Code Informatique Arduino

```
/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial
#include <CCS811.h>
CCS811 sensor;


#include <ESP8266_Lib.h>
#include <BlynkSimpleShieldEsp8266.h>

#define SensorPin A2
float sensorValue = 0;
// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "z3YKMcwuD7sOS4L14oMeu0tsw7rP0fbl";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "arduino";
char pass[] = "1234abcd";
int motor2pin1 = 4;
int motor2pin2 = 5;
float dataBlynk;
float dataBlynk2=1;
int switcher = 11;                // Connect Tilt sensor to Pin11
int buzzer=10;//set digital IO pin of the buzzer
byte sensorPin = 12;
int Time;

// Hardware Serial on Mega, Leonardo, Micro...
//#define EspSerial Serial1

// or Software Serial on Uno, Nano...
//#include <SoftwareSerial.h>
//SoftwareSerial EspSerial(2, 3); // RX, TX

// Your ESP8266 baud rate:
#define ESP8266_BAUD 115200

ESP8266 wifi(&Serial);
WidgetLCD lcd(V12);

BlynkTimer timer;
```

```
// This function sends Arduino's up time every second to Virtual Pin (5).
// In the app, Widget's reading frequency should be set to PUSH. This means
// that you define how often to send data to Blynk App.



void myTimerEvent()
{
  // You can send any value at any time.
  // Please don't send more that 10 values per second.


  // Water level code
  float sensorData1 = analogRead(A0);
  Blynk.virtualWrite(V5, sensorData1);


  //CO2 sensor code
  float sensorData2 = sensor.getCO2PPM();
  Blynk.virtualWrite(V6, sensorData2);
  Serial.print(sensor.getCO2PPM());
  //TVOC sensor code
  float sensorData4=sensor.getTVOCPPB();
  Blynk.virtualWrite(V7, sensorData4);



  // Temperature
   float sensorData3 = analogRead(A1);
   float senso=(sensorData3-26);
  Blynk.virtualWrite(V8, senso);



  // moisture content
   for (int i = 0; i <= 100; i++)
 {
   sensorValue = sensorValue + analogRead(SensorPin);
   delay(1);
 }
 sensorValue = sensorValue/100.0;
 Blynk.virtualWrite(V9,sensorValue);


 if (dataBlynk>1)
 {
```

```
 motorcontrol();
 }
 else
 {
  dataBlynk=dataBlynk;
 }
// Antivol
 Antivol();
 //Presence
 Presence();
}



void setup()
{
  // Debug console
  Serial.begin(9600);


  // Set ESP8266 baud rate
  Serial.begin(ESP8266_BAUD);
  delay(10);

  Blynk.begin(auth, wifi, ssid, pass);
  // You can also specify server:
 // Blynk.begin(auth, wifi, ssid, pass, "blynk-cloud.com", 80);
  //Blynk.begin(auth, wifi, ssid, pass, IPAddress(192,168,1,100), 8080);

  // Setup a function to be called every second
  timer.setInterval(1000L, myTimerEvent);


  //Motor
  pinMode(motor2pin1, OUTPUT);
  pinMode(motor2pin2, OUTPUT);
  // Anti vol
 pinMode(buzzer,OUTPUT);// set digital IO pin pattern, OUTPUT to be output
 pinMode(switcher, INPUT);       // Set digital pin 3 to input mod
 // Presence
 pinMode(sensorPin,INPUT);

 //CO2
 while(sensor.begin() != 0){
        Serial.println("failed to init chip, please check if the chip connection is fine");
   }
   /**
```

```
    * @brief Set measurement cycle
    * @param cycle:in typedef enum{
    *               eClosed,      //Idle (Measurements are disabled in this mode)
    *               eCycle_1s,    //Constant power mode, IAQ measurement every second
    *               eCycle_10s,   //Pulse heating mode IAQ measurement every 10
seconds
    *               eCycle_60s,   //Low power pulse heating mode IAQ measurement every
60 seconds
    *               eCycle_250ms  //Constant power mode, sensor measurement every
250ms
    *               }eCycle_t;
    */
   sensor.setMeasCycle(sensor.eCycle_250ms);
}



BLYNK_WRITE(V10){
  dataBlynk= param.asInt();
}

BLYNK_WRITE(V11){
  dataBlynk2= param.asInt();
}



void deelay(){

 delay(dataBlynk);
}




void motorcontrol(){

 if (dataBlynk > 1){
  digitalWrite(motor2pin1, HIGH);
  digitalWrite(motor2pin2, LOW);
  delay(5000);

  digitalWrite(motor2pin1,LOW);
  digitalWrite(motor2pin2,LOW);
  deelay();
  digitalWrite(motor2pin1, LOW);
  digitalWrite(motor2pin2, HIGH);
  delay(5000);

  digitalWrite(motor2pin1,LOW);
  digitalWrite(motor2pin2,LOW);
  delay(10000);
  }
  else {
    delay(50000);
  }
}


void notification (){
    Blynk.notify("Urgent!!La plante se fait voler !");
}






void Antivol(){
if(dataBlynk2!=0){
  Time=0;
   if(digitalRead(switcher)==HIGH) //Read sensor value
     {

      unsigned char i,j;//define variable
        while(Time<100)
          { for(i=0;i<80;i++)// output a frequency sound
          { digitalWrite(buzzer,HIGH);// sound
          delay(1);//delay1ms
          digitalWrite(buzzer,LOW);//not sound
          delay(1);//ms delay

          }
        for(i=0;i<100;i++)// output a frequency sound
          {
          digitalWrite(buzzer,HIGH);// sound
          digitalWrite(buzzer,LOW);//not sound
          delay(2);//2ms delay
          }
         Time=Time+1;
          }
          }
   else
     {
        digitalWrite(buzzer,LOW);//not sound
        delay(2);//2ms delay
     }
 }
 else
 {
  dataBlynk2=dataBlynk2;
 }
}

void Presence ()
{
byte state = digitalRead(sensorPin);
  if(state == 1)
  {

    lcd.clear();
    lcd.print(0,0,"Somebody is here!");
    delay(10000);
  }
  else if(state == 0)
  {
  lcd.clear();
  lcd.print(0,0,"No one is here!");
  }
  delay(500);

 }


void loop()
{
  Blynk.run();
  timer.run(); // Initiates BlynkTimer



}
```