

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

Задача 1: Модуль системы наведения

В рамках задания требуется разработать независимый модуль спутника форм фактора cubesat. Для выбранного модуля требуется реализовать весь его необходимый функционал, а также корпус cubesat 1u и систему крепления разрабатываемого модуля в нем.

1. Условие задачи:

Реализация прототипа модуля системы наведения на объект для космического аппарата. Модуль состоит из:

- радиомодуля;
- лазерного целеуказателя;
- системы из манипуляторов (моторов, шаговых двигателей, сервоприводов и необходимых для их управления компонентов).

Разрабатываемый модуль должен производить наведение луча лазера из геометрического центра юнита в двух перпендикулярных осях вращения (ось наклона и поворота) с углом обзора 80 градусов по каждой из осей. Ось поворота для наклона проходит через геометрический центр юнита, а также центры двух противоположных боковых сторон, ось поворота проходит через геометрический центр юнита и центры верхней и нижней сторон юнита (Рисунок 1).

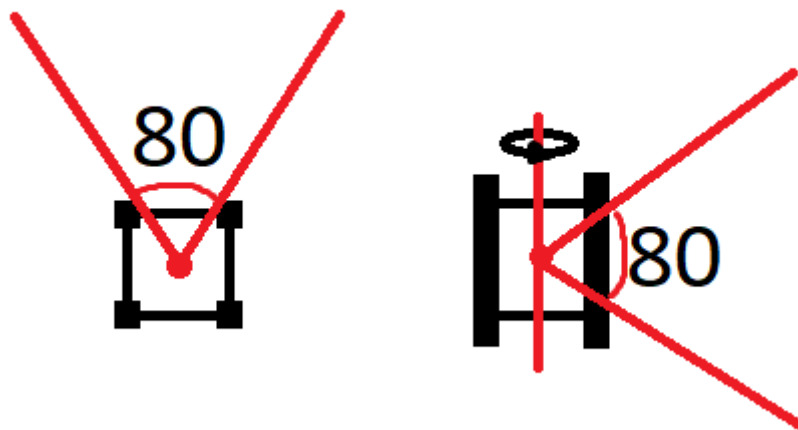


Рисунок 1. Схема угла обзора полезной нагрузки, вид сверху и сбоку.

При получении управляющего радиосигнала устройство должно исполнять тестирующий алгоритм. Алгоритм работы устройства:

1. Включение и инициализация систем;
2. Установка начального состояния устройства;
3. Начала прослушивания радиоканала;
4. Прием и расшифровка входящих сообщений;
5. При получении сообщения со станции:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

- a. Переход в режим вещания сообщений;
- b. До тех пор, пока не окажется во всех позициях:
 - i. Установка нового положения углов наклона и поворота каждые 3 секунды;
 - ii. Отправка сообщения на станцию с параметрами заданных углов наклона и поворота;

6. Возврат к пункту 2.

За время выполнения тестового алгоритма устройство полезной нагрузки должно установить лазер в следующие положения:

1. Устройство сначала выставляется “слева направо” в положение 0 градусов относительно горизонтальной оси и в диапазоне от -40 до +40 градусов относительно вертикальной оси с шагом в 10 градусов;
2. Аналогично “снизу вверх” для положения 0 градусов по вертикальной оси и в диапазоне от - 40 до 40 градусов по горизонтальной оси с шагом в 10 градусов.
3. Затем, “по диагонали” из положения -40, -40 по обеим осям в положение 40, 40 с шагом в 10 градусов по обеим осям.

Проверка работоспособности будет производиться по мишени, с точками положения рассчитанными так, что луч лазера на протяжении испытания исходит из геометрического центра кубсата, расположенного на расстоянии 1 метра от мишени. Данную особенность требуется учитывать программно, если конструкция перемещает точку испускания лазера в пространстве.

Отправляемые сообщения кодируются по определённым правилам перед отправкой на станцию приема. Вышеперечисленные требования к сообщению, а также пример станции приема, на основе радиомодуля nrf24 приведены в приложении 1. Описанная станция будет предоставлена участникам на очной защите, однако, если для выполнения кейса участниками используется иной модуль им также потребуется реализовать и собственную станцию приема для демонстрации на защите.

2. Требования к конструкции устройства.

