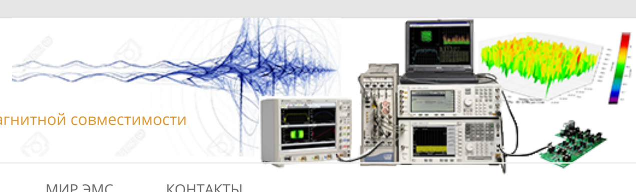


Занятие 3 — Как собрать и обновить прошивку NodeMCU для ESP8266?

На прошлом занятии мы разобрались с вопросом как начать работать с отладочной платой NodeMCU ESP8266 и залили в нее прошивку NodeMCU. Также мы обратили внимание, что версия прошивки была 0.9.5, в то время как на официальном сайте доступна версия NodeMCU 1.5.4.1 https://github.com/nodemcu/nodemcu-firmware
Для того чтобы получить максимум стабильности и производительности от отладочной платы NodeMCU ESP8266, а также воспользоваться библиотекой модулей (а их сейчас целых 48 штук) необходимо обновить прошивку до актуальной версии. Вот тут и возникает первая загвоздка.

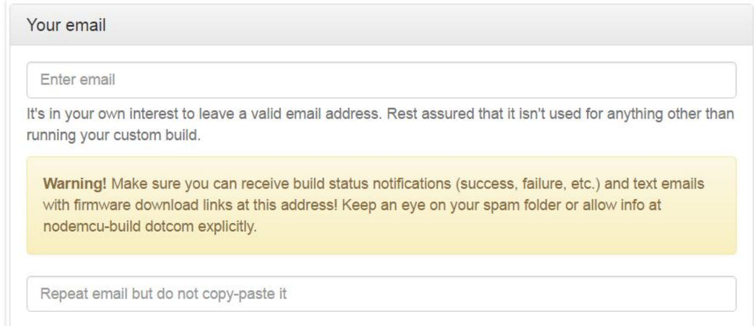
Если вы вспомните занятие 2, то готовая прошивка NodeMCU шла в комплекте с прошивальщиком NodeMCU Flasher и мы ничего не компилировали, а сразу приступали к загрузке прошивки. Сейчас же нам придется вначале скомпилировать (собрать) нужную нам прошивку и только после этого залить ее в отладочную плату NodeMCU ESP8266. Для этого действия существует три способа:

- облачный сервис компилирования
- Docker Image
- компилирование в среде Linux

У каждого из этих способов есть свои преимущества и недостатки. Для новичков NodeMCU ESP8266, а также разработчиков прикладных приложений наиболее удобным будет первый способ. По двум остальным способам более подробно можно почитать тут https://nodemcu.readthedocs.io/en/master/en/build/

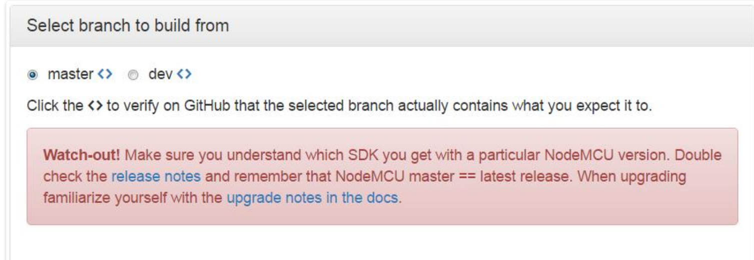
Как собрать прошивку NodeMCU ESP8266 в облачном сервисе

Сам облачный сервис расположен здесь https://nodemcu-build.com/, обладает простым и понятным интерфейсом. Создание прошивки NodeMCU ESP8266 происходит непосредственно на главной странице и начинается с указания email (вашего email) на который в дальнейшем придет два письма – первое с сообщением о начале компиляции прошивки NodeMCU ESP8266 и второе со ссылкой для скачивания самой прошивки. Для исключения ошибки email необходимо ввести дважды (*копировать/вставить не работает, придется вводить руками*)



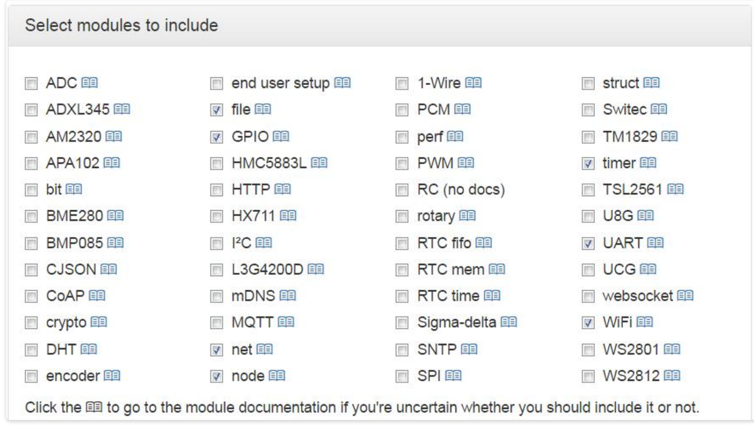
Облачный сервис компилирования прошивки NodeMCU ESP8266 – вводим email

Далее предлагается выбрать ветвь прошивки – *master* (стабильная прошивка) или *dev* (тестируемая прошивка). Для обучения и основной массы проектов рекомендуется выбрать *master*.



