

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/332381105>

ПРОГРАММИРОВАНИЕ ESC РЕГУЛЯТОРОВ ПРОШИВКАМИ SIMONK И BLHEL1 ЧЕРЕЗ ARDUINO И ПОЛЕТНЫЙ КОНТРОЛЛЕР

Article · April 2019

CITATIONS

0

READS

3,067

1 author:



Alexandr Myasischev

Khmelnytskyi national university

12 PUBLICATIONS 1 CITATION

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Полетные контроллеры летающих роботов. Удаленное управление оборудованием через TCP/IP сеть [View project](#)

ПРОГРАММИРОВАНИЕ ESC РЕГУЛЯТОРОВ ПРОШИВКАМИ SIMONK И BLHELI ЧЕРЕЗ ARDUINO И ПОЛЕТНЫЙ КОНТРОЛЛЕР

1. Регуляторы Hobbypower ESC-30A, Readytosky ESC-30A, Simonk-30A

Обычные регуляторы имеют внутри программный и аппаратный фильтры, сглаживающий реакцию на резкие скачки управляющего воздействия. Принцип стабилизации мультироторных аппаратов основан на изменении тяги каждого ротора в отдельности. Чем более отзывчив будет регулятор, тем лучше будет стабилизация, меньше колебания, и т.п.

Минусом "родных" прошивок регуляторов является низкое разрешение. Т.е. регулятор весь диапазон оборотов проходит, например, за 100 шагов. Кастомные прошивки обеспечивают большее разрешение. После перепрошивки регулятор теряет большую часть своих функций. Т.е. невозможно настроить отсечку, ограничитель напряжения батареи и т.д. Этих функций просто нет. Прошитые регуляторы предназначены исключительно для работы на мультироторах. Исключением можно считать прошивку BLHeli.

Также замечено, что у регуляторов Hobbypower ESC-30A, Readytosky ESC-30A с собственными прошивками наблюдаются срывы синхронизации при оборотах ниже средних. Поэтому при резком сбросе оборотов возможна остановка двигателя, что приведет к аварии коптера. Прошивки китайских регуляторов Simonk 30A не всегда имеют последнюю версию.

Следует отметить ненадежную работу регуляторов Hobbypower ESC-30A и Readytosky ESC-30A китайского производителя, которые способны самопроизвольно сбрасывать настройки после программирования его через пульт управления БПЛА, а также являются очень ненадежными в случае запуска при температуре ниже +15 град. Поэтому при подключении батареи необходимо внимательно прослушивать мелодии ESC о его готовности и на основании этого делать вывод о возможности запуска БПЛА.

Перечисленные выше замечания требуют выполнения перепрошивки перечисленных выше регуляторов. Опыт показал, что лучшей прошивкой для них при использовании преимущественно в квадрокоптере является прошивка Simonk. На рисунке 1. представлены эти регуляторы, которые собраны по однотипным схемам.



Рис.1. Перепрошиваемые регуляторы

На рисунке 2 представлены регуляторы Hobbypower ESC-30A со снятой термоусадкой. Их отличие от

Readytosky ESC-30A и Simonk 30A в том, что у последних регуляторов присутствуют площадки для MOSI, MISO, SCK, RESET, GND, которые используются для подключения программатора.

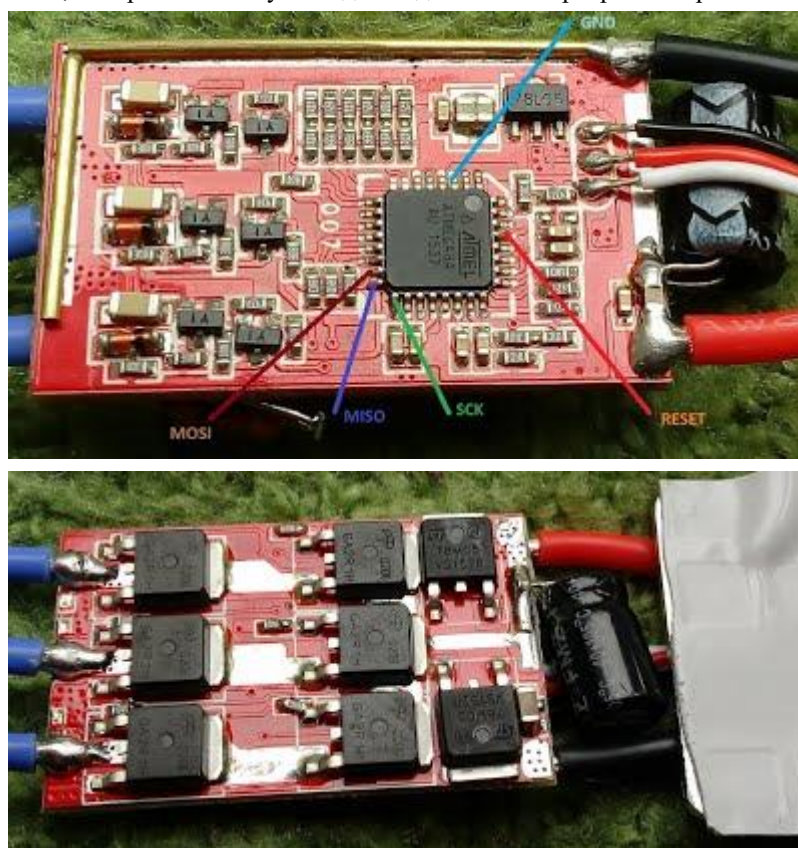


Рис.2. Регулятор Hobbypower ESC-30A с снятой термоусадкой

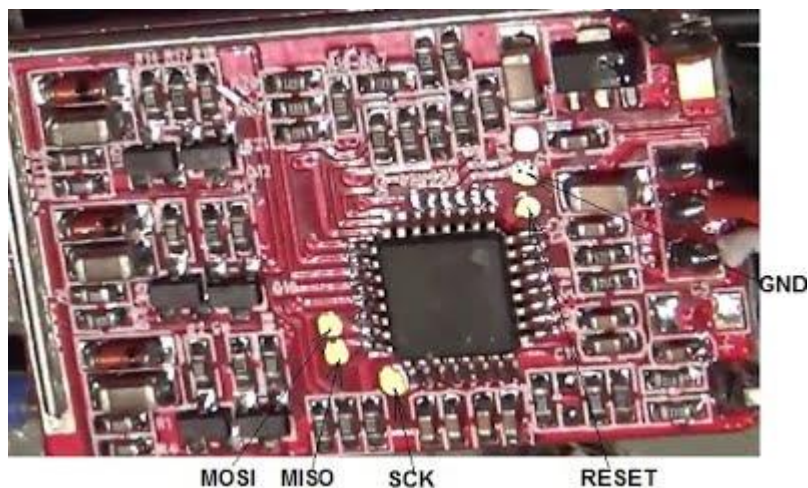


Рис.3. Регуляторы Readytosky ESC-30A и Simonk 30A

В рекламе продажи регулятора Simonk 30A было отмечено наличие кварцевого генератора на повышение в частности температурной стабильности его работы. Однако при разборке не выявлено наличия внешнего кварцевого резонатора (рис.3).

Для перепрошивки регуляторов необходимо скопировать с сайта <http://0x.ca/tgy/downloads/> файл tgy_2015-09-12_103edb5.zip и распаковать его в каталоге. Найти среди распакованных файлов tgy.hex и с помощью программатора и адаптера переслать его в микроконтроллер ATmega8 регулятора.

В качестве программатора воспользуемся программой PonyProg2000, которая должна быть предварительно установлена на компьютере со свободным COM портом. Схема адаптера для работы через

COM порт представлена на рисунке 4. Выводы адаптера подключаются к выводам, припаянным к соответствующим контактам, как на рис.2,3. Пайка должна быть особенно для Hobbypower ESC-30A аккуратной, чтобы не повредить микроконтроллер.

