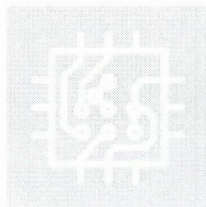


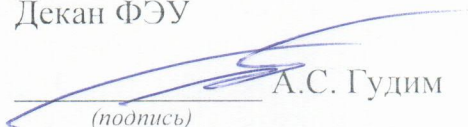
Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»



Работа выполнена в СКБ «Электроника и робототехника»

СОГЛАСОВАНО

Декан ФЭУ

  
(подпись) А.С. Гудим

«17» 06 20 22 г.

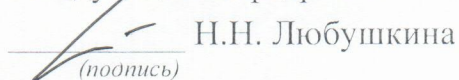
УТВЕРЖДАЮ

Начальник отдела ОНиПКРС

  
(подпись) В.В. Солецкий

«17» 06 20 22 г.

Заведующий кафедрой ПЭ

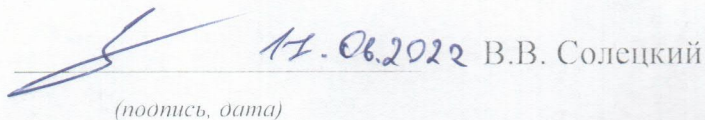
  
(подпись) Н.Н. Любушкина

«17» 06 20 22 г.

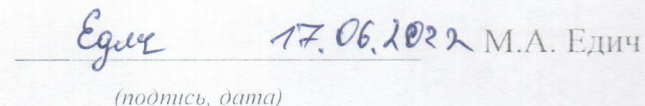
Комплексный увлажнитель воздуха с функцией видеозаписи для  
террариума с науками

Комплект конструкторской документации

Руководитель проекта

  
(подпись, дата) 17.06.2022 В.В. Солецкий

Ответственный исполнитель

  
(подпись, дата) 17.06.2022 М.А. Едич

Комсомольск-на-Амуре 2022

## Карточка проекта

<b>Название</b>	Комплексный увлажнитель воздуха с функцией видеозаписи для террариума с пауками
<b>Тип проекта</b>	<u>По заказу</u> (инициативный, по заказу, в рамках конкурса, учебная работа, другое)
<b>Исполнители</b>	<u>Едич М.А. – 0МР6-1</u> ответственный исполнитель Самков Д.С. – 0ЭЛ6-1
<b>Срок реализации</b>	<u>04.2022-6.2022</u> Месяц, год

## Использованные материалы и компоненты

<b>Наименование</b>	<b>Количество, шт</b>
Плата «ESP-CAM» на контроллере ESP32-S	1
Камера OV2640	1
Wemos Battery Shield	1
TFT SPI дисплей	1
Насос (название)	1
Пластик	1
Оргстекло	1
Мосфета	1

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»



## ЗАДАНИЕ на разработку

**Выдано студентам:** Едич М.А. – 0МРБ-1, Самков Д.С. – 0ЭЛБ-1.

**Название проекта:** Комплексный увлажнитель воздуха с функцией видеозаписи для террариума с пауками.

**Назначение:** Облегчить уход за тропическими пауками.

**Область использования:** Для домашнего пользования.

**Функциональное описание устройства:** Проект представляет собой компактное устройство в виде небольшой коробки, из которого выходят две съемных трубки и коннектор питания. Одна трубка подводится к любому резервуару воды, другая выводится в сам террариум с надетым на нее распылителем. Само устройство обладает защитой от влаги, поэтому располагать его можно как в террариум, так и за ним. Управление данным аппаратом осуществляется через смартфон или ПК удаленно, а также через дисплей находящимся на нем. В процессе работы устройство считывает показания среды террариума, строит на этих данных статистику, регулирует подходящий уровень влажности для пауков и снимает их на камеру. Вся эту информацию можно посмотреть в мобильном приложении или же на сайте в любой точке мира. Устройство автономно и может работать без питания.

**Техническое описание устройства:** Данная коробка представляет из себя распечатанный корпус для МК-типа ESP32 со встроенной камерой, дисплея, небольшого насоса, аккумуляторного шилда, мосфеты, повышающего



модуля и датчика влажности. ESP ориентируясь на уровень влажности и заданных настроек регулирует мощность насоса и расписание его запуска. А также в процессе работы ведет статистику уровня влажности и температуры, включая видеотрансляцию. Все это передается в мобильное приложение, где это все можно посмотреть, и произвести настройку устройства.

**Требования:** Устройство должно быть безопасным, надежным, мобильным, удобным в использовании, практичным, иметь эстетичный внешний вид.

**План работ:**

Наименование работ	Срок
Собрать прототип на макетной плате	09.2021
Написать управляющую программу	10.2021
Поднять сервер на хостинге	11.2021
Реализовать интернет соединение	12.2021
Разработать приложение	01.2022
Собрать опытный образец устройства	02.2022
Тестирование опытного образца	03.2022
Написать документацию по проекту	04.2022

**Комментарии:**

**Перечень графического материала:**

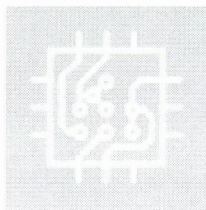
1. Структурная схема;
2. Блок-схема работы устройства;
3. Принципиальная схема;
4. Внешний вид устройства.

