**Сдать до 23:59 08.10.23 надо уже готовый результат преподу , дедлайн выполнения 6.10.23 10:00**

Уточнить пункты 1.4,7.2,

Пункт 3.2.4- сфоткать блок для требований по напряжению

Богдан: пункты 4.1;4.2- посмотри и скажи сделаешь или мы сами разберемся?

Сделать: 6.3-6.6

1. Наименование ОКР, основание, исполнитель и сроки выполнения ОКP

1.1 Наименование ОКР

«Управление яркостью светодиода с помощью клавиатуры»

«Мартышкин пинг-понг»

«Система детекции нарушения климатических условий»  
1.2 Основание для выполнения ОКР

Приложение1.pdf

1.3 Исполнитель ОКР

Финансовый университет, УЦИ20-1, Группа Негры:

- Дуденко Богдан Андреевич

- Васильев Иван Сергеевич

-Грюнер Георгий Андреевич  
1.4 Срок выполнения ОКР

Выполнен:

С 11.09.23 по 25.09.2023

2. Цель выполнения ОКР, наименование и индекс изделия   
2.1 Цель ОКР

Создать программно-аппаратный комплекс осуществляющий мониторинг климата в помещении с возможностью оповещать пользователя о нарушении предельных значений.

2.2 Наименование и индекс образца, объект исследования ОКР:

Наименование - «Управление яркостью трехцветного светодиода с клавиатуры», Индекс-АПК001

Наименование - «Мартышкин пинг-понг», Индекс-АПК002

Наименование - «Система детекции нарушения климатических условий», Индекс АПК003

3. Технические требования к изделию

3.1Состав изделия:

АПК001состав изделия: Аппаратно-программный комплекс, включающий в себя микроконтроллер, светодиоды, клавиатуру, источник питания.

АПК002 состав изделия: Резистивные датчики изгиба, 2 шт., кнопочные выключатели, 2 шт., резисторы номиналом 4 кОм, 4 шт., беспечная макетная плата, 1 шт., совместимая с Arduino плата (на рис. 2.11 вверху показана плата MKR1000, а внизу — плата Arduino 101), 1 шт.

Используемые возможности платы: цифровой ввод, аналоговый ввод, УАПП (UART), персональный компьютер.

АПК003 состав изделия: Компьютер - 1 Arduino UNO – 1, провод для подключения Arduino UNO к компьютеру – 1, небольшая макетная доска - 1 DHT11 – 1, ЖК-дисплей 16X2 I2C – 1, MQ-135- 1, датчик пламени – 1, резистор 220 Ом – 1, СВЕТОДИОД - 2 Зуммер – 1, водяной насос – 1, реле – 1, батарея 9 В – 1, датчик влажности грунта - 1

3.2 Требования назначения

3.2.1 Назначение:

Мониторинг нарушения климатических условий и оповещение с помощью видимых и ощущаемых сигналов характерных для каждого из климатических условий.

3.2.2 Функции:

АПК001- Тестирование возможностей RGB светодиода

АПК002- Тестирование возможностей ультразвукового датчика для возможности взаимодействия с пользователем

АПК003- обнаружение огня, измерение количество углекислого газа в воздухе, измерение уровня влажности почвы, включение мелодии при нарушении критических значений, вывод данных на дисплей, на основе двух предыдущих работ включение светового сигнала при нарушении критических значений.

3.2.3 Метрологические характеристики:

1. Датчики температуры и влажности DHT11

Диапазон измерения температуры: от 0℃ до 50℃, допуск ± 2℃;  
Диапазон измерения влажности: 20% ~ 95%, допуск ± 5%;  
Размеры платы: 29 х 18 мм;  
Монтажное отверстие: 2.0мм.

2. УЗ-датчик расстояния HC-SR04

диапазон измерения — от 2 до 500 см;

точность — 0,3 см;

эффективный рабочий угол — <15°;

угол измерения: 30 градусов;

Длительность импульса срабатывания: 10 микросекунд;

3 Датчик пламени YG1006

Напряжение питания (Vcc): 3.5 - 5.5 В  
Диапазон чувствительности 760 нм - 1100 нм  
Угол обнаружения пламени: 60 °  
Сигнал на выходе компаратора LM393: 15 мА  
Два выхода: DO цифровой (0 или 1) и AO аналоговый  
Размеры: 36 x 16 мм  
Масса: 5 гр.  
Датчик чувствителен к обычному свету.  
Расстояние тестирования 80 см, чтобы не повредить датчик, чем больше пламя, тем больше должно быть расстояние.

4. Датчик влажности почвы YL-38

Сенсор при работе потребляет ток около 35 мА.

Напряжение питания 3,3—5 В.

Возвращаемый сигнал при питании от 5 В: 0—4,2 В.

Отобразив эти значения на 10-битный диапазон, можно воспользоваться следующими приближениями:  
  
0—300: сухая почва  
  
300—700: влажная почва  
  
700—950: датчик в воде

5. Датчик газа MQ-135

датчик качества воздуха чувствительный к аммиаку (NH3), оксиду азота (NOx), алкоголю, бензолу, дыму, углекислому газу (CO2)

Характеристики: Напряжение питания нагревателя: 5 В Напряжение питания датчика: 3,3–5 В

Потребляемый ток: 150 мА Габариты: 25,4×25,4 мм.

