#include <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

BlynkTimer timer;

#define DEBUG\_SW 1

// Pins of Fan Regulator Knob

#define F1 13

#define F2 12

#define F3 14

#define F4 27

// Pins of Switches ( S - Switches )

#define S1 35

#define S2 34

#define S3 39

#define S4 36

#define S5 32

#define S6 33

#define S7 25

#define S8 26

// Pins of Relay (A - Appliances Control)

#define A1 15

#define A2 2

#define A3 4

#define A4 5

#define A5 18

#define A6 19

#define A7 21

#define A8 3

// Pins of Relay (Fan Speed Control)

#define Speed1 1

#define Speed2 22

#define Speed4 23

// By default the mode is with\_internet

int MODE = 0;

// You should get Auth Token in the Blynk App.

// Go to the Project Settings (nut icon).

char auth[] = "AUTH\_TOKEN";

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = "SSID";

char pass[] = "PASS";

bool speed1\_flag = 1;

bool speed2\_flag = 1;

bool speed3\_flag = 1;

bool speed4\_flag = 1;

bool speed0\_flag = 1;

int switch\_ON\_Flag1\_previous\_I = 0;

int switch\_ON\_Flag2\_previous\_I = 0;

int switch\_ON\_Flag3\_previous\_I = 0;

int switch\_ON\_Flag4\_previous\_I = 0;

void setup()

{

// put your setup code here, to run once:

pinMode(F1, INPUT);

pinMode(F2, INPUT);

pinMode(F3, INPUT\_PULLUP);

pinMode(F4, INPUT);

pinMode(S1, INPUT);

pinMode(S2, INPUT);

pinMode(S3, INPUT);

pinMode(S4, INPUT);

pinMode(S5, INPUT);

pinMode(S6, INPUT);

pinMode(S7, INPUT);

pinMode(S8, INPUT);

pinMode(A1, OUTPUT);

pinMode(A2, OUTPUT);

pinMode(A3, OUTPUT);

pinMode(A4, OUTPUT);

pinMode(A5, OUTPUT);

pinMode(A6, OUTPUT);

pinMode(A7, OUTPUT);

pinMode(A8, OUTPUT);

pinMode(Speed1, OUTPUT);

pinMode(Speed2, OUTPUT);

pinMode(Speed4, OUTPUT);

Serial.begin(9600);

WiFi.begin(ssid, pass);

timer.setInterval(2000L, checkBlynk);

// check if connected to Blynk server every 2 seconds

Blynk.config(auth);//, ssid, pass);

}

BLYNK\_WRITE(V0)

{

int fan\_speed = param.asInt();

// assigning incoming value from pin V0 to a variable

if (fan\_speed == 0)

{

speed0();

}

if (fan\_speed == 1)

{

speed1();

}

if (fan\_speed == 2)

{

speed2();

}

if (fan\_speed == 3)

{

speed3();

}

if (fan\_speed == 4)

{

speed4();

}

}

**BLYNK\_WRITE(V1)**

{ // assigning incoming value from pin V1 to a variable

int pinValue = param.asInt();

digitalWrite(A1, pinValue); // process received value

}

**BLYNK\_WRITE(V2)**

{ // assigning incoming value from pin V2 to a variable

int pinValue = param.asInt();

digitalWrite(A2, pinValue); // process received value

}

**BLYNK\_WRITE(V3)**

{ // assigning incoming value from pin V3 to a variable

int pinValue = param.asInt();

digitalWrite(A3, pinValue); // process received value

}

**BLYNK\_WRITE(V4)**

{ // assigning incoming value from pin V4 to a variable

int pinValue = param.asInt();

digitalWrite(A4, pinValue); // process received value

}

**BLYNK\_WRITE(V5)**

{ // assigning incoming value from pin V5 to a variable

int pinValue = param.asInt();

digitalWrite(A5, pinValue); // process received value

}

**BLYNK\_WRITE(V6)**

{ // assigning incoming value from pin V6 to a variable

int pinValue = param.asInt();

digitalWrite(A6, pinValue); // process received value

}

**BLYNK\_WRITE(V7)**

{ // assigning incoming value from pin V7 to a variable

int pinValue = param.asInt();

digitalWrite(A7, pinValue); // process received value

}

**BLYNK\_WRITE(V8)**

{ // assigning incoming value from pin V8 to a variable

int pinValue = param.asInt();

digitalWrite(A8, pinValue); // process received value

}

----------------------------------------------------------------------------------------------------------------------------------------------------------------------

**void loop()**

{

if (WiFi.status() != WL\_CONNECTED)

{

if (DEBUG\_SW) Serial.println("Not Connected");

}

else

{

if (DEBUG\_SW) Serial.println(" Connected");

Blynk.run();

}

**//Check Blynk Connectivity every 3 seconds**

**void checkBlynk()**

{

bool isconnected = Blynk.connected();

if (isconnected == false)

{

MODE = 1;

}

if (isconnected == true)

{

MODE = 0;

}

}

**timer.run(); // Initiates SimpleTimer**

if (MODE == 0)

with\_internet();

else

without\_internet();

