Муниципальное бюджетное общеобразовательное учреждение

«Бесединская средняя общеобразовательная школа»

Курского района Курской области

|  |
| --- |
| 305501, Курская область, Курский район, с. Беседино, д.265 А, телефоны: 59-71-30, 59-71-31,  эл. почта [kurskii92@mail.ru](mailto:kurskii92@mail.ru) ОГРН 1024600617029 ИНН 4611004895 КПП 461101001 ОКПО 21832638 |

**Интернет вещей, технология IoT**

**Выполнил:**

Глазов Илья 9 класс

МБОУ «Бесединская средняя общеобразовательная школа»

**Руководитель:**

Гладких Александр Викторович

учитель информатики

МБОУ «Бесединская средняя общеобразовательная школа»

Курский район, 2022

**Содержание:**

**Паспорт проекта....................................................................................................... 3**

**Введение..................................................................................................................... 6**

**Краткое описание......................................................................................................7**

**Глава 1.Теоретическая часть**

* 1. Что такое «Интернет вещей» или IoT?....................................................**......** 9
  2. Что такое "Умный Дом Йотик М2"?.............................................................10

1.3. Что такое Arduino IDE?**.................................................................................**..12

1.4. Что такое Blynk?............................................................................................. 12

**Глава 2. Практическая часть**

2.1 Сборка каркаса макета и установка датчиков................................................. 14

2.2.Работа с сервисами Blynk и приложениями.................................................... 15

**Заключение.............................................................................................................. 32**

**Список информационных ресурсов.................................................................... 33**

**Паспорт проекта:**

|  |  |
| --- | --- |
| Тема | Интернет вещей, технология IoT |
| Автор | Глазов Илья |
| Руководитель проекта | Гладких Александр Викторович |
| Актуальность | С того момента, когда интернет стал доступен для общества, люди получили возможность накапливать и обмениваться огромным количеством информации. А на данном этапе в интернете ведут целые отрасли дистанционно управляя бизнесом. В данном проекте на примере "Умный дом ЙоТик М2" я буду рассматривать возможность управления собственным домом из любой точки планеты. А пультом управления будет являться смартфон. |
| Основные понятия | **Интернет вещей** (IoT)[2] – это огромное количество «**вещей**», которые подключены к Интернету для обмена данными с другими **вещами**  - приложениями IoT, подключенными устройствами, промышленными машинами и многим другим.  **Набор "Умный Дом Йотик М2"**[1] - создан для  изучения основ автоматизации, программирования и Интернета вещей (1оТ). Данный набор - это идеальная интеграция реального и цифрового пространства.  **Arduino IDE**[3] — интегрированная среда разработки для Windows, MacOS и Linux, разработанная на Си и C ++, предназначенная для создания и загрузки программ на Arduino-совместимые платы, а также на платы других производителей.  **Blynk** - Это платформа Интернета вещей (IoT), предназначенную для создания проектов, подключенных к Интернету и доступных с телефона или мобильного устройства. |
| Цель | Познакомиться с принципами работы технологии IoT на примере собранного набора "Умный Дом Йотик М2", попробовать создать скетч и прошить основной контролер в среде программирования Arduino IDE, создать среду управления устройствами умного дома в приложении Blynk. |
| Задачи проекта | 1. Изучить основные элементы "Умный Дом Йотик М2".  2. Согласно инструкции полностью собрать устройство.  3. Познакомиться с основными принципами работы в среде программирования Arduino IDE  4. Создать скетч для собранного макета и прошить контролер.  5. Познакомиться с основными принципами работы платформы Blynk.  6. В приложении Blynk создать пульт управления макетом "Умный Дом Йотик М2" и попробовать управлять его устройствами. |
| Гипотеза | В связи с тем, что основной контролер набора "Умный Дом Йотик М2" оснащен WI-FI – модулем, макет должен иметь возможность подключаться к среде интернет через беспроводную точку доступа. Это должно позволить управлять устройством из любой точки планеты, следить за показаниями датчиков в режиме реального времени через платформу Blynk и созданный на ней пульт управления. |
| Сроки реализации проекта | * Изучение основных элементов и характеристик набора "Умный Дом Йотик М2" (ноябрь-декабрь 2021). * Сбор макета "Умный Дом Йотик М2" (декабрь 2021) * Изучение основ и принципов программирования в среде Arduino IDE. Установка необходимых библиотек (январь-февраль 2022). * Регистрация в сервисе Blynk, знпкомство с приложением, создание пульта и скетча (февраль-март 2022). |
| Объект исследования | Технологии и принципы организации IoT. |
| Предмет исследования: | Набор "Умный Дом Йотик М2", Blynk, Arduino IDE. |
| Практическая значимость | Проект позволит познакомиться с принципами работы IoT и применять их в повседневной жизни. |

**Введение.**

В настоящее время общество может почувствовать, что находится в некотором переходном моменте на более высокий уровень развития. Мир вокруг нас становится все более систематизированным, интерактивным и автоматизированным, позволяя нам тратить освободившееся время на саморазвитие.

Современные технологии позволяют сделать автоматизированным и интеллектуальным не только производства но и собственное жилье. Потому, что если раньше интернет представлял из себя информационную сеть, то сейчас в этой системе участвуют так же различные вещи и устройства. Они тоже образуют целую сеть взаимодействующих друг с другом технических изделий. Вот здесь и появляется новый термин IoT.

На рынке появилось уже много провайдеров предоставляющих услуги IoT. К примеру Ростелеком [4] или Триколор [5].

Давайте вместе попробуем разобраться что это за система.

