

Микроконтроллер ESP32

Микроконтроллер ESP32 – это одна из самых доступных и мощных платформ для создания умных ардуино-проектов с поддержкой WiFi. Придя на смену ESP8266, этот чип дал новые возможности для разработчиков, хотя по-прежнему остались старые проблемы с поддержкой и документацией.

Описание микроконтроллера ESP32

Фирма Espressif выпустила мощный недорогой микроконтроллер ESP32 летом 2016 года. Устройство представляет собой систему на кристалле, построенную по технологии TSMC 40 нм, с Wi-Fi и Bluetooth контроллерами. Оно оснащено двухъядерным 32-битным процессором, который работает на частотах 80, 160 или 240 МГц. Также в систему интегрированы антенные коммутаторы, радиочастотные компоненты, фильтры, усилители, модули управления питанием. Подключается ESP32 к компьютеру через обычный USB провод.

Микроконтроллер ESP32



Характеристики чипа

Технические характеристики ESP32:

- Двух- или одноядерный 32-битный процессор Tensilica Xtensa LX6;
- Тактовая частота – 160 или 240 МГц;
- 520 Кб SRAM;
- Максимальный ток потребления 260 мА, в спящем режиме – 10 мА;
- Стандарты беспроводной связи – Wi-Fi: 802.11 b / g / N, Bluetooth: v4.2 BR/EDR and BLE;
- Наличие датчиков температуры, Холла, тач-сенсоров;
- Инфракрасное дистанционное управление;
- Можно подключать двигатели и светодиоды через ШИМ разъем;
- Стандарт IEEE 802.11 с поддержкой WPA, WPA/WPA2 и WAPI;
- Возможность безопасной загрузки;
- Шифрование флэш диска.



Также в модуле традиционно присутствует встроенное управление энергопитанием. Для этого используются линейный регулятор, индивидуальное питание для RTC (ядро низкого энергопотребления), пробуждение по таймеру или сенсорному датчику.

Программирование модуля ESP32 может производиться на самых разных платформах, и средах, вот небольшой список наиболее популярных вариантов:

- Arduino IDE;
- Espressif IoT Development Framework;
- Espruino;
- PlatformIO;
- PyMaker IDE.

Большинство проектов реализуется на Arduino IDE и Espruino.

Использование ESP 32

Хоть плата ESP32 появилась недавно, она уже активно используется в коммерческих проектах, связанных с мобильными приложениями, электроникой и задачами IoT. Например, на ее базе построен светодиодный браслет IoT группы Alibaba. Он представляет собой живой беспроводной экран, в котором каждый браслет работает как пиксель. Также на основе микроконтроллера реализована биометрическая система отслеживания посещаемости и проекты, связанные с анализом климатических условий. Имея плату ESP32 и датчики температуры, влажности и давления, можно самостоятельно собрать метеостанцию. ESP32 используется в музыкальных плеерах, помощниках с голосовым управлением, аудиогарнитуре.