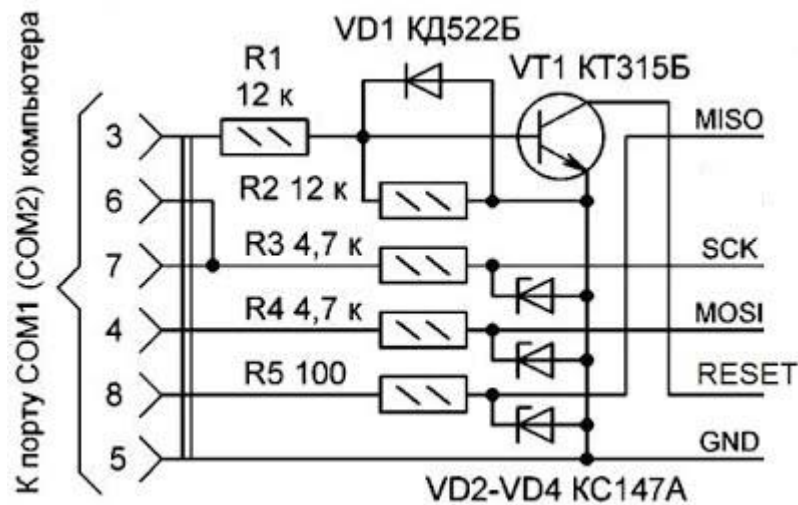
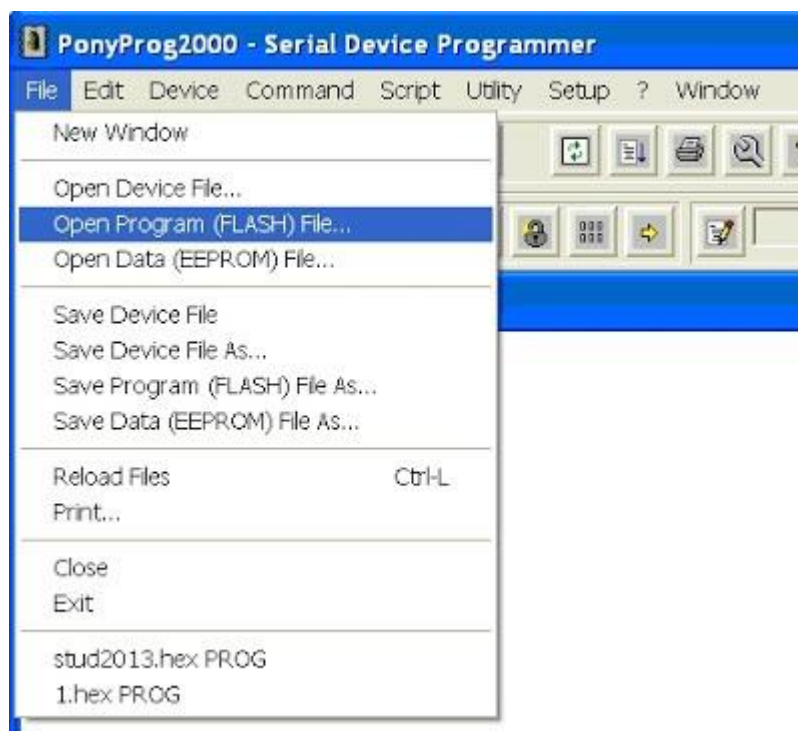


Рис.4. Схема адаптера для COM порта

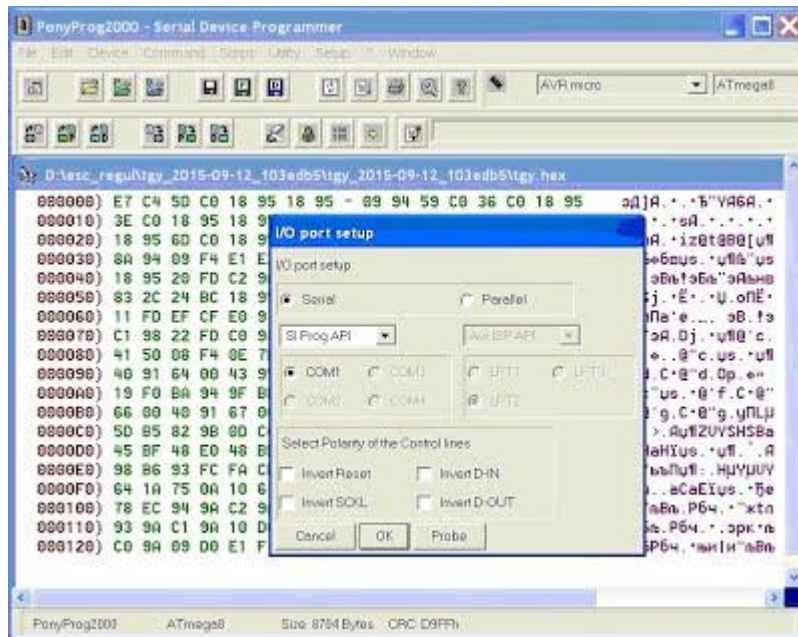
После установки PonyProg необходимо выполнить процедуру ее калибровки. Для этого необходимо зайти в закладки Setup->Calibration и нажать на Yes.

Рассмотрим следующие этапы использования программы PonyProg.

1. Запускаем программу и открываем файл с расширением .hex. В данном случае это файл tgy.hex.



2. После установки PonyProg необходимо выполнить процедуру ее калибровки. Для этого необходимо зайти в закладки Setup->Calibration и нажать на Yes.
3. Заходим во вкладки Device -> AVR micro и выбираем ATmega8
4. Настраиваем интерфейс связи между компьютером и микроконтроллером:



5. После загрузки файла tgy.hex можно зайти в закладку Command->Security and Configuration Bits и просмотреть фьюз-биты (биты конфигурации) для ATmega8.



Например, для Readytosky будут установлены следующие биты (стоят галочки): SPIEN, BOOTSZ1, BOOTSZ0, SUT0, CKSEL3, CKSEL1, CKSEL0. Для Hobbypower ESC-30A: SPIEN, BODEN, SUT0, CKSEL3, CKSEL1, CKSEL0. При загрузке прошивки tgy.hex фьюз-биты менять не стоит. Эти биты установлены при изготовлении регуляторов.



Рис.5. Установка кварцевого резонатора 16 МГц

На рисунке 5 показана модернизация регуляторов Readytosky ESC-30A и Simonk 30A путем добавления кварцевого резонатора на 16МГц. Он был подпаян к выводам микроконтроллера, как показано на рис.6. В этом случае перед прошивкой необходима замена фьюз-битов. Необходимо проверить установку SPIEN(хотя он всегда установлен и в программе PonyProg200 он не переустанавливается), а все остальные биты сбросить и перепрошить микроконтроллер той же прошивкой. Опыт показал, что добавление кварцевого резонатора при эксплуатации в диапазоне температур -8,...+25 град. при установке на квадрокоптер 2-х регуляторов с кварцем и 2-х без кварца не привели к заметным изменениям. Двигатели работали одинаково с регуляторами с внешним кварцем и без него т.е. на внутреннем генераторе из RC-цепочки.

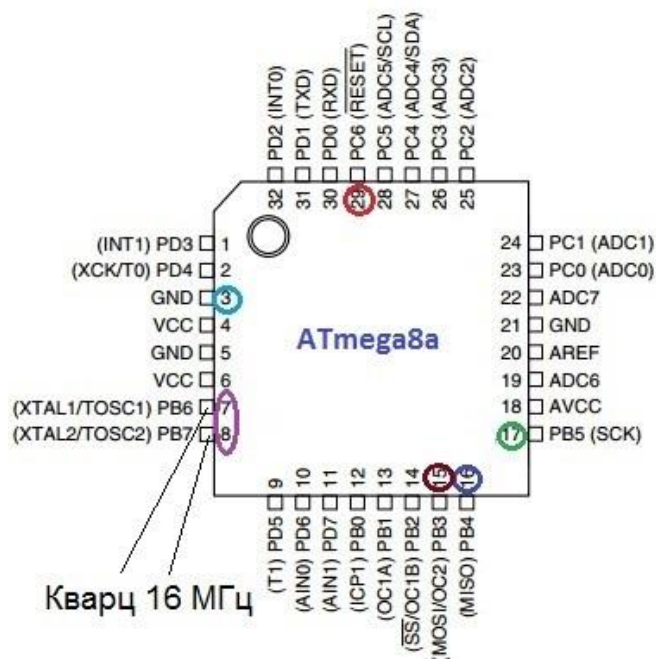


Рис.6. ATmega8, подключение

На рисунке 7 представлено фото ESC регулятора с установленным микроконтроллером ATmega8 в DIP28 корпусе с внешним кварцевым резонатором в связи с выходом из строя ATmega8 в корпусе TQFP32. Такая переделка не вызвала изменений в работе регулятора.