Руководитель проекта

  
Подпись/дата

17.06.2021 В.В. Солецкий

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»



### ПАСПОРТ

Комплексный увлажнитель воздуха с функцией видеозаписи для  
terrariuma с пауками

Руководитель проекта

17.08.2022 В.В. Солецкий

Подпись/дата

Ответственный исполнитель

11.08.2022 М.А. Едич

Подпись/дата

Комсомольск-на-Амуре 2022

## Содержание

1	Общие положения .....	3
1.1	Наименование изделия .....	3
1.2	Наименования документов, на основании которых ведется проектирование системы.....	3
1.3	Перечень организаций, участвующих в разработке системы .....	3
1.4	Сведения об использованных при проектировании нормативно-технических документах .....	4
2	Назначение и принцип действия .....	6
2.1	Назначение изделия .....	6
2.2	Области использования изделия .....	6
2.3	Принцип действия.....	6
3	Состав изделия и комплектность.....	7
4	Технические характеристики .....	8
4.1	Основные технические характеристики устройства .....	8
5	Устройство и описание работы изделия.....	9
5.1	Устройство изделия .....	9
5.2	Описание работы изделия .....	10
6	Условия эксплуатации .....	11
6.1	Правила и особенности размещения изделия .....	11
6.2	Меры безопасности.....	11
6.3	Правила хранения и транспортирования.....	12
	ПРИЛОЖЕНИЕ А .....	13

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						2
Изм.	Лист.	№ документа	Подп.	Дата.		



## 1 Общие положения

Настоящий паспорт является документом, предназначенным для ознакомления с основными техническими характеристиками, устройством, правилами установки и эксплуатации устройства «Комплексный увлажнитель воздуха с функцией видеозаписи для террариума с пауками» (далее «изделие»).

Паспорт входит в комплект поставки изделия. Прежде, чем пользоваться изделием, внимательно изучите правила обращения и порядок работы с ним. В связи с постоянной работой по усовершенствованию изделия, повышающей его надежность и улучшающей условия эксплуатации, в конструкцию могут быть внесены изменения, не отраженные в данном издании.

### 1.1 Наименование изделия

Полное наименование системы – Комплексный увлажнитель воздуха с функцией видеозаписи для террариума с пауками.

### 1.2 Наименования документов, на основании которых ведется проектирование системы

Создание комплексного увлажнителя воздуха с функцией видеозаписи для террариума с пауками осуществляется на основании требований и положений следующих документов:

- задание на разработку.

### 1.3 Перечень организаций, участвующих в разработке системы

Заказчиком создания комплексного увлажнителя воздуха с функцией видеозаписи для террариума с пауками является Федеральное

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						3
Изм.	Лист.	№ документа	Подп.	Дата.		

государственное бюджетное образовательное учреждение высшего образования «Комсомольский-на-Амуре государственный университет» (далее заказчик), находящийся по адресу: 681013, Хабаровский край, г. Комсомольск-на-Амуре, Ленина пр-кт., д. 27.

Исполнителями работ по созданию комплексного увлажнителя воздуха с функцией видеозаписи для террариума с пауками являются Конструкторы студенческого конструкторского бюро электротехнического факультета (далее СКБ ФЭУ), студенты группы ОМРБ-1 Едич Максим Андреевич, ОЭЛБ-1 Самков Даниил Сергеевич.

#### **1.4 Сведения об использованных при проектировании нормативно-технических документах**

При проектировании использованы следующие нормативно-технические документы:

ГОСТ 2.001-2013. Единая система конструкторской документации. Общие положения.

ГОСТ 2.102-2013. Единая система конструкторской документации. Виды и комплектность конструкторских документов.

ГОСТ 2.105-95. Единая система конструкторской документации. Общие требования к текстовым документам.

ГОСТ 2.610-2006. Единая система конструкторской документации. Правила выполнения эксплуатационных документов.

ГОСТ 2.004-88. Единая система конструкторской документации. Общие требования к выполнению конструкторских технологических документов на печатающих и графических устройствах вывода ЭВМ.

ГОСТ 2.051-2006. Единая система конструкторской документации. Электронные документы. Общие положения.

ГОСТ 2.052-2006. Единая система конструкторской документации. Электронная модель изделия. Общие положения.

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						4
Изм.	Лист.	№ документа	Подп.	Дата.		



ГОСТ 2.601-2013. Единая система конструкторской документации.  
Эксплуатационные документы.

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		5

## **2 Назначение и принцип действия**

### **2.1 Назначение изделия**

Комплексный увлажнитель воздуха с функцией видеозаписи для террариума с пауками – устройство, предназначенное для облегчение ухода за тропическими пауками.

В состав изделия входят: устройство, трубки.

### **2.2 Области использования изделия**

Изделие может применяться дома.

### **2.3 Принцип действия**

Устройство прикрепляется к любой поверхности к террариума, откуда имеется хороший обзор. После производится первичная настройка через приложение. По завершению устройство готово к работе

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						6
Изм.	Лист.	№ документа	Подп.	Дата.		

### 3 Состав изделия и комплектность

В комплект поставки входит:

- Устройство.
- Трубки.
- Паспорт.

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						7
Изм.	Лист.	№ документа	Подп.	Дата.		

## 4 Технические характеристики

### 4.1 Основные технические характеристики устройства

Основные технические характеристики устройства приведены в таблице 1.

