

**Государственное бюджетное общеобразовательное учреждение города
Москвы “Школа №1544”**

Система стабилизации малых спутников CubeSat

Выполнил:

Павлов Максим Вадимович, 10 “Б” класс, Школа №1544

Научный руководитель работы:

Москва, 2025

Оглавление

1. Введение

2. Актуальность проекта
3. Цель и задачи
4. Методика выполнения
5. Практическая и теоретическая значимость
6. Будущее развитие проекта
7. Список литературы

Введение

CubeSat — это миниатюрный спутник формата 10×10×10 см (1U), разработанный для упрощения и удешевления доступа к космосу. Такие спутники активно применяются в университетских и научных миссиях, образовательных проектах и экспериментах на орбите.

Почему CubeSat набирают популярность:

- Низкая стоимость производства и запуска
- Возможность быстрой разработки и тестирования
- Простота масштабирования и модульности
- Широкая поддержка открытого ПО и документации

В космосе CubeSat подвергается различным видам колебаний — как во время запуска ракеты, так и на орбите.

Во время старта спутник испытывает мощные механические вибрации, которые могут:

- Повредить чувствительные датчики и электронику;
- Нарушить калибровку гироскопов и акселерометров;
- Привести к ошибкам в ориентации аппарата.

Даже на орбите возможны **микровибрации**, вызванные:

- Работой двигателей ориентации и маховиков;
- Перемещением подвижных частей конструкции (например, антенн или солнечных панелей);
- Тепловыми деформациями материалов при переходах из тени в освещённые области.

Для обеспечения стабильной ориентации спутника необходимо **подавление вибраций и колебаний**.

Это можно реализовать как механическими способами (демпферами и упругими креплениями), так и программными — с помощью **алгоритмов фильтрации данных датчиков и PID-регуляторов**, которые автоматически компенсируют колебания и удерживают спутник в заданном положении.

Актуальность проекта

Системы стабилизации ориентации широко применяются во многих областях современной техники.

Они используются:

- в **авиации** — для удержания самолётов и дронов в устойчивом положении;
- в **морских и наземных платформах** — для компенсации колебаний и вибраций;
- в **робототехнике** — для точного позиционирования манипуляторов и подвижных систем;
- в **космической технике** — для управления положением спутников, телескопов и зондов.

Современные малые спутники формата **CubeSat** представляют собой один из наиболее перспективных и доступных примеров применения таких систем.

Цель и задачи проекта

Цель проекта:

Разработка и экспериментальное исследование системы стабилизации ориентации CubeSat с использованием инерциальных датчиков, фильтрации данных и PID-регулирования.

Задачи проекта:

1. Изучить принципы работы систем ориентации и стабилизации малых спутников.
2. Изучить работу инерциальных датчиков (акселерометра, гироскопа) и освоить их использование для измерения угловых параметров.
3. Реализовать фильтр **Madgwick** для объединения данных акселерометра и гироскопа и получения устойчивых значений ориентации.
4. Настроить **PID-регуляторы** для автоматической стабилизации CubeSat по осям Roll и Pitch.
5. Разработать схему управления моторами через драйвер **TB6612FNG**.
6. Собрать и протестировать установку, имитирующую работу системы стабилизации в условиях земной гравитации.
7. Провести анализ работы системы и оценить эффективность применённого алгоритма.

Методика выполнения

Основная идея стабилизации спутника заключается в **управлении моментом вращения** — то есть создании такого крутящего момента, который изменяет или компенсирует текущее угловое движение аппарата.

Момент вращения (момент силы) определяется формулой:

$$M = r * F$$

M - момент силы (в Н·м),

r - радиус-вектор от центра масс до точки приложения силы,

F - приложенная сила.

Если на спутник действует момент, то изменяется его **угловая скорость** по закону динамики вращательного движения:

$$M = I \cdot d\omega / dt$$

I - момент инерции спутника,

ω - вектор угловой скорости.

Как это работает на практике

Когда CubeSat начинает вращаться, например, из-за внешнего воздействия (удар, несимметричный выброс, солнечное давление), на нём возникает угловая скорость.

Чтобы погасить это вращение, необходимо создать **противоположный момент**.

- Если спутник вращается по часовой стрелке, система управления создаёт момент против часовой стрелки.
- Когда скорости уравниваются, результирующий момент становится нулевым, и спутник стабилизируется.

