

Занятие 2 — Как начать работать с отладочной платой NodeMCU ESP8266?

На прошлом занятии мы разобрались с вопросом [как подключить отладочную плату NodeMCU ESP8266 к компьютеру](#). Теперь настало время разобраться как начать работу с отладочной платой NodeMCU ESP8266, т.е. как залить прошивку в отладочную плату NodeMCU ESP8266, как загружать скрипты (исполняемые программы) в NodeMCU ESP8266 и как управлять отладочной платой NodeMCU ESP8266.

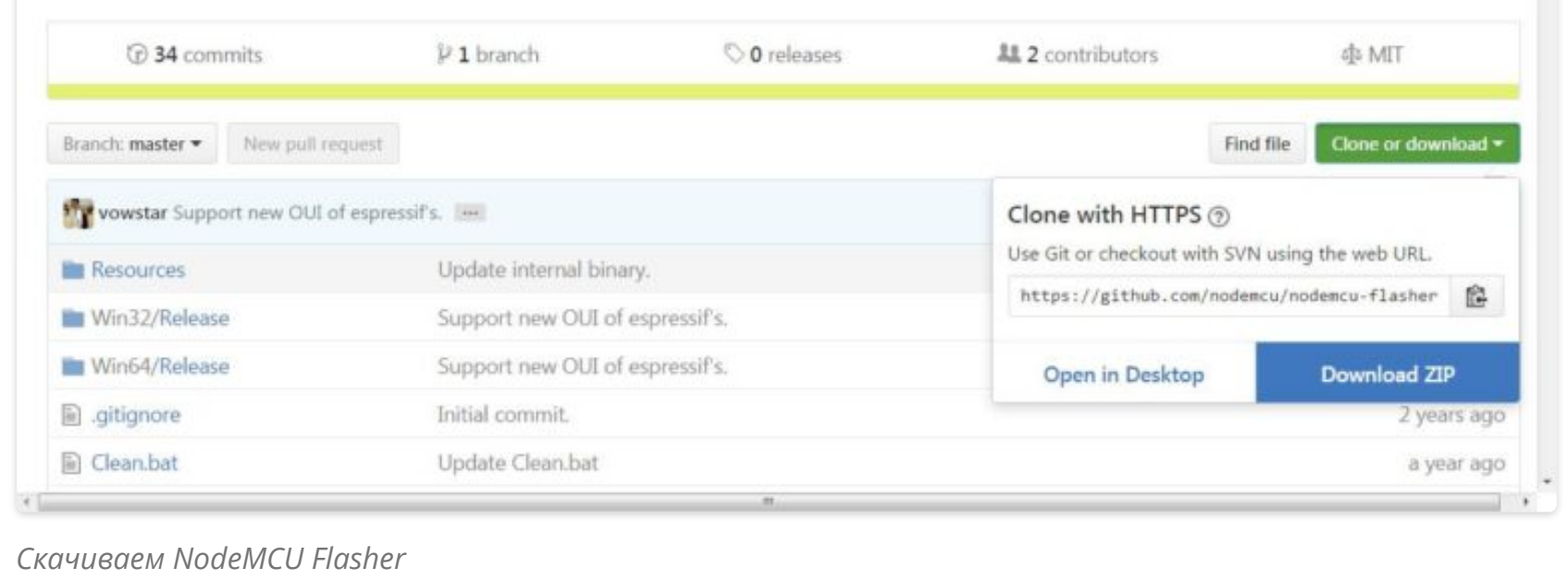
Еще раз напомним, что **предет NodeMCU** состоит из самой отладочной платы NodeMCU ESP8266 и прошивки NodeMCU. Прошивка заливается в NodeMCU ESP8266 и в дальнейшем управляет отладочной платой по написанным нами инструкциям (скриптам) на скриптовом языке Lua. Прошивка NodeMCU создаст подобие файловой системы на основе `spiffs`, что дает определенное удобство при разработки управляющих программ. Вот чем еще хороша прошивка NodeMCU:

- легкость программирования беспроводного клиента и/или точки доступа;
- основана на Lua 5.1.4 (без отладочного модуля);
- асинхронная событийно-ориентированная модель программирования;
- более 40 встроенных модулей для наиболее распространенных датчиков и задач;
- прошивка доступна как с поддержкой плавающей точки так и в варианте экономии памяти с целочисленной поддержкой;
- обновляемая документация на <https://nodemcu.readthedocs.io>

Но для того, чтобы воспользоваться всеми прелестями этой чудесной прошивки ее необходимо залить в нашу отладочную плату NodeMCU ESP8266.