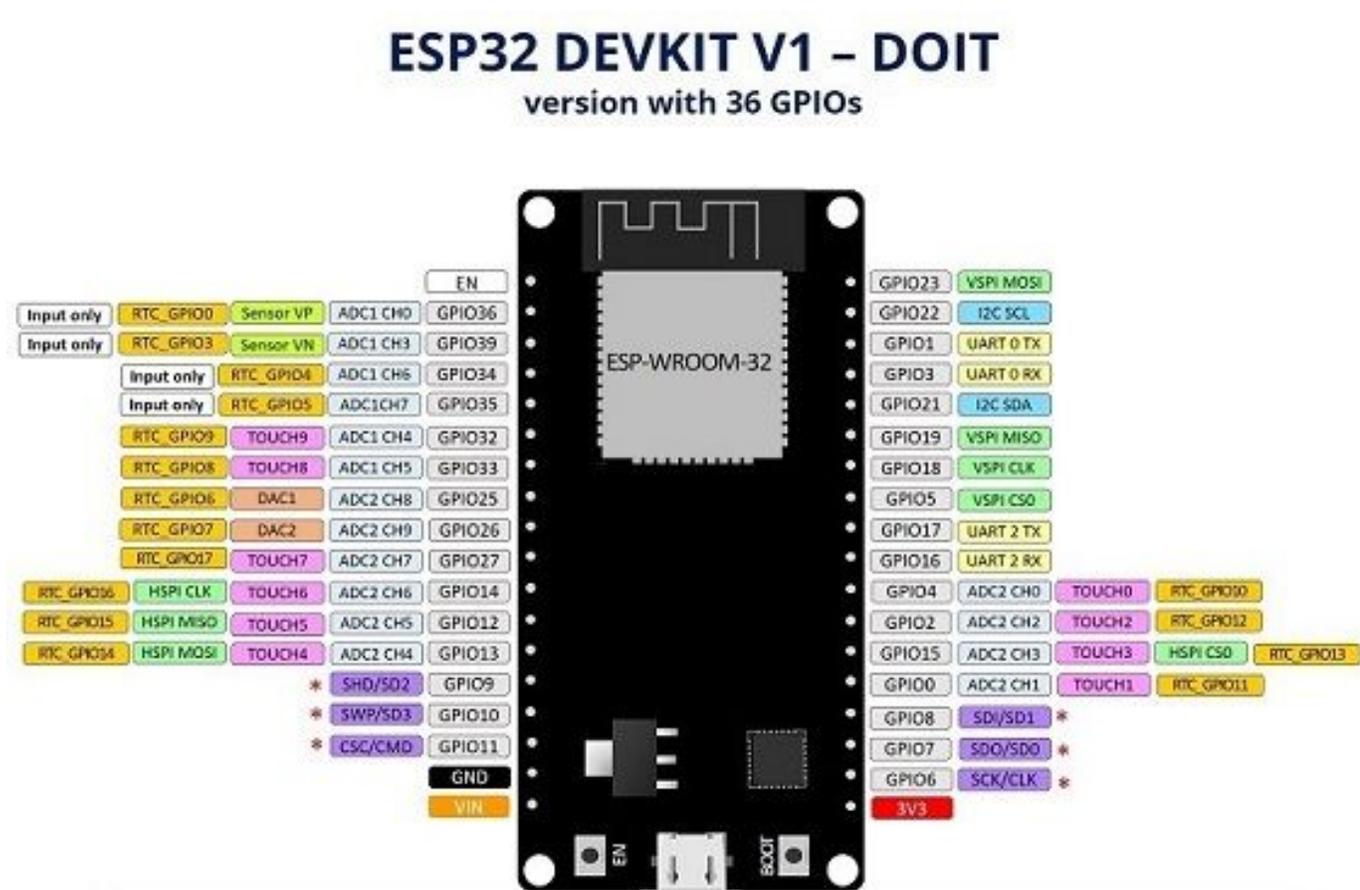
Распиновка ESP32

Микроконтроллер оснащен 48 контактами плюс 1 большой тепловой контакт, которые обладают разными функциями.

Выводы микросхемы:

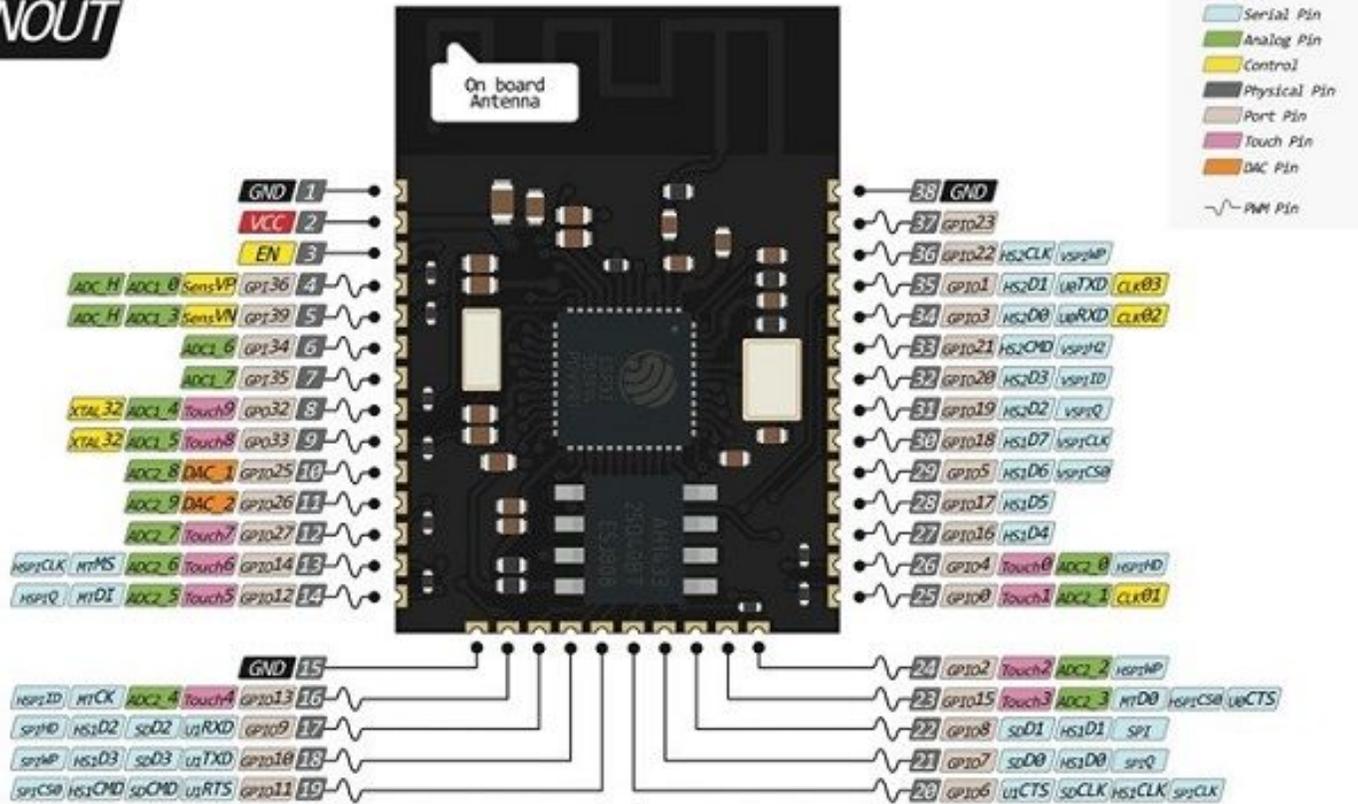
- 18 каналов 12-разрядного АЦП;
- 10 GPIO портов;
- 3 SPI;
- 3 UART;
- 2 I2C;
- 16 ШИМ выводов;
- 2 8-битных ЦАП вывода;
- 2

Расположение выводов зависит от производителя. Например, есть плата ESP32 DEVKIT V1 DOIT, у которой 36 контактов. Распиновка представлена на рисунке:



Одним из самых популярных модулей является ESP-WROOM-32. Распиновка также приведена на картинке.

ESP32
PINOUT



К портам GPIO 0, 4, 2, 15, 13, 12, 14, 27, 33 и 32 подключены сенсорные выводы. Они могут использоваться для вывода ESP32 из глубокого сна. Они фиксируют любое изменение электрического заряда.

Порты с 34 по 39 используются только для ввода информации. На них отсутствуют подтягивающие резисторы, поэтому их нельзя использовать как выходы.

На некоторых платах отображаются контакты 6-11. Они подсоединены к к интегрированной SPI flash. Их не используют в проектах.

Есть различные версии платы ESP32 размерами 5x5 мм или 6x6 мм. Также на основе ESP32 существуют модули SMT для интегрирования в другие платы.