Исполнительный механизм

Для создания этого момента применяются различные устройства:

Реакционные колёса — небольшие маховики, которые раскручиваются электродвигателями.

По третьему закону Ньютона, изменение скорости вращения маховика создаёт обратный момент на корпус спутника.

$$M = -I \cdot d\omega / dt$$

I - момент инерции ротора,

ω - его угловая скорость.

Таким образом, ускоряя или замедляя вращение колеса, можно точно управлять положением спутника.

Работа с датчиком MPU6050

Датчик **MPU6050** — это инерциальный модуль, объединяющий в одном корпусе **трёхосевой акселерометр** и **трёхосевой гироскоп**.

Он позволяет измерять линейные ускорения и угловые скорости CubeSat по трём осям: **X, Y, Z**.

В проекте MPU6050 подключён к микроконтроллеру по интерфейсу **I²C** и используется для определения текущего положения спутника в пространстве.

В начале программы производится инициализация датчика и калибровка его нулевых смещений.

Затем в основном цикле данные акселерометра и гироскопа считываются и передаются в фильтр **Madgwick** для расчёта углов ориентации (Roll и Pitch).

```
MPU6050 mpu(Wire);

float ax = mpu.getAccX();
float ay = mpu.getAccY();
float az = mpu.getAccZ();
float gx = mpu.getGyroX() * DEG_TO_RAD;
float gy = mpu.getGyroY() * DEG_TO_RAD;
float gz = mpu.getGyroZ() * DEG_TO_RAD;

filter.updateIMU(gx, gy, gz, ax, ay, az);
```

Фильтр Madgwick

Для получения углов ориентации CubeSat используется **алгоритм Madgwick**, который объединяет данные акселерометра и гироскопа.

Если просто складывать угловые скорости гироскопа каждую итерацию, ошибки измерения будут накапливаться, что приведёт к **дрейфу** — спутник «потеряет» правильное направление уже через несколько секунд.

Акселерометр, наоборот, даёт стабильные, но шумные данные, чувствительные к вибрациям и ускорениям.

Фильтр **Madgwick** решает эту проблему, комбинируя преимущества обоих сенсоров:

- гироскоп обеспечивает плавность и динамическую точность,
- акселерометр — корректирует накопленную ошибку.

В результате мы получаем устойчивые и точные значения **Roll** и **Pitch**, даже при наличии вибраций и внешних возмущений.

PID-регулятор

Для стабилизации CubeSat используется **PID-регулятор** (Proportional-Integral-Derivative), который управляет скоростью вращения моторов, создавая необходимый момент для компенсации отклонений по осям **Roll** и **Pitch**.

PID-регулятор вычисляет управляющее воздействие по трём составляющим:

- **Р (пропорциональная)** — зависит от текущей ошибки, быстро возвращает систему к нужному положению.
- **І (интегральная)** — учитывает накопленные ошибки за время, устраняя постоянное смещение.
- **Д (дифференциальная)** — реагирует на скорость изменения ошибки, сглаживая колебания и предотвращая перерегулирование.

$$M_{\text{упр}} = -K_p \cdot \omega - K_i \int \omega dt - K_d \frac{d\omega}{dt}$$

В коде контроллер получает текущие углы наклона от фильтра Madgwick, сравнивает их с заданным значением (например, 0° — горизонтальное положение), и вычисляет значение PWM для каждого мотора.

Это позволяет системе плавно гасить колебания и удерживать спутник в устойчивом положении.

```

float q0, q1, q2, q3;
filter.getQuaternion(q0, q1, q2, q3);

float norm = sqrt(q0*q0 + q1*q1 + q2*q2 + q3*q3);
q0 /= norm; q1 /= norm; q2 /= norm; q3 /= norm;

// направление "верха" устройства
float roll = 2.0f * (q1*q3 - q0*q2);
float pitch = 2.0f * (q2*q3 + q0*q1);

// --- PID вычисляем скорость моторов ---
float pwmRoll = pidRoll.compute(roll);
float pwmPitch = pidPitch.compute(pitch);

```

Был реализован отдельный класс работы PID-регулятора

```

float PID::compute(float input) {
    // Ошибка
    float error = setpoint - input;

    // Интегральная часть
    integral += error * dt;

    // Дифференциальная часть
    float derivative = (error - prevError) / dt;

    // PID
    float output = kp * error + ki * integral + kd * derivative;

    // Ограничим диапазон
    output = constrain(output, outMin, outMax);

    // Сохраняем ошибку
    prevError = error;

    return output;
}

```

Схема управления моторами через драйвер TB6612FNG

Для реализации стабилизации CubeSat используются два электродвигателя, подключённые через драйвер **TB6612FNG**. Этот драйвер управляет направлением и скоростью вращения моторов с помощью сигналов **PWM** и цифровых входов **IN1/IN2**.

