

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра комплексной информационной безопасности электронно-
вычислительных систем (КИБЭВС)

RGB-ПОДСВЕТКА ДЛЯ РАСТЕНИЙ

Курсовая работа
по дисциплине
«Технологии Интернета вещей»
Пояснительная записка

Обучающийся гр.720-1

(подпись) В.А. Кикоть
(И.О. Фамилия)

(дата)

Руководитель
ст.преп. кафедры КИБЭВС

(подпись) О.В. Пехов
(И.О. Фамилия)

(оценка)

(дата)

Томск 2023

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра комплексной информационной безопасности электронно-
вычислительных систем (КИБЭВС)

УТВЕРЖДАЮ

Заведующий кафедрой
КИБЭВС, д-р техн. наук,
профессор

_____ А.А.

Шелупанов

“ ____ ” _____ 2022

г.

ЗАДАНИЕ

на курсовую работу по дисциплине «Технологии Интернета вещей»
студенту Кикоть Валерии Артемовне группы 720-1 факультета безопасности.

1 Тема работы: RGB-подсветка для растений

2 Исходные данные к работе: научная статья «ВЛИЯНИЕ
ОСВЕЩЕНИЯ НА ВЫРАЩИВАНИЕ РАСТЕНИЙ В ТЕПЛИЦАХ С
СИСТЕМОЙ ЗАКРЫТОГО ГРУНТА» Кирилович Ю.Г., Латкова Е.В., Латков
Н.Ю.

2.1 Данные по предметной области: пользователь, сервер,
приложение, устройство

3 Срок сдачи студентом законченной работы: 01.07.2023

4 Содержание курсовой работы:

4.1 Проектирование инфологической модели данных:

- описание и структуризация предметной области (описание бизнес-процессов, диаграммы IDEF0);
- описание взаимодействия в системе (IML).

4.2 Написание программы получения и обработки данных:

- Преобразование данных для работы со светодиодной лентой.

5 Содержание пояснительной записки:

- титульный лист;
- реферат на русском языке;
- задание;
- содержание;
- введение;
- актуальность, обзор аналогов;
- формирование требований к своему решению;
- описание архитектуры решения;
- разработка функционала;
- заключение;
- список использованных источников;
- приложения (экранные формы, листинг программы и др.).

Пояснительная записка должна быть оформлена в соответствии со стандартом ТУСУР.

6 Дата выдачи задания:

« » _____ 2023 г.

Задание согласовано:

Руководитель работы

Пехов О.В., старший преподаватель кафедры КИБЭВС

“ ” _____ 202 г.

Задание принято к исполнению

“ _____ ” _____ 2023 г. _____ Кикоть В.А.

Реферат

Курсовая работа, 23 страницы, 5 рисунков, 4 источника, 1 приложение.

ПОЛЬЗОВАТЕЛЬ, MQTT, C#, MBED, ПРИЛОЖЕНИЕ, УСТРОЙСТВО.

Объект разработки: устройство, которое является дополнительной подсветкой при необходимости с настройкой пользователя через приложение при помощи команд.

Цель работы: разработать устройство, которое будет помогать в уходе за растениями или рассадой.

В процессе работы было проведено изучение предметной области, изучены характеристики света и растений.

Было собрано устройство из компонентов: плата STM32Nucleo, модуль датчика освещенности на основе OPT3001 (плата UMDK-LIT), лента светодиодная RGB, модуль транзисторных ключей UMDK-6FET, гнездо питания 2.1 мм с клеммником, импульсный блок питания 12 В, Wi-Fi модуль ESP8266.

Программа для реагирования на освещенность и чтение команды пользователя была написана на C# Mbed.

Пояснительная записка выполнена в текстовом редакторе Microsoft Word 2021 и по ОС ТУСУР 01-2021.[4]

В результате было разработано устройство, которым можно управлять через заданные команды и которое реагирует на освещенность в месте работы, так на ярком солнце лента не будет светит или в пасмурную погоду будет ярче, чем в обычный день.

