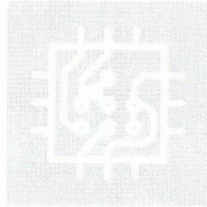


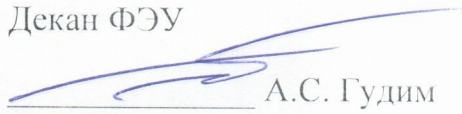
Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»



Работа выполнена в СКБ «Электроника и робототехника»

СОГЛАСОВАНО

Декан ФЭУ

  
А.С. Гудим  
(подпись)

« 17 » 06 20 22 г.

УТВЕРЖДАЮ

Начальник отдела ОНИПКРС

  
В.В. Солецкий  
(подпись)

« 17 » 06 20 22 г.

Заведующий кафедрой ПЭ

  
Н.Н. Любушкина  
(подпись)


« 17 » 06 20 22 г.

**Разработка автономного с/х робота**

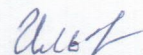
**«Agro RoboPlant»**

**Комплект конструкторской документации**

Руководитель проекта

  
17.06.2022 А.С. Гудим  
(подпись, дата)

Ответственный исполнитель

  
17.06.2022 Е.С. Ильченко  
(подпись, дата)

Комсомольск-на-Амуре 2022

## Карточка проекта

<b>Название</b>	Разработка автономного с/х робота «Agro RoboPlant»
<b>Тип проекта</b>	<u>В рамках конкурса</u> (инициативный, по заказу, в рамках конкурса, учебная работа, другое)
<b>Исполнители</b>	<u>Ильченко Е.С. – 0БМб-1</u> ответственный исполнитель
<b>Срок реализации</b>	<u>10.2021-06.2022</u> Месяц, год

## Использованные материалы и компоненты

Наименование	Количество, шт.
Arduino UNO R3	1
WiFi модуль	1
WiFi антенна	1
ИК датчики	2
Ультразвуковой датчик	1
Шасси в сборе	1
Кронштейн камеры	1
Сервопривод	4
Лампочки фары	2
HD камера	1
Аккумулятор	1
Детали роборуки	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:** Ильченко Е.С. - ОБМб-1

**Название проекта:** Разработка автономного с/х робота «Agro RoboPlant».

**Назначение:** агроробот, предназначен для сборки урожая, обработки почвы, автополива.

**Область использования:** Изделие может применяться дома, на ферме, в сельском хозяйстве.

**Функциональное описание устройства:** По команде оператора робот автономно выполняет заданные ему задачи: автополив, сбор урожая, срывание сорняков. По заданному ему периметру он выполняет функции, которые поручил оператор. Тем временем оператор через приложение может наблюдать за его маршрутом и то, что выполняет робот.

**Техническое описание устройства:** Система управления, лежащая в основе «Agro-RoboPlant» является универсальной и может быть установлена на любую спецтехнику или трактор. Электроника, антенны, датчики и вспомогательное оборудование «Agro-RoboPlant» расположены на специальном корпусе, который устанавливается вместо привычной кабины на новую или уже существующую основу. Контролер передает информацию в мобильный телефон, который может контролировать одновременную работу сразу нескольких десятков единиц техники.

**Требования:** Агроробот, должен быть универсальным, платформонезависимым, безопасным, иметь возможность проверки через приложение, автономно управляемым в сложных условиях.

**План работ:**

Наименование работ	Срок
Разработать структурную схему	10.2021
Определить список комплектующих	10.2021
Собрать прототип на макетной плате	11.2021
Составить блок-схемы и написать программы	02.2022
Разработать плату расширения	03.2022
Собрать опытный образец	05.2022
Составить паспорт	05.2022
Провести испытания и демонстрацию готового изделия	05.2022

**Комментарии:**

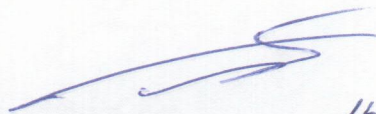
---

---

**Перечень графического материала:**

1. Принципиальная схема
2. Блок-схема
3. Чертежи изделия
4. Внешний вид изделия

Руководитель проекта



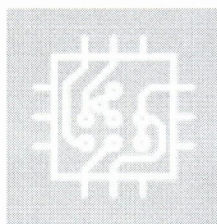
16.09.2021

А.С. Гудим

(подпись, дата)

Министерство науки и высшего образования Российской Федерации

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



## ПАСПОРТ

Разработка автономного с/х робота

«Agro RoboPlant»

Руководитель проекта

Подпись/дата

*17.06.2022* А.С. Гудим

Ответственный исполнитель

Подпись/дата

*17.06.2022 Ильченко*

Е.С. Ильченко

Комсомольск-на-Амуре 2022

## Содержание

1	Общие положения .....	7
1.1	Наименование изделия .....	7
1.2	Наименования документов, на основании которых ведется проектирование системы.....	7
1.3	Перечень организаций, участвующих в разработке системы .....	7
1.4	Сведения об использованных при проектировании нормативно-технических документах .....	8
2	Назначение и принцип действия .....	9
2.1	Назначение изделия .....	9
2.2	Области использования изделия .....	9
2.3	Принцип действия.....	9
3	Состав изделия и комплектность.....	10
4	Технические характеристики .....	11
4.1	Основные технические характеристики блока агробота .....	11
4.2	Основные технические характеристики роборуки.....	12
5	Устройство и описание работы изделия.....	13
5.1	Устройство изделия .....	13
5.2	Описание работы изделия .....	14
6	Условия эксплуатации .....	15
6.1	Правила и особенности размещения изделия .....	15
6.2	Меры безопасности.....	16
6.3	Правила хранения и транспортирования.....	16

					<b>СКБФЭУ.2.ИП.010000ЭЗ</b>	Лист
						6
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 «Agro RoboPlant» (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 - автономный с/х робот Agro RoboPlant.

