260-522-4.
19. John Soldatos, “Building Blocks for IoT”, <<River Publishers >> 2017, 294 pages. ISBN: 978-87-93519-03-9
20. Mr.M.Anantha Guptha, “Internet of Things & Its applications”, Lecture notes, 2021, 146 pages.
21. Н. В. Папуловская, “Основы интернета вещей”, Учебно-методическое пособие, Уральский федеральный университет, 2022, 104 pages, ISBN 978-5-7996-3537-4
22. П.А. Кокунин, И.И. Латыпов, Л.С. Латыпова, “Введение в интернет вещей”, Учебное пособие 2022, 148 pages, ISBN 978-5-00130-649-8
23. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri, “Internet of Things Architectures, Protocols and Standards”, John Wiley & Sons Ltd -2019, 394 pages, ISBN 9781119359685
24. Francis daCosta, “Rethinking the Internet of Things” a Scalable Approach to Connecting Everything, 2019, 185 pages, ISBN-13 (electronic): 978-1-4302-5741-7
25. Fei Hu, “Security and Privacy in Internet of Things (IoTs)”, Models, Algorithms, and Implementations, 2016, 586 pages.

### **ИНТЕРНЕТ-САЙТЫ.**

1. [Google.com](https://www.google.com)
2. [Yandex.com](https://www.yandex.com)
3. [Tinkercad.com](https://www.tinkercad.com).
4. [Lex.uz](https://lex.uz)
5. [uz.wikipedia.org](https://uz.wikipedia.org)