Оглавление

Введение.....	6
1 АКТУАЛЬНОСТЬ.....	7
2 ОБЗОР АНАЛОГОВ	9
3 ФОРМИРОВАНИЕ ТРЕБОВАНИЙ К СВОЕМУ РЕШЕНИЮ	10
4 ОПИСАНИЕ АРХИТЕКТУРЫ РЕШЕНИЯ	11
5 РАЗРАБОТКА ФУНКЦИОНАЛА.....	13
Заключение	16
Список использованных источников	17
Приложение А (обязательное) Код программы с устройства	18

Введение

Целью выполнения данной курсовой работы является разработка умного светильника, который будет освещать растения в необходимой мере в зависимости от этапа развития растения.

Область применения разрабатываемого объекта проращивание растений, их выращивание и цветение и плодоношение. Подходит для людей, занимающихся выращиванием огорода как на подоконнике, так и для дальнейшей высадки в землю, или для тех, кто занимается растениями для дома.

Целесообразность состоит в том, что растениям необходима определенная длина волны при проращиваниях, выращивании и цветении.

1 АКТУАЛЬНОСТЬ

В современном мире есть много статей, посвященных зависимости растений от света. Так его недостаток может привести к тому, что растения ведут себя непредсказуемо: фикус сбросил нижние листья, гибискус отказывается цвести, а рассада томатов стала бледной и начала вытягиваться.

Рассада при дефиците света вытягивается, стебли становятся тонкими и слабыми, а листья — бледными. После пикировки молодые растения тяжело приживаются и долго болеют.

Чтобы помочь растениям справиться с недостатком света, используют дополнительное освещение светодиодные фитолампы. Они приносят максимум пользы при минимальном расходе электроэнергии. Кроме того, в отличие от люминесцентных и обычных лампочек, светодиодные фитосветильники почти не нагревают воздух.

Какой уровень освещенности будет достаточным? Это зависит от конкретного сорта и культуры. Примерные параметры таковы:

1000-3000 лк — для тенелюбивых растений;

до 4000 лк — для культур, которым достаточно рассеянного света;

до 6000 лк — для растений, нуждающихся в прямом освещении;

до 12000 лк — для плодоносящих и тропических культур.

При использовании RGB подсветки выделяются преимущества каждого света.

Красный:

- ускоряет прорастание рассады, необходим на начальном этапе;
- стимулирует вертикальный рост.

Синий и фиолетовый:

- активизируют процесс фотосинтеза;
- повышают скорость деления клеточной массы;

- если синего спектра достаточно, побеги не вытягиваются, а стебель становится более толстым и плотным.

Также стоит отметить что, фитолампы нельзя держать включёнными круглосуточно: для взрослых растений короткого светового дня он не должен длиться дольше 8–10 часов; длинного светового дня — 14 часов; для рассады в период активного роста — 12–16 часов.

2 ОБЗОР АНАЛОГОВ

В ходе изучения интернет-магазинов было выявлено, что полных аналогов не существует. Так на рынке нет ламп, которые будут подстраиваться под этап развития растения и работало в зависимости от освещения, есть лишь «универсальные», или лампы с таймером 3, 6, 9 часов работы.

3 ФОРМИРОВАНИЕ ТРЕБОВАНИЙ К СВОЕМУ РЕШЕНИЮ

Для проекта необходимо три основных блока работы:

1. Сборка

Само устройство из необходимых компонентов:

- STM32Nucleo
- модуль датчика освещенности на основе OPT3001 (плата UMDK-LIT)
- лента светодиодная RGB
- модуль транзисторных ключей UMDK-6FET
- гнездо питания 2.1 мм с клеммником
- импульсный блок питания 12 В
- Wi-Fi модуль ESP8266

2. Программирование схемы

Программа, которая выполняет следующее: получает данные с сервера и затем управляет работой светодиодной лентой в зависимости от пришедших данных.

3. Система для работы пользователя с устройством

Для полнофункциональной работы необходимо web-приложение, которое было бы удобное для взаимодействия с пользователем и учитывала бы нюансы каждого вида растения, предпочтения в свете и тип: рассада или комнатное растение.

4 ОПИСАНИЕ АРХИТЕКТУРЫ РЕШЕНИЯ

Для описания структура была составлена схема в методологии IDEF0, которая приведена на рисунке 4.1. На рисунке видно, что на входе имеется логин с паролем и режимы работы; участниками являются пользователь, сервер (MQTT), устройство, приложение; ограничивают правила регистрации и на выходе свечение выбранного режима на устройстве.