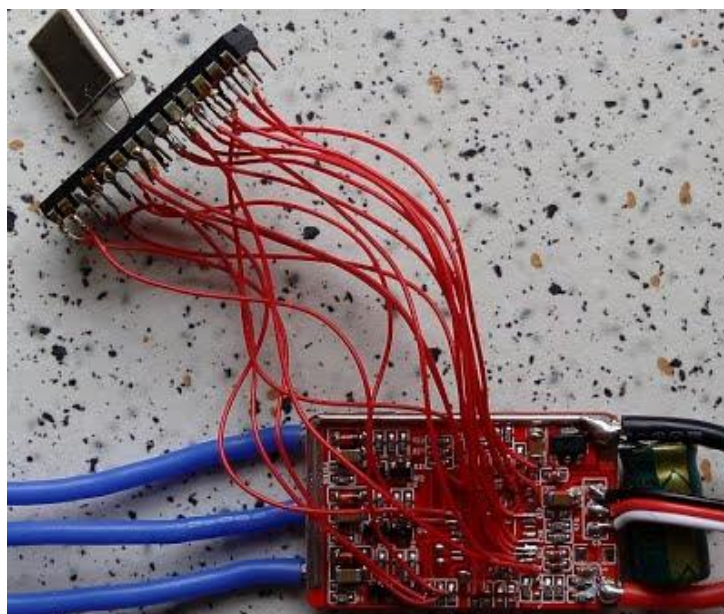
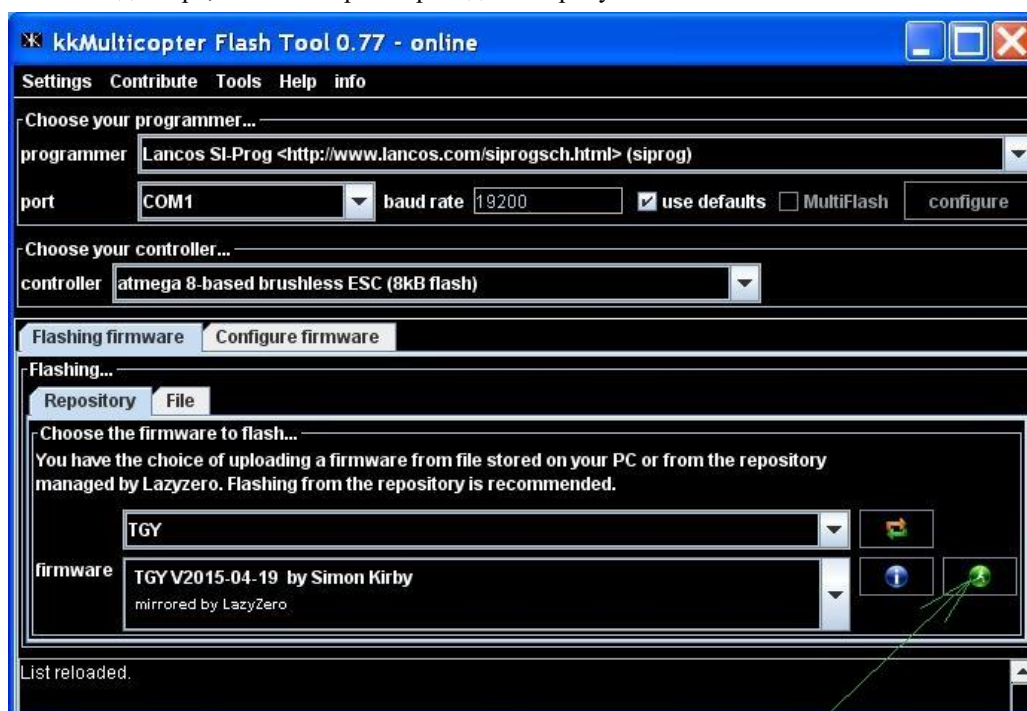


Рис.7. ESC регулятор с ATmega8 в DIP28 корпусе

6. Выполняем передачу файла tgy.hex микроконтроллеру по команде Ctrl/P. Или можно зайти во вкладку Command и курсором мышки выбрать Program. В случае успешной передачи программы в микроконтроллер должно появиться окно с надписью Program successful. При программировании необходимо не забыть на регулятор подать напряжение 7,...12В. Если компьютер для программирования микроконтроллера не имеет COM порта, но имеет USB порты, необходимо использовать более сложные программаторы.

Можно выполнить прошивку с помощью программы KKmulticopter Flash Tool, но она не устанавливает фьюз-биты. Программу можно скопировать с ссылки <http://lazyzero.de/en/modellbau/kkmulticopterflashtool>. Предварительно на компьютере должна быть установлена Java. Установку прошивки можно выполнить с репозитория или с файла. На рисунке 8 представлены выбранные опции для загрузки прошивки с репозитория при использовании адаптера, схема которого приведена на рисунке 4.



Нажать при прошивке

Рис.8. Прошивка программой kkMulticopter Flash Tool

3. Регуляторы SIMONK-30A И EMAX SIMON-12A

Известно, что ESC это Electronic Speed Controller, который необходим для контроля скорости двигателей. ESC получает сигнал дроссельной заслонки аппаратуры управления от контроллера полета и приводит в движение бесщеточный двигатель с оптимальной скоростью, обеспечивая необходимый уровень электрической мощности. Качественные ESC с качественной прошивкой обеспечивают надежный и бесперебойный полет. Большинство ESC поставляются с программным обеспечением (прошивкой) производителя. Однако такие прошивки не всегда оптимальны для этого ESC. Часто на ESC устанавливается прошивки с открытым исходным кодом SimonK и BLHeli, однако версии этих прошивок являются устаревшими и поддерживают старые протоколы связи полетного контроллера и ESC. Например, устаревший протокол PWM. Бывают случаи, когда ESC поставляется с прошивкой BLHeli, не рекомендованной для этого ESC. Например регуляторы EMAX Simon-12A приходят с прошивками EMAX20A Rev.-14.3.-Multi. Хотя рекомендацией BLHeli для EMAX Simon-12A является прошивка XP-12A Rev.-14.9.-Multi. Также видно, что и прошивка Rev.-14.3. является устаревшей. Аналогичная ситуация с ESC SimonK 30A, которые используют устаревшую прошивку Simonk, не поддерживающую протокол OneShot125. Следует отметить кроме SimonK и BLHeli также и новые прошивки:

-BLHeli_S является вторым поколением прошивки BLHeli, разработанной специально для ESC с процессорами Busybee с аппаратным PWM. Прошивки Simonk и BLHeli формируют программный PWM.

-BLHeli_32 для ESC является третьим и последним поколением BLHeli. Прошивка написана специально для 32-битных процессоров ESC и ее программный код был закрыт. Эти ESC с более мощными процессорами обеспечивают более плавную, точную и надежную работу, чем предыдущие ESC.

Прошивка SimonK начиная с 2016 года не поддерживается. Ее последняя версия была выпущена 12-09-2015г. Перечисленные выше замечания требуют выполнения перепрошивки перечисленных выше регуляторов.

Протокол ESC используется для связи между FC (контроллером полета) и ESC, в основном для того, чтобы контролировать скорость вращения двигателя. Первоначально протокол был основан на широтно-импульсной модуляции, имеющей аббревиатуру ШИМ или PWM. Это способ получения аналогового результата цифровыми средствами. Цифровые средства здесь используются для получения импульсов квадратной формы. Сигнал переключается между двумя состояниями, On и Off. В случае прошивки полетного контроллера прошивкой LibrePilot, сигнал PWM используется для указания ESC положения позиции стика Газа на пульте управления. На рисунке 1 показан стандартный сигнал PWM с частотой следования импульсов 490Гц (с периодом ~2000 мкс).

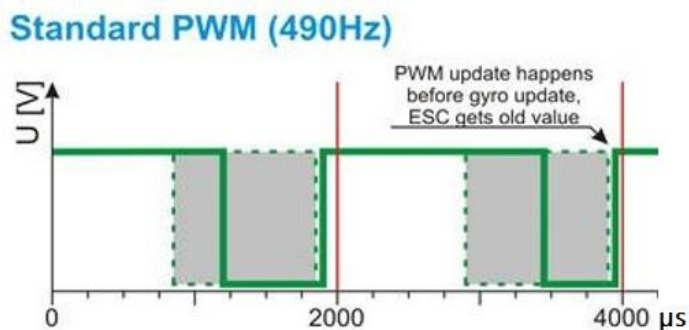


Рис.1. Схема Standard PWM - протокол управления ESC.

Для ESC, используемых в системах радиоуправления(RC), длина импульса стандартного RC - PWM сигнала используется в процентах. Длина импульса в 1000мкс соответствует 0% , а 2000мкс - 100% частоте вращения мотора. Например, если ESC должен обеспечить 50% частоты вращения мотора, он пошлет сигнал PWM длительностью 1500мкс. Следует отметить, что PWM не синхронизирует PWM сигнал, который контроллер полета посылает ESC с работой гироскопа летательного аппарата. Поэтому обновление PWM происходит до обновления сигнала с гироскопа и ESC получает старый сигнал от контроллера полета.

