

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА
ШКОЛЬНИКОВ
ГБОУ ШКОЛА №2009

Команда: ъёшогшх

По теме:

Профиль "Инженерия"

Создание образовательного проекта по теме "Модуль системы наведения",
кейс 1

Авторы проекта,
учащиеся 8 и 9 класса,
Струков Егор
Мамедов Леонид
Бабенко Иван

Научный руководитель,
учитель технологии,
Мырадов Мырат Вепаевич

Москва, 2026

РЕФЕРАТ

Ключевые слова: система наведения, CubeSat 1U, ESP32, ESP-NOW, сервоприводы, лазерный целеуказатель, 3D-печать, Arduino IDE.

Цель работы: разработка и создание функционирующего прототипа модуля системы наведения для космического аппарата формата CubeSat 1U с дистанционным управлением по радиоканалу.

В процессе работы были выполнены: проектирование 3D-модели корпуса в Autodesk Fusion 360, подбор и монтаж электронных компонентов (ESP32, сервоприводы SG90, лазерный модуль, преобразователь Mini360), разработка программного обеспечения в Arduino IDE, сборка и испытания системы.

В результате создан автономный модуль, способный по радиокоманде выполнять сканирование лазерным лучом в двух взаимно перпендикулярных осях с углом обзора $\pm 40^\circ$.

Предполагаемое применение: учебно-демонстрационный стенд для изучения основ спутниковых систем, участие в инженерных олимпиадах и конкурсах.

СОДЕРЖАНИЕ

- 1. РЕФЕРАТ**
- 2. ВВЕДЕНИЕ**
 - 2.1. Актуальность
 - 2.2. Цель и задачи
 - 2.3. Ожидаемые результаты
- 3. АНАЛИТИЧЕСКИЙ И КОНСТРУКТОРСКИЙ РАЗДЕЛ**
 - 3.1. Требования к модулю
 - 3.2. Выбор компонентов и обоснование
 - 3.3. Конструкция устройства
 - 3.3.1. 3D-моделирование и изготовление корпуса
 - 3.3.2. Электронная схема и сборка
- 4. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И АЛГОРИТМ РАБОТЫ**
 - 4.1. Средства разработки
 - 4.2. Алгоритм работы
- 5. ИСПЫТАНИЯ И РЕЗУЛЬТАТЫ**
 - 5.1. Методика испытаний
 - 5.2. Полученные результаты
 - 5.3. Соответствие требованиям
- 6. ЗАКЛЮЧЕНИЕ И ВЫВОДЫ**
 - 6.1. Выводы
 - 6.2. Перспективы развития
- 7. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**
- 8. ПРИЛОЖЕНИЯ**
 - Приложение А. Фотографии готового устройства и процесса сборки
 - Приложение Б. Принципиальная схема электрических соединений
 - Приложение В. Код программы
 - Приложение Г. Фрагменты исходного кода
 - Приложение Д. Схема алгоритма

1. ВВЕДЕНИЕ

1.1. Актуальность

Разработка компактных и недорогих систем наведения для малых космических аппаратов (CubeSat) является актуальной задачей в области космического образования и прототипирования. Проект позволяет отработать навыки 3D-моделирования, программирования микроконтроллеров и работы с беспроводными интерфейсами.

1.2. Цель и задачи

Цель: Создание рабочего прототипа модуля системы наведения, совместимого с габаритами CubeSat 1U.

Задачи:

1. Изучить требования к форм-фактору CubeSat 1U.
2. Спроектировать и изготовить корпус модуля методом 3D-печати.
3. Подобрать электронные компоненты и собрать устройство.
4. Разработать программное обеспечение для управления модулем и пультом на базе ESP32 с использованием протокола ESP-NOW.
5. Провести испытания работоспособности и точности наведения.

1.3. Ожидаемые результаты

- Функционирующий модуль, размещаемый внутри куба 100×100×100 мм.
- Система управления с пульта по радиоканалу (ESP-NOW).
- Выполнение циклов сканирования лазерным лучом с шагом 10°.