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

Создание АПК Agro RoboPlant осуществляется на основании требований и положений следующих документов:

- задание на разработку.

### **1.3 Перечень организаций, участвующих в разработке системы**

Заказчиком создания АПК Agro RoboPlant является Федеральное 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.

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

Исполнителем работ по созданию Agro RoboPlant являются Конструкторы студенческого конструкторского бюро электротехнического факультета (далее СКБ ФЭУ), студент группы) ОБМБ-1 Ильченко Елизавета Сергеевна.

#### **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.

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

					<b>СКБФЭУ.2.ИП.010000ЭЗ</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		8



## 2 Назначение и принцип действия

### 2.1 Назначение изделия

Аграрный с/х робот. Предназначен для минимизации человеческого труда на ферме.

В состав изделия входят: машина на гусеничном ходу, роборука, источник питания, электроника.

### 2.2 Области использования изделия

Изделие может применяться дома или на ферме.

### 2.3 Принцип действия

По команде оператора робот автономно выполняет заданные ему задачи: автополив, сбор урожая, срывание сорняков. По заданному ему периметру он выполняет функции, которые поручил оператор. Тем временем оператор через приложение может наблюдать за его маршрутом и то, что выполняет робот.

					<b>СКБФЭУ.2.ИП.010000ЭЗ</b>	Лист
						9
Изм.	Лист.	№ документа	Подп.	Дата.		

### 3 Состав изделия и комплектность

В комплект поставки входит:

- Паспорт.
- Агроробот.
- Батарея.
- Модуль питания машины.
- Комплект кабелей.

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

## 4 Технические характеристики

### 4.1 Основные технические характеристики блока агробота

Основные технические характеристики агробота приведены в таблице 1.

Таблица 1 – Технические характеристики агробота

Наименование параметра	Значение
Драйвер мотора	ТВ6612
Частота WiFi	2.400 ~ 2.4835 ГГц
Питание, В	литиевый аккумулятор 1800 мАч 12 В
Камера	WiFi камера высокого разрешения
Дальность сигнала	< 80 м
Протокол беспроводной связи	802.11b/g/n, до 150 Мбит/с
Кнопка выключения	есть
Кнопка выключения	Удаленная, через приложение
Управление	Через приложение
Функциональные возможности	видеосъемка и трансляция видео, движение по линии, обнаружение и объезд препятствий, движение за объектом
Габариты, мм	220 мм x 256 мм x 260 мм
Масса нетто, кг	2

## 4.2 Основные технические характеристики роборуки

Основные технические характеристики роборуки приведены в таблице 2.

Таблица 2 – Технические характеристики роборуки

Наименование параметра	Значение
Функциональные возможности	Возможность захвата предмета
Материал	пластик
Питание, В	7,5 В
Тип индикации	LCD дисплей, Пьезодинамик
Кнопка выключения	есть
Кнопка сброса	есть
Габариты ,мм	285 мм х 120 мм х 465 мм
Масса нетто, кг	1,24

## 5 Устройство и описание работы изделия

### 5.1 Устройство изделия

Изделие состоит из двух объектов машина на гусеничном ходу и роборука прикрепленная к каркасу машины. Объект машины это железный каркас, где два гусеничных хода по бокам с моторами. Объект роборуки напечатан на 3D принтере, состоит из пластика, собранный с сервомоторами. Структурная схема изделия представлена на рисунке 1.

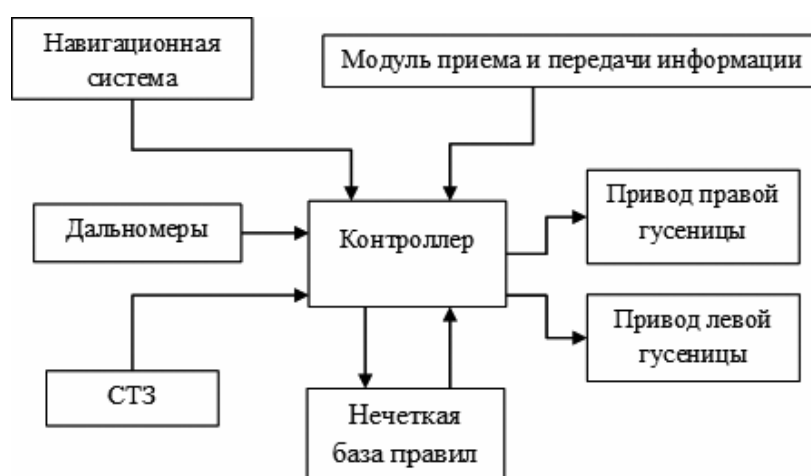


Рисунок 1 – Структурная схема изделия

Управление агроботом выполняется микроконтроллером Arduino UNO R3, через мобильное приложение, с помощью ПО, ПК. Агробот имеет кнопку включения, после включается модуль Wi-Fi или Bluetooth. На выбор предлагается в приложении способ подключения к машине. По заданному маршруту агробот проходит путь и выполняет заданные на выбор функции. Принципиальная схема оружия представлена в Приложении А.