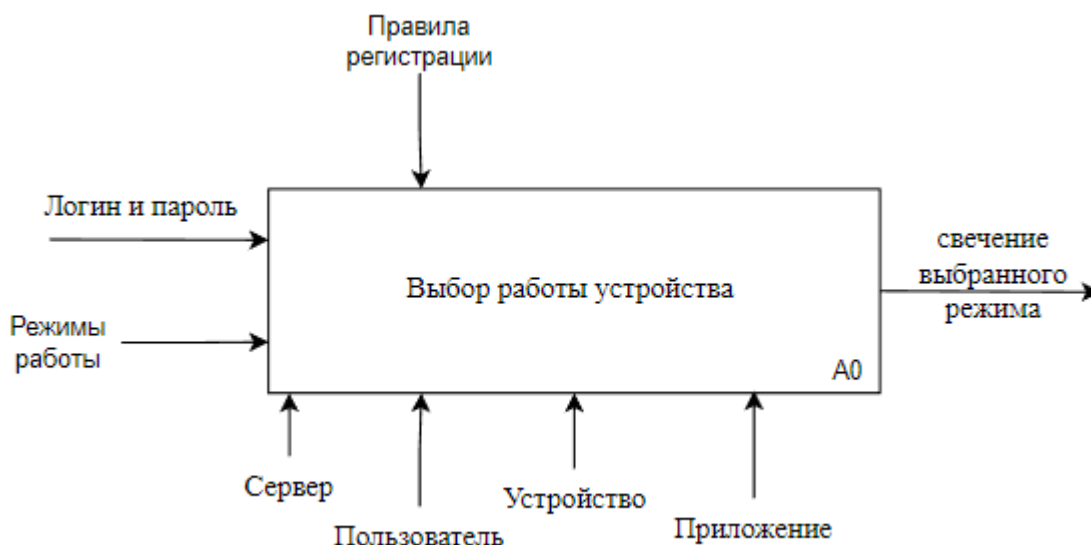


Рисунок 4.1 – Модель «черного ящика»

Для более наглядной работы была составлена декомпозиция «черного ящика», которая представлена на рисунке 4.2. Представлено, что пользователь может выполнить сразу вход на сайт или через регистрацию, если пользователь ввел данные верно, то он может выбрать режим, который будет передан на сервер, с сервера на устройство при помощи команды, если пользователь ввел данные неверно, то ему будет отказано в доступе, в этом случае он может попробовать снова войти, зарегистрироваться или не входит в систему вовсе.

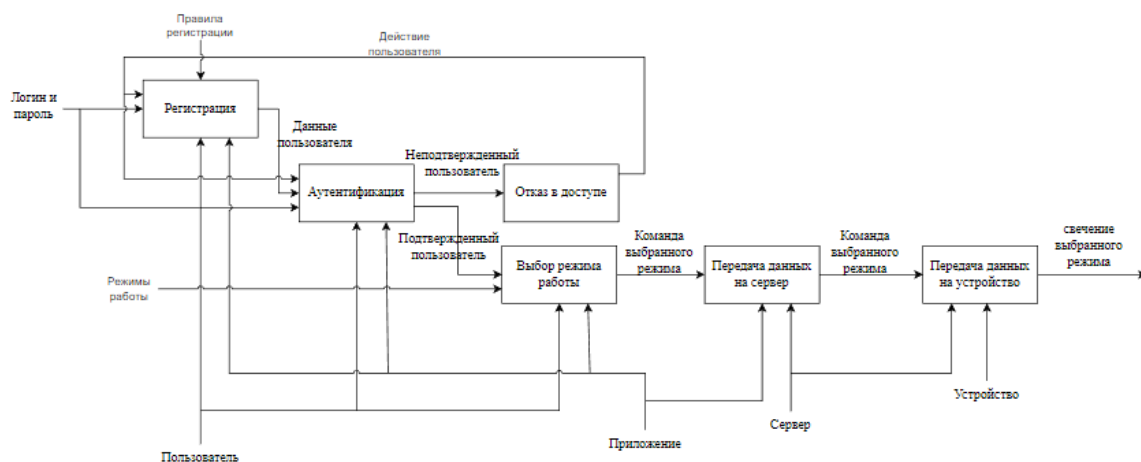


Рисунок 4.2 – Декомпозиция черного ящика.

Затем для описания того, как должна работать система была составлена UML диаграмма прецедентов, представленная на рисунке 4.3. Где пользователь взаимодействует с приложением посредством выбора команды, которая затем и на MQTT и далее на устройство.