АНАЛИТИЧЕСКИЙ РАЗДЕЛ

2.1. Требования к модулю

Параметр	Значение/Требование
Габариты	100×100×100 мм (совместимость с CubeSat 1U)

Угол обзора	$\pm 40^\circ$ по осям наклона и поворота
Шаг перемещения	10°
Управление	Дистанционное, по радиоканалу
Протокол связи	ESP-NOW
Основной контроллер	ESP32 (модуль и пульт)
Приводы	Сервоприводы SG90 (2 шт.)
Источник питания	Батарея 9В
Преобразователь	Mini360 (понижающий)
Корпус	PLA, 3D-печать

2.2. Выбор компонентов и обоснование

- Микроконтроллер ESP32: Выбран за наличие встроенного модуля Wi-Fi/Bluetooth, поддержку протокола ESP-NOW для простой и быстрой связи между устройствами, достаточное количество GPIO.
- Сервоприводы SG90: Оптимальны по соотношению размеров, момента и стоимости для создания двухосевой системы поворота.
- Лазерный модуль (650 нм): Используется в качестве имитатора целеуказателя для визуализации наведения.
- Преобразователь Mini360: Необходим для получения стабильного напряжения 5В для питания сервоприводов и ESP32 от батареи 9В.
- Материал корпуса (PLA): Доступный, лёгкий и удобный для 3D-печати материал.

2.3. Конструкция устройства

2.3.1. 3D-моделирование и изготовление корпуса

Проектирование выполнено в Autodesk Fusion 360.

- Создана модель основного корпуса с посадочными местами для ESP32, платы преобразователя и батарейного отсека.

- Разработаны поворотные узлы для крепления сервоприводов, обеспечивающие вращение вокруг двух осей.
- Важно: В конструкции корпуса предусмотрены и вырезаны технологические окна и пазы, обеспечивающие его установку внутрь стандартного каркаса CubeSat 1U с соблюдением зон под направляющие рельсы.
- Все детали распечатаны на 3D-принтере Creality Ender 3 из материала PLA.

2.3.2. Электронная схема и сборка

Принципиальная схема соединений представлена на рисунке (см. Приложение Б). Печатная плата не разрабатывалась; сборка выполнена на макетной плате и с использованием проводов.

- **Цепь питания:** Батарея 9В → Преобразователь Mini360 (настроен на 5В) → ESP32 и сервоприводы.
- **Цепь управления:** ESP32 (выводы GPIO) → Сервоприводы.
- **Система связи:** Две платы ESP32 (одна в модуле, одна в пульте), запрограммированные на обмен данными по протоколу **ESP-NOW**.

3. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И АЛГОРИТМ РАБОТЫ

3.1. Средства разработки

- Среда программирования: Arduino IDE.
- Используемые библиотеки: ESP32 ESP-NOW, Servo.

3.2. Алгоритм работы

1. Инициализация: После включения модуль и пульт инициализируют радиомодуль ESP-NOW и устанавливают соединение.

2. Ожидание команды: Модуль находится в режиме ожидания радиосигнала от пульта.
3. Выполнение цикла сканирования: При получении команды "Старт" модуль последовательно выполняет заданные перемещения сервоприводов, перемещая лазерный луч. Реализованы базовые режимы:
 - Горизонтальное сканирование.
 - Вертикальное сканирование.
4. Передача телеметрии: После каждого шага модуль может отправлять на пульт данные о текущих углах (в демонстрационном режиме).

Блок-схема алгоритма представлена в Приложении В.

4. ИСПЫТАНИЯ И РЕЗУЛЬТАТЫ

4.1. Методика испытаний

1. Проверка механической сборки: Убедились в плавности хода осей, отсутствии заеданий.
2. Тест связи: Проверена устойчивая дальность связи между пультом и модулем в условиях помещения (до 20-25 метров).
3. Тест наведения: На расстоянии 1 метр от модуля размещена мишень. Проверялось попадание лазерной точки в заданные сектора при выполнении программы сканирования.

4.2. Полученные результаты

- Устройство успешно собрано и помещается внутри габаритов CubeSat 1U.
- Система выполняет циклы сканирования. Точность позиционирования луча соответствует шагу в 10° .