Диапазон измерений: Аммиак: 10—300 ppm Бензин: 10—1000 ppm Алкоголь: 10—300 ppm.

3.2.4 Требования к электропитанию: **(Написать требования к питанию с блока к которому подключают в лабе)**

3.3 Требования электромагнитной совместимости:

В связи отсутствия корпуса с заземлением запрещается использовать устройство возле высоко индуктивных контуров и источников сильного переменного магнитного поля

3.4 Требования живучести и стойкости к внешним воздействиям:

Поскольку это образец требования по безопасности не выдвигались поэтому живучесть не обеспечена

3.5 Требования надежности: АПК должен обеспечивать надежную работу в течение всего срока эксплуатации

Поскольку это образец требования по надежности не выдвигались поэтому работа в течении всего срока эксплуатации не обеспечена

3.6 Требования эргономики, обитаемости и технической эстетики:

Поскольку это образец требования по эргономике и технической эстетике не выдвигались

3.7 Требования к эксплуатации, хранению, удобству технического обслуживания и ремонта:

Поскольку это образец требования по хранению удобству и техническому обслуживанию не выдвигалось

3.8 Требования транспортабельности

АПК должен соответствовать размерам вместимым в упаковку набора «Умный дом на базе Arduino. Большой набор» ISBN 978-5-9775-6608-7

3.9 Требования безопасности: АПК должен соответствовать требованиям безопасности при эксплуатации. (**Написать госты на напряжение до 15В)**

3.10 Требования стандартизации и унификации:

На уровне прототипа требования не выдвигались, соответствие стандартизации и унификации не обеспечено.

3.11 Требования технологичности: АПК должен быть легким в производстве и сборке.

1. Датчики температуры и влажности DHT11

2. УЗ-датчик расстояния HC-SR04

3. Датчик пламени YG1006

4. Датчик влажности почвы YL-38

5. Датчик газа MQ-135

3.12 Конструктивные требования:

На уровне прототипа не предусматривает защитного кожуха поэтому требования выдвигаются на основе техники безопасности

4. Технико-экономические требования

4.1 Аппаратная составляющая

Аппаратно-программный комплекс, включающий в себя микроконтроллер, светодиоды, клавиатуру, источник питания.

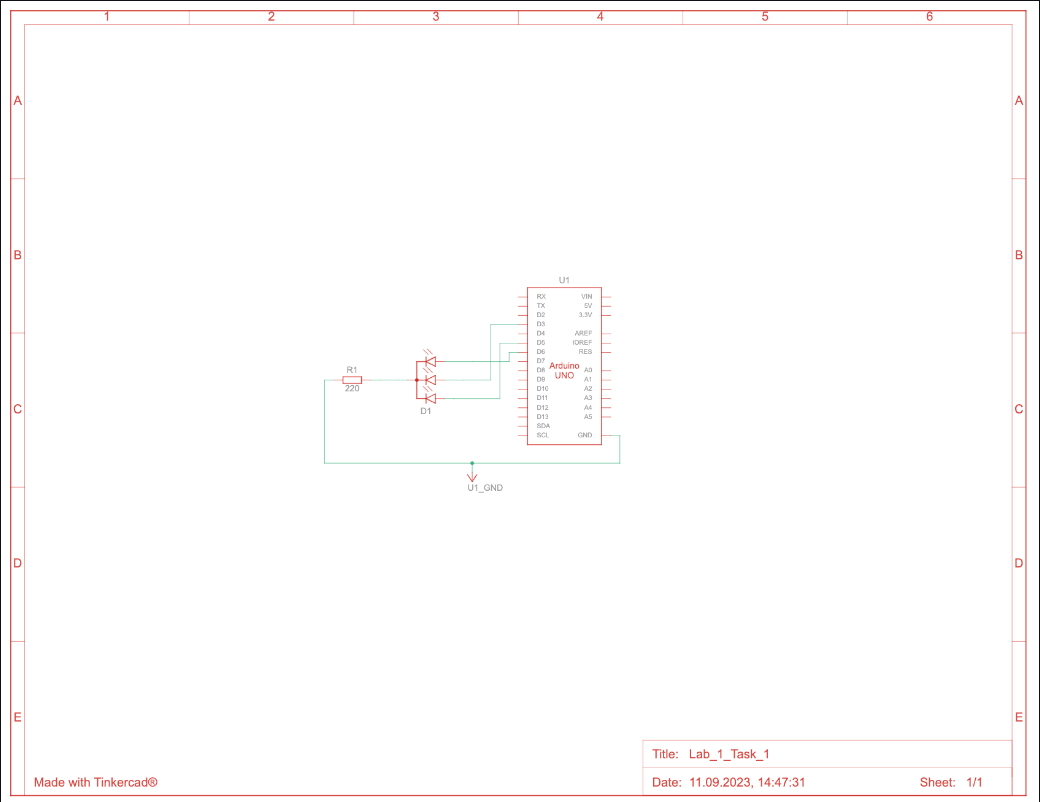
Резистивные датчики изгиба, 2 шт., кнопочные выключатели, 2 шт., резисторы номиналом 4 кОм, 4 шт., беспечная макетная плата, 1 шт., совместимая с Arduino плата (на рис. 2.11 вверху показана плата MKR1000, а внизу — плата Arduino 101), 1 шт.

