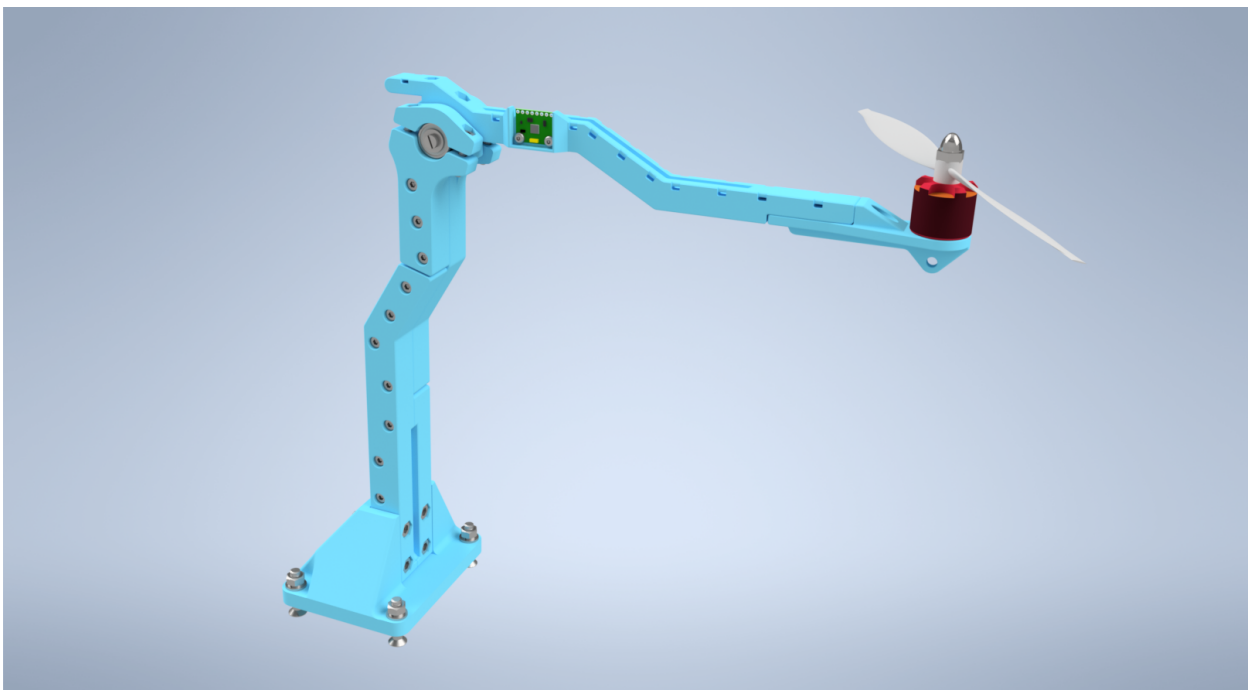


Ещё один ПИД-регулятор

Даниил Барков

Лето 2025



Меня не оставляет желание сделать свой маленький квадрокоптер, и хотелось бы прийти к этому планомерно и методично. В [прошлой статье](#) я описал способ определения ориентации с помощью инерциального датчика. Следующий этап - разработка алгоритма автоматической стабилизации. Простейшей реализацией такого алгоритма является пресловутый ПИД-регулятор, подробно разобранный на всех тематических ресурсах, например, [тут](#). Сходу было решено сделать стенд, на котором можно попрактиковаться в отладке алгоритма и подборе коэффициентов. Очевидно, нужно что-то вращающееся и с пропеллерами, но чтобы не улетело. Так было принято в разработку устройство со следующими требованиями:

- Устройство должно быть динамичным
- Устройство должно быть экстремальным в эксплуатации
- Устройство должно выглядеть симпатично
- Устройство должно быть достаточно надёжным, чтобы можно было успеть наиграться
- В качестве вычислительного устройства должен использоваться микроконтроллер STM32 с ядром Cortex-M4 или круче
- Необходима связь с ПК для настройки коэффициентов и отображения графиков

Устройство

В этой статье расскажу об устройстве стенда, о его механических свойствах, процессе проектирования и электронике.