Отличительной чертой спутника формата cubesat являются направляющие рельсы, предназначенные для помещения спутника в пусковой контейнер. Внутреннее содержимое спутника поддается интерпретации, но размеры и рельсы остаются неизменными. Корпус cubesat должен быть реализован с соблюдением замечаний, указанных на чертеже (Рисунок 2):

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

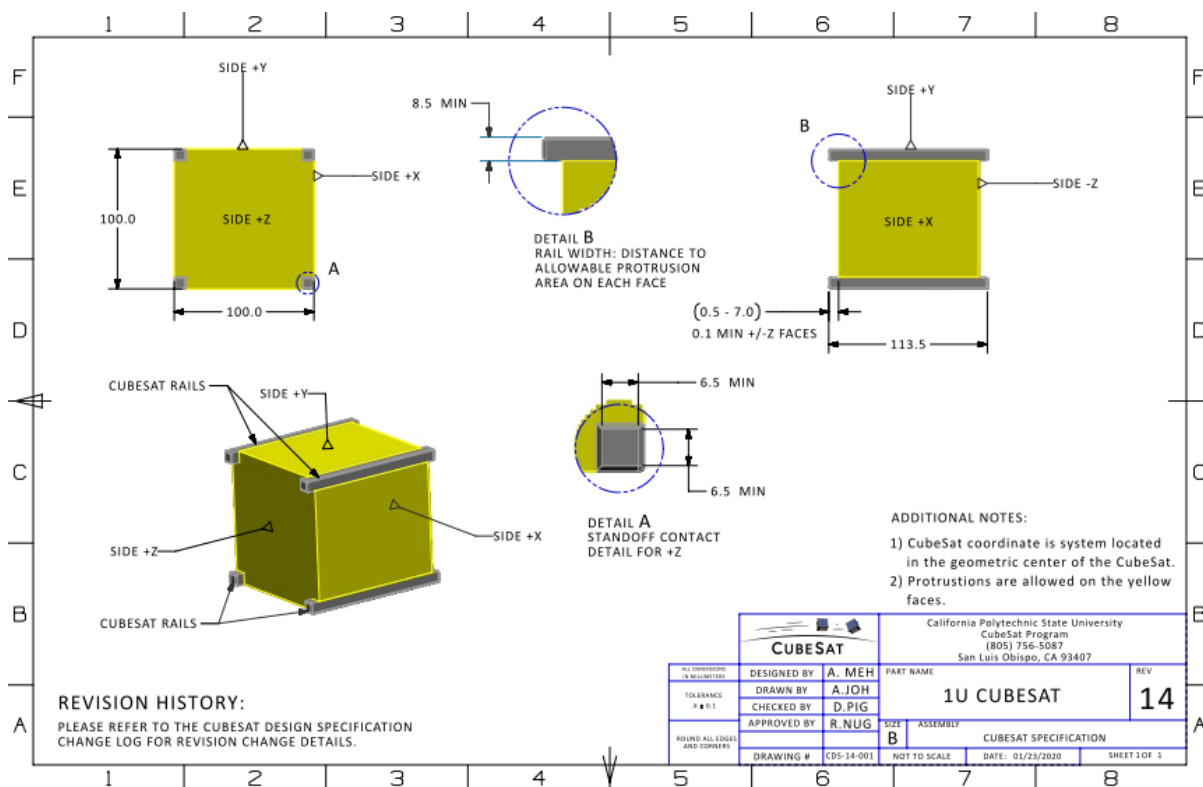


Рисунок 2. Чертеж спутника формфактора Cubesat 1u.

В дополнение к чертежу при проектировании устройства необходимо соблюдать следующие требования:

- Никакие компоненты на сторонах, заштрихованных желтым цветом, не должны выступать дальше, чем на 6,5 мм по нормали к поверхности от плоскости рельса.
- Рельсы должны иметь минимальную ширину 8,5 мм, измеренную от края рельса до первого выступа на каждой грани.
- Края реек должны быть закруглены до радиуса не менее 1 мм.
- Концы направляющих на стороне +/- Z должны иметь минимальную площадь контакта 6,5 мм x 6,5 мм с соседними направляющими CubeSat.
- Необходимо предоставить трёхмерные модели:
 - Корпус модуля приёмника, осуществляющий жёсткое крепление всех элементов модуля (контроллера, приёмника, макетных/паечных плат и т. д.);
 - Корпус спутника cubesat формфактора 1U;
 - Итогового устройства в сборке.

Допускаются конструкторские расхождения в трёхмерной модели и физической реализации. Созданная модель должна удовлетворять требованиям функционирования итогового устройства;

- Необходимо представить программный код для разработанного устройства (формат .txt, .py, .c, .cpp, .h, .hpp, .cxx, .cc, .hxx или иного

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

расширения, в названии файла должен быть указан язык программирования) для каждого этапа;