OneShot125 является развитием стандартного сигнала RC PWM. OneShot125 состоит из двух частей:

-Первая часть OneShot125 синхронизирует импульсы RC (PWM сигнал, который контроллер полета посылает ESC) с выходом сигнала с гироскопа. Это позволяет ESC иметь самые актуальные последние данные из полетного контроллера без больших задержек;

-Вторая часть OneShot125, это передать на ESC самые свежие актуальные данные без задержек.

Стандартный PWM импульс находится в промежутке между 1000 и 2000 микросекундами. OneShot125 короче PWM импульса и находится в диапазоне между 125 и 250 мкс. Это означает, что ESC получает полный сигнал PWM в 8 раз быстрее. Для ESC важно иметь самые свежие данные. Полетный контроллер постоянно меняет данные для ESC на основании данных, полученных с гироскопов. Свежие актуальные без задержек данные позволяют ESC получить необходимые значения газа быстрее. Это позволяет осуществлять более точное управление, устанавливать более высокие значения PID и добиться большей стабильности особенно для гоночных квадрокоптеров.

На рисунке 2 показан стандартный PWM - с частотой обновления 490Hz при ширине импульса 1000-2000мкс. Этот PWM не синхронизируется с выходом гироскопа. В настоящее время все ESC работают с этим протоколом. PWM импульс выдается на частоте обновления равной 490Hz.

Standard PWM (490Hz)

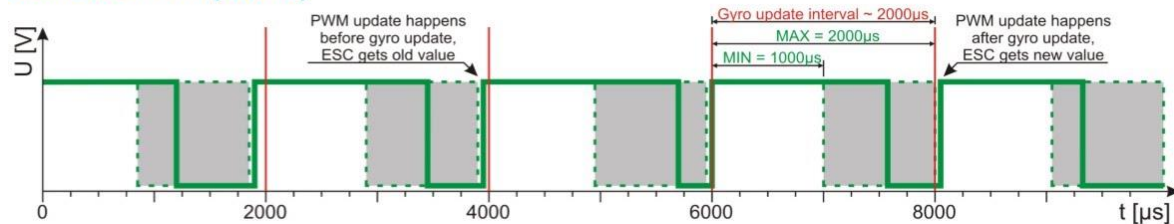


Рис.2. Стандартный PWM для Multicopter

На рисунке 3 показан OneShot125 с частотой обновления 500Hz и шириной импульса 125-250μs. Для его работы требуется ESC с поддержкой OneShot125. Свободно распространяемые прошивки BLHeli версии 13.0+, SimonK 2015-02-26 и новее поддерживают OneShot125, если их можно установить на ESC.

OneShot125 (500Hz synced to gyro)

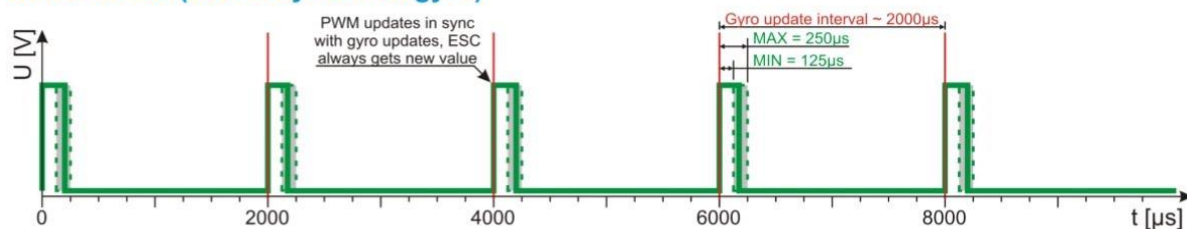


Рис.3. OneShot125 для Multicopter

Таким образом:

- Oneshot125 устанавливает скорость таймера на выходе двигателя в 8 раз быстрее. Выходной сигнал двигателя теперь намного короче (125-250 мкс), чем исходный PWM (1000 мкс-2000 мкс), это означает, что для отправки данных в ESC требуется меньше времени. Теперь проблема, связанная с обновлением сигнала PWM с опозданием устранена.
- Oneshot125 посылает только «один выстрел» сигнала в каждом цикле контроллера полета и делает это, как только у контроллера полета появляется новое значение для ESC.
- ESC теперь записывает новое значение в двигатель, даже до того, как FC запускает следующий цикл управления, поэтому имеет место более быстрый отклик FC и ESC. В связи с этим значение P для PID регулятора может быть немного увеличено. Отмечается более быстрым изменением скорости двигателя в соответствии с реакцией газа на пульте управления. Это важно для гоночных дронов, огибающих препятствия.

Ниже перечислены все протоколы ESC, доступные для мультикоптера, и соответствующая им ширина сигнала - время, необходимое для отправки одного пакета данных:

-Стандартный PWM (1000мкс - 2000мкс)	-Dshot150 (106,8мкс)
-Oneshot125 (125мкс - 250мкс)	-Dshot300 (53,4мкс)
-Oneshot42 (42мкс - 84мкс)	-Dshot600 (26,7мкс)
-Multishot (5мкс - 25мкс)	-DShot1200 (13,4мкс)

Протоколы Oneshot125, Oneshot42 и Multishot представляют собой аналоговые сигналы, похожие на стандартный PWM, но намного быстрее. Эти протоколы синхронизируются с циклом PID-регулятора, чтобы уменьшить дрожание, повысить производительность и уменьшить задержку между вводом данных со стика аппаратуры управления и реакцией корабля.

DShot - это новейший протокол ESC, представляющий собой цифровой сигнал. Это будущее протокола ESC, который имеет лучшую надежность и производительность, а также способности посылать на ESC не только параметры скорости двигателя, но и конкретные команды.

Аппаратное обеспечение ESC регуляторов и полетных контроллеров, выпущенное после середины 2017 года, должно поддерживать все протоколы ESC, кроме DShot1200, который является относительно новым и совместим только с некоторыми 32-разрядными ESC.

Рассмотрим аппаратную часть ESC, основой которой является микроконтроллер. Существует три основных семейства микроконтроллеров (MCU) - это Atmel, Silabs и ARM Cortex. Различные MCU имеют разные спецификации и функции, которые позволяют запускать разные прошивки.

- 8-битный ATMEL поддерживается прошивками SimonK и BLHeli ESC
- SILABS 8-битный может работать только BLHeli или BLHeli_S
- Atmel ARM Cortex 32-битный (точнее STM32 F0) - может работать BLHeli_32

ESC на основе ATMEL были более распространены до того, как на рынке начали доминировать SILABS. ESC Silabs, как правило, превосходят 8-битные ATMEL по производительности. В 2017 году Atmel Arm Core MCU стал использоваться для 32-битных ESC.

В ESC на основе SiLabs установлены разные процессоры, которые обеспечивают разную производительность. Наиболее используемые настоящее время - F330 и F39X. MCU F330 имеет более низкую тактовую частоту и может иметь проблемы с работой двигателей с высоким KV. F39X не имеет этих проблем и поддерживает протоколы Multishot и Oneshot42 для ESC.

Busybee MCU - это обновленные микроконтроллеры F330 и F39X. Прошивка BLHeli_S для ESC обычно работает на BusyBee1 - BB1 (EFM8BB10F8) и BusyBee2 - BB2 (EFM8BB21F16). Преимущество BusyBee в том, что вместо использования программного PWM, у них есть специальное оборудование для аппаратной генерации PWM-сигнала, который синхронизируется с рабочим циклом MCU, в результате получается гораздо более плавный отклик на газ аппаратуры управления. Эти микроконтроллеры поддерживают новейшие протоколы DShot ESC.

Оценивая производительность микроконтроллеров, можно записать ряд от лучшего к худшему:
1.BB2, 2.BB1, 3.F330, 4.F396, 5.Atmel 8-разрядный.

В настоящее время большинство ESC используют 8-разрядные процессоры (F330, F39X, Busybee и т. д.). Но с 2016 года начали появляться некоторые 32-разрядные ESC на основе STM32. Эти мощные 32-разрядные процессоры имеют много новых функций, которые были невозможны из-за ограниченной вычислительной мощности и возможностей 8-разрядного MCU.