4.3. Соответствие требованиям

Созданный прототип соответствует поставленным задачам по габаритам, функциональности (управление по радиоканалу, двухосевое сканирование) и демонстрирует принцип работы системы наведения.

5. ЗАКЛЮЧЕНИЕ И ВЫВОДЫ

5.1. Выводы

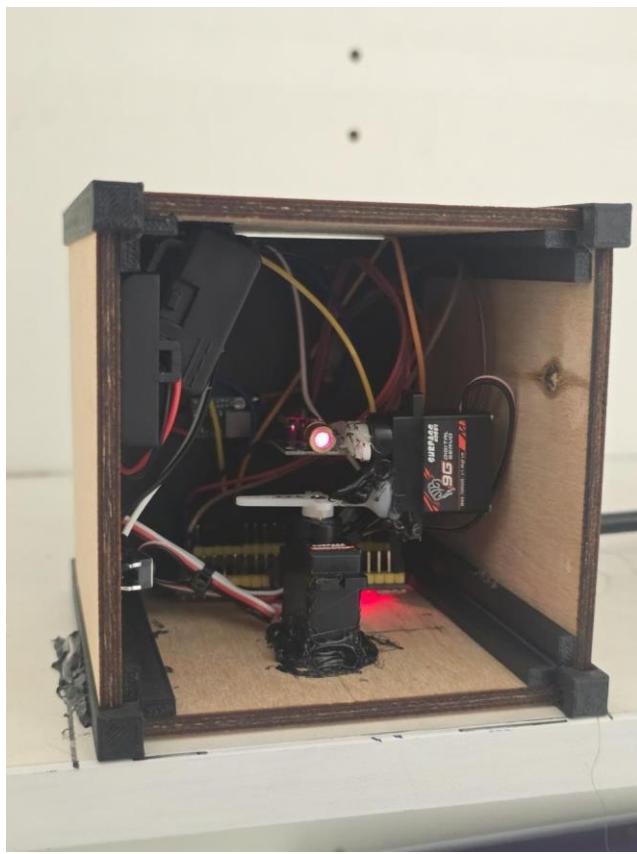
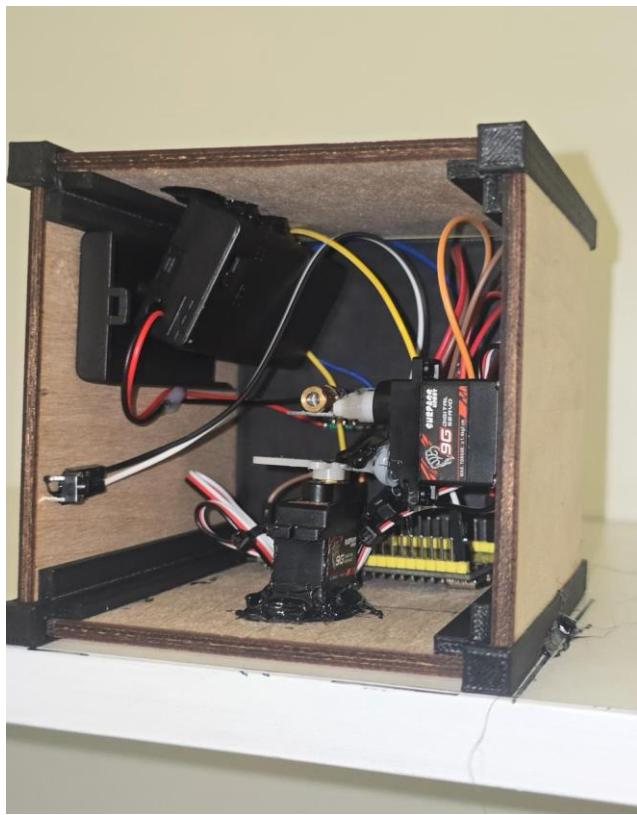
В ходе проекта был успешно разработан, собран и протестирован прототип модуля системы наведения для CubeSat 1U. Цель проекта достигнута, все основные задачи выполнены. Проект позволил получить практический опыт в области 3D-моделирования, программирования микроконтроллеров ESP32 и организации беспроводной связи.