}

--------------------------------------------------------------------------------------------------------------------------------------------------------------------

**void with\_internet()**

{

// FOR FAN

if (digitalRead(F1) == HIGH && digitalRead(F2) == HIGH && digitalRead(F3) == HIGH && digitalRead(F4) == HIGH && speed0\_flag == 1)

{

speed0();

Blynk.virtualWrite(V0, 0);

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 0;

}

if (digitalRead(F1) == LOW && speed1\_flag == 1)

{

speed1();

Blynk.virtualWrite(V0, 1);

speed1\_flag = 0;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F2) == LOW && digitalRead(F3) == HIGH && speed2\_flag == 1)

{

speed2();

Blynk.virtualWrite(V0, 2);

speed1\_flag = 1;

speed2\_flag = 0;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F2) == LOW && digitalRead(F3) == LOW && speed3\_flag == 1)

{

speed3();

Blynk.virtualWrite(V0, 3);

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 0;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F4) == LOW && speed4\_flag == 1)

{

speed4();

Blynk.virtualWrite(V0, 4);

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 0;

speed0\_flag = 1;

}

**// FOR SWITCH**

if (digitalRead(S1) == LOW)

{

if (switch\_ON\_Flag1\_previous\_I == 0 )

{

digitalWrite(A1, HIGH);

if (DEBUG\_SW) Serial.println("Relay1- ON");

Blynk.virtualWrite(V1, 1);

switch\_ON\_Flag1\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch1 -ON");

}

if (digitalRead(S1) == HIGH )

{

if (switch\_ON\_Flag1\_previous\_I == 1)

{

digitalWrite(A1, LOW);

if (DEBUG\_SW) Serial.println("Relay1 OFF");

Blynk.virtualWrite(V1, 0);

switch\_ON\_Flag1\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch1 OFF");

}

if (digitalRead(S2) == LOW)

{

if (switch\_ON\_Flag2\_previous\_I == 0 )

{

digitalWrite(A2, HIGH);

if (DEBUG\_SW) Serial.println("Relay2- ON");

Blynk.virtualWrite(V2, 1);

switch\_ON\_Flag2\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch2 -ON");

}

if (digitalRead(S2) == HIGH )

{

if (switch\_ON\_Flag2\_previous\_I == 1)

{

digitalWrite(A2, LOW);

if (DEBUG\_SW) Serial.println("Relay2 OFF");

Blynk.virtualWrite(V2, 0);

switch\_ON\_Flag2\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch2 OFF");

//delay(200);

}

if (digitalRead(S3) == LOW)

{

if (switch\_ON\_Flag3\_previous\_I == 0 )

{

digitalWrite(A3, HIGH);

if (DEBUG\_SW) Serial.println("Relay3- ON");

Blynk.virtualWrite(V3, 1);

switch\_ON\_Flag3\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch3 -ON");

}

if (digitalRead(S3) == HIGH )

{

if (switch\_ON\_Flag3\_previous\_I == 1)

{

digitalWrite(A3, LOW);

if (DEBUG\_SW) Serial.println("Relay3 OFF");

Blynk.virtualWrite(V3, 0);

switch\_ON\_Flag3\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch3 OFF");

//delay(200);

}

if (digitalRead(S4) == LOW)

{

if (switch\_ON\_Flag4\_previous\_I == 0 )

{

digitalWrite(A4, HIGH);

if (DEBUG\_SW) Serial.println("Relay4- ON");

Blynk.virtualWrite(V4, 1);

switch\_ON\_Flag4\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch4 -ON");

}

if (digitalRead(S4) == HIGH )

{

if (switch\_ON\_Flag4\_previous\_I == 1)

{

digitalWrite(A4, LOW);

if (DEBUG\_SW) Serial.println("Relay4 OFF");

Blynk.virtualWrite(V4, 0);

switch\_ON\_Flag4\_previous\_I = 0;

}

if (DEBUG\_SW) Serial.println("Switch4 OFF");

//delay(200);

}

if (digitalRead(S5) == LOW)

{

if (switch\_ON\_Flag1\_previous\_I == 0 )

{

digitalWrite(A5, HIGH);

if (DEBUG\_SW) Serial.println("Relay1- ON");

Blynk.virtualWrite(V1, 1);

switch\_ON\_Flag1\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch1 -ON");

}

if (digitalRead(S5) == HIGH )

{

if (switch\_ON\_Flag1\_previous\_I == 1)

{

digitalWrite(A5, LOW);

if (DEBUG\_SW) Serial.println("Relay1 OFF");

Blynk.virtualWrite(V1, 0);

switch\_ON\_Flag1\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch1 OFF");

}

if (digitalRead(S6) == LOW)

{

if (switch\_ON\_Flag2\_previous\_I == 0 )

{

digitalWrite(A6, HIGH);

if (DEBUG\_SW) Serial.println("Relay2- ON");

Blynk.virtualWrite(V2, 1);

switch\_ON\_Flag2\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch2 -ON");

}

if (digitalRead(S6) == HIGH )