Тактовый генератор

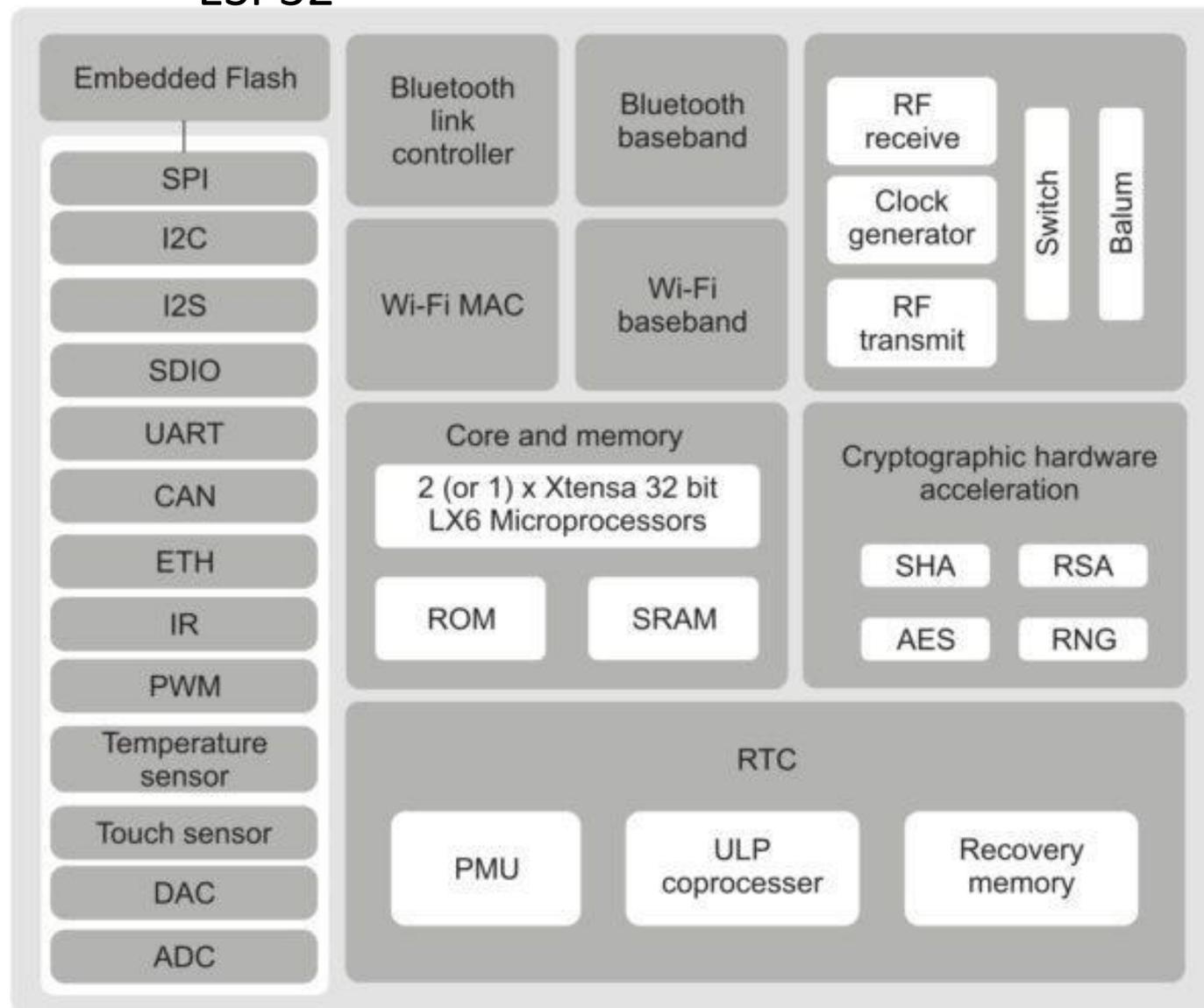
FLASH-память



Антенна

Микроконтроллер

Функциональный состав чипа ESP32



Распределение внешней памяти ESP32

Bus Type	Boundary Address		Size	Target	Comment
	Low Address	High Address			
Data	0x3F40_0000	0x3F7F_FFFF	4 MB	External Flash	Read
Data	0x3F80_0000	0x3FBF_FFFF	4 MB	External SRAM	Read and Write
Bus Type	Boundary Address		Size	Target	Comment
	Low Address	High Address			
Instruction	0x400C_2000	0x40BF_FFFF	11512 KB	External Flash	Read

UART0	UART1	UART2
SPI1	SPI2	SPI3
I2S0	I2S1	
SDIO Slave	SDMMC	
EMAC		
BT	WIFI	

Структура входных и выходных дескрипторов

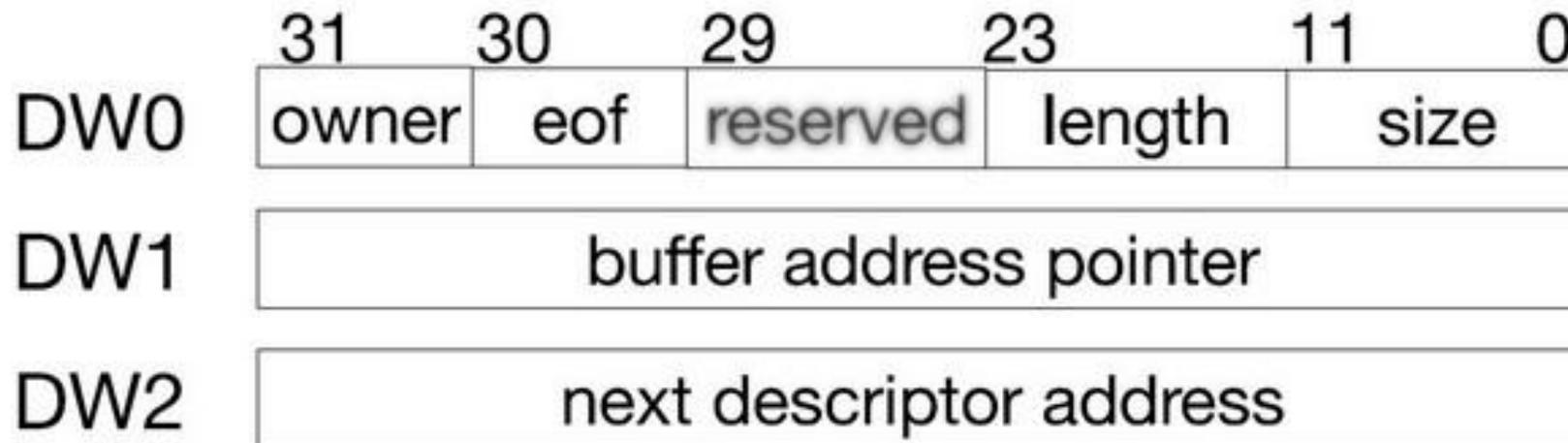


Table 15: APB_CLK Derivation

CPU_CLK Source	APB_CLK
PLL_CLK	80 MHz
APLL_CLK	CPU_CLK / 2
XTAL_CLK	CPU_CLK
RTC8M_CLK	CPU_CLK

Отличия esp32 от esp8266

Платформы ESP8266 и ESP32 произведены одной компанией Espressif. Микроконтроллер ESP32 отличается от своего предшественника улучшенными характеристиками, увеличенным функционалом и большим объемом памяти. Стоимость нового прибора дороже примерно в 2 раза.

Важным преимуществом ESP32 является более быстрый Wi-Fi и Bluetooth. В ESP32 установлен более мощный процессор, позволяющий реализовывать сложные проекты. Эта платформа подходит для приложений, в которых требуется интернет или новые интерфейсы. Для более дешевых разработок используется ESP8266.