Таблица 1 – Технические характеристики устройства

Наименование параметра	Значение
Сенсор	DHT11
Интерфейсы	USB C, WiFi
Питание, В	5
Тип индикации	TFT SPI дисплей
Емкость аккумулятора, мАч	10000
Габариты, мм <sup>3</sup>	100*100*50
Масса нетто, кг	0.3



## 5 Устройство и описание работы изделия

### 5.1 Устройство изделия

Изделие состоит из двух независимых блоков: блока ПО и блока устройства. Блок ПО – это мобильное приложение позволяющее настроить и получить данные у устройства. Блок устройства – небольшая коробка выполняющая поддержание заданного уровня влажности в террариуме. Структурная схема изделия представлена на рисунке 1.

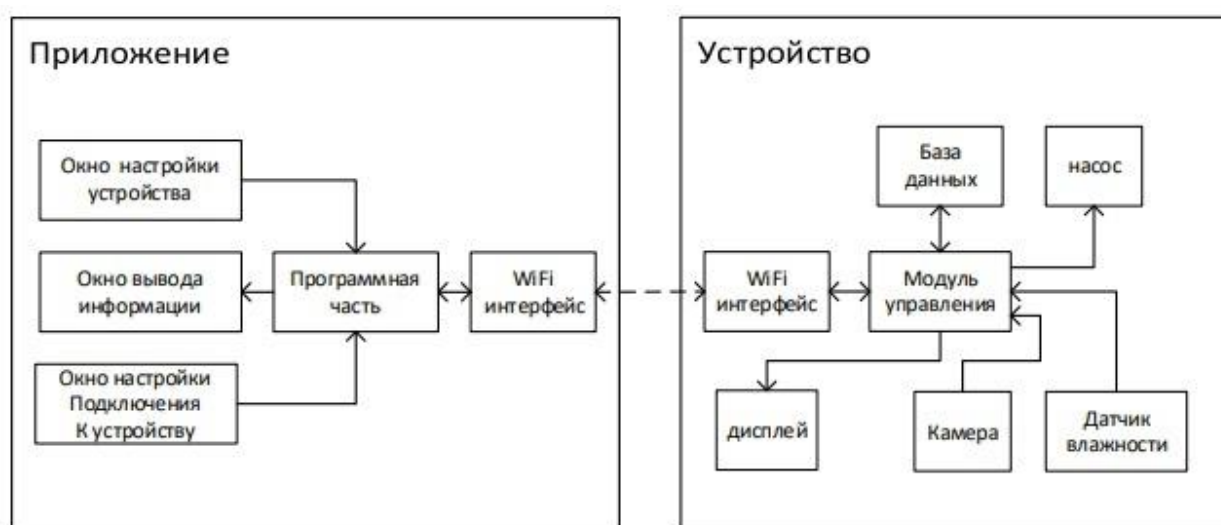


Рисунок 1 – Структурная схема изделия

Устройство управляется микроконтролером ESP-32S с возможностью подключения камеры OV2640. Для вывода информации имеется TFT SPI дисплей 160x128 на базе драйвера ILI9163, на котором отображается краткая информация текущего состояния. Для автономного питания используется плата wemos battery shield v8, позволяющую получать стабильный ток, производить зарядку, а также дающую защиту от переразряда и короткого замыкания. Передача данных осуществляется по беспроводному интерфейсу связи WiFi. Напряжение для зарядки 5 В . Внутри делится с помощью повышающего модуля sp6009 на 12В для насоса и на 5 и 3.3В для остальной

периферии. Для управления насосом используется мосфета. Принципиальная схема представлена в Приложении А.

## 5.2 Описание работы изделия

Перед началом использования изделия необходимо произвести его настройку. Для этого нужно в окне приложения для настройки подключения найти названия устройства отображаемое на дисплее и выбрать его. Далее в окне настройки устройства задать идентификационные данные от домашней точки доступа, а также поддерживаемый уровень влажности и расписание полива и сохранить.

После первичной настройки изделие можно устанавливать в террариуме в любом месте и направлении, где от камеры имеется хорошая видимость.

Автономная работа рассчитана на неделю, по истечению данного срока устройство следует поставить на зарядку.

Вся краткая информация о текущем состоянии отображается на доисплее. Для более подробной, а также для просмотра изображения с камеры нужно открыть в приложении окно вывода информации.

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						10
Изм.	Лист.	№ документа	Подп.	Дата.		

## 6 Условия эксплуатации

Изделие выпускается в климатическом исполнении УХЛ 4.2 по ГОСТ 15150-69 и предназначен для использования в стационарных условиях в закрытых помещениях при соответствующих климатических условиях:

- интервал температур от +10 до +35 °С;
- относительная влажность воздуха до 80 % при температуре +25 °С;
- высота над уровнем моря не более 2000 м;
- атмосферное давление от 86,6 до 106 кПа (от 650 до 800 мм рт. ст.).

В помещении, где используется изделие не должно возникать условий для конденсации влаги (выпадения росы). Изделие является электронным прибором, требующим бережного обращения.

Для обеспечения безотказной работы, сохранения точности и его сбережения необходимо соблюдать следующие правила:

- изучить паспорт, прежде чем приступить к работе с изделием;
- предохранять изделие от ударов и повреждений;
- не прикасаться руками;
- не допускать самостоятельную разборку изделия.