СПИСОК ИСТОЧНИКОВ

1. CubeSat Design Specification. – California Polytechnic State University, 2022.
2. NASA CubeSat Launch Initiative. – URL: <https://www.nasa.gov/cubesat>
3. Ардуино: от азов до создания проектов. – М.: ДМК-Пресс, 2021.
4. Основы проектирования малых космических аппаратов. – М.: МГТУ им. Баумана, 2020.
5. NRF24L01+ Datasheet. – Nordic Semiconductor, 2021.
6. Autodesk Fusion 360 Tutorials. – URL: <https://www.autodesk.com>
7. ГОСТ Р 60.0.0.4-2023. Роботы и роботизированные устройства.
8. ГОСТ 2.702-2011. Правила выполнения электрических схем.

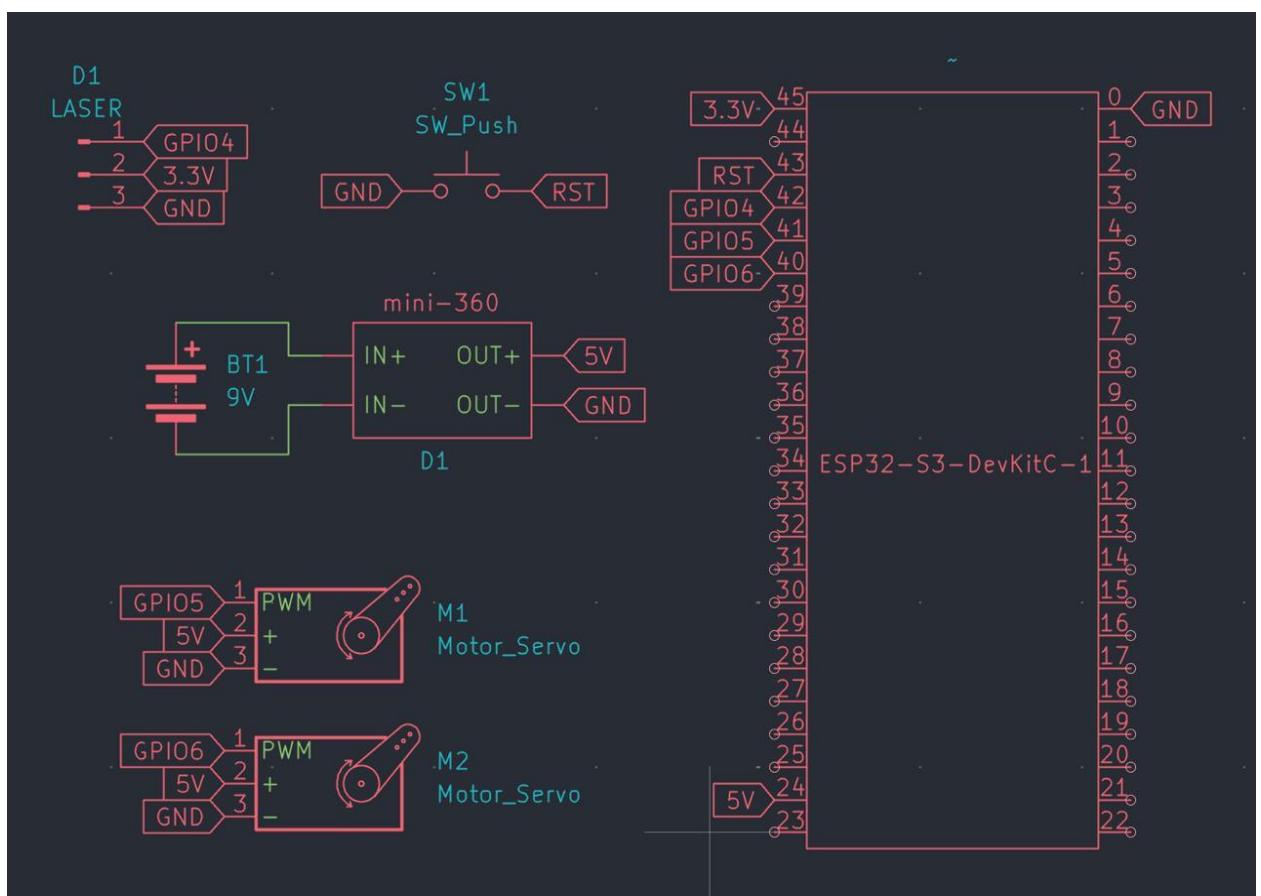
9. Материалы конкурса «Инженеры будущего». – М.: Департамент образования, 2024.

