```
/*************************************************************************
*********************************
   IoT DHT Temp/Hum Station using NodeMCU ESP-12 Develop Kit V1.0
    DHT connected to NodeMCU pin D3
    DHT Data on local OLED Display

    Developed by MJRovai 12 October 2017

*************************************************************************
************************************************/
/* station<defines*/
#include "stationDefines.h"
#include <Wire.h>

/* DHT22*/
#include "DHT.h"
DHT dht(DHTPIN, DHTTYPE);

/* TIMER */
#include <SimpleTimer.h>
SimpleTimer timer;

int data[] = {0, 0, 0, 0, 0 , 0, 0}; //empty array where to put the numbers
going to the master
int data2[] = {0};
void setup()
{

  Serial.begin(9600);
  delay(10);

  // Messages accueil
  Serial.println("ArduFarmBot 2");
  Serial.println(".... Starting Setup");
  Serial.println(" ");
  // PINMODES
  pinMode(LED_PIN, OUTPUT);
  pinMode(PUMP_PIN, OUTPUT);
  pinMode(luminosityPin, INPUT);
  pinMode(PUMP_ON_BUTTON, INPUT_PULLUP);
 pinMode(soilMoisterPin2, INPUT);
  dht.begin();
  digitalWrite(PUMP_PIN, LOW);
  digitalWrite(LED_PIN, LOW);
  digitalWrite (soilMoisterVcc, HIGH);
digitalWrite (soilMoisterVcc2, HIGH);

  //CONNEXION
```

```
  Wire.begin(8);                /* join i2c bus with address 8 */
  Wire.onRequest(requestEvent); /* register request event */
  Wire.onReceive(receiveEvent); /* register receive event */


  startTimers();
}
void loop()
{
  readLocalCmd();
  recupererCapteurs();
  checkpump();
  //displayData();

}


/*************************************************************
  Read local commands (Pump and Lamp buttons are normally "HIGH"):
*************************************************************/
void readLocalCmd()
{
  boolean digiValue = debounce(PUMP_ON_BUTTON);
  if (digiValue == 1 )
  {
    data[6]= 1;
    turnPumpOn();
  }
  else
  {
    data[6]= 0;
  }
}

/*****************************************************
  Receive Commands and act on actuators
*****************************************************/
void aplyCmd()
{
  if (pumpStatus == 1)
  {
    digitalWrite(PUMP_PIN, HIGH);
    digitalWrite(LED_PIN, HIGH);
    Serial.println("POMPE ALLUME");
    Serial.println("LED ALLUME");
    delay(6000);
  }
  else
```

```
  {
    digitalWrite(PUMP_PIN, LOW);
    digitalWrite(LED_PIN, LOW);
    Serial.println("POMPE ETEINTE");
    Serial.println("LED ETEINTE");

  }

}

void turnPumpOn()
{
  pumpStatus = 1;
   data[5]= 1;
  aplyCmd();

  pumpStatus = 0;
  aplyCmd();
}

void receiveEvent(int howMany)
{
   if(Wire.available()) {
    int c = Wire.read();
    data2[0]=c;/* receive byte as a character */
    Serial.print("------RECU NODEMCU-----");
    Serial.print(data2[0]);           /* print the character */
    Serial.print("-------");
  }
 Serial.println();             /* to newline */
}

// function that executes whenever data is requested from master
void requestEvent() {
  uint8_t Buffer[7];
  Buffer[0] = data[0];
  Buffer[1] = data[1];
  Buffer[2] = data[2];
  Buffer[3] = data[3];
  Buffer[4] = data[4];
  Buffer[5] = data[5];
  Buffer[6] = data[6];
  Wire.write(Buffer, 7);
}


void recupererCapteurs() {
  getDhtTemp();
```

```
  getDhtHumi();
  getSoilMoisterData();
  getSoilMoisterData2();
  getluminosity();
}
 void checkpump(void){
  if(data2[0]==1){
    Serial.println("Alooo");
    turnPumpOn();
}
else{
    data[5]= 0;
}
}
```