

### **Список использованных источников**

1. Дыдик А.И. Оптический метод неинвазивной глюкометрии / А. И. Дыдик // Новые горизонты – 2023, Минск, 9–10 ноября 2023 г. / Министерство образования Республики Беларусь, Государственный комитет по науке и технологиям, Белорусский национальный технический университет, Институт по науке и технике БНТУ - Минск: БНТУ, 2023 - Т.1 - С. 154-156
2. Дыдик, А.И. Мобильное приложение для оптического глюкометра // Сборник конкурсных научных работ студентов и магистрантов. Ч.1 / УО «Брестский государственный технический университет»; редкол.: Н.Н. Шалобыта [и др.]. – Брест: Издательство БрГТУ, 2024. – С. 73 – 76.
3. Evercare.ru [Электронный ресурс] / Информационный портал в области телемедицины и цифрового здравоохранения – Режим доступа: evercare.ru/GlucoBeam – Дата доступа: 20.04.2023
4. GlucoTrack [Электронный ресурс] / Официальный сайт компании GlucoTrack –Режим доступа: glucotrack.com/ – Дата доступа: 20.04.2023
5. Noviosense [Электронный ресурс] / Официальный сайт компании Noviosense – Режим доступа: <https://www.noviosense.com/> – Дата доступа: 17.05.2023.
6. Патент RU 2368303. Способ неинвазивного определения концентрации глюкозы в крови / А. Д. Эльбаев, Х. А. Курданов, А. Д. Эльбаева – 27.09.2009. – Бюл. № 27
7. Патент RU 2317008. Устройство для неинвазивного определения концентрации глюкозы в крови / А. Д. Эльбаев, Р. И. Эльбаева, Х. А. Курданов, Р. А. Перковский – 20.02.2008. – Бюл. № 5
8. Брейн Бит [Электронный ресурс] / Официальный сайт компании Brainbeat – Режим доступа: <https://brainbeat.ru/o-proekte> – Дата доступа: 21.05.2023.

УДК 004.031.6

## **ПОЛЕТНЫЙ КОНТРОЛЛЕР ДЛЯ ПУСКОВЫХ УСТАНОВОК**

**П. В Жоглов, С. В. Чугунов**

Брестский государственный технический университет, Брест, Беларусь,  
zhoglovpavel21@gmail.com

*Flight controller - an electronic device that controls the flight of an aircraft. The term is applied to unmanned aerial vehicles, including aircraft models. The term autopilot is usually used for manned aircraft. The goal of our work is to develop and create a flight controller model using printed circuit board development technologies and Atmega328P microcontrollers.*

## **Введение**

Занимаясь изучением вопроса любительского ракетостроения, было замечено, что в большом количестве разработанных ракетных установок либо отсутствует бортовой компьютер, либо его действия описываются ещё на этапе программирования микроконтроллера и сборки [1]. В нашей работе мы хотели уделить особое внимание работе пусковых установок ракеты, а именно создать устройство, которое даст возможность не только контролировать, но и управлять полетом ракеты – полетный контроллер.

## **Моделирование полётного контроллера**

Целью нашей работы является разработка и создание модели полетного контроллера с использованием технологий разработки печатных плат и микроконтроллеров Atmega328P.

Мы поставили перед собой следующие задачи:

1. Изучить основы проектирования печатных плат для минимизирования габаритов будущих устройств.
2. Создать код с рациональным использованием типов переменных и низкоуровневых функций для стабильности системы.
3. Изучить основы работы с основными интерфейсами: I2C, UART, SPI.
4. Изучить основы работы с ISP-программатором, а также интегрировать работу с ним в схему устройств.

Полётный контроллер – электронное устройство, управляющее полётом летательного аппарата. Термин применяется к беспилотным летательным аппаратам, в том числе и авиамоделям. Применительно к пилотируемым летательным аппаратам обычно употребляется термин автопилот.

К функциям полётного контроллера относятся:

- стабилизация аппарата в воздухе;
- удержание высоты при помощи барометрического высотомера или иных датчиков и позиции при помощи GPS/GLONASS;
- автоматический полёт по заданным заранее точкам (опционально);
- передача на землю текущих параметров полёта с помощью модема или Bluetooth (опционально);
- обеспечение безопасности полёта (возврат в точку взлета при потере сигнала, автопосадка);
- остановка перед препятствием (для мультикоптеров) или облет препятствий (для самолетов) при наличии датчиков;
- подключение дополнительной периферии: OSD, светодиодной индикации и т.д.

Полётный контроллер представляет собой типичное микропроцессорное управляющее устройство [1]. При этом алгоритмы управления реализуются программно.

Поскольку для подобной техники требуется компактность, то при проектировании платы необходимо учитывать много факторов. Одним из них станет то, что на готовой плате Arduino уже стоит кварцевый источник

тактирования на 16 МГц для работы микроконтроллеров Atmega (в данном случае, Atmega328p). При монтаже компонентов на макетной плате, как показано на рисунке 1, удобен формат DIP, поскольку к ней подключаются компактные порты самого микроконтроллера.