Используемые возможности платы: цифровой ввод, аналоговый ввод, УАПП (UART), персональный компьютер.

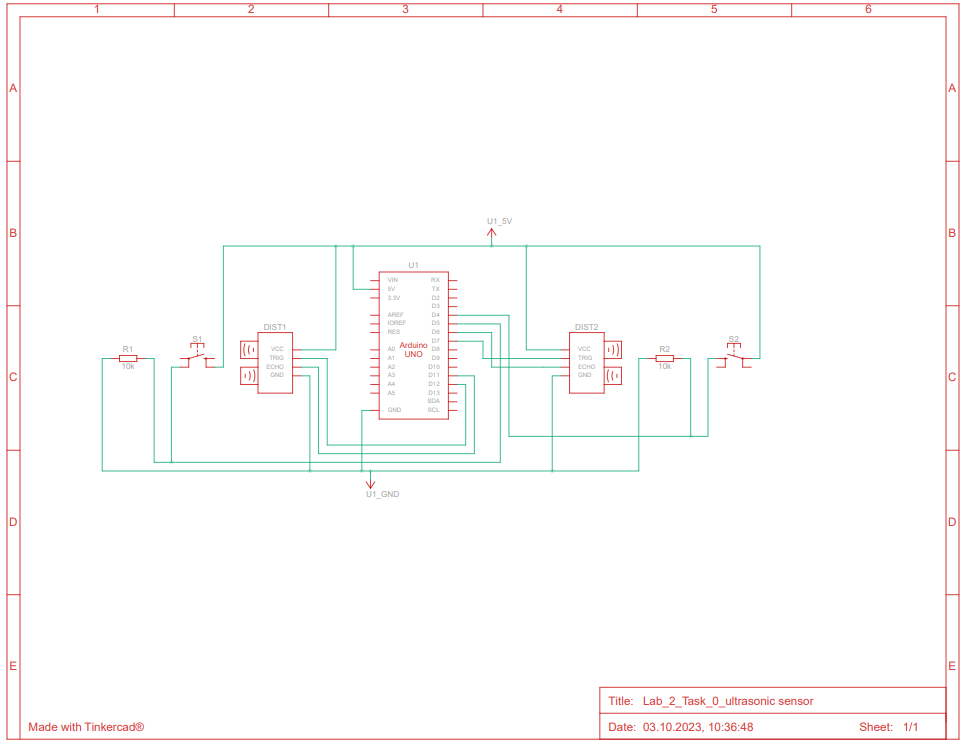
Компьютер - 1 Arduino UNO – 1, провод для подключения Arduino UNO к компьютеру – 1, небольшая макетная доска - 1 DHT11 – 1, ЖК-дисплей 16X2 I2C – 1, MQ-135- 1, датчик пламени – 1, резистор 220 Ом – 1, СВЕТОДИОД - 2 Зуммер – 1, водяной насос – 1, реле – 1, батарея 9 В – 1, датчик влажности грунта - 1