### 6.1 Правила и особенности размещения изделия

Не ронять устройство, может быть нарушена изоляция.

**ВНИМАНИЕ!** При эксплуатации изделия запрещается проводить самостоятельно какие-то либо работы по извлечению и установке внутренних компонентов изделия.

### 6.2 Меры безопасности

Необходимо соблюдать требования техники безопасности и следующие меры предосторожности:

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
						11
Изм.	Лист.	№ документа	Подп.	Дата.		

- не оставлять изделие включенным без наблюдения;
- после транспортировки в холодное время года изделие необходимо выдержать при комнатной температуре не менее двух часов;
- внутренние осмотры и ремонт изделия должны производиться только квалифицированными специалистами;
- не устанавливайте изделие на неустойчивой подставке, стойке или ненадежном кронштейне.

### 6.3 Правила хранения и транспортирования

Транспортирование изделия в упакованном виде может производиться железнодорожным, автомобильным (в закрытых транспортных средствах), воздушным, речным и морским видами транспорта в соответствии с правилами перевозок грузов, действующих на транспорт данного вида. Условия транспортирования изделия по части воздействия климатических факторов должны соответствовать группе 5 по ГОСТ 15150.

После транспортирования изделие должно быть выдержано не менее 2 часов в транспортной таре при температуре  $20\pm 5^{\circ}\text{C}$  и относительной влажности воздуха не более 80%.

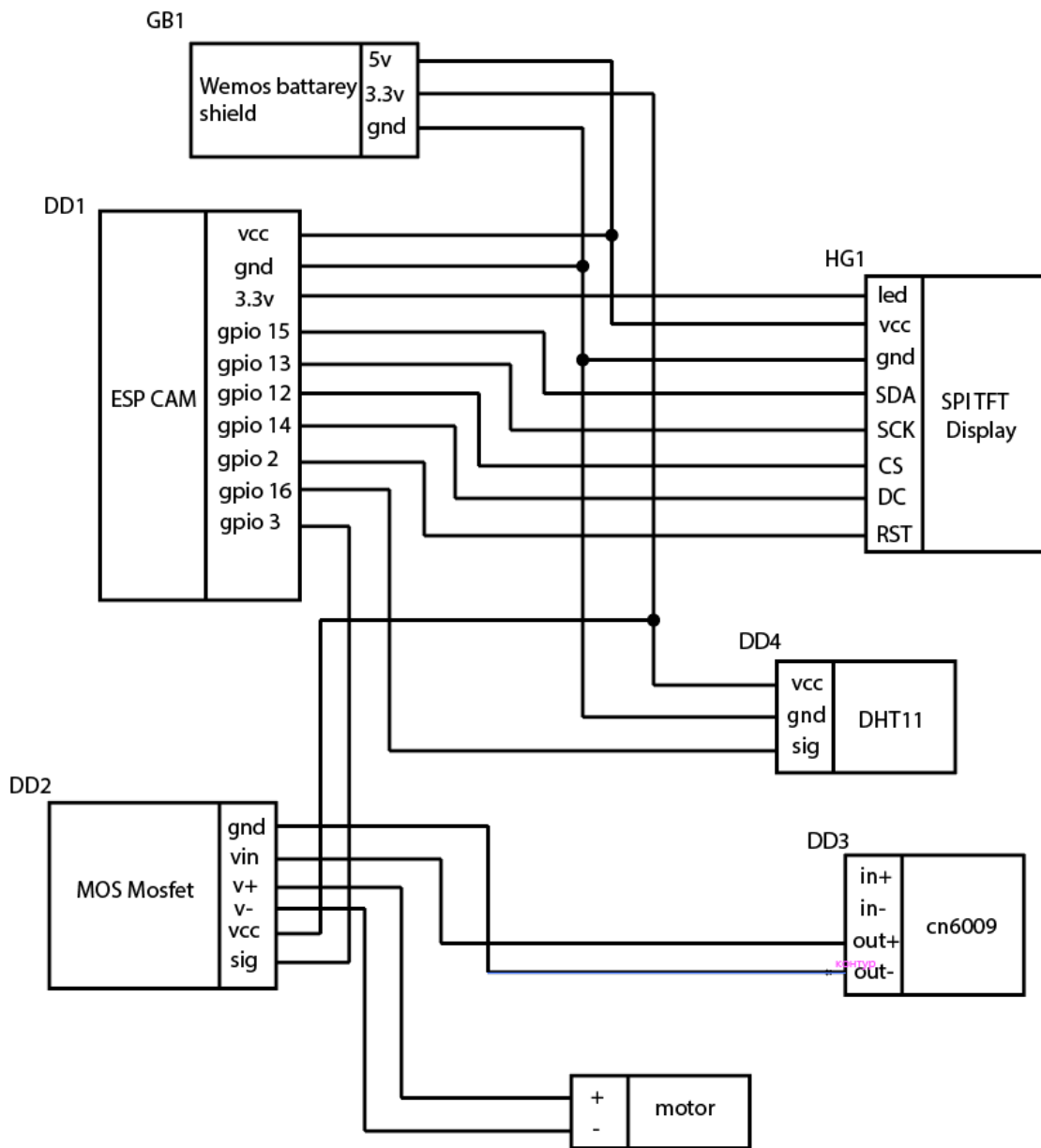
Распакованное изделие должно храниться в отапливаемом и вентилируемом чистом помещении при температуре от  $+5$  до  $+40^{\circ}\text{C}$  и относительной влажности воздуха не более 60 %. При температуре ниже  $25^{\circ}\text{C}$  допускается увеличение относительной влажности до 80 %. Воздух в помещении не должен содержать примесей, вызывающих коррозию металлов, налеты на поверхностях оптических деталей.