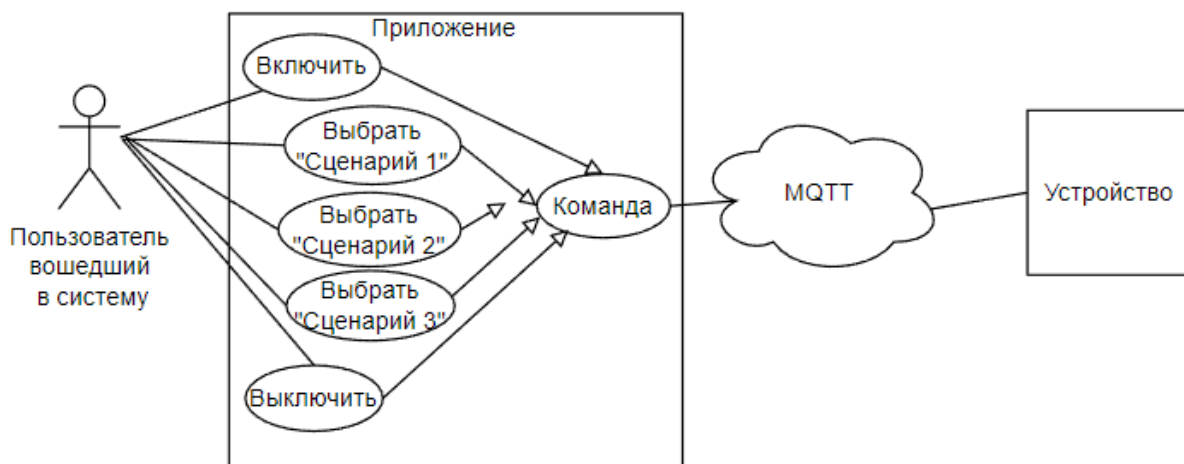


Рисунок 4.3 – UML диаграмма

5 РАЗРАБОТКА ФУНКЦИОНАЛА

Сначала было собрано устройство, представленное на рисунке 5.1. Включающая все необходимые элементы.

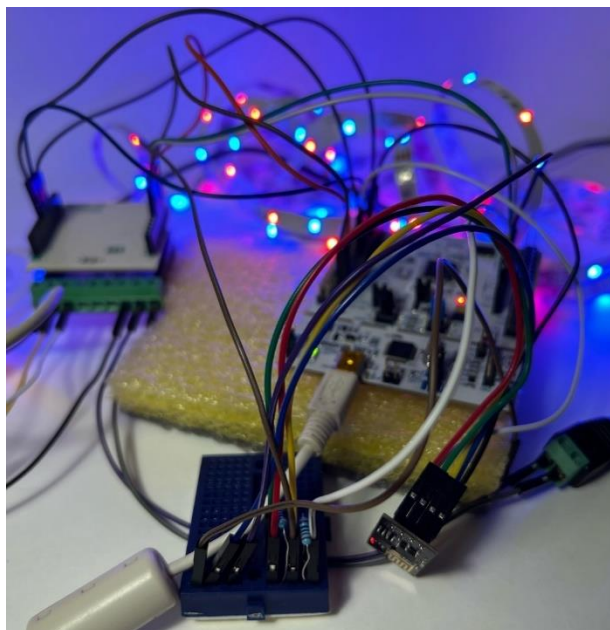


Рисунок 5.1 –

Далее была написана программа. Представленная часть кода, вызываемая при помощи `client.yield()`, записывает полученную команду в переменную.

```
void messageArrived(MQTT::MessageData& md)
{
    MQTT::Message &message = md.message;

    buffer = new char[message.payloadlen + 1];
    memcpy(buffer, message.payload, message.payloadlen);
    buffer[message.payloadlen] = '\0';
}
```

И далее происходит обработка полученного сообщения следующей частью кода, которая в зависимости от полученного сообщения и освещенности на данный момент настраивает работу светодиодной ленты.

```
string ste = "ON", redword = "red", blueword="blue";
if (buffer == ste || buffer == redword || buffer == blueword)
{
    int luminocity = sensor_opt.readSensor();
    brightness += (K * (luminocity_sp - luminocity) / MAXLUM);
    brightness = (brightness < 0.0f)? 0.0f : (brightness > 1.0f)? 1.0f :
brightness;
    green=brightness;
    if (buffer == ste){
        blue=brightness;
        red=brightness;

    }
    if (buffer == redword){
        blue=0;
        red=brightness;

    }
    if (buffer == blueword){
        blue=brightness;
        red=0;

    }
}
else {
    blue=0;
```

```
red=0;
```

```
};
```

Полный код программы представлен в приложении А.