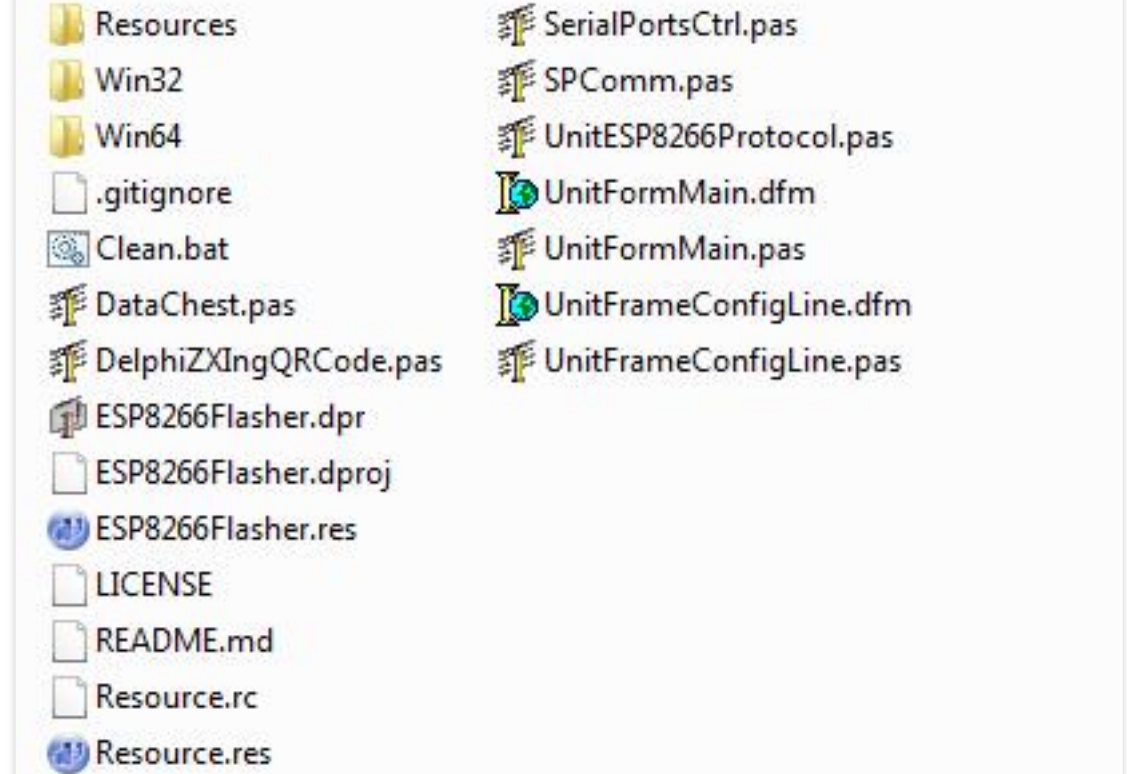
Как прошить NodeMCU ESP8266 прошивкой NodeMCU Lua

Для того прежде всего нам потребуется **прошивальщик NodeMCU Flasher**. Скачать его можно вот отсюда <https://github.com/nodemcu/nodemcu-flasher>. В открывшемся окне слева видим зеленую кнопку `Clone or download`, нажав на нее видим синюю кнопку `Download ZIP`, нажав на которую сохраним архив `nodemcu-flasher-master.zip` в удобное место.



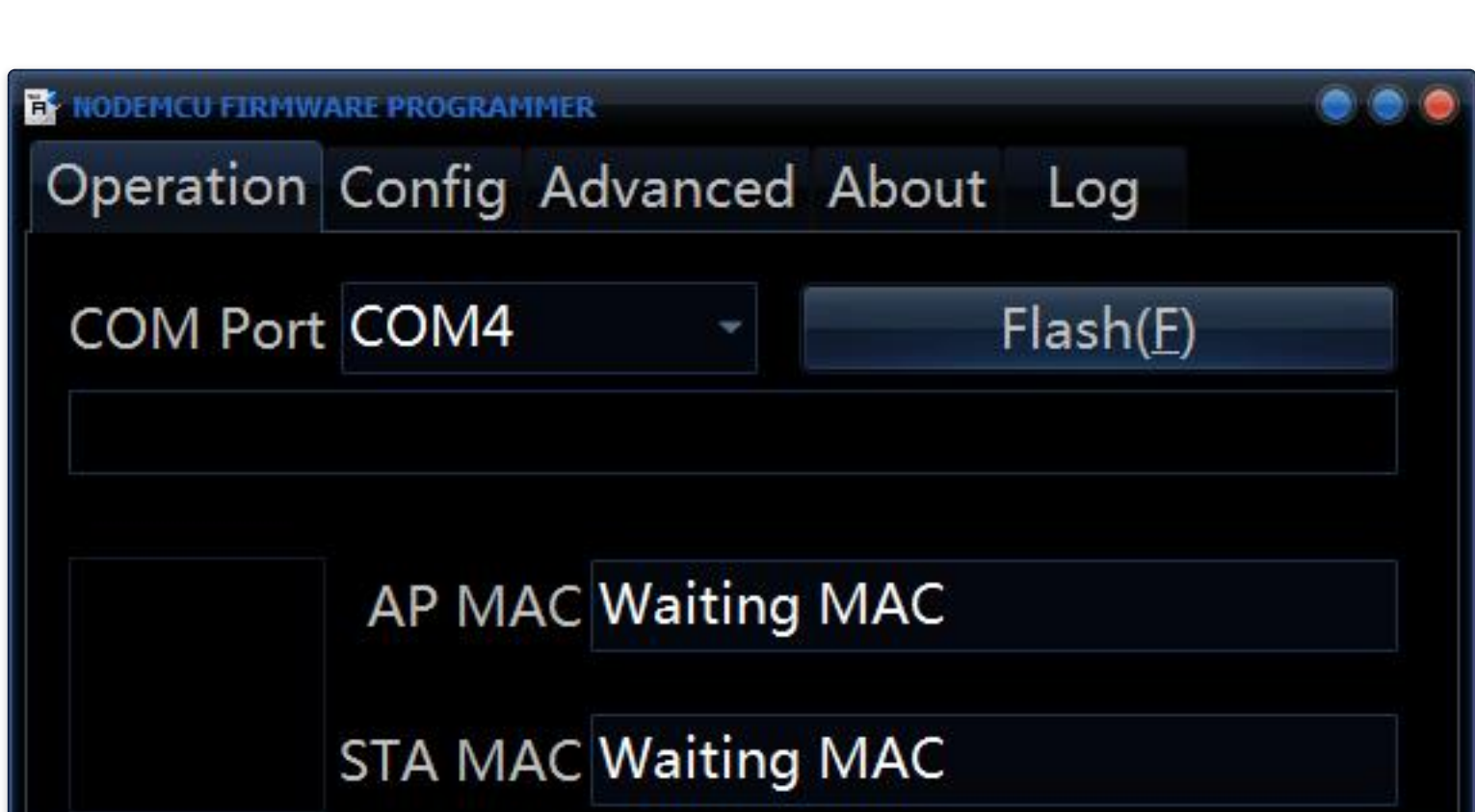
Скачиваем NodeMCU Flasher

Затем необходимо распаковать содержимое архива в отдельную папку, содержимое которой будет следующее