**Актуальность**

С того момента, когда интернет стал доступен для общества, люди получили возможность накапливать и обмениваться огромным количеством информации. А на данном этапе в интернете ведут целые отрасли, дистанционно управляя бизнесом. В данном проекте на примере "Умный дом ЙоТик М2" я буду рассматривать возможность управления собственным домом из любой точки планеты. А пультом управления будет являться смартфон.

**Цель**: Познакомиться с принципами работы технологии IoT на примере собранного набора "Умный Дом Йотик М2", попробовать создать скетч и прошить основной контролер в среде программирования Arduino IDE, создать среду управления устройствами умного дома в приложении Blynk.

**Гипотеза:** В связи с тем, что основной контролер набора "Умный Дом Йотик М2" оснащен WI-FI – модулем, макет должен иметь возможность подключаться к среде интернет через беспроводную точку доступа. Это должно позволить управлять устройством из любой точки планеты, следить за показаниями датчиков в режиме реального времени через платформу Blynk и созданный на ней пульт управления.

**Объект исследования:** Технологии и принципы организации IoT.

**Предмет исследования:** Набор "Умный Дом Йотик М2", Blynk, Arduino IDE.

**Задачи:**

1. Изучить основные элементы "Умный Дом Йотик М2".

2. Согласно инструкции полностью собрать устройство.

3. Познакомиться с основными принципами работы в среде программирования Arduino IDE

4. Создать скетч для собранного макета и прошить контролер.

5. Познакомиться с основными принципами работы платформы Blynk.

6. В приложении Blynk создать пульт управления макетом "Умный Дом Йотик М2" и попробовать управлять его устройствами.

**Практическая значимость:** Проект позволит познакомиться с принципами работы IoT и применять их в повседневной жизни.

**Глава 1.Теоретическая часть.**

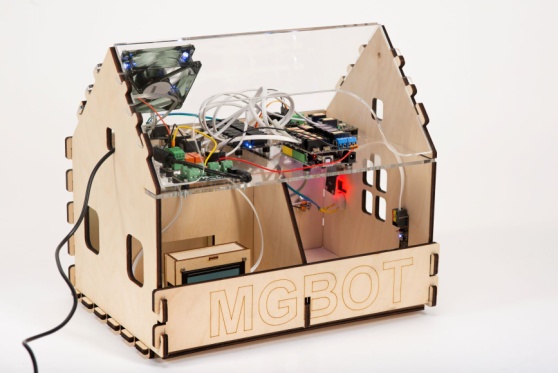
### 1.1 Что такое «Интернет вещей» или IoT?

Термин «Интернет вещей» был придуман предпринимателем Кевином Эштоном, одним из основателей центра Auto-ID в MIT. Эштон была частью команды, которая обнаружила, как связывать объекты с Интернетом с помощью метки RFID. Впервые он использовал фразу «Интернет вещей» в презентации 1999 года, и с тех пор она стала популярной.

Эштон, возможно, был первым, кто использовал термин «Интернет вещей», но концепция подключенных устройств, в частности, подключенных машин, существует давно. Например, машины связывались друг с другом с тех пор, как первые электрические телеграфы были разработаны в конце 1830-х годов. Другими технологиями, которые вошли в IoT, были: радиопередача голоса, беспроводные технологии (Wi-Fi) и программное обеспечение для диспетчерского управления и сбора данных (SCADA). Затем в 1982 году машина по продаже Кока-Кола в Университете Карнеги-Меллона стала первым подключенным интеллектуальным устройством. Используя локальную сеть университета или ARPANET - предшественник современного интернета - студенты могли узнать, какие напитки хранятся на складе, и были ли они холодными.

Сегодня мы живем в мире, где устройств, подключенных к IoT, больше, чем людей. Эти устройства могут быть как умные часы, так и RFID-чипы отслеживания запасов. Устройства, подключенные к IoT, обмениваются данными через сети или облачные платформы, подключенные к Интернету вещей. Информация, полученная в реальном времени из IoT, приближает цифровую трансформацию. Интернет вещей обещает много позитивных изменений в области охраны труда и здоровья, деловых операций, производственных показателей и глобальных экологических и гуманитарных проблем.

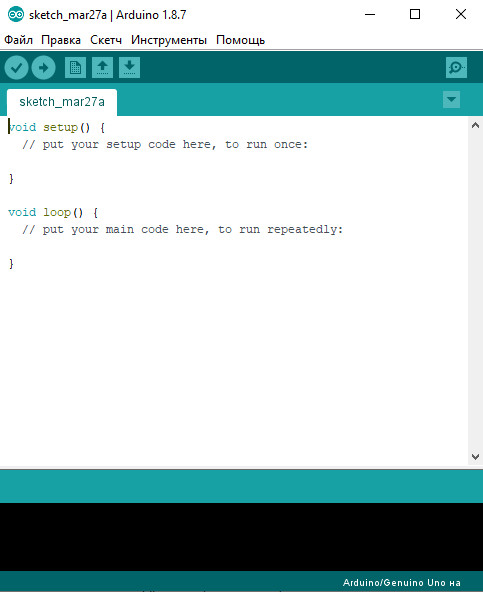
**1.2.Что такое "Умный Дом Йотик М2"?**

Данный набор создан для  изучения основ автоматизации, программирования и Интернета вещей (1оТ). Это идеальная интеграция реального и цифрового пространства.