					<b>СКБФЭУ.2.ИП.010000ПП</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		12



# ПРИЛОЖЕНИЕ А

(обязательное)



Изм.	Лист.	№ документа	Подп.	Дата.

СКБФЭУ.2.ИП.01000033

Лист

13



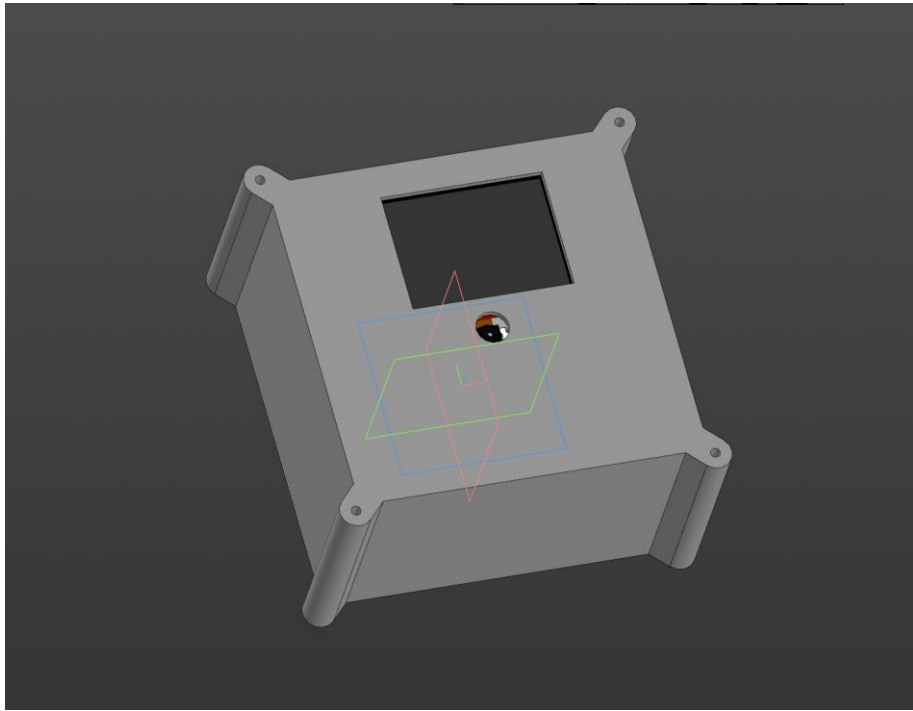


Рисунок А1 – Трехмерная модель изделия в собранном виде

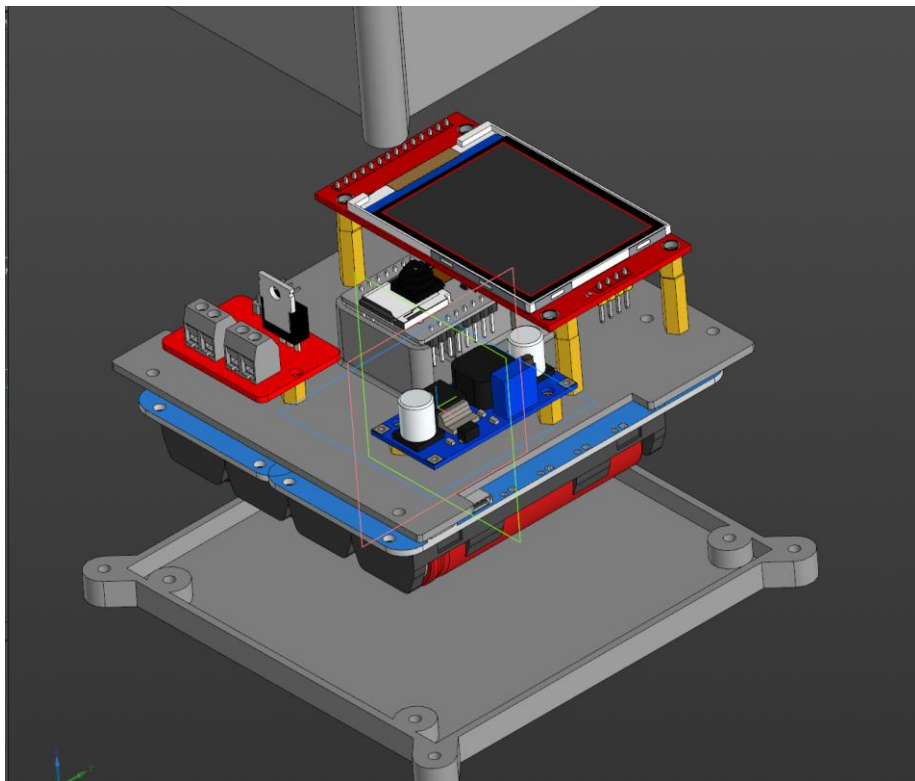


Рисунок А1 – Трехмерная модель изделия в разобранном виде

					СКБФЭУ.2.ИП.010000ИЛ	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		15

## Листинг управляющей программы

```
#include "esp_camera.h"
#include <WiFi.h>
#include <ESPmDNS.h>
#include <WiFiUdp.h>
#include <TFT_eSPI.h>
#include <SPI.h>
#include <EEPROM.h>
#include "DHT.h"

#define CAMERA_MODEL_AI_THINKER // Has PSRAM

#include "camera_pins.h"

#define ApSsid "spider_model-0.1"
#define ApPass "12341234"

TFT_eSPI tft=TFT_eSPI();

WiFiServer server(99);

DHT dht(16, DHT11);

String dataBase = "";

void startCameraServer();

void setup() {
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  Serial.println();

  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
```

					СКБФЭУ.2.ИП.010000ИЛ	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		16



```

config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;

// if PSRAM IC present, init with UXGA resolution and higher JPEG quality
//           for larger pre-allocated frame buffer.
if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

#ifdef CAMERA_MODEL_ESP_EYE
    pinMode(13, INPUT_PULLUP);
    pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		17

```

Serial.printf("Camera init failed with error 0x%x", err);
return;
}

sensor_t * s = esp_camera_sensor_get();
// initial sensors are flipped vertically and colors are a bit saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1); // flip it back
    s->set_brightness(s, 1); // up the brightness just a bit
    s->set_saturation(s, -2); // lower the saturation
}
// drop down frame size for higher initial frame rate
s->set_framesize(s, FRAMESIZE_QVGA);

#if defined(CAMERA_MODEL_M5STACK_WIDE) ||
defined(CAMERA_MODEL_M5STACK_ESP32CAM)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#endif

mainSetup();
SettingWifi();

startCameraServer();

}

WiFiClient client;

unsigned long timer = millis();

void loop() {

Terminal(commandListener());
control();

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		18

```

}

int SettingWifi(){
  int sizea = EEPROM.read(0), sizeb = EEPROM.read(1);
  String s,p;
  for (int i = 0; i<sizea; i++)
    s += char(EEPROM.read(i+2));
  for (int i = 0; i<sizeb; i++)