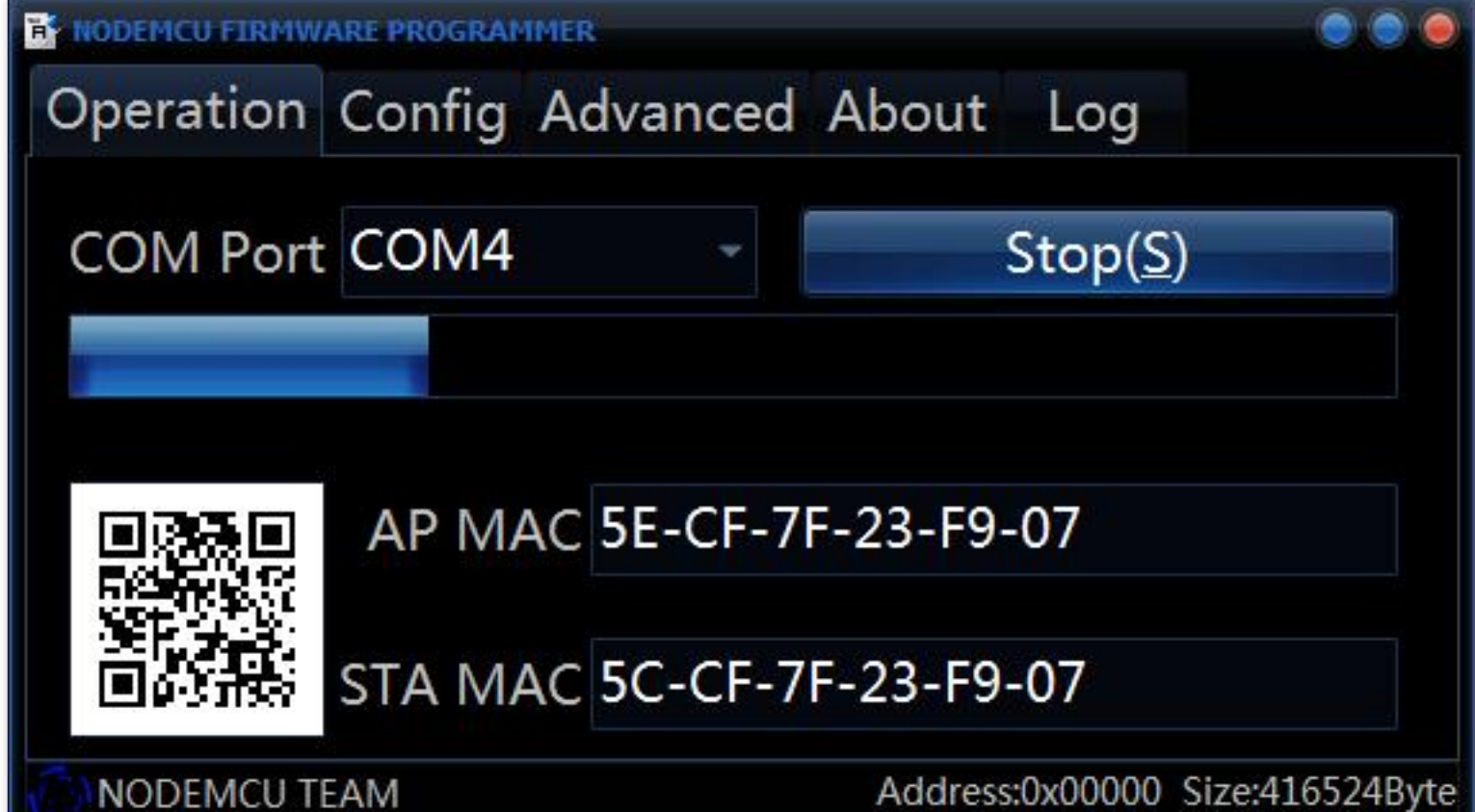
Папка с NodeMCU Flasher

Далее в зависимости от разрядности Windows идём или в папку `Win32` или в `Win64`, в которых есть папка `Release`, внутри которой лежит исполняемый файл `ESP8266Flasher.exe`. Вот от нам и нужен! Теперь самое время подключить к ПК отладочную плату NodeMCU ESP8266 (напомним, что **подключение NodeMCU ESP8266 к компьютеру** было описано в занятии №1) и запустить `ESP8266Flasher.exe`. Должно появиться следующее окно