В состав образовательного набора "Умный дом ЙоТик М2" входит:

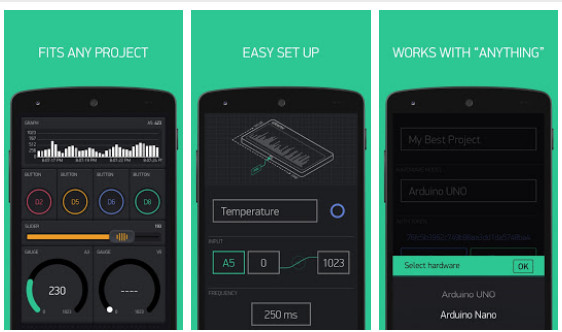
* Контроллер ЙоТик® 32 A v2.0 базовая комплектация
* Плата расширения MGB-DA20 RJ-9 V2.0
* Плата расширения I2C разъем RJ-9 MGB-D14
* Модуль двух реле MGR-2 с разъемом RJ-9
* Датчик освещенности MGS-L75 разъем RJ-9 (BH1750)
* Датчик темп., влажн. воздуха и атм. давления MGS-THP80 разъем RJ-9 (BME280)
* Датчик расстояния лазерный MGS-D20 разъем RJ-9 (VL53L0X)
* Микро сервопривод TowerPro SG90 DIGITAL SERVO
* Переходник MGA-RJ9 с RJ-9 на штырьковые разъемы с внешним питанием
* Датчик лет.орг. Соед. и эквив. Конц. CO2 (CO2eq) MGS-CO30 разъем RJ-9 (SGP30)
* Вентилятор Deepcool WIND BLADE 80 80x80x25
* Модуль RGB светодиода MGL-RGB1 с разъемом RJ-9
* Плата питания с клеммными колодками Power Hub C1
* Датчик цвета, освещённости, приближения и жестов MGS-CLM60 разъём RJ-9 (APDS-9960)
* Амперметр 10А (40х40)
* PBS-28B-W, Кнопка антивандальная OFF-(ON) (2A 250VAC) IP66 Waterproof
* Модуль графического LCD дисплея MGB-LCD12864 128x64 с разъемом RJ-9
* Датчик звука MGS-SND504-A7 с разъемом RJ-9 (INMP504/INMP510)
* Модуль динамика MGB-BUZ1 с разъемом RJ-9
* Датчик протечки воды MGS-WT1-A1 с разъемом RJ-9
* Датчик пламени MGS-FR403 ИК с разъемом RJ-9 (TSL25403)
* Адаптер MGA-FIX1
* Джампер
* блок питания 12В 3А
* Комплект проводов RJ-9
* Комплект провод с НШВИ (15-20 см)
* Кабель для питания вентилятора
* Кабель для считывания скорости вентилятора
* кабель USB (A-B) синий
* Корпус дома из фанеры и акрила
* Отвертка
* Буклет-паспорт с инструкцией

### 1.3. Что такое Arduino IDE?

Аббревиатура IDE расшифровывается как Integrated Development Environment, в переводе – интегрированная среда разработки. С помощью этой программы программисты пишут программы, причем делают это гораздо быстрее и удобнее, чем при использовании обычных текстовых редакторов.

В рамках платформы Ардуино программа Arduino IDE делает то же – помогает программистам писать программы. С ее помощью скетч, написанный на языке Arduino, проверяется, превращается в C++, компилируется, загружается в Arduino. Теоретически, вы можете обойтись без этой программы, но реально других вариантов начать работы с Ардуино для новичка практически не существует. Поэтому первое, что вы должны сделать – это найти и установить  себе эту среду программирования. Это совершенно не трудно и абсолютно бесплатно.

**1.4.Что такое Blynk?**

Blynk представляет собой облачный сервис для создания графических пультов управления и подходит для широкого спектра микрокомпьютеров и микроконтроллеров. Там, где раньше для сбора информации с датчиков нужно было писать полноценный интерфейс ввода-вывода или приобретать дополнительные модули, теперь можно обойтись пятиминутной работой в Blynk.

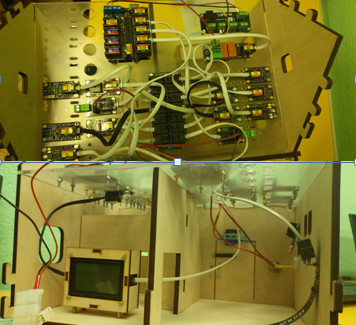
Для создания собственного проекта с управлением через Blynk нужно совсем немного: установить приложение (доступны версии для iOS и Android) или воспользоваться веб-формой. Тут потребуется регистрация в один шаг — ввод email и пароля. Регистрация нужна ввиду того, что Blynk — облачное решение и без неё контроль над железкой может получить любой пользователь.

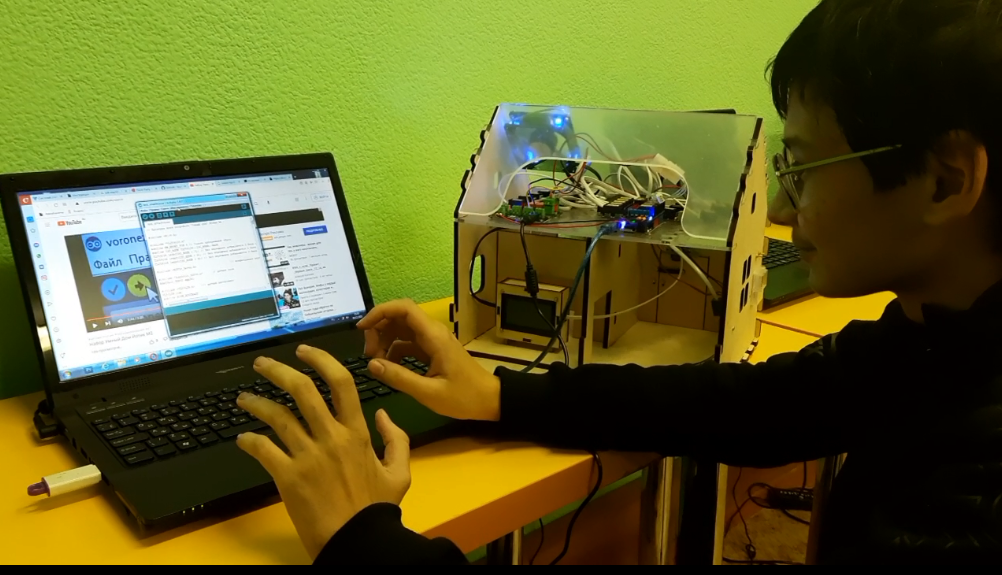
Желающие могут установить сервер локально. В таком случае доступ в интернет не нужен.