Также стоит отметить, что пользователь взаимодействует через сайт <https://dev.righotech.io>, с помощью команда представленных на рисунке 5.2.

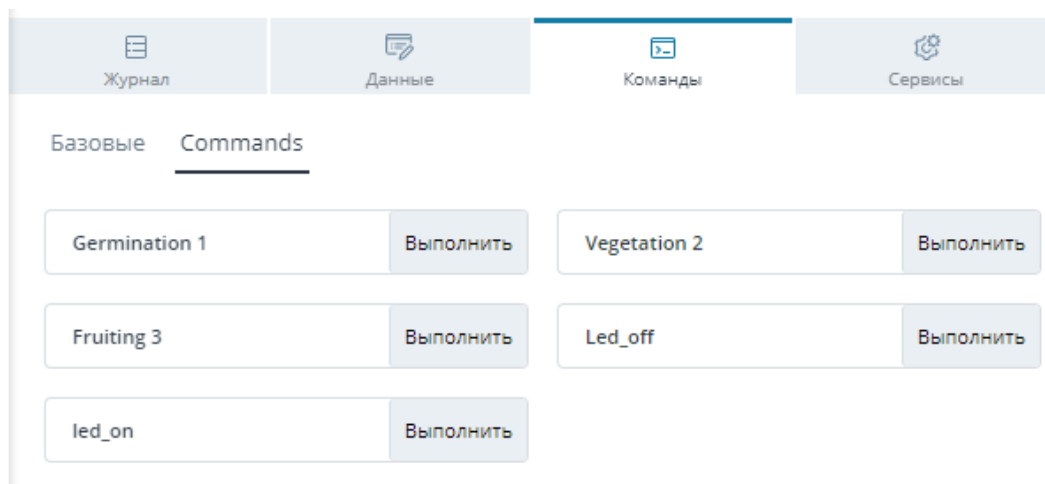


Рисунок 5.2 – Команды на сайте для работы с устройством

Для выполнения работы команд через вкладку «Модели» была настроена отправка слов в топик на сервере, которые потом считываются на устройстве.

Заключение

В ходе выполнения курсовой работы были отработаны навыки работы программирования на C# в MBed, а также работа с электронными устройствами. Помимо этого была изучена предметная область: освещение квартирных растений и рассады, действие красного и синего света на растения. Также был отработан навык по работе с диаграммами.

Список использованных источников

1. Как выбрать фитолампу для комнатных растений и рассады [Электронный ресурс]: сайт Leroy Merlen. URL:https://leroymerlin.ru/advice/osveshchenie/kak-vybrat-fitolampu-dlya-komnatnyh-rasteniy-i-rassady/?utm_referrer=https%3A%2F%2Fyandex.ru%2F (дата обращения: 30.03.2023)
2. Особенности выбора светодиодного освещения для растений [Электронный ресурс]: интернет -магазин электротехнических товаров Сумракoff URL: <https://www.sumrakoff.ru/blog/vibor-svetodiodnogo-osvesheniya-dlya-rassadi/> (дата обращения: 30.03.2023)
3. Больше света: выбираем фитолампы для растений [Электронный ресурс]: интернет-магазин SimaLand. URL:<https://www.sima-land.ru/o-kompanii/novosti-kompanii/3891/> (дата обращения: 30.03.2023)
4. Образовательный стандарт вуза ОС ТУСУР 01-2021. Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления от 25.11.2021 [Электронный ресурс]: сайт ТУСУРа. URL: [https://regulations.tusur.ru/storage/150499/ОС_ТУСУР_01-2021_\(согласовано\).pdf](https://regulations.tusur.ru/storage/150499/ОС_ТУСУР_01-2021_(согласовано).pdf) (дата обращения: 20.06.2023)