- Наличие электрической схемы для каждого из этапов.
- 3. Регламент испытания при демонстрации жюри:**
- Демонстрация продукта без включения;
 - Демонстрация включения продукта;
 - Демонстрация алгоритма работы согласно заданию.
- 4. Форма представления результатов:**
- Программный код в виде текстового файла (формат .txt, .py, .c, .cpp, .h, .hpp, .cxx, .cc, .hxx или иного расширения, в названии файла должен быть указан язык программирования));
 - Трёхмерная модель конструкции в формате .stl;
 - Электрическая схема в формате скриншота или .pdf;
 - Демонстрация работы продукта в виде одной или нескольких видеозаписей, наглядно показывающих продукт и его поведение в условиях, описанных в регламенте испытаний;
 - Отчёт, содержащий следующие пункты:
 1. Титульный лист с ФИО участников, наименованием кейса.
 2. Цель, задачи, гипотезу работы.
 3. Этапы проекта.
 4. Инструменты и методы, описание выбранных аппаратных средств, обоснование их применения.
 5. Полученные результаты, рекомендации по их применению и возможное решение выявленных проблем.
- 5. Примерный перечень материалов для выполнения задания:**
Микроконтроллеры (Arduino, Raspberry и пр.), серводвигатели, шаговые моторы, электродвигатели, модуль лазера, лазерная указка, радиомодуль ардуино, батарейный блок.
- 6. Примерный перечень программного обеспечения для выполнения задания:**
Blender, tinkercad.com для 3d-моделирования;
tinkercad.com, fritzing для моделирования электрических схем (tinkercad.com может быть использован для написания программного кода для Arduino);
PyCharm Edu, Arduino IDE, STM32CubeIDE, Notepad++- как среда программирования.
- 7. Ссылки на рекомендуемые методические материалы для восполнения необходимых навыков:**
- Гук А.П., Евстратова Л.Г. Дистанционное зондирование и мониторинг территорий. Часть 1. Дистанционное зондирование.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

- Теоретические основы и технические средства. КУРС, 2019. – 221 с.
- Сутырина Е.Н. Дистанционное зондирование Земли. Учеб. пособие. ИГУ, 2013. — 165 с.
 - Брайан У. Керниган, Роб Пайк. Практика программирования. Вильямс, 2021. — 288 с.
 - Брайан Керниган, Деннис Ритчи. Язык программирования С. Вильямс, 2019. — 288 с.
 - Дмитриев Д., Фомин А., Кармишин А., Дубровская В., Тяпкин Ю., Фатеев А., Борисевич В. Дистанционное зондирование Земли. — Красноярск : Сибирский федеральный университет, 2014. — 196 с.
 - Груздов В.В. Новые технологии Дистанционного Зондирования Земли из космоса. – : Техносфера, 2019. – 482 с.
 - Работа с датчиками. [Электронный ресурс] Режим доступа: <http://zelectro.cc/>
 - Учебник по работе с устройствами. [Электронный ресурс] Режим доступа: [Учебник по работе с устройствами](#)
 - Учебник: Электроника. В.А. Петин. [Электронный ресурс] Режим доступа: [Электроника. В.А. Петин](#)
 - Уроки по работе с ардуино. [Электронный ресурс] Режим доступа: <https://lesson.iarduino.ru/>
 - Уроки по работе с датчиками. [Электронный ресурс] Режим доступа: [Ардуино. Датчики и сети для связи устройств.](#)
 - Проектная работа по разработке системы. [Электронный ресурс] Режим доступа: <http://arduino-diy.com/>
 - Путеводитель по Ардуино. [Электронный ресурс] Режим доступа: [Радио-ежегодник. Путеводитель по Ардуино.](#)
 - Электронные ресурсы 1. ИТЦ «СКАНЭКС» [Электронный ресурс] Режим доступа: <https://www.scanex.ru/>
 - ООО «Спутникс» [Электронный ресурс] Режим доступа: <http://orbicraft.sputnix.ru/doku.php>

Приложение 1.

Передаваемые данные отправляются в формате массива символов длиной в 32 элемента. Передаваемые параметры в массив записываются подряд, при этом перед значением параметра ставится символ, определяющий тип параметра: 'i', 'f', 'v'.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

После буквы *i* в массив записывается 4 символа, соответствующие четырехзначному целому числу.

После буквы *f* в массив записывается 6 символов, соответствующие вещественному числу с 5 значащими символами и 1 точкой.

После буквы *v* в массив записывается 9 символов, соответствующие трем вещественным числам не больше единицы без разделительного знака и двумя разрядами после запятой.

Все незначащие элементы массив в конце данных заполняются нулями.

Пример сообщения:

Пример 1.

Тип	Число				Тип	Вещественное число					Отсутствующие значения								
<i>i</i>	0	0	1	2	<i>f</i>	0	.	1	2	3	4	0	0	0	0	.	.	.	0

Пример 2.