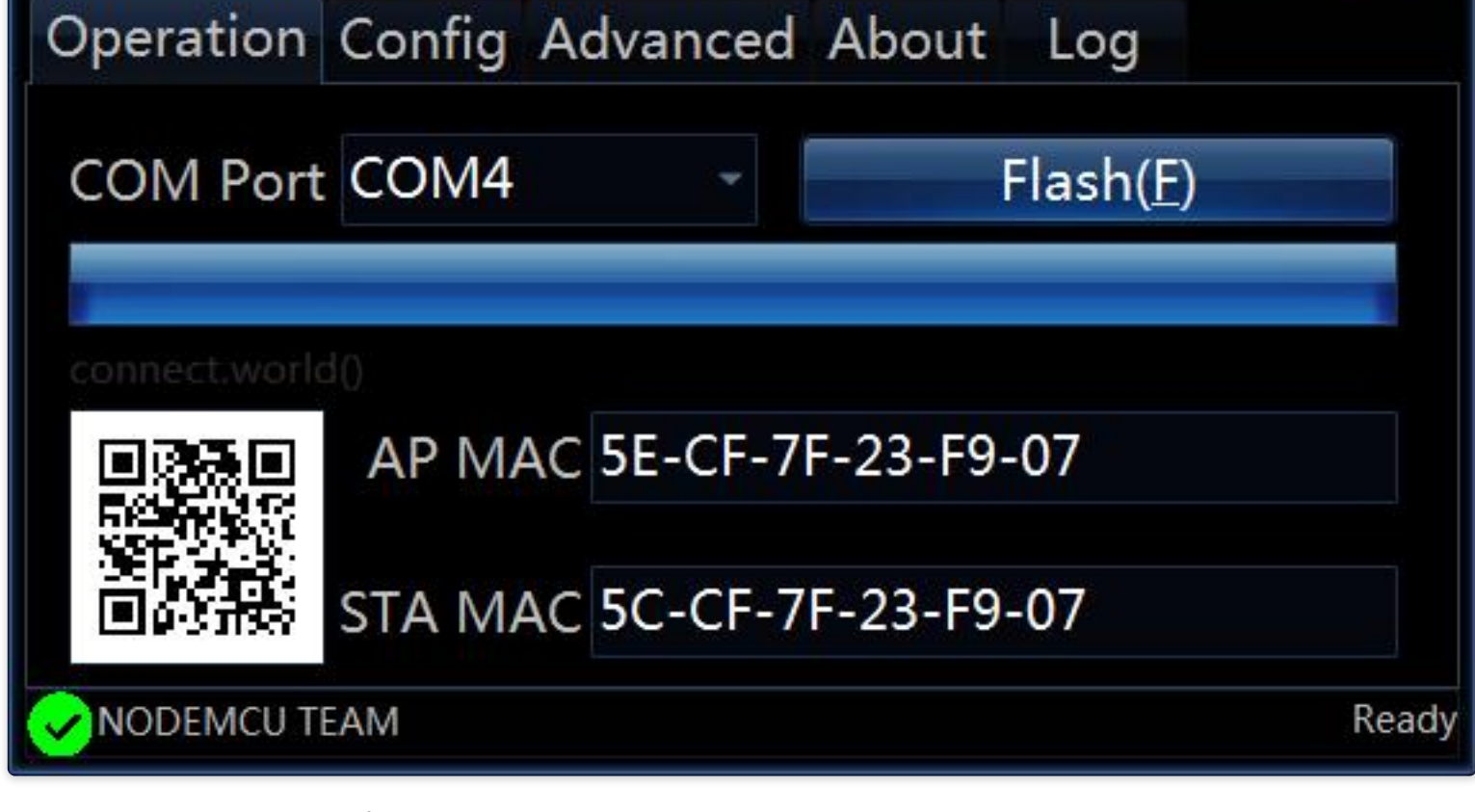
ESP8266 Flasher

Основным признаком того, что NodeMCU Flasher обнаружил отладочную плату NodeMCU ESP8266 является указание верного COM-порта в соответствующем поле (в данном случае это COM4). Скачанный NodeMCU Flasher уже содержит скомпилированную прошивку NodeMCU, поэтому останется просто нажать кнопку `Flash(F)`, после чего начнется прошивка. (не рекомендуется менять *настройки на остальных этапах* без должного понимания их назначения. Программа работает с настройками по умолчанию)



Прошивка ESP8266 через NodeMCU Flasher

Прошивка займет около минуты, а отладочная плата NodeMCU ESP8266 в это время будет бодро мигать встроенным светодиодом. Как видно в процессе заливки отладочной плате NodeMCU ESP8266 было присвоено два MAC адреса, а сама прошивка размещена в ESP8266 с адреса `0x0000` и имеет размер `416524` байта (примерно `406` Кбайт). По окончании прошивки в нижней строке будет отображена зеленая галочка и надпись `Ready`.



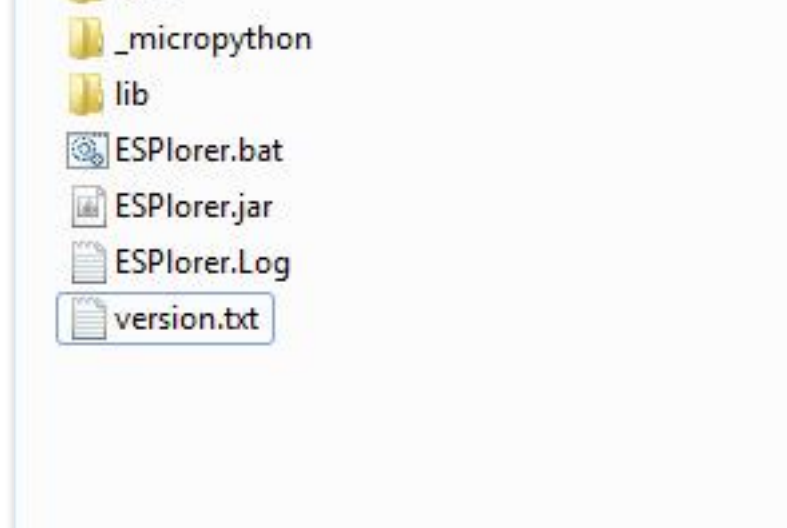
Успешная прошивка NodeMCU

На этом можно считать, что в отладочную плату NodeMCU ESP8266 была успешно залита прошивка NodeMCU и теперь можно переходить к следующему этапу – **настройка отладочной платы NodeMCU ESP8266 для программирования**.