    p += char(EEPROM.read(i+2+sizea));

  WiFi.mode(WIFI_AP_STA);

  char ssid[sizea+1],pass[sizeb+1];
  s.toCharArray(ssid, sizea+1);
  p.toCharArray(pass, sizeb+1);

  WiFi.begin(ssid, pass);
  delay(5000);
  if (WiFi.status() == WL_CONNECT_FAILED){
    WiFi.softAP(ApSsid, ApPass);
  }

  if(WiFi.localIP().toString()=="0.0.0.0"){
    tft.drawString("wifi not connect, enable ap point",10,10);
    tft.drawString("ip: " + WiFi.softAPIP().toString(),10,20);
  }else{
    String pr = "wifi connect: " + s;
    tft.drawString(pr,10,10);
    tft.drawString("ip: " + WiFi.localIP().toString(),10,20);
  }
}

void control(){

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		19

```

float t = dht.readTemperature();
float h = dht.readHumidity();
tft.drawString("temp: " + String(t),10,40);
tft.drawString("hum: " + String(h),10,50);
String hor = String((millis()/3600000)%24);
String minn = String((millis()/60000)%60);
String sec = String((millis()/1000)%60);
tft.drawString("time: " + hor + " h " + minn + " m " + sec + " s ",10,60);

if(millis() - timer>1000){
  dataBase += "[" + String(t) + "," + String(h) + "," + hor + "," + minn + "," + sec + "]" + "\n";
  timer = millis();
}

void mainSetup(){

  pinMode(4,OUTPUT);
  digitalWrite(4,LOW);
  // pinMode(1,OUTPUT);
  // digitalWrite(1,HIGH);

  server.begin();

  tft.init();
  tft.fillScreen(TFT_BLACK);
  tft.setRotation(3);

  dht.begin();
  EEPROM.begin(128);
}

void Terminal(String com){

  String var[5];

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
						20
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		

```

for(int i = 0,b = 0; i<com.length(); i++){
    if(com.charAt(i)==' ')
        b++;
    else
        var[b]+=com.charAt(i);
}

if (var[0]=="!hello"){

    client.write("xey\n");

}else if (var[0]=="!reboot"){

    ESP.restart();

}else if (var[0]=="!getdataBase"){

    for (int i = 0; i<dataBase.length(); i++)
        client.write(dataBase[i]);

}else if (var[0]=="!setwifi"){
    EEPROM.write(0, (byte)var[1].length());
    EEPROM.write(1, (byte)var[2].length());

    for (int i = 0; i<var[1].length(); i++)
        EEPROM.write(i+2, var[1].charAt(i));

    for (int i = 0; i<var[2].length(); i++)
        EEPROM.write(i+2+var[1].length(), var[2].charAt(i));
    EEPROM.commit();

}else if (var[0]=="!wifiinfo"){

    int sizea = EEPROM.read(0), sizeb = EEPROM.read(1);
    String s,p;
    for (int i = 0; i<sizea; i++)
        s+=char(EEPROM.read(i+2));
    for (int i = 0; i<sizeb; i++)

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		21