Приложение А



Фотографии готового устройства и процесса сборки

Приложение Б



Принципиальная схема электрических соединений

Приложение В

```
#include <ESP32Servo.h>
#include <esp_now.h>
#include <WiFi.h>

#define MODE_START 1
#define MODE_DATA 2
#define LASER_PIN 4

Servo hor;
Servo ver;

const int h0 = 1;
const int v0 = 65;

int targets[36][2] = {
    { -2, -46 }, { -1, -33 }, { 0, -22 }, { 0, -10 }, { 0, 0 }, { 0, 12 }, { 0, 25 }, { 0, 36 }, { 0, 48 },
    { 0, -45 }, { -1, -33 }, { -1, 0 }, { -10, 0 }, { 0, 0 }, { 12, 0 }, { 25, 5 }, { 36, 5 }, { 50, 6 },
    { -47, -48 }, { -24, -38 }, { -22, -22 }, { -11, -10 }, { 0, 0 }, { 11, 13 }, { 24, 27 }, { 35, 35 }, { 48, 46 },
    { -43, 37 }, { -30, 30 }, { -21, 22 }, { -11, 10 }, { 0, 0 }, { 11, -10 }, { 25, -21 }, { 38, -20 }, { 50, -33 }
}; uint8_t peerMac[] = { 0x08, 0x85, 0xd5, 0xe5, 0xc7, 0x3e, 0x08 };

typedef struct __attribute__((packed)) {
    int32_t id;
    int32_t hor;
    int32_t ver;
    int32_t mode;
} esp_packet_t;

esp_packet_t outgoing;
esp_packet_t incoming;
bool sendBurst = false;

void OnDataRecv(const esp_now_recv_info_t *info, const uint8_t *data, int len) {
    if (len != sizeof(esp_packet_t)) return;
    memcpy(&incoming, data, sizeof(incoming));
    if (incoming.mode == MODE_START) {
        Serial.println("START RECEIVED");
        sendBurst = true;
    }
}

void setup() {
    for (int i = 4; i <= 6; i++) {
        pinMode(i, OUTPUT);
    }

    hor.attach(5);
    ver.attach(6);

    Serial.begin(115200);

    WiFi.mode(WIFI_STA);
    WiFi.disconnect();

    esp_now_init();
    esp_now_register_recv_cb(OnDataRecv);

    esp_now_peer_info_t peerInfo = {};
    memcpy(peerInfo.peer_addr, peerMac, 6);
    esp_now_add_peer(&peerInfo);
}

void loop() {
    hor.write(h0);
    ver.write(v0);
    digitalWrite(LASER_PIN, HIGH);
    if (sendBurst) {
        digitalWrite(LASER_PIN, HIGH);
        for (auto x : targets) {
            outgoing.id = x[0];
            outgoing.hor = x[0];
            outgoing.ver = x[1];