**Глава 2. Практическая часть**

**2.2 Сборка каркаса макета и установка датчиков.**

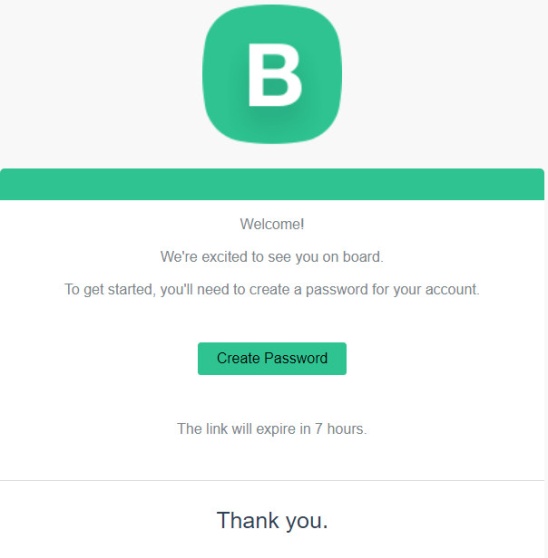
 Изучив теоретические сведения о работе IoT, я приступил к практической части. Ознакомившись со всеми датчиками, элементами конструкции и инструкциями я приступил к сборке макета. Признаюсь было очень интересно соединять все датчики проводами, рассматривать их устройство, знакомиться с принципом работы, прикреплять каждый на свое место.

После того, как макет был собран, я подключил дом к компьютеру. Пришло время устанавливать программное обеспечение под названием Arduino IDE. Это та самая среда для написания алгоритма работы или скетча. Приложение установилось без проблем. Но этого было не достаточно. Дело в том, что к Arduino IDE существуют специальные библиотеки. Их назначение облегчить и сократить написание скетча, а так же познакомить приложение с именно нашим оборудованием. Дополнительные библиотеки были установлены. В приложении были указаны именно наши платы. Теперь можно было попробовать загрузить демо-скетч для проверки соединения с компьютером, а так же проверки всего оборудования.

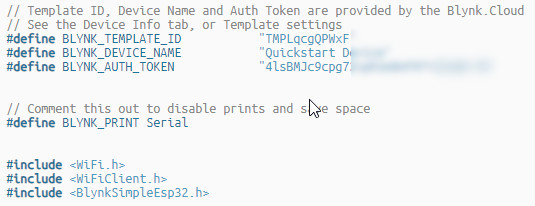
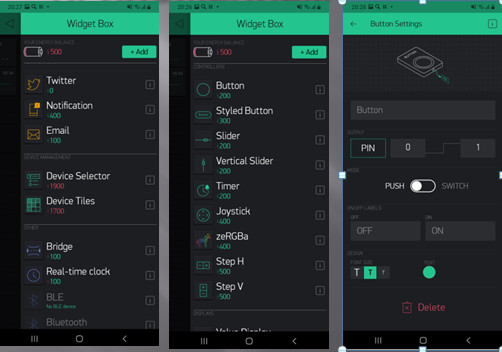


Этот скетч был скомпилирован без проблем. Освещение дома заработало, кулер начал вращаться.

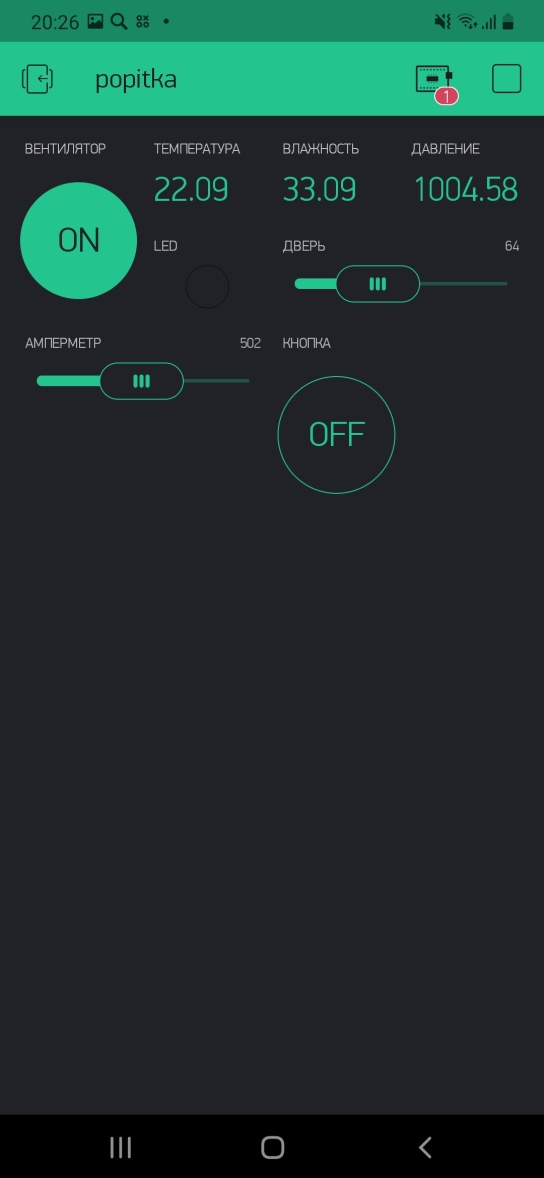
**2.2.Работа с сервисами Blynk и приложениями.**

Теперь можно было переходить к следующему этапу – регистрация на платформе Blynk [7], подключение "Умный Дом Йотик М2" к интернету, установка приложения на смартфон и создание виртуального пульта управления для нашего устройства.