## 5.2 Описание работы изделия

Перед началом использования изделия необходимо задать маршрут агророботу. Дать пройти этот путь агророботу и определить то, что заданный маршрут верный.

Заранее зарядить батарею и подключить ее к агроботу. Включить кнопку питания платы, подключить USB кабель камеры, модуль WIFI. Включить приложение на телефоне или ПК.

Агроботу через приложение задать функцию на выбор: автополив, сбор урожая, уничтожение сорняков. Когда задали, агробот самостоятельно начнет выполнять заданную ему работу. Через приложение можно наблюдать за роботом на расстоянии.

Блок-схемы работы управляющих программ приведены в Приложении А.

					<b>СКБФЭУ.2.ИП.010000ЭЗ</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		14

## 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 допускать самостоятельную разборку изделия.

### 6.1 Правила и особенности размещения изделия

Изделие должно быть расположено на расстоянии не менее 1 м от нагревательных приборов.

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

					<b>СКБФЭУ.2.ИП.010000ЭЗ</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		15

## 6.2 Меры безопасности

Необходимо соблюдать требования техники безопасности и следующие меры предосторожности:

- 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  $20 \pm 5$  °С и относительной влажности воздуха не более 80 %.

Распакованное изделие должно храниться в отапливаемом и вентилируемом чистом помещении при температуре от +5 до +40 °С и относительной влажности воздуха не более 60 %. При температуре ниже 25 °С 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.010000ЭЗ</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		16



# ПРИЛОЖЕНИЕ А

(обязательное)