Рассмотрим обновление прошивки у ESC EMAX-12A. Здесь может быть два варианта. Прошивка с помощью 3-х проводов (интерфейс C2) и по одному сигнальному проводу (требуется загрузчик), который использует ESC для подключения к полетному контроллеру. В первом варианте прошивки для указанных двух случаев в качестве адаптера связи компьютера с ESC используется Arduino Nano v3.0 с преобразователем USB-UART CH340. Во втором варианте прошивка выполняется через полетный контроллер с прошивкой cleanflight.

Для выполнения прошивки вначале необходимо загрузить на компьютер программное обеспечение с прошивками для различных ESC с сайта <https://github.com/bitdump/BLHeli> (BLHeli-master.zip) и программу для выполнения прошивки с <https://www.mediafire.com/folder/dx6kfaasyo24l/BLHeliSuite> (BLHeliSuite16714901.zip).

Рассмотрим прошивку ESC по интерфейсу C2. В этом случае выводы на мотор изолируются и должна быть удалена термоусадка с ESC для напайки трех проводников к контактным площадкам ESC. На рисунке 3 показано подключение Arduino Nano к ESC.

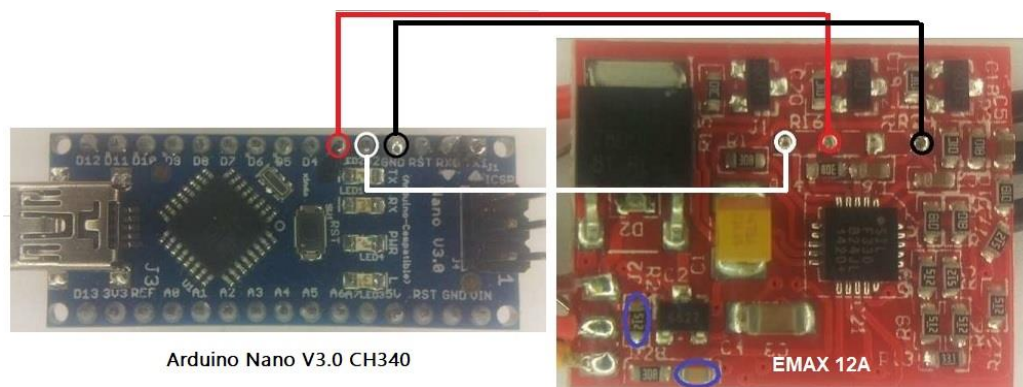


Рис.3. Подключение Arduino Nano к ESC EMAX 12A

Перед прошивкой ESC в Arduino Nano должна быть загружена программа, которая позволит Arduino выполнить процесс программирования. Для этого подключается Arduino к компьютеру через USB кабель, запускается программа BLHeliSuite.exe, выполняется переход на вкладку Make interfaces, выбирается порт и интерфейс (рис.4).

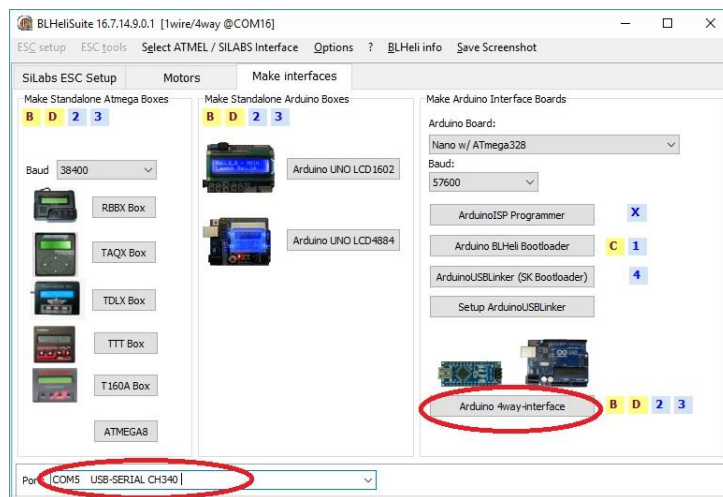


Рис.4. Выбор интерфейса

На рис.2 выбрана плата Arduino Nano со скоростью порта 57600. После нажатия на кнопку Arduino 4way-interface необходимо будет выбрать файл загрузки, как на рисунке 5.

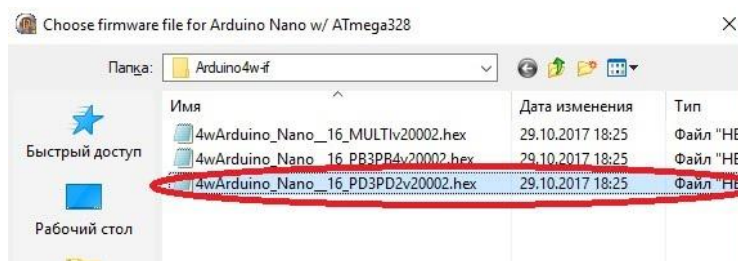


Рис.5. Выбор файла загрузки

После выбора файла выполняется программирование Arduino. При успешной загрузке файла будет выдано соответствующее сообщение. Для загрузки прошивки необходимо перейти во вкладку Silabs ESC Setup и во вкладке Select ATMEL/SILABS Interface выбрать SILABS C2(4way-if). После этого к ESC необходимо подключить батарею и нажать на кнопку Connect.

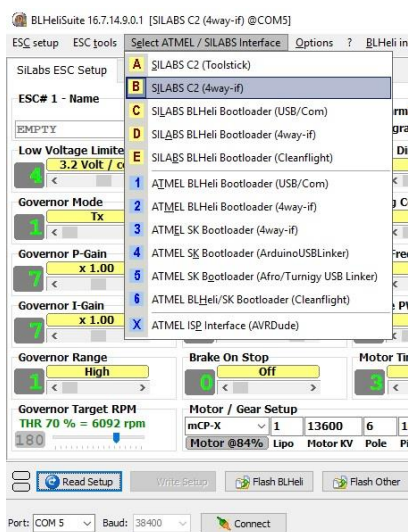


Рис.6. Выбор интерфейса и загрузка прошивки

После успешного подключения к ESC необходимо нажать на кнопку Flash BLHeli. Далее появится окно для выбора файла прошивки. Для EMAX 12A, установленного на квадрокоптере, выбирается файл XP-12A Multi Rev.14.9. После этого выполняется прошивка и результат ее выводится на дисплей. После завершения процедуры и нажатия на кнопку Read Setup выводится обновленный экран с параметрами этой прошивки как на рисунке 7.

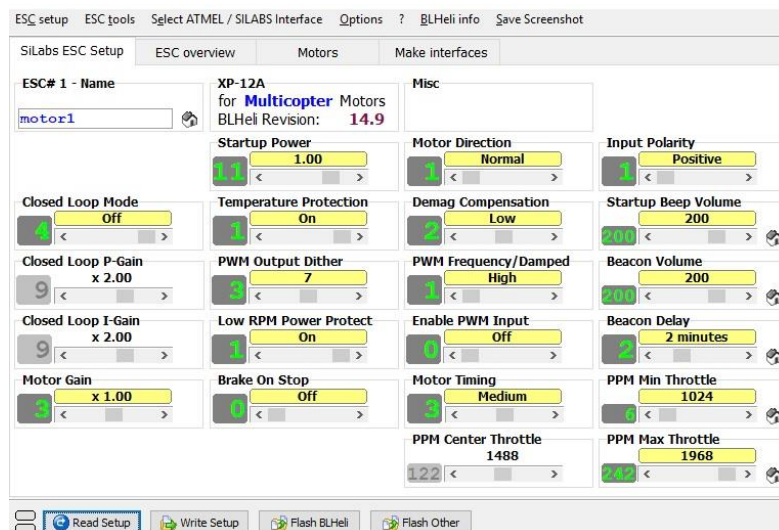


Рис.7. Параметры прошивки

Параметры прошивки можно менять. Для их сохранения необходимо нажать на кнопку Write Setup. Если не отпаивать провода от контактных площадок, то даже после установки ESC на квадрокоптер их можно использовать для изменения параметров или перепрошивки ESC через Arduino Nano. Однако мотор при этих действиях рекомендуется отключать.

Ниже перечислены некоторые параметры с их расшифровкой.

Closed Loop Mode - Режим замкнутого контура - устанавливает диапазон скоростей, с которыми может работать контур управления.

- Для High значения дроссельной заслонки от 0% до 100% линейно соответствуют заданным значениям оборотов т.е. от 0 до 200000 электрических оборотов в минуту (для 2-х полюсного мотора).
- Для Middle диапазона значения дроссельной заслонки от 0% до 100% линейно соответствуют заданным значениям оборотов от 0 до 100000 электрических оборотов в минуту.
- Для Low диапазона значения дроссельной заслонки от 0% до 100% линейно соответствуют заданным значениям оборотов от 0 до 50000 электрических оборотов в минуту.