Как программировать NodeMCU ESP8266

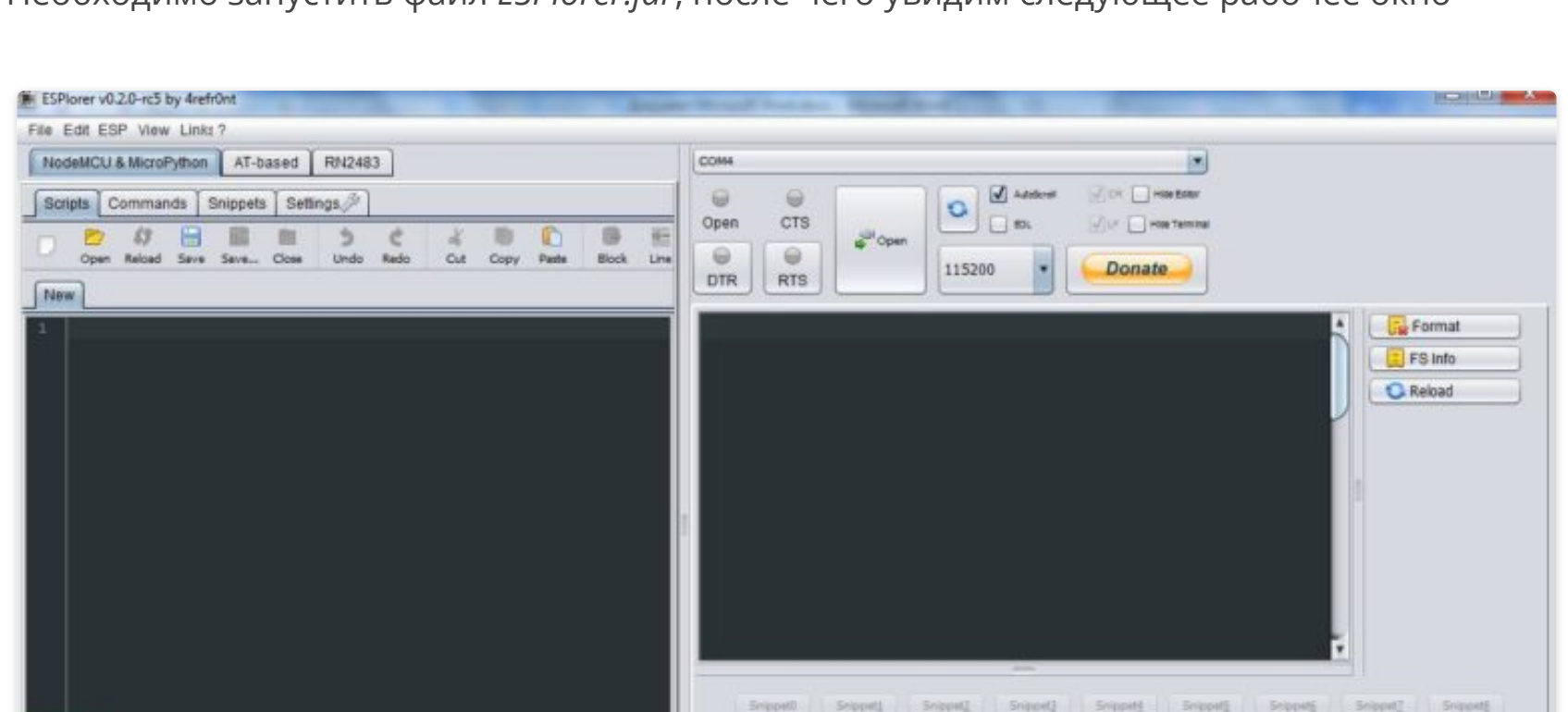
Для программирования NodeMCU ESP8266 существует множество различных программ. На наш взгляд достаточно удобной является IDE для ESP8266 которая называется **ESPLorer**. Это универсальная среда программирования поддерживает языки Lua, Python и AT команды, является мультиплатформенной (Windows, Linux, Solaris, Mac OS X). Для своей работы **ESPLorer требует самую свежую версию Java**, поэтому вначале необходимо обновить ее. Сделать это можно по ссылке https://www.java.com/ru/0/download/windows_installer.jsp. Ничего сложного в этом нет, стандартная процедура установки не должна вызвать вопросы, поэтому не будем на этом останавливаться.