Тип	Ось x вектора			Ось y вектора			Ось z вектора												
<i>v</i>	0	0	1	1	0	0	0	9	8	0	0	0	0	0	0	.	.	.	0

Для приема сообщений с устройства участника реализована приемная станция, использующая следующий программный код, реализованный с применением библиотеки rf24 1.4.6 (<https://nrf24.github.io/RF24/>, доступна к скачиванию через менеджер библиотек Arduino IDE):

```
// MasterSwapRoles

#include <SPI.h>

#include <nRF24L01.h>

#include <RF24.h>

#define CE_PIN 9

#define CSN_PIN 10
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Инженерно-конструкторское направление.
Аэрокосмический профиль.**

```
const byte slaveAddress[5] = {'R','x','A','A','A'};
const byte masterAddress[5] = {'T','X','a','a','a'};

RF24 radio(CE_PIN, CSN_PIN); // инициализация радио

char dataToSend[10] = "Message 0";

char txNum = '0';

char dataReceived[16]; // массив для хранения данных, полученных с
отправителя. Должен совпадать с отправляемым массивом

bool newData = false;

unsigned long currentMillis;

unsigned long prevMillis;

unsigned long txIntervalMillis = 30*1000; // интервал времени, в котором
нужно слушать

//=====

void setup() {

    Serial.begin(9600);

    Serial.println("Старт программы станции");

    radio.begin();

    radio.setDataRate( RF24_250KBPS );

    radio.openWritingPipe(slaveAddress);
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Инженерно-конструкторское направление.
Аэрокосмический профиль.**

```
radio.openReadingPipe(1, masterAddress);

radio.setRetries(3,5); // интервал, счетчик

send(); // для начала работы

prevMillis = millis(); // установка часов
}

//=====

void loop() {

    currentMillis = millis();

    if (currentMillis - prevMillis >= txIntervalMillis) { // отправка команды
раз в интервал

        send();

        prevMillis = millis();

    }

    getData();

    showData();

}

//=====

void send() { // отправка управляющего сообщения

    radio.stopListening();

    bool rslt;

    rslt = radio.write( &dataToSend, sizeof(dataToSend) );

    radio.startListening();
```


**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Инженерно-конструкторское направление.
Аэрокосмический профиль.**

```
Serial.print("Отправляемые данные ");

Serial.print(dataToSend);

if (rslt) {

    Serial.println(" Подтверждение принято");

    updateMessage();

}

else {

    Serial.println(" Не отправилось");

}

}

//=====

void getData() { // получение ответа

    if ( radio.available() ) {

        radio.read( &dataReceived, sizeof(dataReceived) );

        newData = true;

    }

}

//=====

void showData() { // отображение данных

    if (newData == true) {

        Serial.print("Принятые данные: ");

        Serial.println(dataReceived);

        decodeAndPrintData();

    }

}
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Инженерно-конструкторское направление.
Аэрокосмический профиль.**

```
        newData = false;
    }
}

//=====

void updateMessage() {
    // so you can see that new data is being sent

    txNum += 1;

    dataToSend[8] = txNum;
}

//=====

void decodeAndPrintData() { // расшифровка сообщения по шаблону
    int i = 0;
    while (i<sizeof(dataReceived)){
        switch (dataReceived[i]){
            case 'i'://целое число
                char buff_i[4];
                buff_i[0]=dataReceived[i+1];
                buff_i[1]=dataReceived[i+2];
                buff_i[2]=dataReceived[i+3];
                buff_i[3]=dataReceived[i+4];
                Serial.print("Int_data = ");
                Serial.println(atoi(buff_i));
                i+=5;
                break;
        }
    }
}
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Инженерно-конструкторское направление.
Аэрокосмический профиль.**

```
case 'f': //вещественное число

    char buff_f[6];

    buff_f[0]=dataReceived[i+1];
    buff_f[1]=dataReceived[i+2];
    buff_f[2]=dataReceived[i+3];
    buff_f[3]=dataReceived[i+4];
    buff_f[4]=dataReceived[i+5];
    buff_f[5]=dataReceived[i+6];

    Serial.print("Float_data = ");
    Serial.println(atoi(buff_f));

    i+=7;

    break;

case 'v': //вектор

    char buff_vx[4];

    char buff_vy[4];

    char buff_vz[4];

    Serial.print("vector = (");

    buff_vx[0]=dataReceived[i+1];

    buff_vx[1]='.';

    buff_vx[2]=dataReceived[i+2];

    buff_vx[3]=dataReceived[i+3];

    Serial.print(atoi(buff_vx));

    Serial.print(" ");

    buff_vy[0]=dataReceived[i+4];

    buff_vy[1]='.';

    buff_vy[2]=dataReceived[i+5];

    buff_vy[3]=dataReceived[i+6];

    Serial.print(atoi(buff_vy));
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Инженерно-конструкторское направление.
Аэрокосмический профиль.

```
Serial.print(" ");  
  
buff_vz[0]=dataReceived[i+7];  
  
buff_vz[1]='.';  
  
buff_vz[2]=dataReceived[i+8];  
  
buff_vz[3]=dataReceived[i+9];  
  
Serial.print(atoi(buff_vy));  
  
i+=7;  
  
break;  
  
default:  
  
i++;  
  
}  
  
}  
  
}
```