Во время регистрации на сервисе, ко мне на электронную почту приходили различные письма, для формирования логина пароля для личной учетной записи на сервере.

Когда учетная запись была создана, приложение было установлено на смартфон, сервис предоставил мне личный токен для моего устройства, что давало мне право использовать сервис Blynk[8]. А так же частичный скетч для подключения моего устройства через Wi-Fi.

Можно было приступать к созданию пульта на смартфоне. Функционал приложения очень большой. Можно использовать кнопки, скролы, лейблы и многое другое. Задача состоит в том, чтобы при программировании допустим кнопки мы указали входящий цифровой порт на умном доме и выходящий. Тогда сервис Blynk будет подавать на входные порты сигналы через интернет, а так же принимать от них отклик. В результате долгой и кропотливой работы, поскольку опыта у меня еще мало, я сделал следующий пульт:

Как видно на картинке, кнопка под названием «Вентилятор» включает и выключает кулер на крыше макета. Так же на дисплее отображается температура в помещении, влажность и давление. Бегунок отвечает за открытие и закрытие двери дома. Над ней установлен сервопривод. Когда двигаешь бегунок под названием «Амперметр», то на амперметре установленном на «Умном доме» стрелка откланяется влево и вправо. Это схематичный пример управления напряжением дома.

Кнопка в самом низу под названием «Кнопка» отвечает за входной звонок. Когда ее нажимаешь, играет мелодия.

Еще раз хочется уточнить, что данный пульт созданный на смартфоне в приложении Blynk управляет механизмами через интернет по принципу работы IoT. Домом можно управлять, а так же просмотреть его состояние в режиме реального времени из любой точки планеты, где есть доступ к интернету.

На последок, в эту работу я хочу выложить получившийся на данный момент скетч, осованный на том скетче, который шел в комплекте с набором. Он полностью работает. Единственное, что я сделал- это удалил из него свой токен, название сети Wi-Fi и пароль. Если кто-то захочет использовать мой скетч, тогда введите свои значения в этих строках.

#define BLYNK\_PRINT Serial

#include <WiFi.h>

#include <WiFiClient.h>

#include <Adafruit\_ADS1015.h>

#include <BlynkSimpleEsp32.h>

#include <Wire.h>

#include <SPI.h> // конфигурация блинка // Blynk configuration

#include "TLC59108.h"

#define HW\_RESET\_PIN 0 // Только програмнный сброс

#define I2C\_ADDR TLC59108::I2C\_ADDR::BASE

TLC59108 leds(I2C\_ADDR + 7); // Без перемычек добавляется 3 бита адреса

TLC59108 leds2(I2C\_ADDR + 0); // Без перемычек добавляется 3 бита адреса // RGB модуль

TLC59108 leds3(I2C\_ADDR + 6); // Без перемычек добавляется 3 бита адреса // RGB модуль

#include <ESP32\_Servo.h> // конфигурация сервомотора // servo configuration

/\*

#include "Adafruit\_SGP30.h" // датчик газа

Adafruit\_SGP30 sgp30;

\*/

#include <VL53L0X.h> /// датчик расстояния

VL53L0X lox;

#define HIGH\_ACCURACY

#define sensor\_addr 0x39 // датчик пламени

float ir\_data = 0;

float vis\_data = 0;

#include <Adafruit\_MCP4725.h> // динамик

Adafruit\_MCP4725 buzzer;

int ton;

int vol1 = 1000; // Уровень громкости = vol1-vol2

int vol2 = 900; //

#include <Adafruit\_LSM9DS1.h> // гироскоп

Adafruit\_LSM9DS1 lsm = Adafruit\_LSM9DS1();

#include <I2C\_graphical\_LCD\_display.h> // дисплей

I2C\_graphical\_LCD\_display lcd;

#include "mcp3021.h"

uint8\_t adcDeviceId = 0b00000001; // Адрес микросхемы A0

MCP3021 mcp3021;

const float air\_value = 570.0;

const float water\_value = 335.0;

const float moisture\_0 = 0.0;

const float moisture\_100 = 100.0;

#include "MCP3221.h" // микрофон

const byte DEV\_ADDR = 0x4D;

MCP3221 mcp3221(DEV\_ADDR);

#define pump 16 // пин насоса // pump pin

#define wind 17 // пин вентилятора // cooler pin

#define amper 14 // пин амперметра

#define button 4 // пин кнопки

Servo myservo;

int pos = 1; // начальная позиция сервомотора // servo start position

int prevangle = 1; // предыдущий угол сервомотора // previous angle of servo

const byte picture [] PROGMEM = { // картинка

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0x80, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x80, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0xE0,

0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0,

0xF0, 0xF0, 0xE0, 0xE0, 0xE0, 0xC0, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x80,

0x80, 0xC0, 0xE0, 0xE0, 0xE0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xE0, 0xE0, 0xE0, 0xC0, 0xC0, 0x80,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0,

0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0xFE,

0xF8, 0xF0, 0xE0, 0xF0, 0xF8, 0xFC, 0x7F, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,