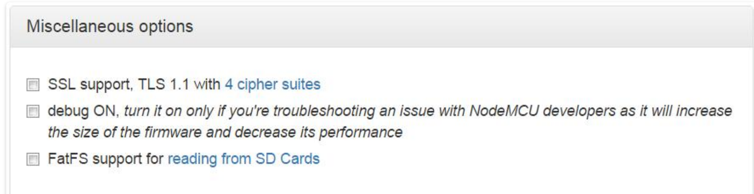
Облачный сервис компилирования прошивки NodeMCU ESP8266 – выбираем master

Затем необходимо выбрать модули, которые будут включены в нашу прошивку. Около каждого модуля есть пиктограмма в виде открытой книжки, при нажатии на которую будет доступна документация с описанием данного модуля. По-умолчанию в прошивку включается несколько модулей (*file, gpio, net, node, timer, uart, wifi*) которые обеспечивают базовый функционал прошивки NodeMCU ESP8266 для начала работы с ней (не рекомендуется их удалять). Остальные модули нужно включать в прошивку только в случае их действительной необходимости, в противном случае размер прошивки может оказаться слишком большим, что приведет к невозможности ее загрузки или выполнения скриптов.



Облачный сервис компилирования прошивки NodeMCU ESP8266 – выбираем необходимые модули

Кроме того в следующем блоке можно выбрать дополнительные опции, такие как поддержка SSL, включение режима отладки или поддержки FatFS для чтения с SD-карт.

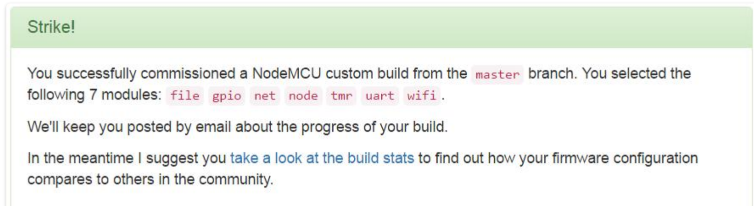


Облачный сервис компилирования прошивки NodeMCU ESP8266 – выбираем дополнительные опции

В данном случае все настройки были оставлены по-умолчанию, после чего можно нажать кнопку *Start your build* в самом низу экрана.

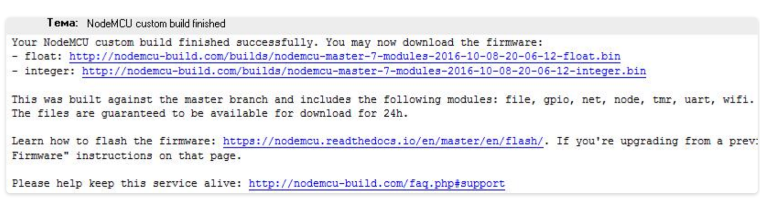
Облачный сервис компилирования прошивки NodeMCU ESP8266 – запускаем сборку прошивки

И получаем окно с сообщением о начале сборки нашей прошивки NodeMCU ESP8266



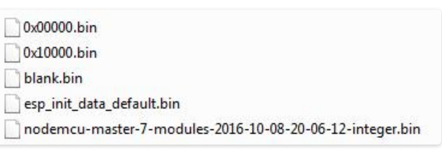
Облачный сервис компилирования прошивки NodeMCU ESP8266 – сборка прошивки началась

Через некоторое время придет первое письмо с сообщением о запуске процесса сборки прошивки NodeMCU ESP8266, а еще через какое-то время второе письмо с ссылками на две версии прошивки – *float* и *integer*. Первая позволяет работать с дробными числами, а вторая с только с целочисленными (*но и размер у нее меньше*). Для обучающих занятий нас вполне устроит вариант *integer*.