4.1 Эскизный проект

АПК001

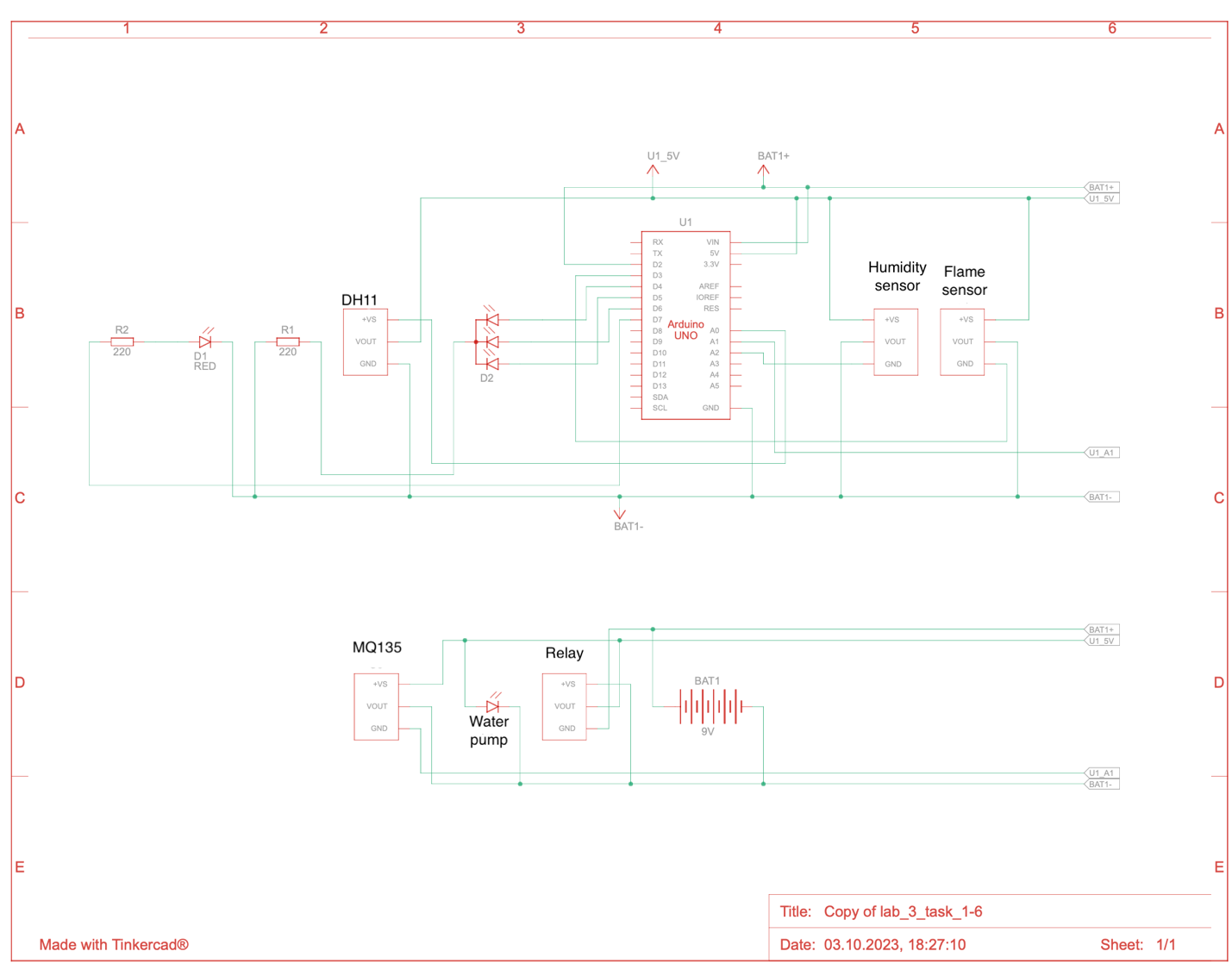


АПК002

****



АПК003



4.2 Описание компьютерных систем, серверов, сетевых устройств, периферийных устройств и других аппаратных средств, а также их характеристики

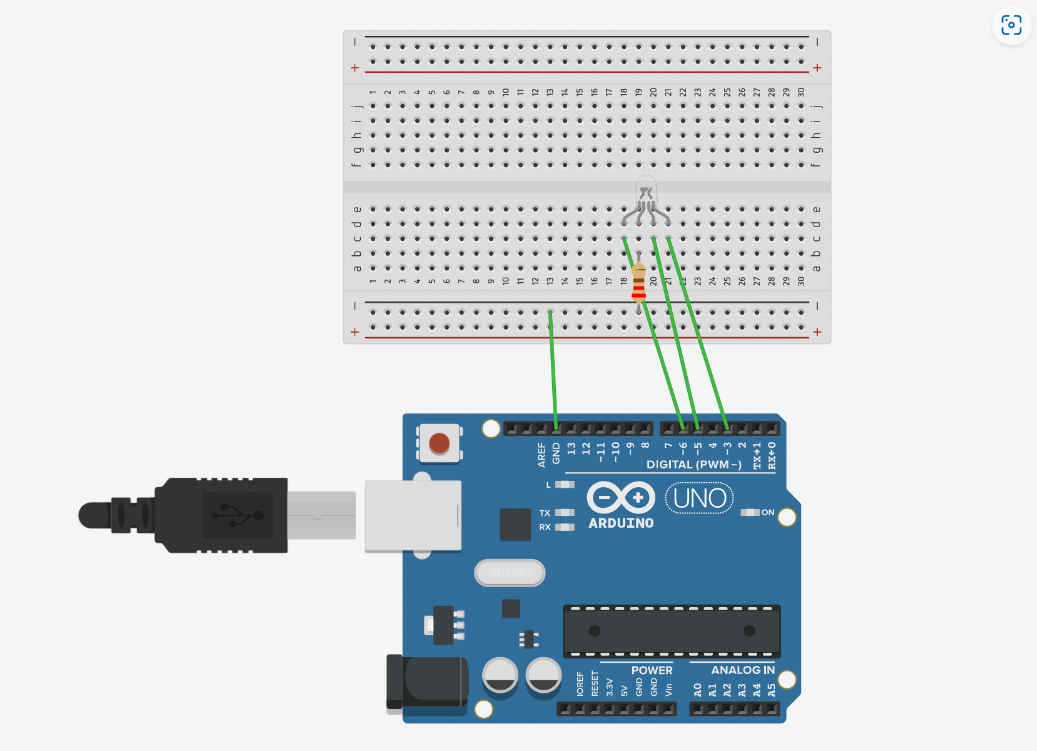
**(Перечень конкретных контроллеров, взять из книги Умный дом на Arduino)— файл скину**