Объем памяти у нового устройства ESP32 увеличен – 512 Кб против 160 Кб ESP8266. Также ESP32 отличается большим количеством выводов GPIO. К нескольким контактам на ESP32 прикреплены емкостные сенсорные датчики и датчик температуры. На обоих устройствах контакты GPIO можно использовать по-разному. ESP32 имеет 18 12-битных АЦП каналов. У его предшественника есть всего 1 10-битный вывод АЦП.

Мощность процессора значительно влияет на скорость работы. Модуль ESP32 показывает рекордную производительность по сравнению с предшественником ESP8266. Загрузка страницы с длинным скетчем и множеством графики занимает секунды.

Из недостатков ESP32 можно выделить отсутствие библиотек для поддержки сенсоров и малое количество драйверов. Это связано с тем, что плата появилась в продаже недавно. Но учитывая все преимущества микроконтроллера и его перспективы, эта проблема будет решена уже в ближайшее время.

Настройка Arduino IDE для работы с ESP32

Изначально изделия от компании Espressif поставляются с прошивкой, позволяющей работать с помощью AT команд. Это не всегда удобно, поэтому лучше запрограммировать плату в привычной среде разработки – Arduino IDE.

Чтобы начать создавать проекты на ESP32, сначала нужно иметь его поддержку в среде разработки Arduino IDE. Поддержка в настоящий момент находится на начальном, но работоспособном уровне. Проблемы могут возникнуть с драйверами устройств, но из-за популярности модуля вскоре будет поддерживаться вся периферия.

В первую очередь нужно скачать Arduino IDE и официальную инструкцию с сайта <https://github.com/espressif/arduino-esp32>. Нужно загрузить файлы дистрибутива и поместить в папку C:\Users\User\Documents\Arduino\hardware\espressif\esp32.

Так как плата ESP32 новая и поддержка в среде разработки Ардуино появилась недавно, то могут возникнуть сложности с установкой. Это связано с тем, что драйвера только разрабатываются и изменен порядок расположения системных директорий. Все скетчи должны быть расположены в папке C:\Users\User\Documents\Arduino. Файлы дистрибутива должны быть размещены внутри этой папки, как советует производитель. Если этого не учесть, то поддержка в Ардуино ESP32 будет отсутствовать.

После установки можно открыть Arduino IDE. Затем нужно перейти в настройки и в менеджере плат выбрать нужную. Теперь можно прошивать модуль.