Облачный сервис компилирования прошивки NodeMCU ESP8266 – ссылки на скачивание прошивки

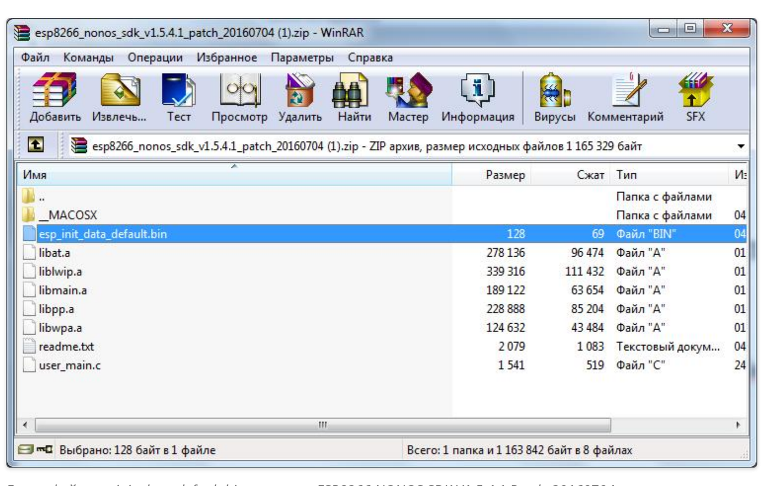
Перейдем по соответствующей ссылке и скачаем *bin* файл, после чего поместим его в NodeMCU Flasher по следующему пути *Resources – Binaries*, а находящийся там файл *nodemcu_integer_0.9.5_20150318.bin* удалим, во избежание путаницы. Таким образом, получим следующую картину



Содержимое папки Binaries прошивальщика NodeMCU Flasher

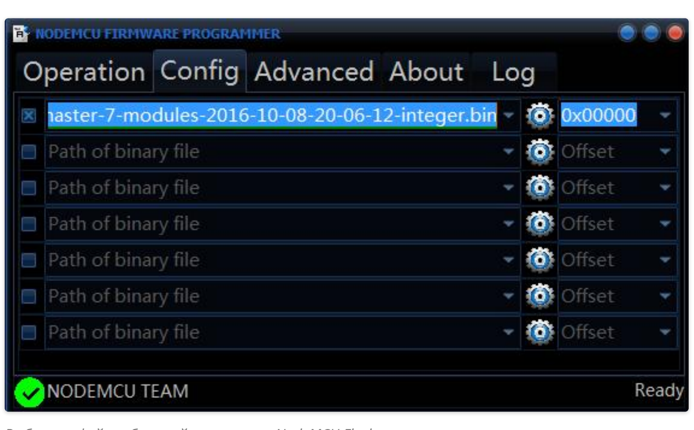
Как обновить прошивку NodeMCU ESP8266 на последнюю версию

С учетом того, что последняя версия прошивки NodeMCU ESP8266 реализована на SDK v1.5.4.1, то для корректной работы отладочной платы NodeMCU ESP8266 и для перехода на новую версию SDK нам необходимо дополнительно перезаписать файл *esp_init_data_default.bin*. Новый *esp_init_data_default.bin* возьмем из патча расположенного на официальном сайте https://espressif.com/sites/default/files/sdks/esp8266_nonos_sdk_v1.5.4.1_patch_20160704.zip
Открыв этот архив увидим нужный нам файл *esp_init_data_default.bin* который и поместим в прошивальщик NodeMCU Flasher по пути *Resources – Binaries* (*старый esp_init_data_default.bin лучше сразу удалить*).



Берем файл esp_init_data_default.bin из патча ESP8266 NONOS SDK V1.5.4.1 Patch_20160704

Теперь можно подключить отладочную плату NodeMCU ESP8266 и переходить к обновлению её прошивки, но в отличие от предыдущего раза нам надо будет сделать кое-какие настройки. Для этого в NodeMCU Flasher перейдем на закладку *Config* и вместо *INTERNAL://NODEMCU* выберем файл собранной нами прошивки (*для выбора файла нажимаем на шестеренку справа*).

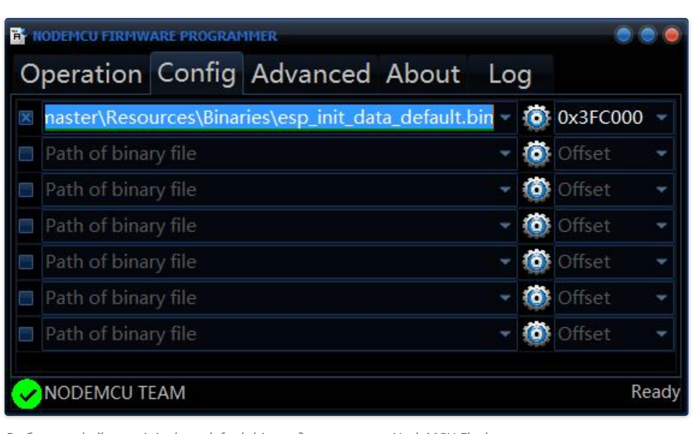


Выбираем файл собранной прошивки в NodeMCU Flasher

Все остальное оставляем без изменений, возвращаемся на вкладку *Operations* и нажимаем *Flash(F)*. После окончания прошивки снова переключаемся на закладку *Config* и указываем в первой строке вместо файла прошивки путь с файлом *esp_init_data_default.bin*. Кроме того необходимо дополнительно указать адрес куда будет помещен этот файл. Существует несколько фиксированных вариантов адресов в зависимости от размера памяти модуля ESP:

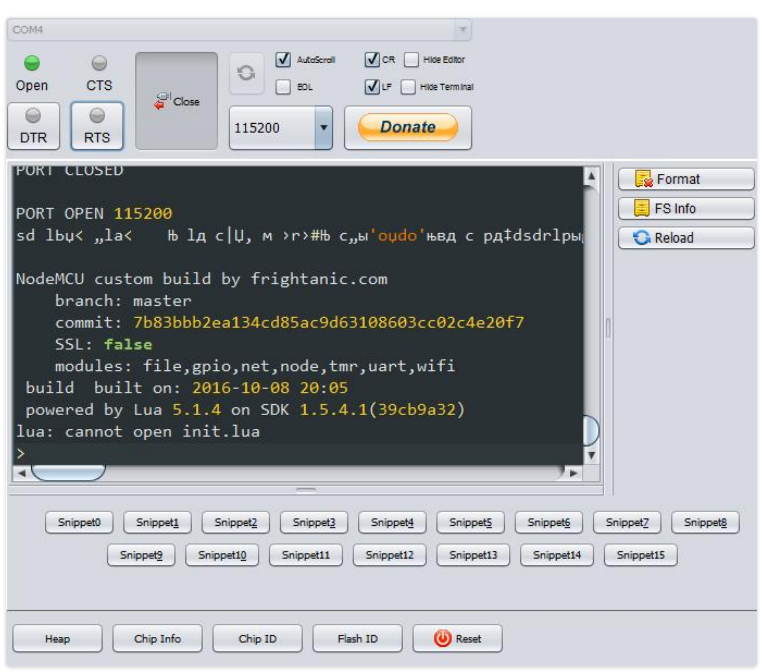
- 0x7C000 для 512 кБ (*модули типа ESP-01, -03, -07 и т.п.*)
- 0xFC000 для 1 МБ (*модули типа ESP8285, PSF-A85*)
- 0x1FC000 для 2 МБ
- 0x3FC000 для 4 МБ (*модули типа ESP-12E, NodeMCU, WeMos D1 mini*)

В нашем случае для отладочной платы NodeMCU LoLin v.3 ESP8266 выбираем последний вариант. Все остальное оставляем без изменений, возвращаемся на вкладку *Operations* и нажимаем *Flash(F)*.



Выбираем файл esp_init_data_default.bin и адрес записи в NodeMCU Flasher

Далее необходимо переформатировать файловую систему отладочной платы NodeMCU ESP8266, в частности из-за изменений размеров прошивки по сравнению со старой версией. Для этого запускаем ESPlorer, **ВАЖНО ! выставляем скорость обмена 115200 !** После перезагрузки отладочной платы NodeMCU ESP8266 должен автоматически запуститься процесс форматирования, а в окне сообщений отобразиться *formatting file system*. Если процесс форматирования не запустился автоматически то вводим команду *file.format()* вручную. При этом все скрипты ранее находящиеся в отладочной платы NodeMCU ESP8266 будут удалены.
После выполнения всех действий в ESPlorer можно будет увидеть новую версию прошивки NodeMCU.



Прошивка NodeMCU powered by Lua 5.1.4 on SDK 1.5.4.1

Для окончательной проверки нормального функционирования отладочной платы NodeMCU ESP8266 можно залить ранее проработанный скрипт мигания светодиодом с именем *init.lua*.

Теперь можно считать, что наша **отладочная плата NodeMCU ESP8266 полностью готова к работе** и мы можем переходить к изучению основных возможностей, которые она предоставляет. На следующем занятии мы узнаем – Как подключить NodeMCU ESP8266 к WiFi

Поиск

СВЕЖИЕ ЗАПИСИ

- Как BURST кладет на лопатки BITCOIN
- Международный симпозиум и выставка по вопросам электромагнитной совместимости EMC Europe 2017
- Международный симпозиум по электромагнитной совместимости, целостности сигнала и мощности EMC+SIPI 2017
- Симпозиум по электромагнетизму в области высоких мощностей ASIAEM 2017
- Международный симпозиум по электромагнитной совместимости APEMC 2017

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

Electromagnetic compatibility laboratory © Лаборатория электромагнитной совместимости