0x00, 0xF8, 0xFE, 0xFF, 0xFF, 0xFF, 0x1F, 0x07, 0x03, 0x03, 0x01, 0xE1, 0xE1, 0xE1, 0xE3, 0xE3,

0xE3, 0xE3, 0xE1, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF8, 0xF8, 0xF8,

0xF8, 0xF9, 0xFF, 0xFF, 0xFF, 0xDF, 0x8F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF0, 0xFC, 0xFF, 0xFF,

0xFF, 0x0F, 0x03, 0xC3, 0xC1, 0xC0, 0xC0, 0xC4, 0xFC, 0xF8, 0xF9, 0xF1, 0x03, 0x07, 0x1F, 0xFF,

0xFF, 0xFE, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0xFF, 0xFF, 0xFF, 0xFF,

0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00,

0x03, 0x07, 0x0F, 0x07, 0x03, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00,

0x00, 0x03, 0x0F, 0x3F, 0x7F, 0x7F, 0xFF, 0xFC, 0xF8, 0xF0, 0xF0, 0xF1, 0xF1, 0xF1, 0xFF, 0xFF,

0xFF, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF1, 0xF1, 0xF1,

0xF1, 0xF1, 0xF1, 0xFF, 0xFF, 0xFF, 0x7F, 0x3F, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,

0xFF, 0xFC, 0xF0, 0xE0, 0xE1, 0xC3, 0xC7, 0xC7, 0xC7, 0xC7, 0xC3, 0xE1, 0xF0, 0xF8, 0xFE, 0xFF,

0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF,

0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,

0x01, 0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01,

0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,

0x01, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01,

0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00

};

char ssid[] = "???????"; // Логин Wi-Fi // Wi-Fi login

char pass[] = "???????"; // Пароль от Wi-Fi // Wi-Fi password

char auth[] = "?????????????????????????"; // Токен // Authorization token

IPAddress blynk\_ip(139, 59, 206, 133); // конфигурация блинка // Blynk configuration

#include <BH1750FVI.h> // добавляем библиотеку датчика освещенности // adding Light intensity sensor library

BH1750FVI LightSensor\_1; // BH1750

#include <Adafruit\_Sensor.h> // добавляем библиотеку датчика температуры, влажности и давления // adding Temp Hum Bar sensor library

#include <Adafruit\_BME280.h> // BME280

Adafruit\_BME280 bme280; //

#define UPDATE\_TIMER 3333

BlynkTimer timer\_update; // настройка таймера для обновления данных с сервера BLynk // Blynk update timer configuration

#define UPDATE\_button 250

#define UPDATE\_sound 25

//////////////////////////////////////////НАСТРОЙКИ/CONFIGURATION/////////////////////////////////////////////////////////////////

void setup()

{

myservo.attach(13); // пин сервомотора // servo pin

init\_sensor();

lcd.begin();

lcd.gotoxy (0, 0);

lcd.clear (0, 0, 128, 64, 0x00); // очищаем поле дисплея

pinMode( pump, OUTPUT );

pinMode( button, INPUT );

pinMode( wind, OUTPUT ); // настройка пинов насоса и вентилятора на выход // pump and cooler pins configured on output mode

digitalWrite(pump, LOW); // устанавливаем насос и вентилятор изначально выключенными // turn cooler and pump off

digitalWrite(wind, LOW);

Serial.begin(115200);

delay(512);

Blynk.begin(auth, ssid, pass, blynk\_ip, 8442); // подключение к серверу Blynk // connection to Blynk server

Wire.begin();

leds.init(HW\_RESET\_PIN);

leds.setLedOutputMode(TLC59108::LED\_MODE::PWM\_IND);

leds2.init(HW\_RESET\_PIN);

leds2.setLedOutputMode(TLC59108::LED\_MODE::PWM\_IND);

leds3.init(HW\_RESET\_PIN);

leds3.setLedOutputMode(TLC59108::LED\_MODE::PWM\_IND);

buzzer.begin(0x61); // С перемычкой адрес будет 0x60

buzzer.setVoltage(0, false); // выключение звука

delay(1000);

/\* if (!sgp30.begin())

Serial.println("Sensor SGP30 not found!");\*/

lsm.begin();

Serial.println("Found LSM9DS1 9DOF");

lsm.setupAccel(lsm.LSM9DS1\_ACCELRANGE\_2G);

lsm.setupMag(lsm.LSM9DS1\_MAGGAIN\_4GAUSS);

lsm.setupGyro(lsm.LSM9DS1\_GYROSCALE\_245DPS);

LightSensor\_1.begin(); // запуск датчика освещенности // turn the light intensity sensor on

LightSensor\_1.setMode(Continuously\_High\_Resolution\_Mode);

lox.init();

lox.setTimeout(500);

#if defined LONG\_RANGE

// lower the return signal rate limit (default is 0.25 MCPS)

lox.setSignalRateLimit(0.1);

// increase laser pulse periods (defaults are 14 and 10 PCLKs)

lox.setVcselPulsePeriod(VL53L0X::VcselPeriodPreRange, 18);

lox.setVcselPulsePeriod(VL53L0X::VcselPeriodFinalRange, 14);

#endif

#if defined HIGH\_SPEED

// reduce timing budget to 20 ms (default is about 33 ms)

lox.setMeasurementTimingBudget(20000);

#elif defined HIGH\_ACCURACY

// increase timing budget to 200 ms

lox.setMeasurementTimingBudget(200000);

#endif

bool bme\_status = bme280.begin();

if (!bme\_status)