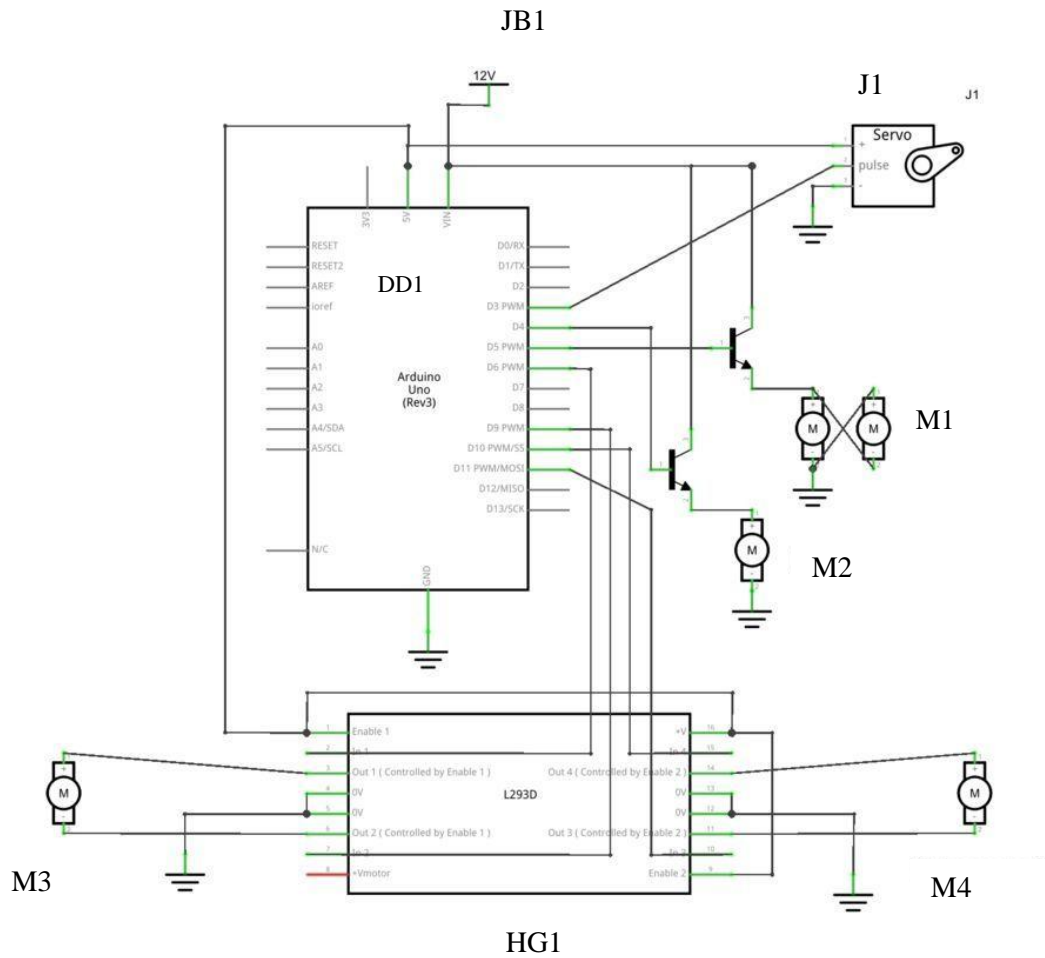


Рисунок А1 – Принципиальная схема агробота

Поз. обознач.	Наименование	Кол.	Примечание
	<u>Микросхемы</u>		
DD1	Arduino UNO	1	
J1	Сервопривод	1	
	<u>Источники питания</u>		
JB1	Li-Ion аккумулятор 12 Вольт, 3000 мА/ч.	1	
	<u>Индикаторы</u>		
HG1	L293D	1	
	<u>Моторы</u>		
M3, M4	Двигатель	2	
	<u>Датчики</u>		
M2	Ультразвуковой датчик	1	
M1	ИК датчик	1	

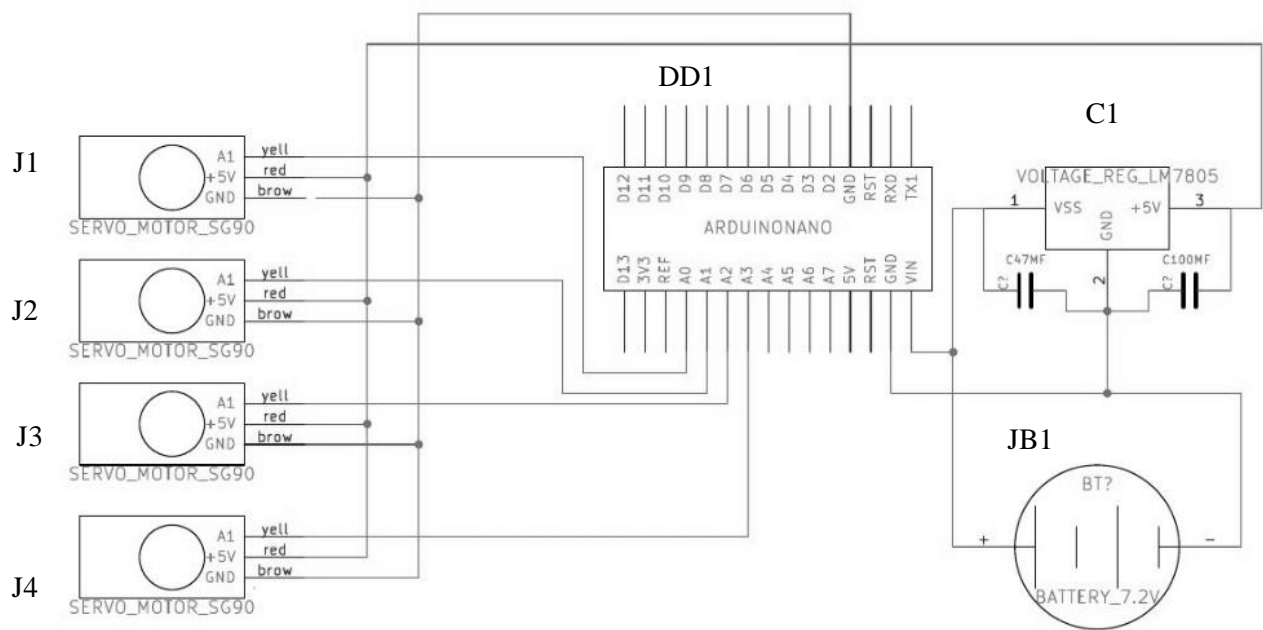


Рисунок А2 – Принципиальная схема роборуки

Поз. обо- знач.	Наименование	Кол.	Примечание
	<u>Микросхемы</u>		
DD1	Arduino NANO	1	
J1-J4	Сервомотор	4	
	<u>Источники питания</u>		
GB1	Батарея	1	
C1	Регулятор напряжения	1	