Сетевое соединение не использовалось

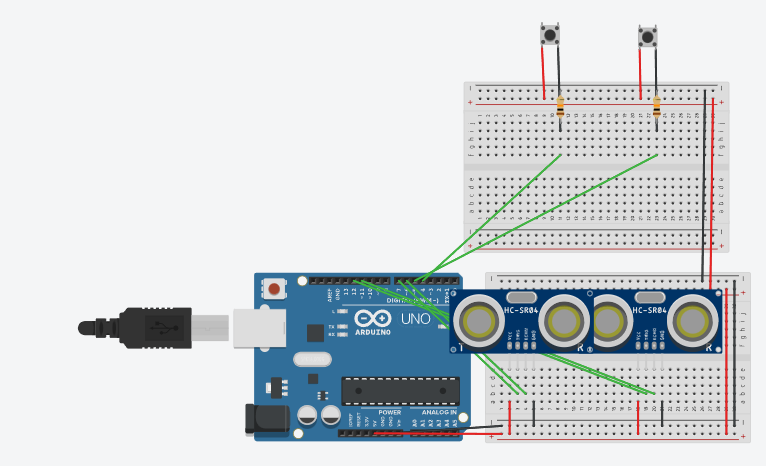
Использовался компьютер для передачи данных на ArduinoUNO(числовой код) книжка умный дом на базе ардуино

4.3 методология оценки правильности испытаний. **(Как именно проводить испытания например подуть на датчик углекислого газа, поднести огонь к датчику огня.)**

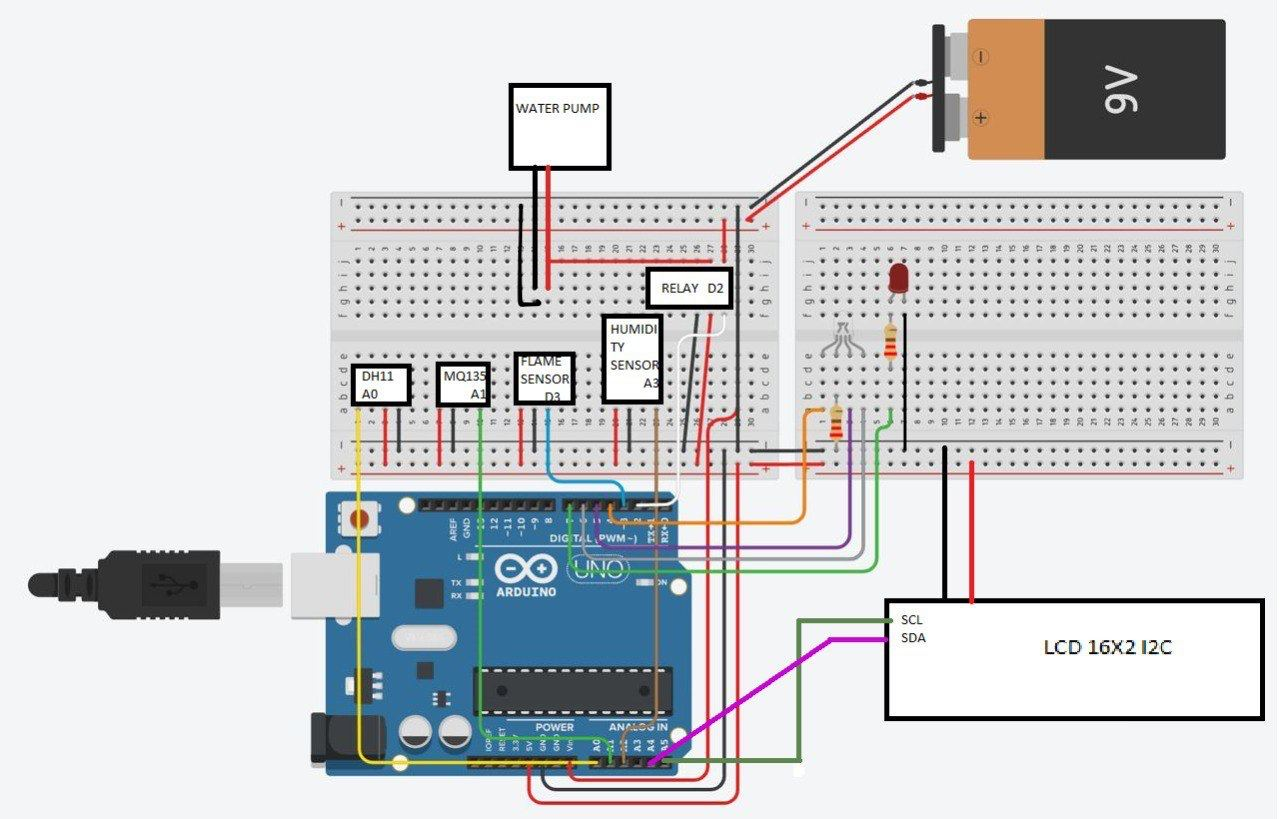
4.4 технический проект

АПК001

АПК002



АПК003



5. Программная составляющая

5.1 Описание разработанных программных продуктов

#include <Wire.h> // Добавляем необходимые библиотеки

#include <LiquidCrystal\_I2C.h>

#include "DHT.h"

#include "MQ135.h"