Serial.println("Could not find a valid BME280 sensor, check wiring!"); // проверка датчика температуры, влажности и давления // checking the temp hum bar sensor

ledcSetup(5, 50, 10);

ledcAttachPin(amper, 5);

timer\_update.setInterval(UPDATE\_TIMER, readSendData); // включаем таймер обновления данных // turn on the update timer

timer\_update.setInterval(UPDATE\_button, butt);

timer\_update.setInterval(UPDATE\_sound, sound);

mcp3021.begin(adcDeviceId);

}

//////////////////

void sound() { // зажигется красный если есть отклонение по оси Х, также открывется картинк на дисплее если был задетекчен звук

float snd = mcp3221.getVoltage();

Serial.println(snd);

lsm.read();

sensors\_event\_t a, m, g, temp;

lsm.getEvent(&a, &m, &g, &temp);

if (a.acceleration.x > 0.00 or a.acceleration.x < -0.20) {

leds.setBrightness(3, 0xff);

}

else {

leds.setBrightness(3, 0x00);

}

if (snd < 1030) {

lcd.blit (picture, sizeof picture);

}

float dist = lox.readRangeSingleMillimeters();

if (dist < 100) {

leds3.setBrightness(2, 0xFF);

}

else {

leds3.setBrightness(2, 0x00);

}

poll\_sensor();

if (ir\_data>700){

leds3.setBrightness(1, 0xFF);

}

else {

leds3.setBrightness(1, 0x00);

}

float l = LightSensor\_1.getAmbientLight();

if(l<150){

leds.setBrightness(0, 0xFF);

leds2.setBrightness(0, 0xFF);

leds3.setBrightness(0, 0xFF);

leds.setBrightness(6, 0xFF);

leds2.setBrightness(6, 0xFF);

leds3.setBrightness(6, 0xFF);

}

else{

leds.setBrightness(0, 0x00);

leds2.setBrightness(0, 0x00);

leds3.setBrightness(0, 0x00);

leds.setBrightness(6, 0x00);

leds2.setBrightness(6, 0x00);

leds3.setBrightness(6, 0x00);

}

}

//////////////// //

void butt() { // при нажатии кнопки загорается светодиод в блинке, загорается красный и уФ в ргб модулях и картинк исчезает

WidgetLED led1(V19);

int ledstate = 0;

ledstate = digitalRead(button);

if (ledstate == HIGH) {

led1.off();

}

else {

led1.on();

for (byte pwm = 0; pwm < 0xff; pwm++) {

leds.setBrightness(3, pwm);

leds2.setBrightness(1, pwm);

leds2.setBrightness(4, pwm);

delay(1);

}

for (byte pwm = 0xfe; pwm < 0xff; pwm--) {

leds.setBrightness(3, pwm);

leds2.setBrightness(1, pwm);

leds2.setBrightness(4, pwm);

delay(1);

}

lcd.clear (0, 0, 128, 64, 0x00);

}

}

//////////////////////////////////////////////////////ЧТЕНИЕ И ЗАПИСЬ ДАННЫХ ДАТЧИКОВ/SENSOR DATA SEND/READ////////////////////////////////////////////////

void readSendData() {

float adc0 = mcp3021.readADC();

float hum = map(adc0, air\_value, water\_value, moisture\_0, moisture\_100);

Blynk.virtualWrite(V1, hum); delay(2); // Отправка данных на сервер Blynk Давление // Pressure data send

float t = bme280.readTemperature();

float h = bme280.readHumidity();

float p = bme280.readPressure() / 100.0F;

Blynk.virtualWrite(V14, t); delay(2); // Отправка данных на сервер Blynk Температура // Temperature data send

Blynk.virtualWrite(V15, h); delay(2); // Отправка данных на сервер Blynk Влажность // Humidity data send

Blynk.virtualWrite(V16, p); delay(2); // Отправка данных на сервер Blynk Давление // Pressure data send

float eco2 = 0;

float tvoc = 0;

/\* sgp30.IAQmeasure();

// if (!sgp30.IAQmeasure())

// {

// Serial.println("Measurement failed!");

// return;

// }

tvoc = sgp30.TVOC;

eco2 = sgp30.eCO2;

Serial.println(tvoc);

Serial.println(eco2);

Blynk.virtualWrite(V12, tvoc); delay(2); // Отправка данных на сервер Blynk Влажность // Humidity data send

Blynk.virtualWrite(V13, eco2); delay(2); // Отправка данных на сервер Blynk Давление // Pressure data send\*/

}

/////////////////////////////////////////////////ГЛАВНЫЙ ЦИКЛ/MAIN LOOP///////////////////////////////////////////////////////////////////////////////

void loop()

{

Blynk.run(); // запуск Blynk // turn Blynk on

timer\_update.run();

}

////////////////////////////////////////////////ДВЕРЬ/DOOR///////////////////////////////////////////////////////////////////////////////////

BLYNK\_WRITE(V0)

{

int angle = param.asInt();

if (prevangle < angle) {

for (pos = prevangle; pos <= angle; pos += 1)

{

myservo.write(pos);

delay(5); // если угол задан больше предыдущего, то доводим до нужного угла в ++ // if the current angle>previous angle then going clockwise

}

prevangle = angle;

}

else if (prevangle > angle) {

for (pos = prevangle; pos >= angle; pos -= 1)

{

myservo.write(pos);

delay(5); // если угол задан меньше предыдущего, то доводим до нужного угла в -- // if the current angle<previous angle then going counter-clockwise

}

prevangle = angle;

}

}

//////////////////////////////////////////////НАСОС/PUMP///////////////////////////////////////////////////////////////

BLYNK\_WRITE(V5)

{

int buttonstate1 = param.asInt ();

if (buttonstate1 == 1) {

digitalWrite(pump, HIGH); // включить если нажата кнопка "Насос" // turn on the pump if button = 1