В проекте используются следующие подключения:

- Для двигателя А — пины **AIN1**, **AIN2** и **PWMA**
- Для двигателя В — пины **BIN1**, **BIN2** и **PWMB**
- Пин **STBY** отвечает за включение или выключение драйвера

```
// --- Пины TB6612FNG ---  
const int AIN1 = 9, AIN2 = 10, PWMA = 11;  
const int BIN1 = 7, BIN2 = 6, PWMB = 5, STBY = 8;
```

Микроконтроллер формирует сигнал PWM, полученный от **PID-регулятора**, и подаёт его на входы драйвера.

В зависимости от знака управляющего сигнала, меняется направление вращения мотора, создавая нужный момент стабилизации.

```
// --- задаём скорость моторам ---  
motorA.setSpeed(pwmPitch);  
motorB.setSpeed(pwmRoll);
```

```
void Motor::setSpeed(int s) {  
    s = constrain(s, -255, 255);  
  
    if (s > 0) {  
        digitalWrite(in1Pin, HIGH);  
        digitalWrite(in2Pin, LOW);  
        analogWrite(pwmPin, s);  
    }  
    else if (s < 0) {  
        digitalWrite(in1Pin, LOW);  
        digitalWrite(in2Pin, HIGH);  
        analogWrite(pwmPin, -s);  
    }  
    else {  
        stop();  
    }  
}
```

Таким образом, при отклонении CubeSat от горизонтального положения один мотор ускоряется, другой замедляется — за счёт этого формируется момент, который возвращает систему в равновесие.

Сборка установки и тестирование проекта

Для проверки работоспособности системы стабилизации была разработана и собрана экспериментальная установка, имитирующая движение CubeSat в условиях земной гравитации.

Конструкция состоит из двух основных частей:

1. Модель корпуса CubeSat 1U

- Печатный корпус размером **10×10×10 см**, выполненный на 3D-принтере.
- Внутри корпуса размещены: микроконтроллер, драйвер моторов, модуль **MPU6050**, а также сама система питания.
- Крепёжные отверстия и направляющие соответствуют реальному формату CubeSat.

2. Двухосевой карданный подвес

- Позволяет свободное вращение модели по двум осям (**Roll** и **Pitch**), имитируя ориентацию спутника в пространстве.
- Каждый уровень подвеса напечатан на 3D-принтере и соединён подшипниками для минимального трения.
- Моторы установлены так, чтобы создавать вращающий момент вдоль соответствующих осей, обеспечивая стабилизацию макета.

3. Опора из деревянной фанеры

- Выступает опорой для карданового подвеса

Во время испытаний установка фиксируется на специальной платформе, а на компьютер выводятся данные с MPU6050: углы **Roll**, **Pitch** и управляющие сигналы **PWM**.

Такой подход позволяет наблюдать, как система реагирует на внешние возмущения и возвращает CubeSat в устойчивое положение.

Теоретическая значимость:

Разработка проекта позволяет глубже понять принципы работы систем стабилизации малых космических аппаратов, в том числе:

- применение инерциальных датчиков для определения пространственного положения;
- использование фильтров ориентации (на примере алгоритма **Madgwick**) для обработки измерений;
- построение и настройку **PID-регуляторов** для управления динамическими системами.

Полученные результаты могут быть использованы при обучении основам управления движением, мехатроники и аэрокосмических технологий.

-

Практическая значимость:

Разработанная система стабилизации на базе доступных компонентов может служить **бюджетным учебным стендом** для отработки алгоритмов ориентации CubeSat и других автономных устройств.