Механика

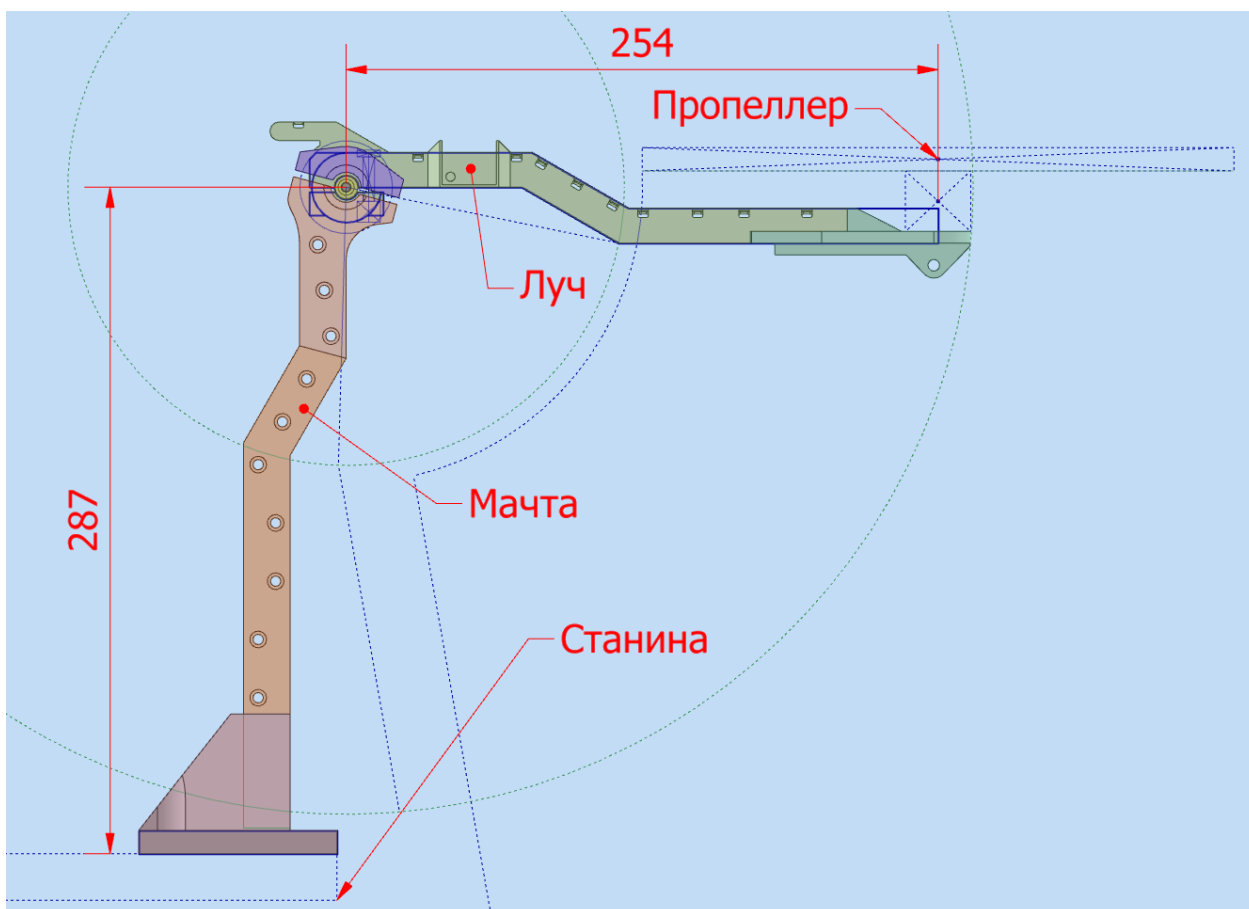


Рис. 1: Эскиз габаритов стенда

Дом строят с печи, а стенд ПИД-регулятора - с пропеллера, купленного 10 лет назад на Али. Тогда же куплены BLDC моторчик с гордой надписью dji 2212/920KV и подходящий ESC. Диаметр винта внушительный - 254 мм. Чтобы как-то соотнести с рамой квадрокоптера, длина луча выбрана соответствующая - 254 мм. Чтобы было

удобно работать со стандом, сидя за столом, ось луча поднята от станины на те же 254 мм и ещё на чуть-чуть.