// For LCD DISPLAY

// SCL\_PIN = A5

// SDA\_PIN = A4

byte degree[8] = // кодируем символ градуса

{

B00111,

B00101,

B00111,

B00000,

B00000,

B00000,

B00000,

};

LiquidCrystal\_I2C lcd(0x27,16,2); // Задаем адрес и размерность дисплея

// Temp and Wet

#define DHTPIN A0 // к какому пину будет подключен сигнальный выход датчика

#define DHTTYPE DHT11 //выбор используемого датчика DHT 11

//#define DHTTYPE DHT22 // DHT 22 (AM2302)

//#define DHTTYPE DHT21 // DHT 21 (AM2301)

//инициализация датчика

int t\_w\_numTones = 10;

// Ноты

int t\_w\_tones[10] = {261, 277, 294, 311, 330, 349, 370, 392, 415, 440};

DHT dht(DHTPIN, DHTTYPE);

//gas

int gas\_numTones = 10;

// Ноты

int gas\_tones[10] = {400, 400, 400, 400, 400, 400, 400, 400, 400, 400};

#define ANALOGPIN A1 //подключение аналогового сигнального пина

//flame

int flame\_numTones = 10;

// Ноты

int flame\_tones[10] = {300, 300, 300, 300, 300, 300, 300, 300, 300, 300};

#define SENSOR\_FLAME\_PIN 4

int flame = 0;

//pump

int pump\_numTones = 10;

// Ноты

int pump\_tones[10] = {200, 200, 200, 200, 200, 200, 200, 200, 200, 200};

#define PUMP\_PIN 2

#define WET\_PIN A2

int soilMoistureValue = 0;

int percentage = 0;

#define LED\_DIOD 12

#define LED\_DIOD1 8

#define LED\_DIOD2 9

#define LED\_DIOD3 10

#define SOUND\_PIN 6

void setup()

{

lcd.init(); // Инициализация lcd

lcd.backlight(); // Включаем подсветку

lcd.createChar(1, degree); // Создаем символ под номером 1

//Устанавливаем пины вход или выход

pinMode(SENSOR\_FLAME\_PIN, INPUT);

pinMode(PUMP\_PIN, OUTPUT);

pinMode(WET\_PIN, INPUT);

pinMode(LED\_DIOD, OUTPUT);

pinMode(LED\_DIOD1, OUTPUT);

pinMode(LED\_DIOD2, OUTPUT);

pinMode(LED\_DIOD3, OUTPUT);

pinMode(SOUND\_PIN, OUTPUT);

Serial.begin(9600);

}

void loop() {

// Reading temperature or humidity takes about 250 milliseconds!

// Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)

dht.begin();

float h = dht.readHumidity();

// Read temperature as Celsius

float t = dht.readTemperature();

delay(3000); //Задержка для считывания

//Отчищаем дисплей

lcd.clear();

// Условие если влажность и температура больше 30

if(h > 10 && t > 30){

// Выводим показания влажности и температуры

lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки

lcd.print("Hum = %"); // Выводим текст

lcd.setCursor(7, 0); // Устанавливаем курсор на 7 символ

lcd.print(h, 1); // Выводим на экран значение влажности

lcd.setCursor(0, 1); // Устанавливаем курсор в начало 2 строки

lcd.print("Temp = \1C "); // Выводим текст, \1 - значок градуса

lcd.setCursor(7, 1); // Устанавливаем курсор на 7 символ

lcd.print(t,1); // Выводим значение температуры

for (int i = 0; i < t\_w\_numTones; i++) { // цикл, чтобы звучала мелодия из нот и мигало в тот-же такт задержки

tone(SOUND\_PIN, t\_w\_tones[i]);

digitalWrite(LED\_DIOD1, HIGH);

delay(800);

}

noTone(SOUND\_PIN); //выключаем звук нот

}

else{

// Выводим показания влажности и температуры

lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки

lcd.print("Hum = % "); // Выводим текст

lcd.setCursor(7, 0); // Устанавливаем курсор на 7 символ

lcd.print(h, 1); // Выводим на экран значение влажности

lcd.setCursor(0, 1); // Устанавливаем курсор в начало 2 строки

lcd.print("Temp = \1C "); // Выводим текст, \1 - значок градуса

lcd.setCursor(7, 1); // Устанавливаем курсор на 7 символ

lcd.print(t,1); // Выводим значение температуры

digitalWrite(LED\_DIOD1, LOW); //выключаем светодиод

}

//Инициализируем переменую датчика MQ135

MQ135 gasSensor = MQ135(ANALOGPIN);

// Считываем данные PPM(количества углекислого газа)

float ppm = gasSensor.getPPM();

// олучите сопротивление RZero датчика для целей калибровки

float rzero = gasSensor.getRZero();

delay(2000);

lcd.clear();

// Если ppm больше 35

if(ppm > 35){

lcd.setCursor(0, 0);

lcd.print("ppm> : % ");

lcd.print(ppm);

for (int i = 0; i < gas\_numTones; i++) {

tone(SOUND\_PIN, gas\_tones[i]);

digitalWrite(LED\_DIOD2, HIGH);

delay(400);

}

noTone(SOUND\_PIN);