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

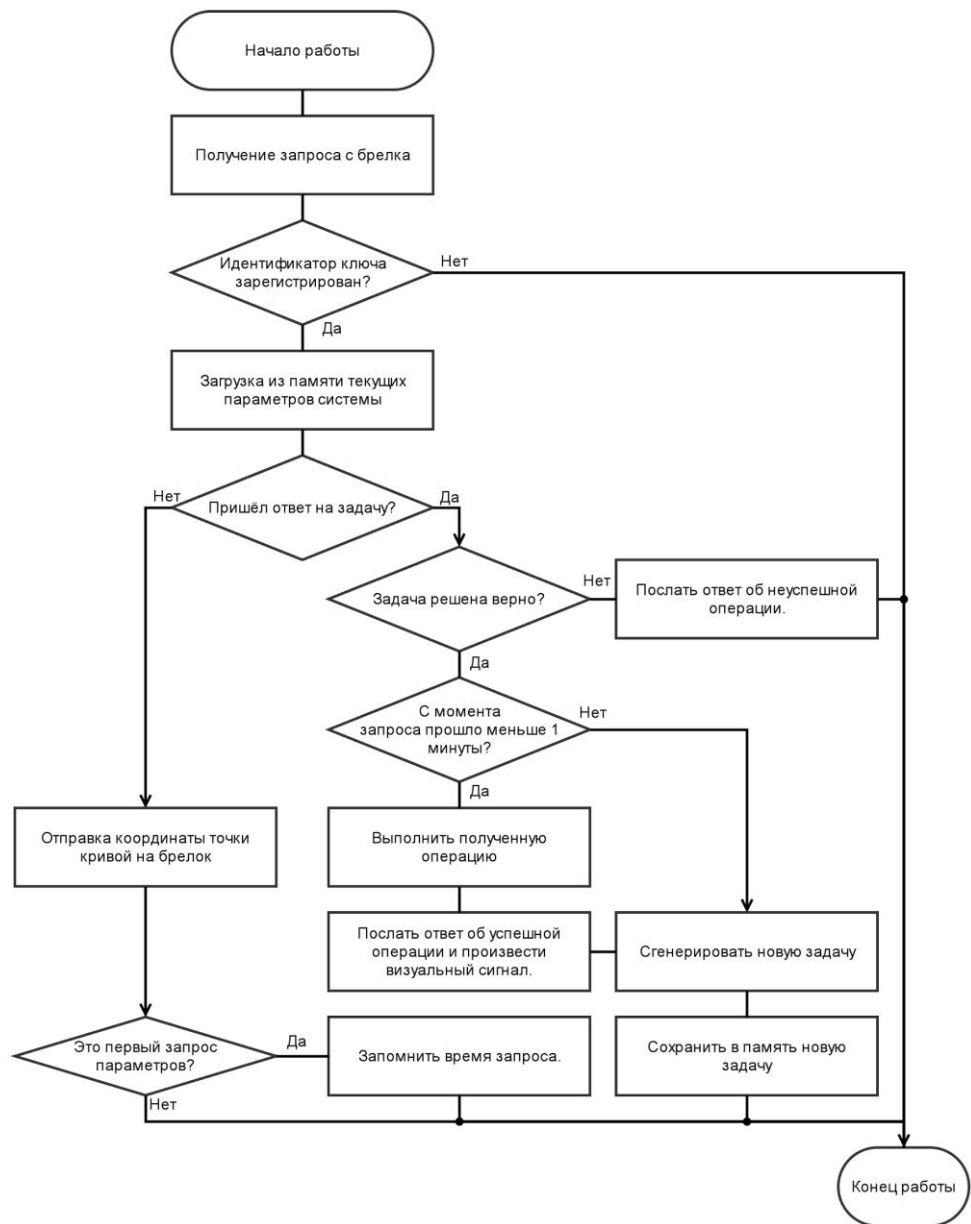


Рисунок А3 – Блок-схема управляющей программы агробота

## Листинг управляющей программы агробота

```
#include <WiFiNINA.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <WiFiServer.h>
#include <WiFiUdp.h>
#include <Servo.h>
#include <EEPROM.h>

int ledpin = A0;
int ENA = 5;
int ENB = 6;
int INPUT2 = 7;
int INPUT1 = 8;
int INPUT3 = 12;
int INPUT4 = 13;
int adjust = 1;
int Echo = A5;
int Trig = A4;
int Input_Detect_LEFT = A3;
int Input_Detect_RIGHT = A2;
int Input_Detect = A1;
int Carled = A0;
int Cruising_Flag = 0;
int Pre_Cruising_Flag = 0 ;
int Left_Speed_Hold = 255;
int Right_Speed_Hold = 255;

Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
//Servo servo5;
//Servo servo6;
Servo servo7;
Servo servo8;

byte angle1 = 70;
byte angle2 = 60;
byte angle3 = 60;
byte angle4 = 60;
//byte angle5 = 60;
//byte angle6 = 60;
byte angle7 = 60;
```

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

```

byte angle8 = 60;
int buffer[3];
int rec_flag;
int serial_data;
unsigned long Costtime;
int IR_R;
int IR_L;
int IR;
#define
MOTOR_GO_FORWARD {
digitalWrite(INPUT1,LOW);
digitalWrite(INPUT2,HIGH);
digitalWrite(INPUT3,LOW);
digitalWrite(INPUT4,HIGH);}
#define
MOTOR_GO_BACK  {
digitalWrite(INPUT1,HIGH);
digitalWrite(INPUT2,LOW);
digitalWrite(INPUT3,HIGH);
digitalWrite(INPUT4,LOW);}
#define
MOTOR_GO_RIGHT {
digitalWrite(INPUT1,HIGH);
digitalWrite(INPUT2,LOW);
digitalWrite(INPUT3,LOW);
digitalWrite(INPUT4,HIGH);}
#define
MOTOR_GO_LEFT  {
digitalWrite(INPUT1,LOW);
digitalWrite(INPUT2,HIGH);
digitalWrite(INPUT3,HIGH);
digitalWrite(INPUT4,LOW);}
#define
MOTOR_GO_STOP  {
digitalWrite(INPUT1,LOW);
digitalWrite(INPUT2,LOW);
digitalWrite(INPUT3,LOW);
digitalWrite(INPUT4,LOW);}
int RevStatus = 0;
int TurnAngle = 0;
int Golength = 0;
void Cruising_Mod()
{