Конструкция несложная, вот основные компоненты:

- Станина
- Мачта, поднимающая ось луча. Крепится к станине широким фланцем.
- Луч, одним концом соединённый с мачтой парой шариковых подшипников. На нём крепятся датчик и мотор
- Держатель мотора, который является интерфейсом между лучом и мотором
- Мотор с пропеллером
- Коробочка с электроникой

Длинные детали пришлось разделить, чтобы они вместились в принтер. Получилось 10 деталей, 8 из которых напечатались без поддержек. Благодаря паре шариковых подшипников, разнесённых на 20 мм, луч вращается свободно там, где надо, сохраняя жёсткость в остальных направлениях. Чтобы луч не опрокинулся, на нём предусмотрен упор. Датчик крепится на луче одним винтом и двусторонним скотчем. Расположен поближе к оси, чтобы центробежные силы меньше влияли на измерения. Пластиковые детали скрепляются либо болтами, либо винтами и резьбовыми втулками. Стенд без станины весит аж 660 грамм, что и является основным критерием надёжности.



Рис. 2: Тяжесть - это хорошо. Тяжесть - это надёжно.

В качестве станины я выбрал алюминиевый рабочий стол с Т-образными пазами. Самоклеящиеся резиновые ножки добавили ему устойчивости и сделали его тише.

В дополнение расскажу о некоторых правилах, которые привык использовать при разработке параметрических 3D-моделей.

Параметрическое моделирование

Работу стоит начать с обмера сложных покупных деталей. Параметрическое моделирование предполагает разработку сложных форм, состоящих из геометрических

примитивов, получаемых из плоских эскизов экструзией, вращением или протягиванием. Эскизы преимущественно состоят из отрезков, дуг окружностей и точек, взаимное расположение которых определяется привязками и геометрическими размерами. Правильное выстраивание зависимостей облегчит редактирование модели в будущем.

Наиболее универсальным решением считаю использование Excel-таблицы, значения из которой можно импортировать в несколько файлов проекта одновременно. Это избавит от необходимости регулярного согласования деталей в случае изменения габаритов покупных деталей.

	A	B
1	blade_radius	127
2	blade_height	10
3	bearing_inner	7
4	bearing_outer	19
5	bearing_width	6
6	beam_length	254
7	motor_height	24
8	motor_diameter	28
9	motor_blade_offset	2
10	shaft_beam_diameter	10
11	M4_screw_innerHex_cap_diam	6,6
12	M4_screw_innerHex_cap_height	4
13	M4_nut_lock_width	6,8
14	M4_nut_lock_height	5

Рис. 3: Таблица параметров

Эскизирование

Почему-то на большинстве курсов по 3D-моделированию учат делать по эскизу для каждой операции создания поверхности (буду называть их элементами 3D), например, выдавливания или вращения. Однако при таком подходе сложно устанавливать взаимосвязи. Приходится проецировать геометрию, устанавливать привязки. Ссылаться на размеры базовой фигуры становится непросто, приходится включать видимость соответствующего эскиза. В результате на дочернем эскизе появляется такое же количество элементов, как на базовом, и это ради того, чтобы при редактировании элемента 3D, проецированные элементы эскиза потеряли зависимость, или того хуже, эскиз потерял опорную плоскость.