}

else{

lcd.setCursor(0, 0);

lcd.print("ppm< : % ");

lcd.print(ppm);

digitalWrite(LED\_DIOD2, LOW);

}

lcd.clear();

//Считываем есть ли пламя

flame = digitalRead(SENSOR\_FLAME\_PIN);

delay(2000);

if (flame == 1){

lcd.setCursor(0, 0);

lcd.print("Fire");

for (int i = 0; i < flame\_numTones; i++) {

tone(SOUND\_PIN, flame\_tones[i]);

digitalWrite(LED\_DIOD3, HIGH);

delay(200);

}

noTone(SOUND\_PIN);

}

else{

lcd.setCursor(0, 0);

lcd.print("No fire");

digitalWrite(LED\_DIOD3, LOW);

}

// Считываем Влажность почвы

soilMoistureValue = analogRead(WET\_PIN);

percentage = map(soilMoistureValue, 490, 1023, 100, 0);

delay(2000);

lcd.clear();

//Если процент влажности почвы меньше 10 или влажность воздуха меньше 35 то включается насос и мигает светодиод и музыка играет

if(percentage < 10 || h < 35){

lcd.setCursor(0, 0);

lcd.print("Pump on % ");

lcd.print(percentage);

digitalWrite(PUMP\_PIN, HIGH);

for (int i = 0; i < pump\_numTones; i++) {

tone(SOUND\_PIN, pump\_tones[i]);

digitalWrite(LED\_DIOD, HIGH);

delay(100);

}

noTone(SOUND\_PIN);

}

if(percentage > 80 && h > 35){

lcd.setCursor(0, 0);

lcd.print("Pump off % ");

lcd.print(percentage);

digitalWrite(PUMP\_PIN, LOW);

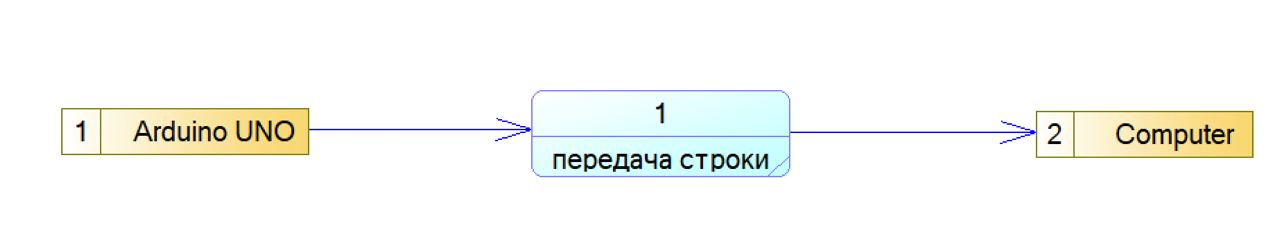
digitalWrite(LED\_DIOD, LOW);

}

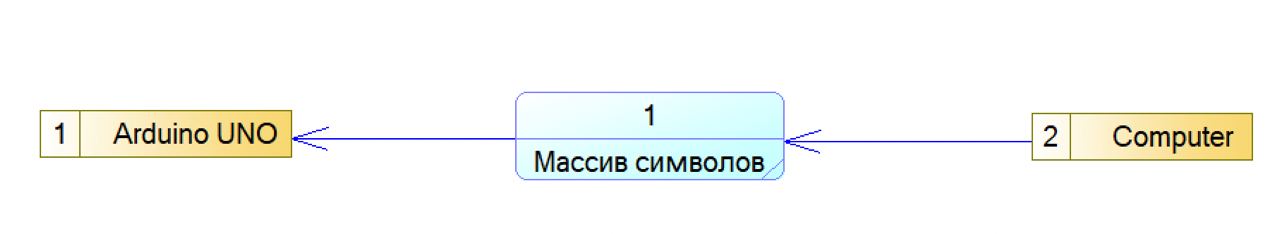
}

5.2 Архитектура работы продукта

АПК001



АПК002



АПК003



5.3 Использованные технологии и инструменты

- Arduino programming language

- Arduino IDE

- Processing  
5.4 Примеры кода продукта

<https://github.com/MinDl1/labs>

6. Требования к пользователю по работе с программой

6.1 Требования к аппаратному обеспечению компьютера, на котором будет запускаться программа отчета по ОКР

Linux X-86-64, Win 10+ 64bit, macos intel, macos Apple Silicon

6.2 Требования к операционной системе компьютера, на котором будет запускаться программа отчета по ОКР

Windows 10+ 64bits, Linux 64bits, macOS intel 10.14: "Mojave"+ 64bits, macOS Apple Silicon, 11: "Big Sur"+ 64bits.

Windows 10:

* Processor: 1 gigahertz (GHz) or faster processor or SoC
* RAM: 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit
* Hard disk space: 16 GB for 32-bit OS or 20 GB for 64-bit OS
* Graphics card: DirectX 9 or later with WDDM 1.0 driver
* Display: 800 x 600

MacOS intel:

* Processor: 3.2 GHz Q. Core Xeon W3565
* VRAM: 1GB
* RAM: 2GB
* Video card: Radeon HD 5770
* HDD minimum :12,5 ГБ

MacOs Apple silicon:

Processor: Apple M1

RAM: 8 ГБ

HDD minimum: 44GB

6.3 Требования к настройкам компьютера, включая настройки безопасности и настройки сетевого соединения, если программа отчета по ОКР использует сетевые ресурсы.