```

					<b>СКБФЭУ.2.ИП.010000ЛП</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		23

```

if (Pre_Cruising_Flag != Cruising_Flag)
{
  if (Pre_Cruising_Flag != 0)
  {
    MOTOR_GO_STOP;
  }
  Pre_Cruising_Flag = Cruising_Flag;
}
switch (Cruising_Flag)
{
  case 2: TrackLine(); return;
  case 3: Avoiding(); return;
  case 4: AvoidByRadar(15); return;
  case 5: Send_Distance(); return;
  case 7: Route(); return;
  default: return;
}
}
void setup()
{
  pinMode(ledpin, OUTPUT);
  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(INPUT1, OUTPUT);
  pinMode(INPUT2, OUTPUT);
  pinMode(INPUT3, OUTPUT);
  pinMode(INPUT4, OUTPUT);
  pinMode(Input_Detect_LEFT, INPUT);
  pinMode(Input_Detect_RIGHT, INPUT);
  pinMode(Carled, OUTPUT);
  pinMode(Input_Detect, INPUT);
  pinMode(Echo, INPUT);
  pinMode(Trig, OUTPUT);
  Delayed();
  servo1.attach(3);
  servo2.attach(4);
  servo3.attach(2);
  servo4.attach(11);
  //servo5.attach(SDA);
  //servo6.attach(SCL);
  servo7.attach(9);
  servo8.attach(10);
  USART_init();
}

```

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



```

Init_Steer();
}

void loop()
{
while (1)
{
UartTimeoutCheck();
Cruising_Mod();
}
}
void Delayed() {
int i;
for(i=0;i<20;i++)
{
digitalWrite(ledpin,LOW);
delay(1000);
digitalWrite(ledpin,HIGH);
delay(1000);
}
for(i=0;i<10;i++)
{
digitalWrite(ledpin,LOW);
delay(500);
digitalWrite(ledpin,HIGH);
delay(500);
}
digitalWrite(ledpin,LOW);
MOTOR_GO_STOP;
}
void Init_Steer()
{
angle1 = EEPROM.read(0x01);
angle2 = EEPROM.read(0x02);
angle3 = EEPROM.read(0x03);
angle4 = EEPROM.read(0x04);
//angle5 = EEPROM.read(0x05);
//angle6 = EEPROM.read(0x06);
angle7 = EEPROM.read(0x07);
angle8 = EEPROM.read(0x08);
if(angle7 == 255 && angle8 == 255)
{
EEPROM.write(0x01,60);
}
}

```

					<b>СКБФЭУ.2.ИП.010000ЛП</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		25

```

EEPROM.write(0x02,60);
EEPROM.write(0x03,60);
EEPROM.write(0x04,60);
//EEPROM.write(0x05,60);
//EEPROM.write(0x06,120);
EEPROM.write(0x07,60);
EEPROM.write(0x08,60);
return;
}
servo1.write(angle1);
servo2.write(angle2);
servo3.write(angle3);
servo4.write(angle4);
//servo5.write(angle5);
//servo6.write(angle6);
servo7.write(angle7);
servo8.write(angle8);
adjust = EEPROM.read(0x10);
if(adjust==0xff)EEPROM.write(0x10,1);
Left_Speed_Hold = EEPROM.read(0x09);
Right_Speed_Hold = EEPROM.read(0x0A);
if(Left_Speed_Hold<55|Right_Speed_Hold<55)
{
Left_Speed_Hold=255;
Right_Speed_Hold=255;
}
analogWrite(ENB,Left_Speed_Hold);
analogWrite(ENA,Right_Speed_Hold);
MOTOR_GO_STOP;
}
void Open_Light()
{
digitalWrite(Carled, HIGH);
delay(1000);
}
void Close_Light()
{
digitalWrite(Carled, LOW);
delay(1000);
}
void forward(int adjust)
{
switch (adjust)

```

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

```

{
    case 1: MOTOR_GO_FORWARD; return;
    case 2: MOTOR_GO_FORWARD; return;
    case 3: MOTOR_GO_BACK; return;
    case 4: MOTOR_GO_BACK; return;
    case 5: MOTOR_GO_LEFT; return;
    case 6: MOTOR_GO_LEFT; return;
    case 7: MOTOR_GO_RIGHT; return;
    case 8: MOTOR_GO_RIGHT; return;
    default: return;
}
}
void back(int adjust)
{
    switch (adjust)
    {
        case 1: MOTOR_GO_BACK; return;
        case 2: MOTOR_GO_BACK; return;
        case 3: MOTOR_GO_FORWARD; return;
        case 4: MOTOR_GO_FORWARD; return;
        case 5: MOTOR_GO_RIGHT; return;
        case 6: MOTOR_GO_RIGHT; return;
        case 7: MOTOR_GO_LEFT; return;
        case 8: MOTOR_GO_LEFT; return;
        default: return;
    }
}
void left(int adjust)
{
    switch (adjust)
    {
        case 1: MOTOR_GO_LEFT; return;
        case 2: MOTOR_GO_RIGHT; return;
        case 3: MOTOR_GO_LEFT; return;
        case 4: MOTOR_GO_RIGHT; return;
        case 5: MOTOR_GO_FORWARD; return;
        case 6: MOTOR_GO_BACK; return;
        case 7: MOTOR_GO_FORWARD; return;
        case 8: MOTOR_GO_BACK; return;
        default: return;
    }
}
void right(int adjust)