Далше необходимо скачать ESPLorer с сайта разработчика по ссылке <http://espressosystems.ru/esplorer/>. Искать нужную ссылку там долго, поэтому можно скачивать сразу с файлового хранилища по ссылке [скачать ESPLorer.zip \(v_0.2.0-rc6\)](#). Затем как обычно сохраняем файл `ESPLorer.zip` в удобное место и распаковываем в отдельную папку, которая будет иметь следующий вид



Папка с ESPLorer

Необходимо запустить файл `ESPLorer.jar`, после чего увидим следующее рабочее окно



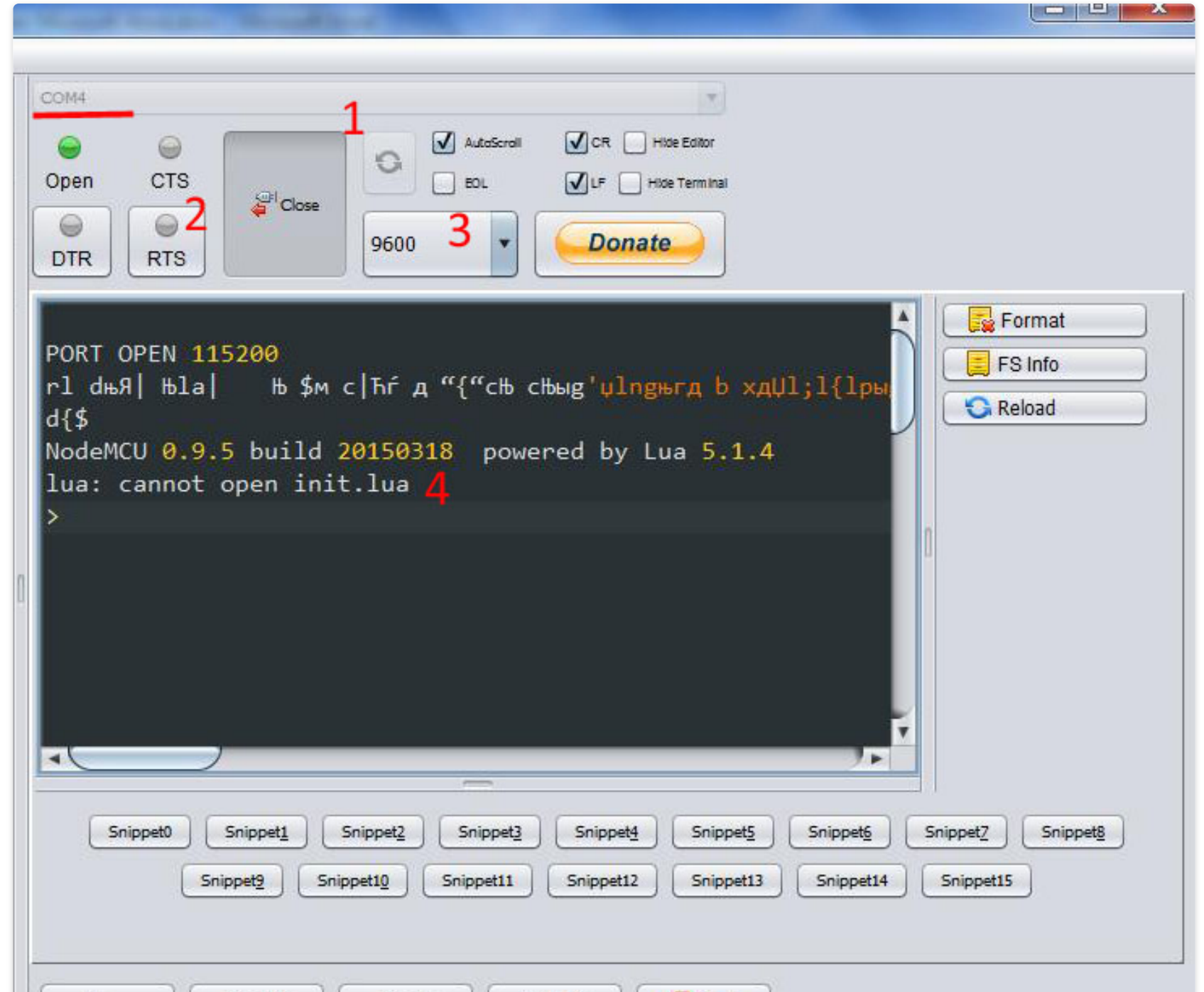
ESPLorer - удобная IDE для NodeMCU ESP8266

Обратите внимание, что ESPLorer обнаружил отладочную плату NodeMCU ESP8266 подключенной к COM-порту 4, это хороший знак и можно попробовать установить связь с NodeMCU ESP8266. Для этого сделаем следующее (см. рисунок ниже)

- 1 – нажимаем кнопку `Open` (открываем порт), в окне сообщений видим `PORT OPEN 115200`, а индикатор `Open` стал зеленым
- 2 – Нажимаем и отжимаем кнопку `RTS`, при этом в окне сообщений появляются «крякозябры»
- 3 – обращаем внимание на скорость обмена, она установлена в `115200`. Меняем на `9600`
- 4 – снова нажимаем и отжимаем кнопку `RTS` и, О чудо!! Видим в окне сообщений текст `NodeMCU 0.9.5 build 20150318 powered by Lua 5.1.4`

`lua: cannot open init.lua`

>



Подключение ESPLorer к отладочной плате NodeMCU ESP8266

Первая строка сообщает о версии загруженной прошивки NodeMCU (в данном случае это `сила устройства 0.9.5`, но об этом поговорим позже). Вторая строка сообщает нам, что скрипт `init.lua` не обнаружен и поэтому NodeMCU ESP8266 приостановит. Действительно, загружая прошивку NodeMCU мы полностью «отформатировали» NodeMCU ESP8266. Поэтому сейчас можно приступить к следующему шагу — **написание и загрузка в отладочную плату NodeMCU ESP8266 своего первого скрипта**.