            outgoing.mode = MODE_DATA;

            hor.write(h0 + x[0]);
            ver.write(v0 + x[1]);

            esp_now_send(peerMac, (uint8_t *)&outgoing, sizeof(outgoing));
            delay(3000);
        }
        digitalWrite(LASER_PIN, LOW);
        sendBurst = false;
    }
}
```

```
#include <esp_now.h>
#include <WiFi.h>

#define BTN_PIN 1

uint8_t cubeMac[] = { 0xDC, 0xB4, 0x09, 0x0A, 0x07, 0x4C };
long lastPressTime = 0;

typedef struct __attribute__((packed)) {
    int32_t id;
    int32_t hor;
    int32_t ver;
    int32_t mode;
} esp_packet_t;

esp_packet_t packet;
esp_packet_t incoming;

bool lastButtonState = false;

void OnDataRecv(const esp_now_recv_info_t *info, const uint8_t *data, int len) {
    if (len != sizeof(esp_packet_t)) return;
    memcpy(&incoming, data, sizeof(incoming));
    Serial.print("ID: ");
    Serial.print(incoming.id);
    Serial.print("\nГоризонталы: ");
    Serial.print(incoming.hor);
    Serial.print("\nВертикали: ");
    Serial.print(incoming.ver);
    Serial.print("\nРежим: ");
    Serial.println(incoming.mode);
    delay(500);
    digitalWrite(2, HIGH);
    delay(500);
    digitalWrite(2, LOW);
}

void setup() {
    Serial.begin(115200);

    pinMode(BTN_PIN, INPUT_PULLUP);
    pinMode(2, OUTPUT);

    WiFi.mode(WIFI_STA);
    WiFi.disconnect();

    esp_now_init();
    esp_now_register_recv_cb(OnDataRecv);

    esp_now_peer_info_t peerInfo = {};
    memcpy(peerInfo.peer_addr, cubeMac, 6);
    esp_now_add_peer(&peerInfo);
}

void loop() {
    bool buttonState = !digitalRead(BTN_PIN);

    if (buttonState and !lastButtonState and millis() - lastPressTime > 50) {
        lastPressTime = millis();

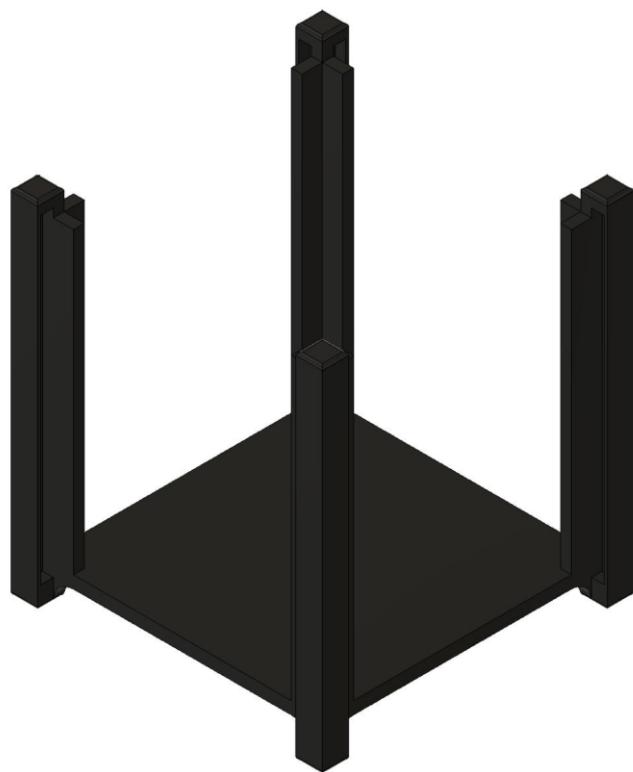
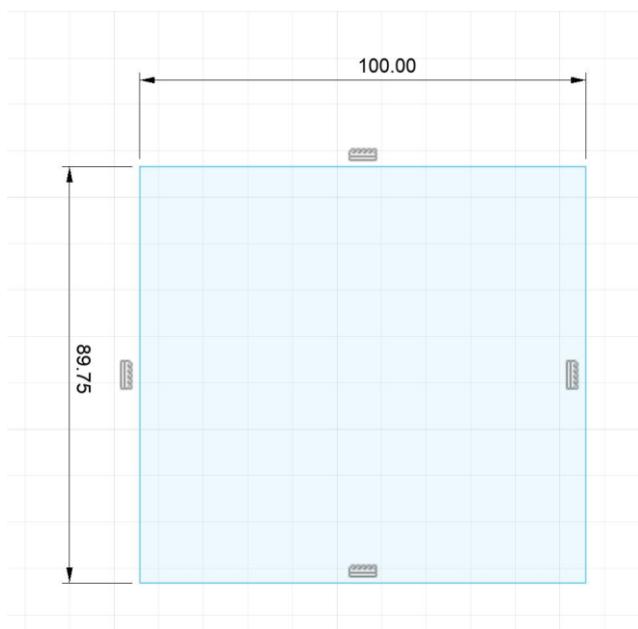
        packet.id = 1;
        packet.hor = 0;
        packet.ver = 0;
        packet.mode = 1;

        Serial.println("START");

        esp_now_send(cubeMac, (uint8_t *)&packet, sizeof(packet));
    }
    lastButtonState = buttonState;
}
```