Такая установка:

- демонстрирует реальные принципы стабилизации и управления моментом вращения;
- позволяет тестировать различные алгоритмы фильтрации и регулирования;
- может быть легко модифицирована под новые эксперименты и исследования.

Таким образом, проект сочетает учебную, исследовательскую и прикладную ценность, способствуя развитию компетенций в области робототехники и систем управления.

Результаты и выводы

В ходе выполнения работы были достигнуты следующие результаты:

1. Изучение принципов стабилизации малых спутников

Были подробно исследованы методы стабилизации CubeSat, включающие работу гироскопов, акселерометров, фильтров ориентации и исполнительных механизмов. Получены практические знания о применении инерциальных датчиков и алгоритмов ориентации в условиях лабораторных экспериментов.

2. Разработана и реализована система стабилизации

Создана рабочая модель системы стабилизации CubeSat на базе датчика MPU6050, микроконтроллера и карданового подвеса. Реализованы алгоритмы фильтрации данных, вычисления углов ориентации и PID-регулирования положения.

3. Проведены испытания и анализ недочётов

Во время тестирования были выявлены ключевые недостатки конструкции и алгоритмов:

- ограниченная точность MPU6050 и отсутствие стабилизации по оси Yaw,
- недостаточный крутящий момент моторов,
- механические люфты и трение карданового подвеса,
- необходимость тонкой настройки PID-регулятора.

Эти проблемы проанализированы, и на их основе сформированы пути дальнейшего развития.

4. Создан макет установки для вращения спутника

Сконструирован и собран функциональный макет, имитирующий поворот CubeSat в пространстве. Он позволяет отрабатывать модели управления, тестировать алгоритмы стабилизации и проводить практические эксперименты с ориентацией.

5. Определены направления будущего развития

Сформирован план модернизации системы: переход на датчик MPU9250 с магнитометром, внедрение фильтра Калмана, добавление третьей оси вращения, установка более мощных моторов, оптимизация внутренней компоновки и улучшенная настройка PID-регулятора.

Будущее развитие проекта

В дальнейшем планируется расширить функциональные возможности системы стабилизации CubeSat, добавив новые компоненты и улучшенные алгоритмы обработки данных.

1. Переход на датчик MPU9250

— В отличие от MPU6050, модуль **MPU9250** включает не только акселерометр и гироскоп, но и **трёхосевой магнитометр**, что позволит реализовать стабилизацию по всем трём осям — **Roll, Pitch и Yaw**.

— Это обеспечит полное определение ориентации CubeSat в пространстве и возможность имитации работы реальных систем навигации малых спутников.

2. Реализация фильтра Калмана (Kalman Filter)

— Для повышения точности ориентации планируется внедрить **расширенный фильтр Калмана (EKF)**, который лучше учитывает шумы и динамику движения.

— В отличие от фильтра Madgwick, EKF позволяет более точно оценивать состояние системы, комбинируя показания нескольких датчиков и предсказания модели движения.

3. Модернизация конструкции установки

— Добавление третьей оси вращения в кардановом подвесе,

— Оптимизация креплений и снижение трения подвижных узлов для более точных экспериментов,

— Возможность подключения внешних интерфейсов (например, радиомодуля) для телеметрии.

4. Увеличение мощности и ресурса системы

— **Установка более мощных моторов** для повышения крутящего момента, стабильности и способности компенсировать внешние воздействия.

— **Изменение внутреннего расположения компонентов внутри куба** для улучшения распределения массы, снижения вибраций и повышения устойчивости системы.

— **Улучшенная настройка PID-регулятора**, включая автоматическую подстройку коэффициентов, что позволит добиться более плавного и точного управления ориентацией.

Таким образом, дальнейшее развитие проекта направлено на создание более точной, трёхосевой и интеллектуальной системы стабилизации, максимально приближённой к условиям работы реальных CubeSat на орбите.

Список литературы (на русском)

1. **Калман Р.** «Новый подход к линейной фильтрации и задачам предсказания» // *Труды по автоматике и телемеханике.* — 1960.
2. **Маджвик** «Эффективный фильтр ориентации для инерциальных датчиков IMU/MARG». Перевод технического отчёта Университета Бристоля.
3. **Техническая документация InvenSense:**
 - *MPU-6050 Product Specification,*
 - *MPU-6050 Register Map,*
 - *MPU-9250 Product Specification.*