{

if (switch\_ON\_Flag2\_previous\_I == 1)

{

digitalWrite(A6, LOW);

if (DEBUG\_SW) Serial.println("Relay2 OFF");

Blynk.virtualWrite(V2, 0);

switch\_ON\_Flag2\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch2 OFF");

//delay(200);

}

if (digitalRead(S7) == LOW)

{

if (switch\_ON\_Flag3\_previous\_I == 0 )

{

digitalWrite(A7, HIGH);

if (DEBUG\_SW) Serial.println("Relay3- ON");

Blynk.virtualWrite(V3, 1);

switch\_ON\_Flag3\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch3 -ON");

}

if (digitalRead(S7) == HIGH )

{

if (switch\_ON\_Flag3\_previous\_I == 1)

{

digitalWrite(A7, LOW);

if (DEBUG\_SW) Serial.println("Relay3 OFF");

Blynk.virtualWrite(V3, 0);

switch\_ON\_Flag3\_previous\_I = 0;

}

if (DEBUG\_SW)Serial.println("Switch3 OFF");

//delay(200);

}

if (digitalRead(S8) == LOW)

{

if (switch\_ON\_Flag4\_previous\_I == 0 )

{

digitalWrite(A8, HIGH);

if (DEBUG\_SW) Serial.println("Relay4- ON");

Blynk.virtualWrite(V4, 1);

switch\_ON\_Flag4\_previous\_I = 1;

}

if (DEBUG\_SW) Serial.println("Switch4 -ON");

}

if (digitalRead(S8) == HIGH )

{

if (switch\_ON\_Flag4\_previous\_I == 1)

{

digitalWrite(A8, LOW);

if (DEBUG\_SW) Serial.println("Relay4 OFF");

Blynk.virtualWrite(V4, 0);

switch\_ON\_Flag4\_previous\_I = 0;

}

if (DEBUG\_SW) Serial.println("Switch4 OFF");

//delay(200);

}

}

**void without\_internet()**

{

// **FOR FAN**

if (digitalRead(F1) == HIGH && digitalRead(F2) == HIGH && digitalRead(F3) == HIGH && digitalRead(F4) == HIGH && speed0\_flag == 1)

{

speed0();

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 0;

}

if (digitalRead(F1) == LOW && speed1\_flag == 1)

{

speed1();

speed1\_flag = 0;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F2) == LOW && digitalRead(F3) == HIGH && speed2\_flag == 1)

{

speed2();

speed1\_flag = 1;

speed2\_flag = 0;

speed3\_flag = 1;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F2) == LOW && digitalRead(F3) == LOW && speed3\_flag == 1)

{

speed3();

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 0;

speed4\_flag = 1;

speed0\_flag = 1;

}

if (digitalRead(F4) == LOW && speed4\_flag == 1)

{

speed4();

speed1\_flag = 1;

speed2\_flag = 1;

speed3\_flag = 1;

speed4\_flag = 0;

speed0\_flag = 1;

}

**// FOR SWITCH**

digitalWrite(A1, !digitalRead(S1));

digitalWrite(A2, !digitalRead(S2));

digitalWrite(A3, !digitalRead(S3));

digitalWrite(A4, !digitalRead(S4));

digitalWrite(A5, !digitalRead(S5));

digitalWrite(A6, !digitalRead(S6));

digitalWrite(A7, !digitalRead(S7));

digitalWrite(A8, !digitalRead(S8));

}

**// Fan Speed Control**

void speed0()

{

//All Relays Off - Fan at speed 0

if (DEBUG\_SW)Serial.println("SPEED 0");

digitalWrite(Speed1, LOW);

digitalWrite(Speed2, LOW);

digitalWrite(Speed4, LOW);

}

void speed1()

{

//Speed1 Relay On - Fan at speed 1

if (DEBUG\_SW)Serial.println("SPEED 1");

digitalWrite(Speed1, LOW);

digitalWrite(Speed2, LOW);

digitalWrite(Speed4, LOW);

delay(1000);

digitalWrite(Speed1, HIGH);

}

void speed2()

{

//Speed2 Relay On - Fan at speed 2

if (DEBUG\_SW)Serial.println("SPEED 2");

digitalWrite(Speed1, LOW);

digitalWrite(Speed2, LOW);

digitalWrite(Speed4, LOW);

delay(1000);

digitalWrite(Speed2, HIGH);

}

void speed3()

{

//Speed1 & Speed2 Relays On - Fan at speed 3

if (DEBUG\_SW)Serial.println("SPEED 3");

digitalWrite(Speed1, LOW);

digitalWrite(Speed2, LOW);

digitalWrite(Speed4, LOW);

delay(1000);

digitalWrite(Speed1, HIGH);

digitalWrite(Speed2, HIGH);

}

void speed4()

{

//Speed4 Relay On - Fan at speed 4

if (DEBUG\_SW)Serial.println("SPEED 4");

digitalWrite(Speed1, LOW);

digitalWrite(Speed2, LOW);

digitalWrite(Speed4, LOW);

delay(1000);

digitalWrite(Speed4, HIGH);

}