```

        p+=char(EEPROM.read(i+2+sizea));
        String r = "ssid: " + s + " pass: " + p + "\n";
        for (int i = 0; i<r.length(); i++)
            client.write(r.charAt(i));
    }
}

```

```

String commandListener(){
    if (!client.connected())
    {
        client = server.available();
        return "";
    }
    else
    {
        String temp2 = "";
        while (client.available() > 0){
            char temp = char(client.read());
            if(temp != '\n')
                temp2 += temp;
        }
        return temp2;
    }
}

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		22

## Листинг управляющего приложения

### Main Activity:

```
package defolt.allspidermater.spider;

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

import android.content.Intent;
import android.media.MediaPlayer;
import android.net.Uri;
import android.util.Log;
import android.view.View;
import android.webkit.WebView;
import android.widget.MediaController;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.ProgressBar;
import android.widget.VideoView;

import com.jjoe64.graphview.GraphView;

import com.jjoe64.graphview.series.DataPoint;
import com.jjoe64.graphview.series.LineGraphSeries;

public class MainActivity extends AppCompatActivity {
    private static String videoSource="http://0.0.0.0";
    private static ClientTcpTask threadConnect;
    private static WebView webView;
    private MediaController mediaController;
    private static int i = 8;
    private static LineGraphSeries<DataPoint> series;
    private static GraphView graph;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        graph = (GraphView) findViewById(R.id.graph);
        series = new LineGraphSeries<DataPoint>();
        graph.addSeries(series);

        webView = findViewById(R.id.videoView);

        webView.getSettings().setJavaScriptEnabled(true);

        String newUA= "Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.0.4) Gecko/20100101 Firefox/4.0";
        webView.getSettings().setUserAgentString(newUA);
        // указываем страницу загрузки
        //webView.loadUrl(videoSource);

    }

    public void startNewActivity(Class c){
        Intent intent = new Intent(this, c);
        startActivity(intent);
    }
}
```

					СКБФЭУ.2.ИП.010000ИЛ	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		23



```

public static void tcpConnection(String ip){
    threadConnect = new ClientTcpTask(ip);
    i++;
    //series.appendData(new DataPoint(i, 8),true,10);
    //graph.notify();
    videoSource="http://" + ip + ":80/stream";
    webView.reload();
    webView.loadUrl(videoSource);

    Log.d("TAG", videoSource);
}

@Override
public boolean onCreateOptionsMenu(Menu menu)
{
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item)
{
    // Операции для выбранного пункта меню
    switch (item.getItemId())
    {
        case R.id.wificonnect_Item:
            startNewActivity(WifiActivity.class);
            return true;
        case R.id.setting_Item:
            startNewActivity(SettingsActivity.class);
            return true;
        case R.id.control_Item:
            startNewActivity(MainActivity.class);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

public void MainReloadEvent(View view) {
    webView.reload();
    Log.d("TAG", videoSource);
    webView.loadUrl(videoSource);
}
}

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		24

## Setting Activity:

```
package defolt.allspidermater.spider;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.widget.AdapterView;
```

```
import android.widget.Spinner;
```

```
public class SettingsActivity extends MainActivity {
```

```
    String[] models = { "SpiderMod_v1.2" };
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_settings);
```

```
        Spinner selectedModel = findViewById(R.id.selectModel);
```

```
        ArrayAdapter<String> adapter = new ArrayAdapter(this, android.R.layout.simple_spinner_item, models);
```

```
        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
```

```
        selectedModel.setAdapter(adapter);
```

```
    }
```

```
}
```

					СКБФЭУ.2.ИП.010000ИЛ	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		25

## WifiActivity

```
package defolt.allspidermater.spider;

import androidx.annotation.NonNull;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.annotation.TargetApi;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.database.Observable;
import android.net.ConnectivityManager;
import android.net.DhcpInfo;
import android.net.Network;
import android.net.NetworkCapabilities;
import android.net.NetworkInfo;
import android.net.NetworkRequest;
import android.net.NetworkSpecifier;
import android.net.Uri;
import android.net.wifi.ScanResult;
import android.net.wifi.WifiConfiguration;
import android.net.wifi.WifiInfo;
import android.net.wifi.WifiManager;
import android.net.wifi.WifiNetworkSpecifier;
import android.net.wifi.WifiNetworkSuggestion;
import android.os.AsyncTask;
import android.os.Build;
import android.os.Bundle;
import android.provider.Settings;
import android.text.Editable;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Switch;
import android.widget.Toast;

import com.google.android.material.snackbar.Snackbar;

import java.io.IOException;
import java.net.InetAddress;
import java.net.InetSocketAddress;
import java.net.NetworkInterface;
import java.net.SocketException;
import java.net.UnknownHostException;
import java.util.ArrayList;
import java.util.List;

import static android.net.wifi.WifiManager.STATUS_NETWORK_SUGGESTIONS_SUCCESS;
```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		26

```

public class WifiActivity extends MainActivity implements CompoundButton.OnCheckedChangeListener {

    private boolean modeSearch = false;
    SearchWifiAdapter Wifiadapter;
    SearchModelAdapter modadapter;
    ArrayList<ItemModel> models;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_wifi);

        Switch sw = findViewById(R.id.switchSearchMode);
        if (sw != null) {
            sw.setOnCheckedChangeListener(this);
        }

        detectModel();