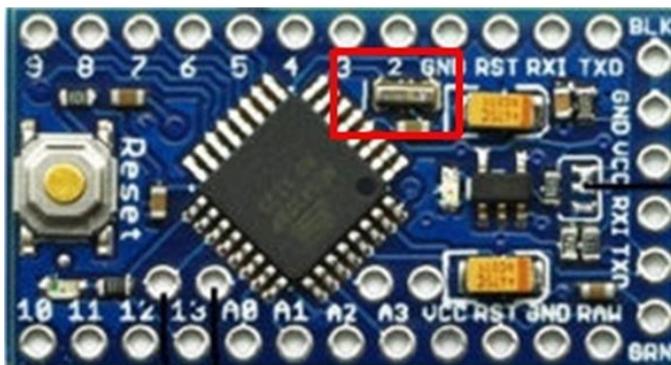


Рисунок 1 – Пример макетной платы

При моделировании возникает ряд проблем с энергопотреблением и габаритами будущего устройства. Проблема с энергопотреблением может быть решена различными способами, один из которых – замена внешнего кварца на внутренний. Это делается в строго определенном порядке с учетом множества факторов.

На рисунке 2 изображен на графике зависимости частоты, при которой контроллер будет работать стабильно, от напряжения на входе.

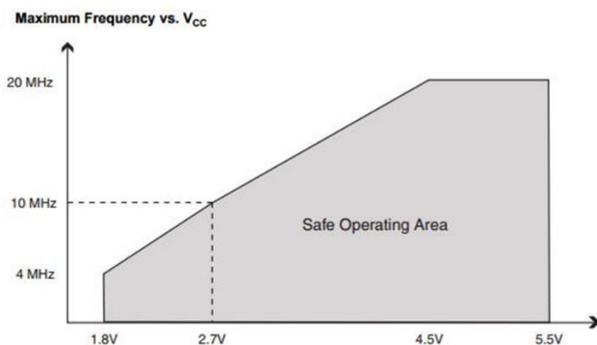


Рисунок 2 – Зависимость допустимых частот контроллера от напряжения

Наглядно показано, что на частоте 10 МГц устройство сможет работать при минимальном напряжении 2.7 В, а для стабильной работы достаточно 8 МГц, что будет важно в будущем. Для уменьшения размеров будем использоваться Li-Po аккумулятор на 3.7 В. Осталось сконфигурировать работу

микроконтроллера на работу с этой частотой, поскольку если просто выпаять микроконтроллер и переставить в схему без кварца, то прошивка будет невозможной. Более того, конфигурацией таких низкоуровневых настроек невозможно базовыми методами прошивки, поэтому на помощь приходит USB-ISP программатор, который работает по SPI и может редактировать фьюзы микроконтроллера в программе AVRDUDE\_PROG [2].

Fuse-биты являются инструкциями для настройки определенных параметров микроконтроллера в конце бинарного кода прошивки [3]. Могут отвечать за следующее:

- частота тактирования;
- запрет на чтение прошивки микроконтроллера;
- включение или выключение аппаратных таймеров;
- деление частоты кварцевого генератора;
- защита энергонезависимой памяти от стирания.

Также Fuse-биты делятся на подтипы:

- Лок-биты для защиты от копирования и перепрошивки.
- Fuse-биты высокого уровня. Основные это EESAVE и RSTDISBL. Первый не стирает память EEPROM при прошивке, а второй позволяет использовать пин RST на микроконтроллер как стандартный цифровой пин, что лишает возможности перепрошивки по SPI, т.к. RST - один из логических пинов этого интерфейса (SCK, MOSI, MISO, RST, VCC, GND).

- Fuse-биты низкого уровня. Вот здесь нам и пригодился программатор. Самым важным для нас фьюзом здесь станет CKDIV8 (CK - clock, DIV - деление на 8). Биты с инверсией, поэтому логическим “да” делению 8 МГц на 8 станет отключение этого параметра, следовательно, нужно оставить включенным, а параметры CKSEL оставить в тех же положениях.

- Extended-биты отвечают за минимальные значения напряжения работы микроконтроллера, при которых он не будет в бесконечной перезагрузке (все, кроме 1 - 1.8В, кроме 2 - 2.7В, кроме 3 - 4.7В).

Основной идеей разработки было создание двух устройств, одно из которых изображено на рисунке 3, позволило бы скорректировать установки непосредственно перед запуском на пусковом контроллере, изображенном на рисунке 4. С помощью первого устройства можно устанавливать таймер срабатывания полетного контроллера, включение и отключение записи log-файлов на SD-карту, а уже сам контроль полета осуществлялся бы полетным контроллером, основным элементом которого является биполярный транзистор PN2222. Решение о добавлении второго транзистора, датчика атмосферного давления и SD карты пришло одним из первых, поскольку для ультимативности, данная разработка может спокойно использоваться для ракеты, где второй транзистор будет активировать запал парашютной системы.