}

else {

digitalWrite(pump, LOW);

}

}

//////////////////////////////////////////ВЕНТИЛЯТОР/COOLER/////////////////////////////////////////////////////////////

BLYNK\_WRITE(V6)

{

int buttonstate2 = param.asInt ();

if (buttonstate2 == 1) {

digitalWrite(wind, HIGH); // включить, если нажата кнопка "Вентилятор" // turn on the cooler if button = 1

}

else {

digitalWrite(wind, LOW);

}

}

/////////////////////////////////////ЭКРАН///////////////////

BLYNK\_WRITE(V4) /// картинк и мелодия тетрис

{

int president = param.asInt ();

if (president == 1) {

lcd.blit (picture, sizeof picture);

note(14, 400); note(2, 100); note(9, 400); note(7, 500); note(14, 300); note(9, 700); buzzer.setVoltage(0, false); // выключение звука

}

else {

lcd.clear (0, 0, 128, 64, 0x00);

buzzer.setVoltage(0, false); // выключение звука

}

}

/////////////////////АМПЕРМЕТР///////////////////

BLYNK\_WRITE(V18)

{

int pwr = param.asInt();

ledcWrite(5, pwr);

delay(10);

}

////////////////////////////

/////////////////////////////////////////////////////////////////////////// все для музыки

int note( int type, int duration) { // нота (какая нота, длительность)

switch (type) {

case 1: ton = 1000; break;

case 2: ton = 860; break;

case 3: ton = 800; break;

case 4: ton = 700; break;

case 5: ton = 600; break;

case 6: ton = 525; break;

case 7: ton = 450; break;

case 8: ton = 380; break;

case 9: ton = 315; break;

case 10: ton = 250; break;

case 11: ton = 190; break;

case 12: ton = 130; break;

case 13: ton = 80; break;

case 14: ton = 30; break;

case 15: ton = 1; break;

}

delay(10);

for (int i = 0; i < duration; i++) {

buzzer.setVoltage(vol1, false);

buzzer.setVoltage(vol2, false);

delayMicroseconds(ton);

}

}

///////////////////////////////////////////////////

void init\_sensor() {

Wire.begin();

Wire.beginTransmission(sensor\_addr);

Wire.write(0x81); // Регистр времени интегрирования АЦП

Wire.write(0b00111111); // 180 мс, 65535 циклов

Wire.endTransmission();

Wire.beginTransmission(sensor\_addr);

Wire.write(0x83); // Регистр времени ожидания

Wire.write(0b00111111); // 180 мс

Wire.endTransmission();

Wire.beginTransmission(sensor\_addr);

Wire.write(0x90); // Регистр усиления

Wire.write(0b00000000); // Усиление 1x

Wire.endTransmission();

Wire.beginTransmission(sensor\_addr);

Wire.write(0x80); // Регистр управления питанием

Wire.write(0b00001011); // Включение ожидания, генератора, АЦП и ALS сенсора

Wire.endTransmission();

}

// Получение данных с датчика

void poll\_sensor() {

unsigned int sensor\_data[4];

Wire.beginTransmission(sensor\_addr);

Wire.write(0x94); // Начальный адрес регистров данных

Wire.endTransmission();

Wire.requestFrom(sensor\_addr, 4);

if (Wire.available() == 4) {

sensor\_data[0] = Wire.read();

sensor\_data[1] = Wire.read();

sensor\_data[2] = Wire.read();

sensor\_data[3] = Wire.read();

}

ir\_data = sensor\_data[3] \* 256.0 + sensor\_data[2];

vis\_data = sensor\_data[1] \* 256.0 + sensor\_data[0];

}

Желаю Вам удачи!

**Заключение.**

Мне кажется что за технологией IoT-будущее. В процессе создания проекта я познакомился с технологиями «Интернет вещей», попробовал свои силы в программировании. «Умный дом» действительно управляется через интернет с помощью пульта на смартфоне. Все цели были достигнуты. Гипотеза подтверждена.

В заключении хотелось бы отметить, что проект мне показался сложным, но очень интересным и увлекательным. После работы над ним, мне хочется по подробней изучить эту тему. Понравилось программировать. Появилось желание изучить Pyton и PHP. Думаю, что эти языки программирования будут востребованы всегда.

Надеюсь, мой проект будет полезен и интересен читателю.

**Список информационных ресурсов:**

1. [**https://mgbot.ru/catalog/obrazovatelnye\_nabory\_iot/umnyy\_dom\_yotik\_m2/**](https://mgbot.ru/catalog/obrazovatelnye_nabory_iot/umnyy_dom_yotik_m2/)
2. [**https://www.sas.com/ru\_ru/insights/big-data/internet-of-things.html**](https://www.sas.com/ru_ru/insights/big-data/internet-of-things.html)
3. [**https://ru.wikipedia.org/wiki/Arduino\_IDE**](https://ru.wikipedia.org/wiki/Arduino_IDE)
4. [**https://domoticzfaq.ru/umnyj-dom-ot-rostelekom/**](https://domoticzfaq.ru/umnyj-dom-ot-rostelekom/)
5. [**https://vash-master.tv/umniy-dom-tricolor**](https://vash-master.tv/umniy-dom-tricolor)
6. [**https://lifehacker.ru/blynk/**](https://lifehacker.ru/blynk/)
7. [**https://blynk.io/en/getting-started**](https://blynk.io/en/getting-started)
8. [**https://secretmag.ru/enciklopediya/chto-takoe-token-obyasnyaem-prostymi-slovami.htm**](https://secretmag.ru/enciklopediya/chto-takoe-token-obyasnyaem-prostymi-slovami.htm)