Этих неудобств можно избежать, укладывая большее количество фигур на эскиз. Туда же можно помещать контуры будущих деталей с заведомо известными габаритами. Например, можно нарисовать контур шайбы, чтобы определить размер необходимой площадки. К тому же это формирует целостное представление о будущей конструкции.

Порядок проектирования

Можно выделить два подхода к созданию сборок.

Первый. Проектирование отдельных деталей с последующим объединением в файле сборки. В таком случае приходится внимательно пользоваться внешней таблицей параметров, импортировать параметры деталей из других файлов, избегать кольцевых зависимостей. Этот подход сложный, однако он позволяет использовать изысканные автоматизированные инструменты, например, генерацию рам из профилей или размещение крепежа, адаптирующегося под ширину скрепляемых деталей.

Второй. Проектирование детали из нескольких твёрдых тел с последующим разделением в сборке. Отлично подходит для моделирования практически с нуля. Например, когда я планирую печатать на FDM-принтере элемент конструкции с большими нависающими частями, удобно разделить деталь на две, напечатать по отдельности и склеить.

Так как основные детали будут изготовлены на FDM-принтере и скреплены винтиками, оптимально будет применить оба подхода. САПР, которым я пользуюсь, предоставляет возможность спроектировать деталь, состоящую из нескольких твёрдых тел, которые будут экспортированы в отдельные файлы. Эти детали будут импортированы в сборку и размещены так, как это было в исходном файле. В этой же сборке можно добавить различные скругления и фаски, отверстия под крепёж. Таким образом, промежуточные детали будут наследовать изменения, произведённые в сборке и в исходном файле, и могут быть отправлены на печать без дополнительных усилий.

При разработке небольших моделей, состоящих преимущественно из изготавливаемых деталей, такая комбинация ускоряет разработку.

Небольшие лайфхаки

Описанные правила стоит применять со скепсисом, потому что их эффективность проверялась на небольших сборках, до пары сотен деталей. А эти лайфхаки будут полезны многим.

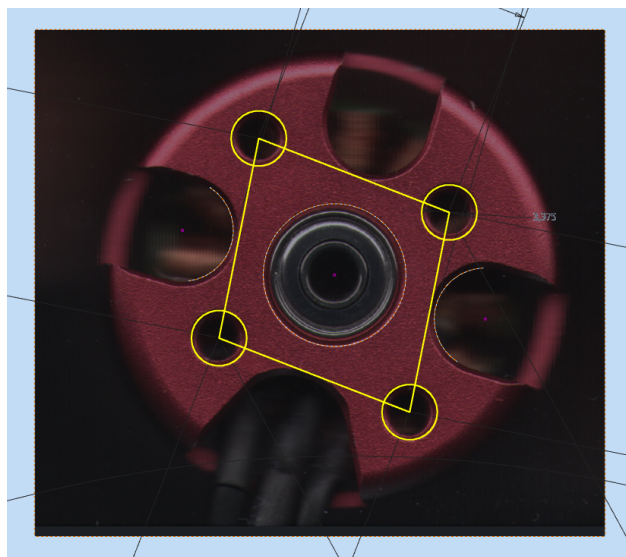


Рис. 4: Фланец мотора

Мотор крепится за статор четырьмя винтами М3. Фланец оказался с подвохом, отверстия расположены на одинаковых расстояниях друг от друга. Хотелось бы считать,

что они находятся на углах квадрата, но нет. Этот квадрат вовсе не квадрат, а лишь параллелограмм. Я не стал возиться со штангенциркулем, а просто отсканировал фланец мотора на домашнем МФУ. Полученную картинку закинул в эскиз САПРа, отмасштабировал, обвёл и получил отличное посадочное место.

Второй полезный совет заключается в том, что в сборных деталях, напечатанных на принтере, практично делать шестигранные отверстия для гаек. Они могут быть расположены на любой поверхности и напечатаны ровно благодаря тому, что FDM-принтеры способны печатать поверхности, нависающие под 60° к вертикали почти из любого пластика. Эти отверстия ускорят сборку, и их можно разместить там, куда не пролезет гаечный ключ.