```

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

```

{
switch (adjust)
{
case 1: MOTOR_GO_RIGHT; return;
case 2: MOTOR_GO_LEFT; return;
case 3: MOTOR_GO_RIGHT; return;
case 4: MOTOR_GO_LEFT; return;
case 5: MOTOR_GO_BACK; return;
case 6: MOTOR_GO_FORWARD; return;
case 7: MOTOR_GO_BACK; return;
case 8: MOTOR_GO_FORWARD; return;
default: return;
}
}
void Avoiding()
{
IR = digitalRead(Input_Detect);
if ((IR == LOW))
{
MOTOR_GO_STOP;
}
}
void TrackLine()
{
IR_L = digitalRead(Input_Detect_LEFT);
IR_R = digitalRead(Input_Detect_RIGHT);
if ((IR_L == LOW) && (IR_R == LOW))
{
forward(adjust);//直行
return;
}
if ((IR_L == LOW) && (IR_R == HIGH))
{
left(adjust);
return;
}
if ((IR_L == HIGH) && (IR_R == LOW))
{
right(adjust);
return;
}
if ((IR_L == HIGH) && (IR_R == HIGH))
{

```

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

```

MOTOR_GO_STOP;
return;
}
}
void AvoidByRadar(int distance)
{
int leng = Get_Distance();
if(distance<10)distance=10;
if((leng>1)&&(leng < distance))
{
while((Get_Distance())>1)&&(Get_Distance() < distance))
{
back(adjust);
}
MOTOR_GO_STOP;
}
}
void Route()
{
if (RevStatus == 0)
{
Sendbyte(0xff);
Sendbyte(0xA8);
Sendbyte(0x00);
Sendbyte(0x00);
Sendbyte(0xff);
delay(500);
}

while (RevStatus)
{
if (RevStatus == 1)
{
RevStatus = 0;
left(adjust);
delay(TurnAngle * 6);
MOTOR_GO_STOP;
forward(adjust);
delay(Golength * 10);
MOTOR_GO_STOP;
break;
}
if (RevStatus == 2)

```

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

```

    {
        RevStatus = 0;
        right(adjust);
        delay(TurnAngle * 6);
        MOTOR_GO_STOP;
        forward(adjust);
        delay(Golength * 10);
        MOTOR_GO_STOP;
        break;
    }
}
}
char Get_Distance()
{
    digitalWrite(Trig, LOW);
    delayMicroseconds(2);
    digitalWrite(Trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(Trig, LOW);
    float Ldistance = pulseIn(Echo, HIGH,5000);
    Ldistance = Ldistance / 5.8 / 10;
    //Serial.println(Ldistance);
    return Ldistance;
}
void Send_Distance()
{
    int dis = Get_Distance();
    Sendbyte(0xff);
    Sendbyte(0x03);
    Sendbyte(0x00);
    Sendbyte(dis);
    Sendbyte(0xff);
    delay(1000);
}
void USART_init()
{
    SREG = 0x80;
    //bitSet(UCSR0A,U2X0);
    bitSet(UCSR0B,RXCIE0);
    bitSet(UCSR0B,RXEN0);
    bitSet(UCSR0B,TXEN0);
    bitSet(UCSR0C,UCSZ01);
    bitSet(UCSR0C,UCSZ00);
}

```

```

UBRR0=(F_CPU/16/BAUD-1);
}
void Sendbyte(char c)
{
loop_until_bit_is_set(UCSR0A,UDRE0);
UDR0=c;
}
ISR(USART_RX_vect)
{
UCSR0B &= ~(1 << RXCIE0);
Get_uartdata();
UCSR0B |= (1 << RXCIE0);
}
void SendString(char *str)
{
while(*str)
{
Sendbyte(*str);
str++;
}
Sendbyte(0x0d);
Sendbyte(0x0a);
}
void Get_uartdata(void)
{
static int i;
serial_data = UDR0;
if (rec_flag == 0)
{
if (serial_data == 0xff)
{
rec_flag = 1;
i = 0;
Costtime = 0;
}
}
else
{
if (serial_data == 0xff)
{
rec_flag = 0;
if (i == 3)
{