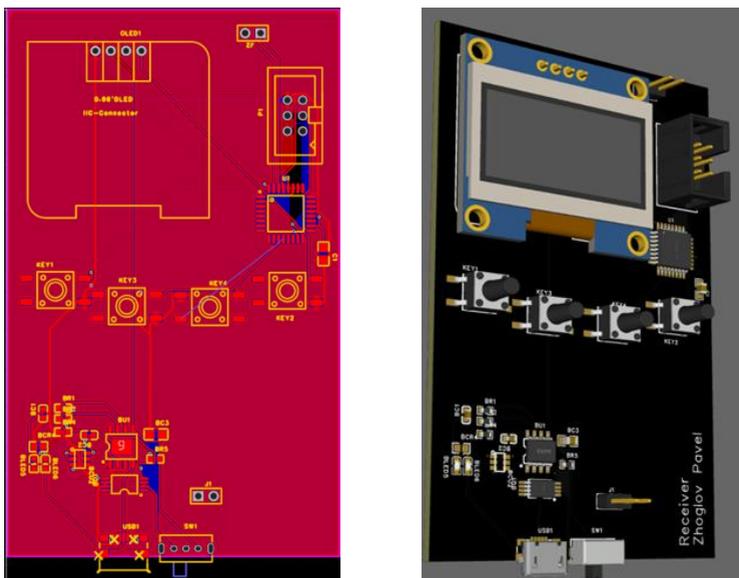


Рисунок 3 – Платы управляющего устройства

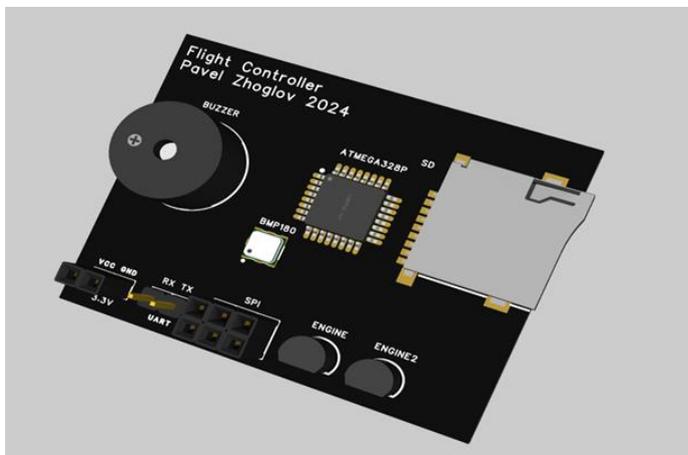


Рисунок 4 – Плата полётного контроллера

Передача необходимых нам установок с пульта в полётный контроллер производится передачей одного пакета с тремя данными по UART (T0-255; L0/1; N0/1), где T обозначает таймер от 0 до 255 секунд срабатывания первого транзистора после отключения пульта, L - включить или отключить запись высоты на SD карту и обозначение точки Апогея, N - это тестовый режим, при включении которого через указанный таймер сработает только парашют

для проверки его работы. Это и стало одной из причин отказа от прошивки привычным способом с загрузчиком. USB-TTL конвертор использует те же пины RX1 И TX1 для перепрошивки, поэтому в схеме SPI была указана распиновка, т.к. при ней устройство работает стабильно.

### **Заключение**

По итогам выполненной работы были получены и усвоены знания о проектировании и трассировке печатных плат малых и средних размеров, показана необходимость учёта особенностей при настройке тактирования будущего устройства и его прошивке нестандартными методами.

### **Список использованных источников**

1. [https://ru.wikipedia.org/wiki/Полётный\\_контроллер](https://ru.wikipedia.org/wiki/Полётный_контроллер)
2. <https://www.yourdevice.net/proekty/avrdude-prog>
3. <https://www.alldatasheet.com/>

УДК 004.82

## **ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ДЛЯ СИСТЕМ УПРАВЛЕНИЯ**

**В. И. Хведчук, Чжан Сяотянь**

Брестский государственный технический университет, Брест, Беларусь

*To solve the problem of describing the law of functioning of a typical control element, a system of classification of control units by types and subspecies is used. The description of the links uses temporal logic. For each subspecies, the parameters are set in accordance at each time interval. Subspecies compositions are possible. The developed software tools can be used in solving problems of describing the program code of control systems.*

**Введение** Задача проектирования ПО систем управления, впрочем как и самих систем управления остается весьма сложной и трудоемкой задачей. Для решения используются такие приложения, как: Keil, Proteus, CoDeSys, Altium Designer. Известны также сервисные средства, распространяемые производителями управляющего оборудования: Siemens, Omron и т.д. Вместе с тем растет число различных функциональных приложений, приводящих к использованию аддитивных технологий, вызывающих новые способы применения электронных управляющих систем. В частности, можно отметить успех Internet of Thing, поддерживаемых и в рамках внутрифирменных инструментариев. При этом далеко не всегда используются стандартные средства и интерфейсы, предлагаются интегрированные разработки. Примером может служить и некоторая продукция компании Intel. Поэтому появляется задача управления такого рода системами.