Недостатком является то, что гайка не способна двигаться по плоскости, это может вызвать заклинивание винта при сборке. Чтобы минимизировать риски, не стоит использовать винты с потайной головкой.

Наконец, расскажу о том, как запрессовать метрический крепёж в подшипник. При отсутствии токарного станка, самым очевидным валом для подшипника является резьбовая шпилька, так как она металлическая, вместо стопорного кольца можно использовать гайку. Но диаметры шпильки и подшипника не подходят для соединения с натягом, что плохо. С другой стороны, можно напечатать небольшую ось на 3D-принтере, но ожидать значительной прочности от неё не стоит.

Напрашивается очевидное решение.



Подобрав параметры печати, получил вал, который продевается в подшипник с небольшим усилием, но при вкручивании винта, вал расширился, и соединение стало жёстким.

Электроника

Тут напишу, какой микроконтроллер выбран, почему, на какой плате собран. Расскажу, что знаю о моторе, контроллере хода, блоке питания. Расскажу о выборе платы с датчиком, как проложил проводку.

В качестве микроконтроллера выбран STM32F411CEU6, собранный на модуле Black-Pill. В отличие от использованного в предыдущей статье F103, этот обладает ускорителем вычислений с плавающей точкой единичной точности и значительно большей

тактовой частотой процессора. Также в качестве приятного бонуса, порт USB исполнен в разъёме Type-C. Так как я планирую использовать USB в режиме CDC, частоту придётся понизить с максимальных 100 МГц до 96 МГц. Модуль поддерживает питание от 5 В, которое, в отсутствии подключения к ПК, может быть получено от ESC.

Модель ESC я сообщить не могу, так как этикетка давно отпала. Знаю лишь, что он состоит из МК ATmega328, шести транзисторов и радиатора, завернутых в термоусадку. Интерфейс управления стандартный - двухсотпятидесятигерцовая ШИМ с заполнением 0.25 - 0.5.

Моторчик бесколлекторный, аутраннер. Трехпроводное подключение без обратной связи.

В качестве датчика был выбран MPU6050 на модуле GY-521. В нём нет магнитометра, как в используемом ранее MPU9255. Ну и ладно. Зато в модуле есть отверстие для винтика. Даже два отверстия. Но второе находится впрытик к компонентам, и шляпка винтика не помещается. В остальном модули практически идентичны. Модуль питается от 5 В и общается с МК посредством I²C.

Проводка проложена абы как, провода торчат в разные стороны с целью уменьшения оседания ЭМИ на сигнальные провода. Всё же I²C предназначен для обмена данными компонентами, находящимися в пределах платы, а векторное управление мотором, регулируемое ШИМом, при токах до 10 А RMS - это существенный источник помех.

Впрочем, для адекватной работы стенда достаточно потребления 4 А в пике. Поэтому используется небольшой импульсный БП на 12 В, примерно соответствующим трёхбаночному литий-ионному аккумулятору.

Теория

Тут расскажу о том, как происходит вычисление угла луча. Как происходит изначальная проверка положения, защита от нештатной работы. Как я пообщался с жипити и упростил расчёты. Расскажу о том, как прикинул дифур, как натянул ПИД на численное решение в матлабе. Покажу теоретические графики. Перечислю неучтённые свойства, например гироскопические силы, временные задержки между поворотом и измерением, сигналом и изменением скорости мотора.

Реализация и эксперименты

Сделаю измерения и получу таблицу соответствия сигнала для мотора и его тяги. Подгоню кривульку, сделаю градуировку. Расскажу, как расширил оригинальный алгоритм ПИД-контроллера, как это улучшило работу, что из этого действительно применимо к квадрокоптерам. Проведу эксперименты с разными коэффициентами, сравню с симуляцией. Расскажу, как хочу измерить неучтённые параметры для моделирования. Сделаю намёк в сторону машинного обучения.