

Rapport

Entre les deux séances, j'ai pu rédiger à nouveau le code reliant le module de température DH22 et l'écran LCD 20x4. Ensuite lors de la séance vous m'avez donné deux modules Bluetooth HC05 HC06 déjà connectés entre eux, du coup j'ai essayé d'écrire leur programme respectif pour que le maître puisse communiquer avec l'esclave. Cependant lors de la séance j'ai rencontré une difficulté qui concernait la mauvaise compatibilité entre mon Mac et l'application Arduino. En mettant en marche mes deux programmes, j'aurais dû pouvoir ouvrir les deux moniteurs afin de voir si les deux Bluetooth communiquaient bien mais malheureusement un seul moniteur s'ouvrait. Heureusement, Tiska m'a gentiment prêté son ordinateur ce qui m'a permis de me rendre compte qu'il y avait une erreur quelque part dans mon programme car il ne fonctionnait pas correctement. Étant donné que c'était la fin de l'heure, je n'ai pas eu le temps de la corriger, je vais donc m'en occuper pendant les vacances. Et enfin je vais aussi essayer que les données enregistrées par le capteur de température s'affichent correctement sur l'écran LCD par l'intermédiaire du Bluetooth.

Programme de l'écran LCD 20x4 et du capteur de température DH22 :

```
#include <cactus_io_AM2302.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
#define AM2302_PIN 2
AM2302 dht(AM2302_PIN);
float t=0;
float h=0;

void setup() {
  // initialise le nombre de caractères et de lignes du LCD
  lcd.begin(20, 4);
  Serial.begin(9600);
  dht.begin();
}

void loop(){
  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  dht.readHumidity();
  h=dht.humidity;
  dht.readTemperature();
  t=dht.temperature_C;
  // Check if any reads failed and exit early (to try again).
  if (isnan(dht.humidity) || isnan(dht.temperature_C)) {
    Serial.println("AM2302 sensor read failure!");
    return;
  }

  //Affichages :
  // Ligne 1:
  lcd.setCursor(0, 0);
  // positionne le curseur à la colonne 1 et à la ligne 1
  lcd.print("Humidite: ");
  lcd.setCursor(10, 0);
  lcd.print(h);
  lcd.setCursor(15, 0);
  lcd.print("%");
  lcd.setCursor(0,2);
  lcd.print("mode :");
  lcd.setCursor(10,2);
  lcd.print("sport");

  // Ligne 2:
  lcd.setCursor(0, 1);
  // positionne le curseur à la colonne 1 et à la ligne 2
```

```

lcd.print("Temp: ");
lcd.setCursor(10, 1);
lcd.print(t);

Serial.print(h); Serial.print(" %\t\t");
Serial.print(dht.temperature_C); Serial.print(" *C\t");
Serial.print(dht.computeHeatIndex_C()); Serial.print(" *C\t");

// Wait a few seconds between measurements. The AM2302 should not be read at a higher frequency of
// about once every 2 seconds. So we add a 3 second delay to cover this.
delay(3000);
}

```

Programme du module Bluetooth HC06 :

```

#include<SoftwareSerial.h>
#define RX 3
#define TX 2
SoftwareSerial BlueT(RX,TX);
String answer;
String message;

void setup() {
  Serial.begin(9600);
  Serial.println("6");
  Serial.println("Bonjour -Prêt pour les commandes AT");
  BlueT.begin(9600);
}

void loop() {
  readSerialPort();

  //lire les réponses de l'esclave
  while (BlueT.available()){
    delay(10);
    if (BlueT.available()>0) {
      char c = BlueT.read();
      answer += c;
    }
  }

  //envoyer data vers esclave
  if(message!=""){
    Serial.print("Le maître envoie : ");
    Serial.println(message);
    BlueT.print(message);
    message="";
  }

  //Send answer to monitor
  if(answer!=""){
    Serial.print("L'esclave reçoit : ");
    Serial.println(answer);
    answer="";
  }
}

void readSerialPort(){
  while (Serial.available()) {
    delay(10);
    if (Serial.available() >0) {
      char c = Serial.read(); //gets one byte from serial buffer
      message += c; //makes the string readString
    }
  }
}

```

```
}  
}  
Serial.flush();  
}
```

Programme du module Bluetooth HC05 :

```
#include<SoftwareSerial.h>  
#define RX 10  
#define TX 11
```

```
String message;
```

```
SoftwareSerial BlueT(RX,TX);
```

```
void setup() {  
  Serial.begin(9600);  
  Serial.println("5");  
  BlueT.begin(9600);  
}
```

```
void loop() {  
  readSerialPort();  
  
  //envoyer une réponse au maître  
  if (message!=""){  
    Serial.print("Le maître envoie : ");  
    Serial.println(message);  
    BlueT.println(message);  
    message="";  
  }  
}
```

```
void readSerialPort(){  
  while (BlueT.available()){  
    delay(10);  
    if (BlueT.available()>0){  
      char c = BlueT.read();  
      message += c;  
    }  
  }  
  BlueT.flush();  
}
```