Код программы

Приложение Г



фрагменты чертежа и 3D модели

Приложение Д

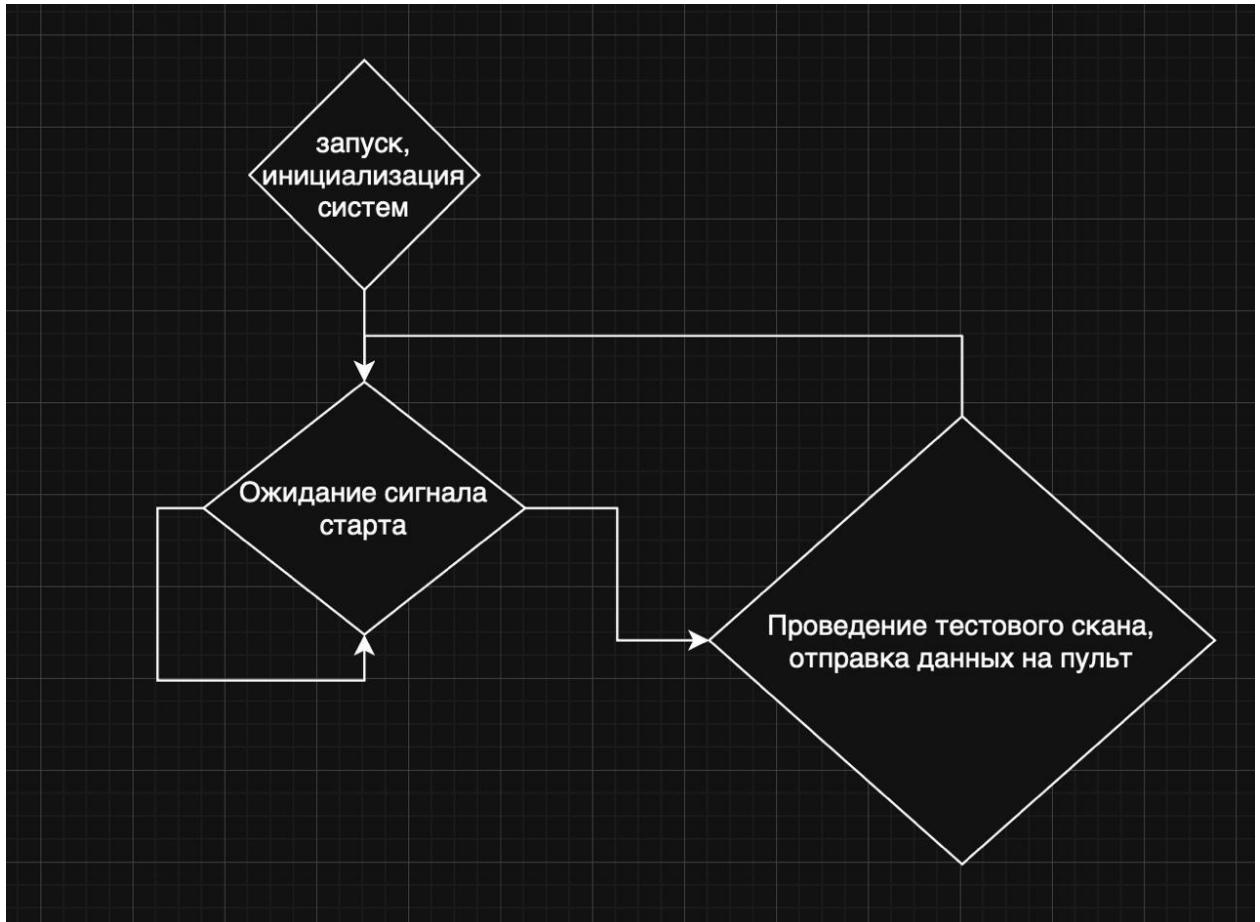


Схема алгоритма