Когда Closed Loop Mode Off, контур управления отключен.

Таким образом с помощью этого параметра ESC пытается управлять крутящим моментом в зависимости от текущих оборотов. Если установить HiRange, это соответствует 200 тыс. оборотов в минуту при полном открытии дроссельной заслонки. Например, если мотор 1000KV, батарея 4S то $4,2\text{Вольт} \times 4S \times 1000KV = 16.8$ тыс. оборотов в минуту на полном газу на заряженных аккумуляторах. То есть нужно установить или HiRange или вовсе выключить. По умолчанию режим выключен. В этом случае при максимальной дроссельной заслонке мотор работает с максимальными оборотами для заданного напряжения.

Motor Gain - ограничивает максимальную мощность в зависимости от входного сигнала.

Startup Power - мощность импульса при старте мотора.

Temperature Protection - защита от перегрева.

PWM Output Dither - это параметр, который добавляет некоторое изменение к длительности цикла PWM отключения двигателя. Это может уменьшить проблемы (например, дискретность сигнала дроссельной заслонки или вибрации) в областях частот вращения, где частота PWM равна гармоникам частоты коммутации двигателя. Полезно в режиме активного торможения (DampedLight).

Low RPM Power Protect - если включить этот параметр, то ESC отключит мотор, если пропеллеры за что-то зацепились (мотор увяз при низких оборотах). Эта исключит пожар квадрокоптера.

Brake On Stop - при включении этого параметра выполняется резкий останов пропеллера при выключении мотора пультом управления.

Motor Direction - направление вращения мотора. Если мотор вращается не в ту сторону, необходимо изменить этот параметр.

Demag Compensation - защита от срыва синхронизации. Для небольших двигателей параметр не актуален.

PWM Frequency - частота входящего PWM-сигнала. Значение Low - около 8кГц, High - около 20кГц.

PWM Frequency - Damped и Damped Light - режимы торможения двигателя. Не все регуляторы имеют поддержку Damped. Это зависит от скорости работы силовых ключей регулятора.

Enabe PWM input - включить PWM вход. В режиме "OFF" ESC будет ждать на входе только протокол OneShot125, а в режиме "ON" будет выбирать автоматически между OneShot и PWM.

Motor Timing - синхронизация двигателя. Low - 0 градусов, MediumLow - 8 градусов, Medium - 15 градусов, MediumHigh - 23 градуса, High - 30 градусов. Практика показывает, что если мотор на низких оборотах начинает дергаться при резком увеличении газа, то увеличивают Motor Timing.

Синхронизация двигателя, настраиваемая ESC, важна для эффективности двигателя, надежности его работы, крутящего момента и числа оборотов. Для вращения мотора пропускается электрический ток через катушку, которая создает магнитную силу и притягивает магнит ротора. Когда включается и выключается ток через катушку, происходит индукция. При получении тока катушкой, требуется время для создания магнитного поля и достижения его идеального уровня, когда ток отключается, потребуется также время, чтобы магнитное поле полностью исчезло. Целью синхронизации двигателя является раннее переключение катушки двигателя, для создания магнитного поля в нужное время. Если включить катушку раньше, когда она еще далека от магнита, это увеличит обороты двигателя, но уменьшит крутящий момент. Когда запускается катушка поздно, она находится близко к магниту, увеличивается крутящий момент двигателя, но уменьшается число оборотов.

Синхронизация двигателя во многом зависит от индукции, а индукция зависит например от тока, сопротивление катушки, число витков катушки и так далее. Разные двигатели будут иметь разное время индукции, и, следовательно, Motor Timing должен подстраиваться.

Input Polarity - полярность PWM сигнала. Если при нулевом газе мотор работает на полных оборотах - меняется значение этого параметра.

Startup Beep Volume - громкость сигнала при подаче питания на квадрокоптер, издаваемая моторами.

Beacon Volume - громкость сигнала при бездействии.

Beacon Delay - задержка перед подачей сигнала бездействия.

PPM Min Throttle - минимальное значение PPM-сигнала.

PPM Max Throttle - максимальное значение PPM-сигнала.

Рассмотрим программирование (прошивку) ESC регулятора EMAX-12A по одному сигнальному проводу прошивкой BLHeli. В этом случае предыдущее программное обеспечение(прошивка) ESC регулятора должно иметь также и загрузчик. В противном случае прошивка должна быть выполнена предыдущим способом для прошивки BLHeli или как работе[13] для прошивки SimonK. Обычно современные ESC поставляются с прошивками, имеющими начальный загрузчик(bootloader) но с устаревшими версиями SimonK и BLHeli, которые могут не поддерживать протоколы OneShot. Однако даже при наличии bootloader-а ESC необходимо доработать, чтобы он мог работать с начальным загрузчиком. Например, в EMAX-12A требуется выпаять конденсатор и резистор, обведенные синим цветом на рисунке 3 (аналогично на рис.8).

Для программирования с помощью Arduino по сигнальному проводу необходимо выполнить подключение Arduino к EMAX-12A, как на рисунке 8 (вывод D3 Arduino к сигнальному проводу ESC, GND Arduino к GND ESC).

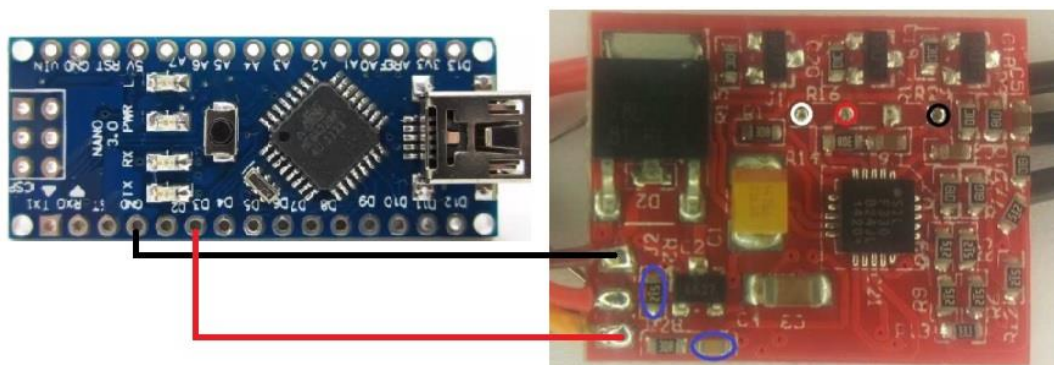


Рис.8. Программирование с помощью Arduino по сигнальному проводу ESC EMAX-12A

Программирование Arduino Nano выполняется как в предыдущем случае по рисункам 4 и 5. Для загрузки прошивки необходимо перейти во вкладку Silabs ESC Setup и во вкладке Select ATMEL/SILABS Interface выбрать SILABS BLHeli Bootloader(4way-if). После этого к ESC подключить батарею через и нажать на кнопку Connect.



Рис.9. Выбор интерфейса

После успешного подключения к ESC необходимо нажать на кнопку Flash BLHeli. Далее появиться окно для выбора файла прошивки. Для EMAX 12A, установленного на квадрокоптере и выбирается файл XP-12A Multi Rev.14.9. После этого выполняется прошивка и результат ее выводится на дисплей. После завершения процедуры и нажатия на кнопку Read Setup выводится обновленный экран с параметрами этой прошивки как было представлено на рисунке 7.

Рассмотрим программирование (прошивка) ESC регулятора SimonK-30A по одному сигнальному проводу прошивкой SimonK (tgy_2015-09-12). В регуляторе SimonK-30A должна быть установлена прошивка с Bootloader-ом. Arduino Nano подключается к ESC как на рисунке 10. Необходимо учесть, что ESC Simonk-30A работает под управлением микроконтроллера ATMEGA-8A.