Не использует сетевые ресурсы, у компьютера должен быть доступ к USB портам. Драйвер для ардуино.

6.4 Инструкции по установке и запуску программы отчета по ОКР

Инструкция по установки для администратора:

Скачать Arduino IDE, библиотеки необходимые для работы скетча перечисленные выше, Processing.

Сначала необходимо залить Arduino sketch на Arduino UNO, после этого закрыть Arduino IDE и открыть Processing и запустить скетч для Lab2. Для Lab1 и Lab3 только залить Arduino скетч.

6.5 Инструкция по использования программы ОКР

Для АПК002 использования Processing нужно:

Скачать Processing открыть файл для Processing, проверить правильность считывания порта (с того ли порта считывает компьютер) после запустить скетч и начать играть с помощью датчиков(В зависимости от компьютера нужно будет зайти в «набросок» в верхнем меню и выбрать «импортировать библиотеку» «serial»). Для АПК003 не нужны приложения для конечного пользователя. Для АПК001 нужно иметь скаченную Arduino IDE в верхнем меню выбрать «Tools» и «Serial Monitor».

6.6 Инструкции по обращению за технической поддержкой, включая рекомендации по описанию проблемы

Если у вас не после запуска processing не двигаются ракетки, нужно проверить правильность подключения порта, в коде программы нужно поменять в функции void setup() во второй строчке String portName = Serial.list()[1], на 0, 2, 3 и тд, в зависимости от количество портов в вашем устройстве.

Пример отправки ошибки:

Консольный лог (Ошибка если есть).

Код программы.

Фото собранной схемы.

Описать что вы сделали перед подключением.

Отправлять ошибки на почту: [b0gdandudenk0@mail.ru](mailto:b0gdandudenk0@mail.ru) или TG: @MinDl1

7 Результаты тестирования  
7.1 Оценка полноты решения поставленной задачи

**(Наша схема выполняет задачи из-т пункта 2)🡨 надо написать это более грамотным языком**

Во время выполнения поставленной задачи мы собрали все элементы схемы в одну рабочую схему, затем проверили каждый датчик на работоспособность по отдельности, после того как мы убедились, что все датчики работают исправно и без ошибочно, мы проверили общую работоспособность схемы.

7.2 Оценка достоверности полученных результатов

**(Внести результаты с использованием методологий в виде таблиц и прочей херни разброс в районе 7% формулы не указываем)🡨 Уточню у препода на днях**

Погрешность DHT11 = ±5%

Погрешность MQ135 = 10 - 15 ppm

Погрешность YL-38 = ±2%

В связи с ограниченностью ресурсов нам пришлось заменить гибкие датчики на ультразвуковые, в остальном полученные результаты соответствуют техническому заданию.  
7.3 Сравнение с аналогичными результатами отечественных и зарубежных работ )

**(Надо сделать классическое сравнение в виде таблицы, и написать выводы по типу-« мы сделали почти как у конкурентов» или «нихуя не сделали мы чмо»)**

7.4 Недостатки

**(Недостатки нашей разработке)🡨 недостатки конкретно нашей разработки**

Во время выполнения второй лабораторной работы мы не сделали схему с четырьмя датчиками, также мы не сделали схему с WI-FI соединением, потому что нам зачли всю лабораторную работу после хорошего сбора схемы с двумя датчиками.   
7.5 Предложения по дальнейшим направлениям работ или обоснование о необходимости их прекращения

**(Написать как устранить недостатки)🡨 либо как устранить недостатки либо пиши это полный пиздец мы нихуя сделать не можем деньги вернем хуй соснем**

Добавить в набор дополнительные датчики, установить датчик шума посередине комнаты на потолке для определения шума, если шум превышает определенное значение то загорается светодиод(Установлен новый светодиод).

8 Заключение  
8.1 Выводы по результатам ОКР **(Пишем общий вывод ориентир общая цель)**

8.2 Оценка полноты решения поставленных задач

**(Либо все поставленные задачи и еще + доп, либо наш ОКРК не выполняет следующее….))**

8.3 Разработка рекомендаций по конкретному использованию ОКР

**(Пишем куда внедрить зачем и как)**

8.4 Оценка технико-экономической эффективности внедрения.

**(SWOT-анализ)🡨 надо сделать грамотно а не отьебись**  
8.5 Оценка технического уровня выполненной ОКР

(Ничего не пишем)

9 Приложения

9.1 Промежуточные и итоговые схемы

Промежуточные схемы из тинкера  
9.2 Таблицы

(Если нет таблиц ничего не пишем)  
9.3 Протоколы испытаний и тестирования

(Оставляем пустым)

9.4 Иллюстрации

[MinDl1/labs: Labs is a project based on different hardware and software. (github.com)](https://github.com/MinDl1/labs) вставить фотки DFD  
9.5 Промежуточные и итоговые программы

Ссылка на гит и куски кода