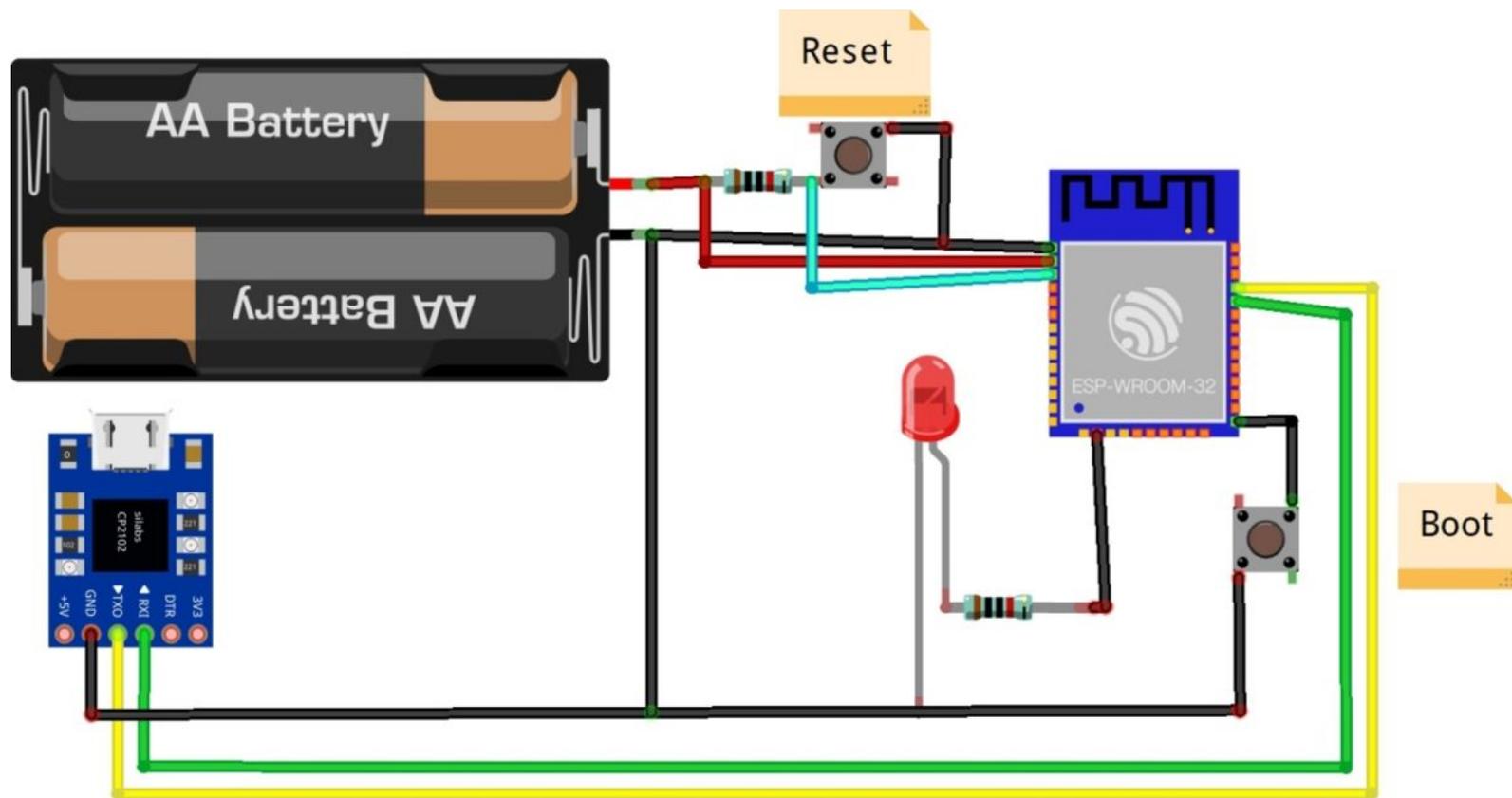
Проверка модуля и подключение к Arduino IDE

Чтобы произвести проверку работоспособности модуля, можно собрать проект с мигающим светодиодом. Для подключения потребуются:

- Модуль ESP32;
- Источник питания на 3 В;
- Кнопка;
- Резисторы;
- USB кабель для подключения к компьютеру;
- Провода;
- Светодиод;
- USB-TTL конвертор.

Затем нужно загрузить тестовый код, прописав в нем номер COM порта, к которому подключен модуль, и тип платы. После загрузки скетча должен замигать светодиод.

Собирается это все согласно схеме ниже.



Настройка официальной среды разработки ESP-IDF (Espressif IoT Development Framework)

Среду ESP-IDF можно установить на Windows. Для этого нужно:

- Скачать с официального сайта среду разработки;
 - Распаковать архив, поместить его на диск C;
 - Затем появится каталог `msys32`, в который нужно перейти и запустить программу `mingw32.exe`;
 - Появится окно терминала, где нужно создать каталог с именем `esp`;
 - Нужно перейти в созданный каталог;

 - Подключить необходимые библиотеки с помощью команды `git clone --recursive https://github.com/espressif/esp-idf.git`;
 - Затем устанавливаются пакеты Python командой `python -m pip install --user -r $IDF_PATH/requirements.txt`.
- После этого можно начинать работу. В каталоге `\esp-idf\examples\` есть различные примеры скетчей для работы с модулем.