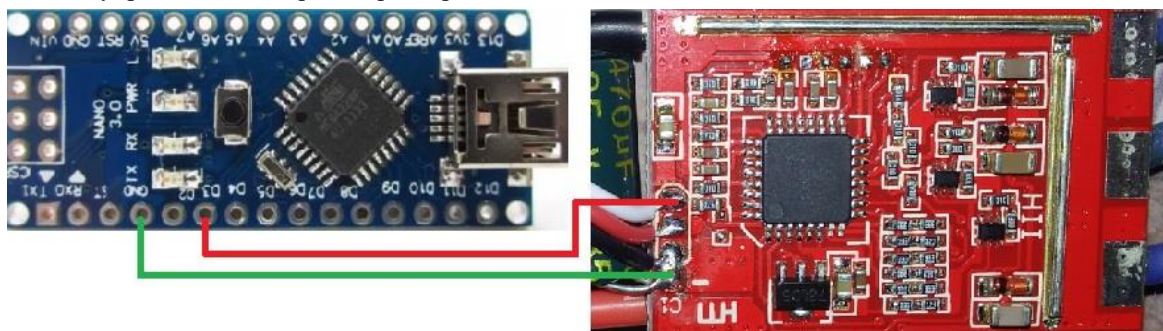


Рис.10. Программирование с помощью Arduino по сигнальному проводу ESC Simonk-30A

Программирование Arduino Nano выполняется как в предыдущем случае по рисункам 4 и 5. Для загрузки прошивки необходимо перейти во вкладку Silabs ESC Setup и во вкладке Select ATMEL/SILABS Interface выбрать ATMEL BLHeli Bootloader(4way-if) - рис.11. После этого к ESC подключить батарею через и нажать на кнопку Connect.

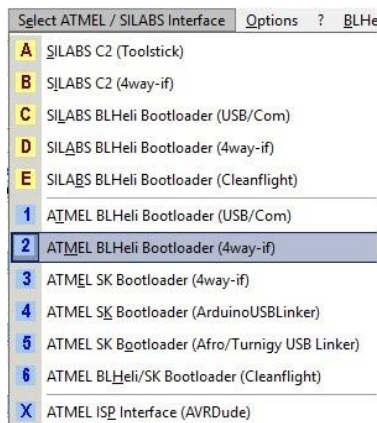


Рис.11. Выбор интерфейса

Далее через 15-20сек. необходимо нажать на кнопку Read Setup, после чего должно появиться окно (рис.12).

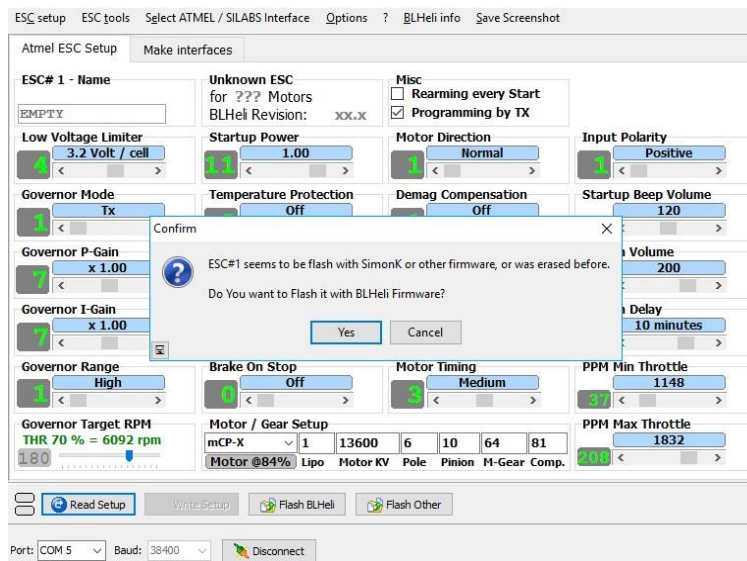


Рис.12. Окно с идентификацией прошивки Simonk или другой неизвестной

После нажатия на кнопку Cancel, а потом на Flash Other, появиться запрос на выбор файла прошивки. Прошивку необходимо скопировать с сайта <http://0x.ca/tgy/downloads/> - файл tgy_2015-09-12_103edb5.zip и распаковать его в каталоге. Для прошивки выбирается файл tgy.hex. Далее выполняется прошивка, ход которой отображается на экране. В случае успешной прошивки появляется соответствующее сообщение. После этого ESC Simonk-30A будет поддерживать протокол OneShot125.

После установки ESC и полетного контроллера на квадрокоптер, ESC можно перепрошивать по сигнальному проводу после отключения его от полетного контроллера и переподключении к Arduino Nano. Однако перепрошивку можно делать и через полетный контроллер, если там установлена прошивка CleanFlight последних версий. Для этого вначале подключается полетный контроллер через USB к компьютеру, затем подключается к квадрокоптеру батарея и запускается программа BLHeliSuite.exe(рис.13)

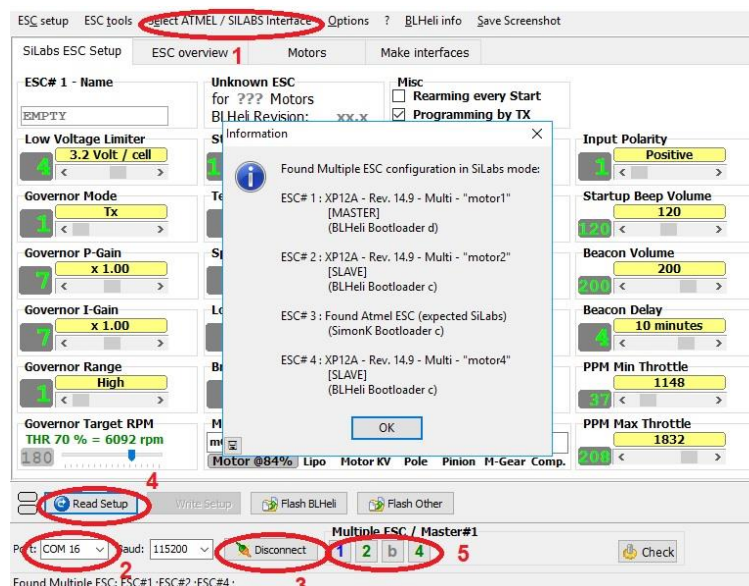


Рис.13. Подключение ко всем ESC через полетный контроллер

Для загрузки прошивок с 4-х ESC необходимо перейти во вкладку Select ATMEL/SILABS Interface и выбрать ATMEL BLHeli Bootloader(CleanFlight) - рис.11. На втором этапе выбирается номер порта полетного контроллера - COM 16 со скоростью 115200. На третьем этапе нажимается кнопка Connect. Примерно через 10-15 сек. нажимается кнопка Read Setup. Это приводит к чтению параметров всех ESC. 3-й ESC был определен с начальным загрузчиком Simonk, микроконтроллером Atmel и неизвестной прошивкой, остальные с прошивкой XP12A BLHeli. Выбор нужного регулятора можно выполнить в соответствии с меткой 5 рисунка 13. Для выбора, например 2-го ESC необходимо нажать на кнопки не нужных ESC, например 1,4. Останется 2-й ESC который можно перепрошить или поменять у него параметры.

На рисунке 14 представлено фото квадрокоптера с рамой 250мм, у которого установлено 3 регулятора EMAX-12A с прошивкой XP12A BLHeli и один регулятор SimonK-30A с последней прошивкой SimonK. Полетный контроллер SP Racing F3 - Асро управляется прошивкой Cleanflight. В качестве протокола связи с ESC используется OneShot125. Тестирование квадрокоптера показало достаточную его устойчивость в полете.

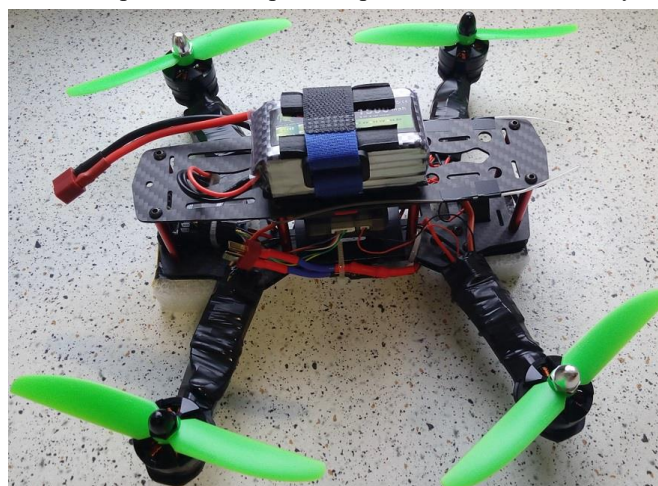


Рис.14. Фото квадрокоптера с рамой 250

Выводы

1. Показана необходимость перепрограммирования ESC регуляторов более современными прошивками, например BLHeli или SimonK в зависимости от аппаратной части регулятора. Если регуляторы основаны на

микроконтроллерах ATMEGA(ATmega8a и др.) то используются прошивки BLHeli или SimonK, если используются микроконтроллеры SILABS(F330, F39X) то они прошиваются только BLHeli.