Приложение А

Обязательное

Код программы с устройства

```
#include <mbed.h>
#include "TCPSocket.h"
#include "MQTTmbed.h"
#include "MQTTClientMbedOs.h"
#include "OPT3001.h"

OPT3001 sensor_opt(I2C_SDA, I2C_SCL);
const int MAXLUM = 5000;      // Максимальная освещенность
const int luminocity_sp = 500; // Желаемый уровень освещенности
const float K = 0.1;         // Коэффициент пропорциональности

PwmOut blue(D5);
PwmOut red(D3);

WiFiInterface *wifi;

char* buffer = 0;
float brightness = 0.0f;

void messageArrived(MQTT::MessageData& md)
{
    MQTT::Message &message = md.message;

    buffer = new char[message.payloadlen + 1];
```

```

memcpy(buffer, message.payload, message.payloadlen);
buffer[message.payloadlen] = '\0';

}

const char *sec2str(nsapi_security_t sec)
{
    switch (sec) {
        case NSAPI_SECURITY_NONE:
            return "None";
        case NSAPI_SECURITY_WEP:
            return "WEP";
        case NSAPI_SECURITY_WPA:
            return "WPA";
        case NSAPI_SECURITY_WPA2:
            return "WPA2";
        case NSAPI_SECURITY_WPA_WPA2:
            return "WPA/WPA2";
        case NSAPI_SECURITY_UNKNOWN:
        default:
            return "Unknown";
    }
}

void mqtt_demo(NetworkInterface *net)
{

    char* topic = "base/relay/led1";
    char* topiccommand = "base/relay/command";

```

```

TCPSocket socket;
MQTTClient client(&socket);

SocketAddress a;
char* hostname = "dev.rightech.io";
net->gethostbyname(hostname, &a);
int port = 1883;
a.set_port(port);

printf("Connecting to %s:%d\r\n", hostname, port);

socket.open(net);
printf("Opened socket\n\r");
int rc = socket.connect(a);
if (rc != 0)
    printf("rc from TCP connect is %d\r\n", rc);
printf("Connected socket\n\r");

MQTTPacket_connectData data = MQTTPacket_connectData_initializer;
data.MQTTVersion = 3;
data.clientID.cstring = "8f226dcea860406e92a6888b9ee74fb0";
data.username.cstring = "kikot";
data.password.cstring = "2000";
if ((rc = client.connect(data)) != 0)
    printf("rc from MQTT connect is %d\r\n", rc);

if ((rc = client.subscribe(topiccommand, MQTT::QOS2, messageArrived))
!= 0)
    printf("rc from MQTT subscribe is %d\r\n", rc);

```

```

MQTT::Message message;

OPT3001 sensor_opt(I2C_SDA, I2C_SCL);

char buf[100];

message.retained = false;
message.dup = false;
message.qos = MQTT::QOS1;

while(true){
    sprintf(buf, "%d", sensor_opt.readSensor());
    message.payload = (void*)buf;
    message.payloadlen = strlen(buf)+1;
    rc = client.publish(topic, message);
    client.yield();
    string ste = "ON", redword = "red", blueword="blue";
    if (buffer == ste || buffer == redword || buffer == blueword)
    {
        int luminocity = sensor_opt.readSensor();
        brightness += (K * (luminocity_sp - luminocity) / MAXLUM);
        brightness = (brightness < 0.0f)? 0.0f : (brightness > 1.0f)? 1.0f :
brightness;
        green=brightness;
        if (buffer == ste){
            blue=brightness;

```

```

        red=brightness;

    }
    if (buffer == redword){
        blue=0;
        red=brightness;

    }
    if (buffer == blueword){
        blue=brightness;
        red=0;

    }
}

else {
    blue=0;
    red=0;
};
}

if ((rc = client.unsubscribe(topic)) != 0)
    printf("rc from unsubscribe was %d\r\n", rc);

if ((rc = client.disconnect()) != 0)
    printf("rc from disconnect was %d\r\n", rc);

return;

```

```

}

int main()
{

    wifi = WiFiInterface::get_default_instance();
    if (!wifi) {
        printf("ERROR: No WiFiInterface found.\n");
        return -1;
    }

    printf("\nConnecting to %s...\n", MBED_CONF_APP_WIFI_SSID);
    int      ret      =      wifi->connect(MBED_CONF_APP_WIFI_SSID,
MBED_CONF_APP_WIFI_PASSWORD, NSAPI_SECURITY_WPA_WPA2);
    if (ret != 0) {
        printf("\nConnection error: %d\n", ret);
        return -1;
    }

    mqtt_demo(wifi);

}

```