    }

    public int intToIp(int i) {

        return
            (((i >> 24) & 0xFF) + "." +
            ((i >> 16) & 0xFF);
            //+ "." + ((i >> 8) & 0xFF) + "." +
            // (i & 0xFF);
    }

    private class ipScan extends AsyncTask<Void, String, Void> {

        /*
        Scan IP 192.168.1.100~192.168.1.110
        you should try different timeout for your network/devices
        */
        static final String subnet = "192.168.0.";
        static final int lower = 100;
        static final int upper = 255;
        static final int timeout = 500;

        @Override
        protected void onPreExecute() {
            models.clear();
            modadapter.notifyDataSetChanged();
            Log.d("TAG", "Scan IP...");
            WifiManager wifii= (WifiManager) getApplicationContext().getSystemService(Context.WIFI_SERVICE);
            DhcpInfo d= wifii.getDhcpInfo();
            Log.d("TAG", "Subnet Mask: "+intToIp(d.ipAddress));
        }

        @Override
        protected Void doInBackground(Void... params) {

            for (int i = lower; i <= upper; i++) {
                String host = subnet + i;

                try {

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		27

```

        InetAddress inetAddress = InetAddress.getByName(host);
        if (inetAddress.isReachable(timeout)){
            publishProgress(inetAddress.getHostByName(),inetAddress.getCanonicalHostName());
        }

    } catch (UnknownHostException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

return null;
}

@Override
protected void onProgressUpdate(String... values) {
    models.add(new ItemModel(values[0], values[0]));
    modadapter.notifyDataSetChanged();
    Log.d("TAG", values[1]);
}

@Override
protected void onPostExecute(Void aVoid) {
    Log.d("TAG", "Done");
}
}

public void detectModel() {
    models = new ArrayList<ItemModel>();

    modadapter = new SearchModelAdapter(this, R.layout.select_wifi_item, models);
    ListView modelList = findViewById(R.id.modelList);
    modelList.setAdapter(modadapter);

    new ipScan().execute();

}

public void detectWifi() {
    ArrayList<ItemWiFi> ap = new ArrayList<ItemWiFi>();
    WifiManager wifiManager = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);
// Get List of Available Wifi Networks
    List<ScanResult> results = wifiManager.getScanResults();

    Log.d("TAG", results.toString());

    for (int i = 0; i < results.size(); i++) {
        String item = results.get(i).toString();
        String[] vector_item = item.split(",");
        String item_essid = vector_item[0];
        String item_capabilities = vector_item[2];
        String item_level = vector_item[3];
        String ssid = item_essid.split(": ")[1];
    }
}

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		28

```

String security = item_capabilities.split(": ")[1];
String level = item_level.split(":")[1];
ap.add(new ItemWiFi(ssid, security, level));
}

Wifiadapter = new SearchWifiAdapter(this, R.layout.select_wifi_item,
ap,this.findViewById(android.R.id.content));
ListView modelList = findViewById(R.id.modelList);
modelList.setAdapter(Wifiadapter);
}

public void resetSearchEvent(View view) {
    if (this.modeSearch) {
        Wifiadapter.disconnectFromNetwork();
        detectWifi();
    } else {
        detectModel();
    }
}

@Override
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
    // Toast.makeText(this, "Отслеживание переключения: " + (isChecked ? "on" : "off"),
    //     Toast.LENGTH_SHORT).show();
    this.modeSearch = isChecked;
    if (this.modeSearch) {
        detectWifi();
    } else {
        detectModel();
    }
}
}

```

					<b>СКБФЭУ.2.ИП.010000ИЛ</b>	Лист
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		29

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

УТВЕРЖДАЮ

Декан ФЭУ

  
А.С. Гудим

« 17 » 06 2022 г.

СОГЛАСОВАНО

Заведующий кафедрой ФЭУ

  
Н.Н. Любушкина

« 17 » 06 2022 г.

### АКТ

о приемке в эксплуатацию аппаратно-программного комплекса  
увлажнителя воздуха с функцией видеозаписи для террариума с пауками

г. Комсомольск-на-Амуре

« 17 » 06 2022 г.

Комиссия в составе представителей:

заказчика

- В.В. Солецкий – руководитель СКБ ФЭУ,
- Н.Н. Любушкина – Заведующий кафедрой ФЭУ,
- А.С. Гудим – декана ФЭУ

исполнителя

- М.А. Едич – 0МРб-1,
- Д.С. Самков – 0ЭЛб-1

составила акт о нижеследующем:



«Исполнитель» передает аппаратно-программный комплекс «увлажнитель воздуха с функцией видеозаписи для террариума с пауками», в составе:

Оборудование, в составе:

- Устройства
- приложения

Программное обеспечение, в том числе:

- Рабочие программы управления изделием.


Эксплуатационная документация:

- Паспорт изделия

Аппаратно-программный комплекс «увлажнитель воздуха с функцией видеозаписи для террариума с пауками» прошел опытную эксплуатацию с «25» 04 по «10» 05 2022 г. и признан годным к эксплуатации. Были протестированы все режимы функционирования, отказы системы, а также аварийные отключения по вине системы не наблюдались.

Руководитель СКБ ФЭУ

Ответственный исполнитель

 / В.В. Солецкий /

 / М.А. Едич /