2. Показано, что протокол PWM связи полетного контроллера с ESC регулятором не синхронизируется с работой гироскопа летательного аппарата. Поэтому ESC получает старый сигнал от контроллера полета. Протокол OneShot125 синхронизирует импульсы аппаратуры управления с выходом сигнала с гироскопа. Это позволяет ESC иметь самые актуальные последние данные из полетного контроллера без больших задержек, что стало возможным потому, что длина управляющего импульса OneShot125 в 8 раз меньше чем у PWM. Синхронизация позволяет осуществлять более точное управление, устанавливать более высокие значения PID и добиться большей стабильности полета особенно для гоночных квадрокоптеров.

3. Показано, что прошивка SimonK способна поддерживать относительно медленные протоколы PWM и OneShot125 и в настоящее время не поддерживается. Работает только на микроконтроллерах ATMEGA. Прошивка BLHeli поддерживает всю линейку высокопроизводительных современных протоколов. Она работает на микроконтроллерах ATMEGA, SILABS. В зависимости от модели микроконтроллера SILABS поддерживаются разные по производительности протоколы. Например SILABS F330 работает с самыми медленными протоколами PWM и OneShot125, F396 с более быстрыми OneShot42 и MultiShot и т.д.

4. Показана возможность использования Arduino Nano для программирования ESC EMAX-12A прошивкой BLHeli через три проводника в случае невозможности использования начального загрузчика из-за аппаратных ограничений регулятора ESC, позволяющего выполнить прошивку по сигнальному кабелю. Здесь на контактные площадки ESC напаиваются три проводника, которые подключаются к Arduino. Перед прошивкой ESC Arduino предварительно программируется.

5. При аппаратной доработке ESC регулятора EMAX-12A(и других ESC из этой серии) показана возможность программирования ESC через сигнальный провод, который используется для подключения к полетному контроллеру. В этом случае программирование ведется также через Arduino Nano.

6. Рассмотрена возможность программирования ESC регулятора Simonk-30A самой последней версией прошивки SimonK через Arduino Nano по одному сигнальному проводу с помощью программы BLHeliSuite.exe.

7. Установлено, что если все ESC регуляторы(или несколько) подключены к полетному контроллеру с прошивкой CleanFlight (опробована ver.2.5.0), то ESC регуляторы можно прошивать новыми прошивками, необязательно BLHeli, а также менять параметры регуляторов, если используется прошивка BLHeli. Если используется полетный контроллер например APM 2.6, то перепрошивку ESC можно выполнять по сигнальному проводу через Arduino, предварительно отключив ESC от полетного контроллера.

Литература

1. Paweł Sychalski. PWM, OneShot125, OneShot42, Multishot and DSHOT comparison. [Электронный ресурс]. – Режим доступа: <https://quadmeup.com/pwm-oneshot125-oneshot42-and-multishot-comparison/> , 2016.
2. Oscar Liang. WHAT IS ONESHOT ESC PROTOCOL – ACTIVE BRAKING. [Электронный ресурс]. – Режим доступа: <https://oscarliang.com/oneshot125-esc-quadcopter-fpv/> , 2016
3. Использование OneShot125 или PWMSync. [Электронный ресурс]. – Режим доступа: <http://fpv-community.ru/applications/core/interface/file/attachment.php?id=2204> , 2016.
4. Oscar Liang. WHAT IS DSHOT ESC PROTOCOL. [Электронный ресурс]. – Режим доступа: <https://oscarliang.com/dshot/> , 2017.
5. Programming adapters for BLHeli. [Электронный ресурс]. – Режим доступа: <https://flyingmodsc.com.files.wordpress.com/2017/06/blheli-programming-adapters.pdf>, 2017.
6. Oscar Liang. FLASH AND PROGRAM BLHELI ESC VIA CLEANFLIGHT – FLIGHT CONTROLLER. [Электронный ресурс]. – Режим доступа: <https://oscarliang.com/flash-esc-via-cleanflight-fc/>, 2015.
7. ESCs supported by BLHeli SiLabs. [Электронный ресурс]. – Режим доступа: <https://www.mediafire.com/folder/dx6kfaasyo24l/BLHeliSuite> ,2018.

8. Pin Configuration for BLHeliSuite 4-Way Interfaces (4w-if) on Arduino boards for Atmel/SiLabs ESC (v3). [Електронний ресурс]. – Режим доступа: <https://www.mediafire.com/folder/dx6kfaasyo24l/BLHeliSuite> , 2018.
9. BLHeliSuite MultipleESC mode 14.3.0.2 and later. [Електронний ресурс]. – Режим доступа: <https://www.mediafire.com/folder/dx6kfaasyo24l/BLHeliSuite> , 2018.
10. Operation manual for BLHeli SiLabs Rev14.x. [Електронний ресурс]. – Режим доступа: <https://www.mediafire.com/folder/dx6kfaasyo24l/BLHeliSuite> , 2018.
11. Oscar Liang. HOW TO CHOOSE ESC FOR RACING DRONES AND QUADCOPTERS. [Електронний ресурс]. – Режим доступа: <https://oscarliang.com/choose-esc-racing-drones/> , 2017.
12. Мясичев А.А. Перепрошивка регуляторов Hobbypower ESC-30A, Readytosky ESC-30A и Simonk 30A прошивкой Simonk. [Електронний ресурс]. – Режим доступа: https://sites.google.com/site/webstm32/esc_reg
13. Мясичев А. А. Использование платы RobotDyn Mega2560 Pro для построения полетного контроллера гексакоптера / А. А. Мясичев // Вісник Хмельницького національного університету. Технічні науки. – 2018. – № 3. – С. 172-180.
14. Мясичев А. А. Настройка PID регуляторов и GPS модуля для прошивки MegapirateNG полетного контроллера на базе Arduino mega2560 / А. А. Мясичев, Е. С. Ленков, В. М. Полозова // Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. – Київ : ВІКНУ, 2017. – Вип. 57. – С. 143-152.
15. Мясичев, А.А. Построение БПЛА на базе полетного контроллера APM 2.6 // Вісник Хмельницького національного університету. Технічні науки. – 2016. – № 5. – С. 225-230.
16. Мясичев А.А. НАСТРОЙКА PID РЕГУЛЯТОРОВ ДЛЯ ПОЛЕТНОГО КОТРОЛЛЕРА НА БАЗЕ ARDUINO MEGA256 И ПРОШИВКИ MEGAPIRATENG. Materialy XIII mezinarodni vedecko - prakticka konference « Moderni vymozenosti vedy– 2017». - Dil 8. Technicke vedy. Fyzika. Matematika. Stavebnictvi a architektura. Modernich informacnich technologii: Praha. Publishing House «Education and Science» s.r.o - 86 stran, с 76-83,2017
17. Мясичев А.А. Использование BLHeliSuite для прошивки ESC регуляторов SimonK 30A и EMAX Simon-12A прошивками SimonK и BLHeli. [Електронний ресурс]. – Режим доступа: <https://sites.google.com/site/webstm32/blhelisuite>, 2019.
- 18 Мясичев А. А. Возможности полетного контроллера CC3D с прошивкой INAV // Вісник Хмельницького національного університету. Технічні науки. – 2019. – № 1. – С. 129-136.
19. Мясичев О, А. Режимы полёту контролерів полёту APM 2.6 і Pixhawk БПЛА [Текст] / А. Мясичев О, В. В. Швець // Вісник Хмельницького національного університету. Технічні науки. – 2018. – № 1. – С. 78-82.
20. Мясичев А.А. О возможности построения БПЛА длительного полёта с использованием солнечных элементов. Materials of the XIII International scientific and practical Conference «Science without borders - 2017», Volume 11 : Chemistry and chemical technology . Physics . Technical science . Sheffield. Science and education LTD -80 p. - с. 49-56
21. Мясичев А.А. (ХмНУ), Ленков С.В. (ВІКНУ) ВЫЧИСЛИТЕЛЬНЫЕ ВОЗМОЖНОСТИ МИКРОКОНТРОЛЛЕРОВ STM32F4. ЗБІРНИК НАУКОВИХ ПРАЦЬ ВІЙСЬКОВОГО ІНСТИТУТУ КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКА Виходить 4 рази на рік Випуск № 51 КІЇВ – 2016, 185-192 с
22. Мясичев А.А. ПОСТРОЕНИЯ БПЛА ДЛИТЕЛЬНОГО ПОЛЕТА С ИСПОЛЬЗОВАНИЕМ СОЛНЕЧНЫХ МОДУЛЕЙ/ ВІСНИК ХНУ. Технічні науки.-Хмельницький: ХНУ, 2017. - №2.-с. 132-136.