```

					<b>СКБФЭУ.2.ИП.010000ЛП</b>	Лист
						31
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		

```

        Communication_Decomde();
    }
    i = 0;
}
else
{
    buffer[i] = serial_data;
    i++;
}
}
}
void UartTimeoutCheck(void)
{
    if (rec_flag == 1)
    {
        Costime++;
        if (Costime == 100000)
        {
            rec_flag = 0;
        }
    }
}
void Communication_Decomde()
{
    if(buffer[0]==0x00)
    {
        switch(buffer[1])
        {
            case 0x01:MOTOR_GO_FORWARD; return;
            case 0x02:MOTOR_GO_BACK; return;
            case 0x03:MOTOR_GO_LEFT; return;
            case 0x04:MOTOR_GO_RIGHT; return;
            case 0x00:MOTOR_GO_STOP; return;
            default: return;
        }
    }
    else if(buffer[0]==0x01) {
        // if (buffer[2] > 170)return;
        switch(buffer[1])
        {
            case 0x01:angle1 = buffer[2];servo1.write(angle1);return;
            case 0x02:angle2 = buffer[2];servo2.write(angle2);return;
            case 0x03:angle3 = buffer[2];servo3.write(angle3);return;
        }
    }
}

```

					<b>СКБФЭУ.2.ИП.010000ЛП</b>	Лист
						32
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		



```

        case 0x04:angle4 = buffer[2];servo4.write(angle4);return;
        //case 0x05:angle5 = buffer[2];servo5.write(angle5);return;
        //case 0x06:angle6 = buffer[2];servo6.write(angle6);return;
        case 0x07:angle7 = buffer[2];servo7.write(angle7);return;
        case 0x08:angle8 = buffer[2];servo8.write(angle8);return;
        default:return;
    }
}
else if(buffer[0]==0x02) {
    if(buffer[2]>100)return;
    if(buffer[1]==0x01)
    {
        Left_Speed_Hold=buffer[2]*2+55;
        analogWrite(ENB,Left_Speed_Hold);
        EEPROM.write(0x09,Left_Speed_Hold);
    }
    if(buffer[1]==0x02)
    {
        Right_Speed_Hold=buffer[2]*2+55;
        analogWrite(ENA,Right_Speed_Hold);
        EEPROM.write(0x0A,Right_Speed_Hold);
    }else return;
}
else if(buffer[0]==0x33)
{
    Init_Steer();return;
}
else if(buffer[0]==0x32) {
    EEPROM.write(0x01,angle1);
    EEPROM.write(0x02,angle2);
    EEPROM.write(0x03,angle3);
    EEPROM.write(0x04,angle4);
    //EEPROM.write(0x05,angle5);
    //EEPROM.write(0x06,angle6);
    EEPROM.write(0x07,angle7);
    EEPROM.write(0x08,angle8);
    return;
}
else if(buffer[0]==0x13)
{
    switch(buffer[1])
    {
        case 0x02: Cruising_Flag = 2; return;
    }
}

```

```

    case 0x03: Cruising_Flag = 3; return;
    case 0x04: Cruising_Flag = 4; return;
    case 0x05: Cruising_Flag = 5; return;
    case 0x07: Cruising_Flag = 7;
        analogWrite(ENA,115);
        analogWrite(ENB,115);
        return;
    case 0x00: Cruising_Flag = 0;RevStatus=0;
        analogWrite(ENA,Left_Speed_Hold);
        analogWrite(ENB,Right_Speed_Hold);
        return;
    default:Cruising_Flag = 0; return;
}
}

else if(buffer[0]==0x04)
{
    switch(buffer[1])
    {
        case 0x00:Open_Light(); return;
        case 0x01:Close_Light(); return;
        default: return;
    }
}
else if(buffer[0]==0x40)
{
    adjust=buffer[1];
    EEPROM.write(0x10,adjust);
}
else if(buffer[0]==0xA0)
{
    RevStatus = 2;
    TurnAngle=buffer[1];
    Golength=buffer[2];
}
else if(buffer[0]==0xA1)
{
    RevStatus = 1;
    TurnAngle=buffer[1];
    Golength=buffer[2];
}
}
}

```

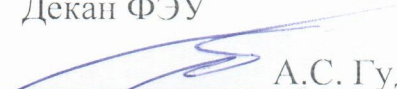
					<b>СКБФЭУ.2.ИП.010000ЛП</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		34

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  
(подпись)

« 17 » 06 20 22 г.

СОГЛАСОВАНО

Заведующий кафедрой ПЭ

  
Н.Н. Любушкина  
(подпись)

« 17 » 06 20 22 г.

### АКТ

о приемке в эксплуатацию аппаратно-программного комплекса

«Agro RoboPlant»

г. Комсомольск-на-Амуре

« 17 » 06 20 22 г.

Комиссия в составе представителей:

заказчика

- А.С. Гудим – руководитель проекта
- Н.Н. Любушкина – Заведующий кафедрой ПЭ
- А.С. Гудим – декан ФЭУ

исполнителя

- Е.С. Ильченко

составила акт о нижеследующем:

«Исполнитель» передает аппаратно-программный комплекс «Агроробот»,

в составе:

Оборудование, в составе:

- Агроробот
- Батарея
- Модуль питания

Программное обеспечение, в том числе:

- Рабочие программы управления изделием.

Эксплуатационная документация:

- Паспорт изделия

Аппаратно-программный комплекс «автономный с/х робот» прошел опытную эксплуатацию с «12» 05 по «23» 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 проекта

  
/ А.С. Гудим /

Ответственный исполнитель

  
/ Е.С. Ильченко /