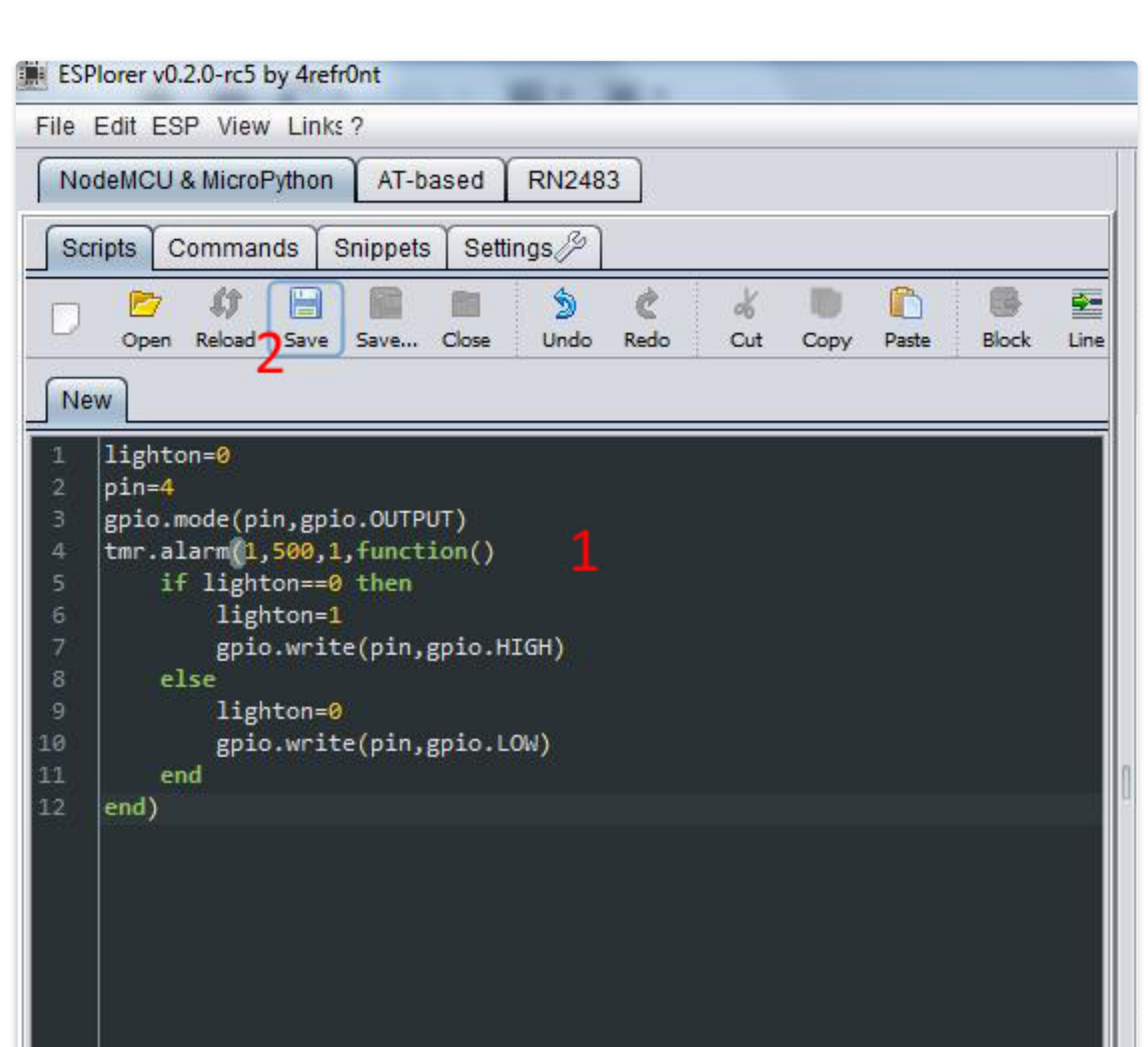
Как загрузить в NodeMCU ESP8266 скрипт (программу)

В качестве демонстрационного скрипта напомним классический пример мигания встроенным светодиодом. Для этого сделаем следующее:

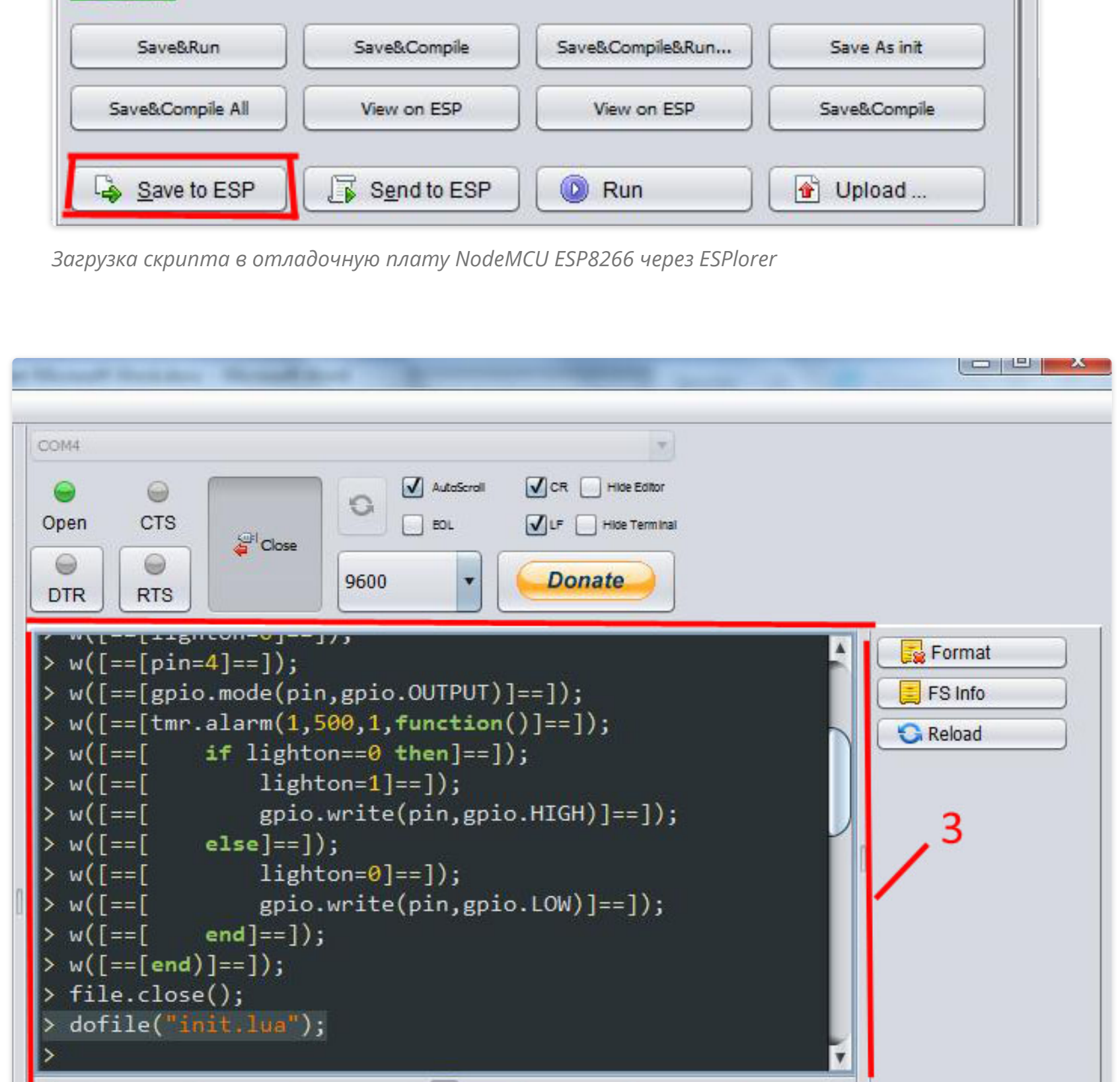
- 1 — в левом окне ESPLorer'a наберем следующий программный код

```
lighton=0
pin=4
gpio.mode(pin,gpio.OUTPUT)
tmr.alarm(1,500,1,function()
  if lighton==0 then
    lighton=1
    gpio.write(pin,gpio.HIGH)
  else
    lighton=0
    gpio.write(pin,gpio.LOW)
  end
end)
```

- 2 – нажмем кнопку `Save` и сохраним скрипт с именем `init.lua`, таким образом создадим автоматический выполняемый при запуске отладочной платы NodeMCU ESP8266 файл.
- 3 – сразу после сохранения файла произойдет автоматическая загрузка скрипта в отладочную плату NodeMCU ESP8266 (в окне сообщений пройдет соответствующий текст) и его автоматическое выполнение.



Загрузка скрипта в отладочную плату NodeMCU ESP8266 через ESPLorer



Процесс выполнения скрипта в отладочной плате NodeMCU ESP8266 через ESPLorer

Если все прошло успешно, то ваша отладочная плата будет раз в полсекунды включать и выключать встроенный светодиод. **Подтвердилось! Ваш первый скрипт был успешно загружен и выполнен NodeMCU ESP8266.**

Разбор программного кода самого скрипта оставим на следующее занятие, а сейчас просто отметим, что изменяя значение `500` в функции `tmr.alarm` на `1000` или `250` можно увеличивать или уменьшать скорость мигания светодиода. Рекомендуем поиграться с разными значениями времени и понаблюдать за изменениями. Обновление скрипта в NodeMCU ESP8266 производится нажатием на кнопку `Save to ESP` (расположена внизу) или просто при сохранении файла.

Обратите внимание, что сейчас отладочная плата NodeMCU ESP8266 выполняет скрипт полностью самостоятельно, т.е. ей не требуются вспомогательные программы. Подключение к USB-порту компьютера используется только в качестве питания. **Благодаря прошивке NodeMCU ESP8266 каждый раз при запуске или перезагрузке отладочной платы NodeMCU ESP8266 будет автоматически выполняться скрипт init.lua.** В реальных задачах тут можно записать программный код создания WiFi точки доступа или подключения к существующей WiFi точке доступа.

А теперь немного вернемся назад и вспомним, что сейчас у нас загружена прошивка NodeMCU 0.9.5 build 20150318, которая хоть и позволяет работать с NodeMCU ESP8266, но использует устаревший SDK 0.9.2 (к сведению, **актуальная версия SDK от Espressif 1.5.4.7**). Разумеется для полноценной работы с отладочной платой NodeMCU ESP8266 и использования всех ее возможностей необходимо научиться самостоятельно **собирать прошивку NodeMCU, содержащую необходимые модули, а затем загружать ее в ESP8266**. Именно этим мы и займемся на следующем занятии — [Как собрать и обновить прошивку NodeMCU для ESP8266](#)

Поиск

СВЕЖИЕ ЗАПИСИ

[Как BURST клавет на плате VITONIN Международной симпозиуми выставка по вопросам электромагнитной совместимости EMC-2017](#)
[Международный симпозиум по электромагнитной совместимости, целостности сигнала и мощности EMC-SIB-2017](#)
[Симпозиум по электромеханизму в области высоких мощностей ASIAEM 2017](#)
[Международный симпозиум по электромагнитной совместимости АРЕМС-2017](#)

Есть чем поделиться? Свяжитесь с нами и разместите свою новость!