



Stage

Initiation à la programmation et à l'électronique avec ARDUINO

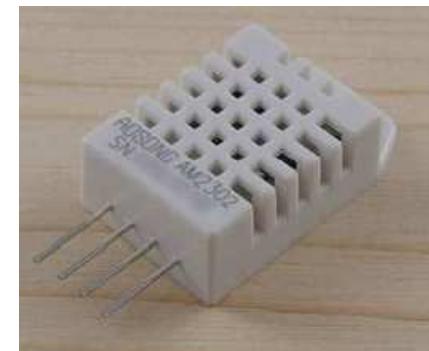
Jour 4



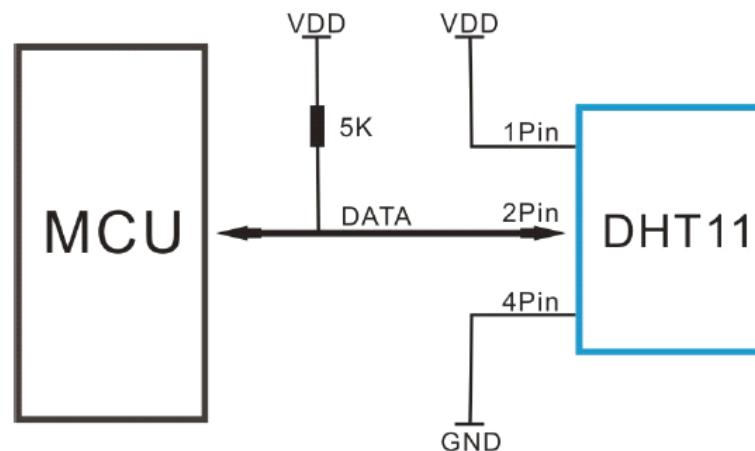
- Projet 7 : Construire un thermomètre hygromètre
 - Acquérir les données du capteur (mesure, conversion)
 - Les afficher sur la console IDE (via liaison série USB)
 - Mettre en œuvre l'afficheur LCD (2 lignes x 12 caractères)
 - Afficher les mesures sur l'afficheur LCD



- Objectif: utiliser un capteur température/humidité et afficher les données sur un écran LCD
- Le capteur utilisé sera le DHT11



- Câblage typique:

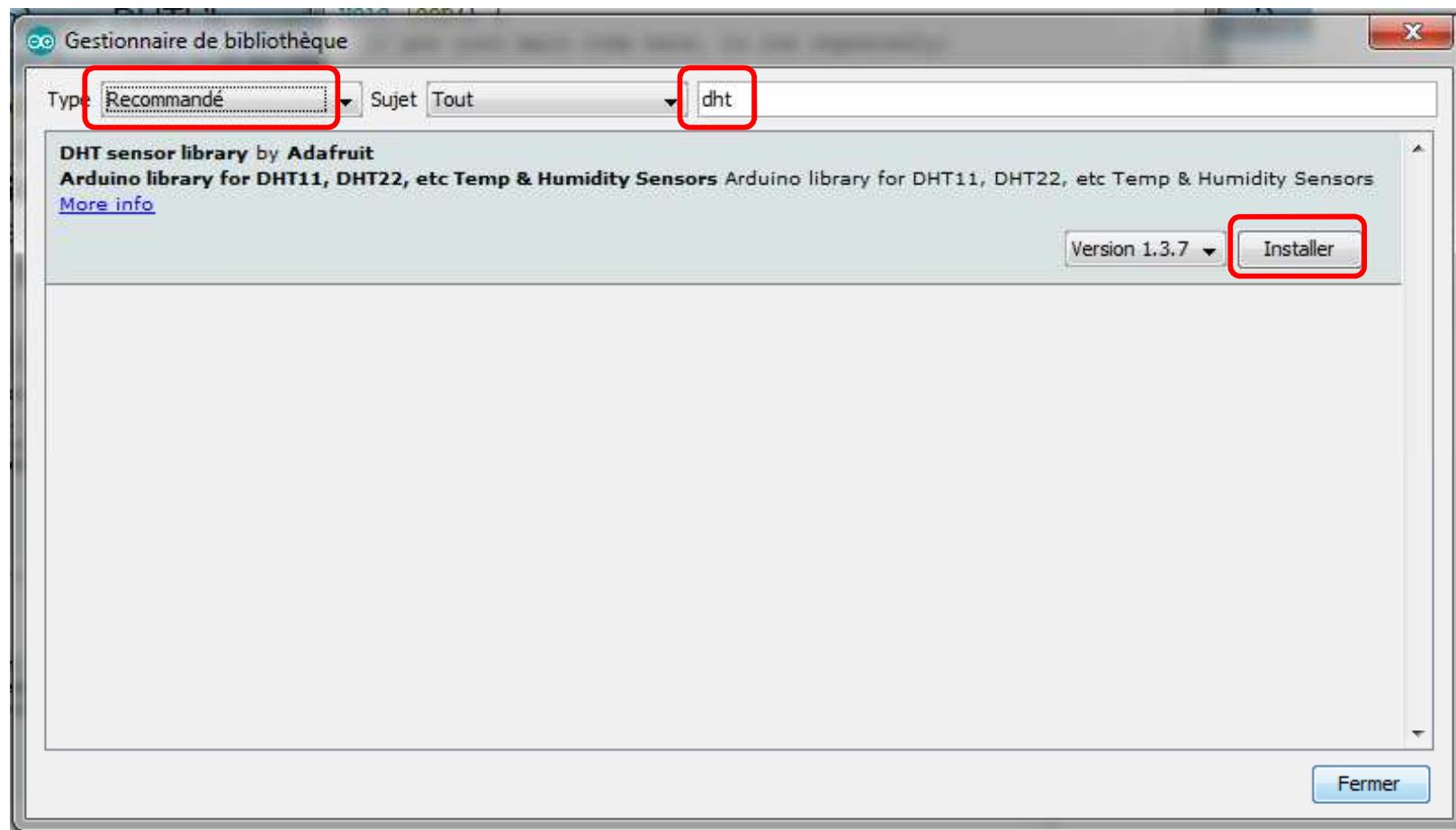




- Pour communiquer avec le capteur DHT11, il faut dans un premier temps télécharger et installer la bibliothèque spécifique à ce capteur
- Les bibliothèques sont des ensembles de fonctions qui s'ajoutent aux fonctions de base du logiciel Arduino.IDE.
- Certaines bibliothèques sont préinstallées, par exemple celle intitulée LiquidCrystal dont nous aurons besoin pour l'affichage sur l'écran LCD, d'autres doivent être installées par l'utilisateur au fur et à mesure de ses besoins.
- La bibliothèque DHT que nous allons utiliser pour ce projet fait partie de cette dernière catégorie et doit donc être installée « à la main ».

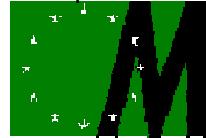


- Pour installer la bibliothèque DHT il suffit d'utiliser le gestionnaire de librairies de l'IDE Arduino comme mentionné ci-dessous en rouge:





- ATTENTION : Ce qui suit est valable si l'on utilise la librairie DHT Adafruit (celle ajoutée précédemment)
- Pour utiliser les fonctions de la librairie DHTLib il faut inclure cette librairie dans le code grâce à l'instruction :
 - **#include <DHT.h>**
- Il faut ensuite créer un objet de type DHT en utilisant l'instruction:
 - **DTH objet_dht(Arduino_pin, type_capteur); // Type = DHT11 ou DHT22**
- La lecture des données du capteur se fera par les instructions :
 - **objet_dht.readTemperature();**
 - **objet_dht.readHumidity();**



- Rappel des fonctions liaison série:

- **Serial.begin(debit);** : Fixe le débit de communication en nombre de caractères par seconde pour la communication série (300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, or 115200).
- **Serial.print(val, format);** : Affiche les données sous le port série sous forme lisible pour les humains (texte ASCII) sans retour à la ligne.
- **Serial.println(val, format);** : Affiche les données sous le port série sous forme lisible pour les humains (texte ASCII) avec retour à la ligne.



- Données d'entrée :
 - Capteur DHT 11
 - Ligne de data du capteur (pin 2) connectée à la pin 7 de l'Arduino
 - Tempo : 1 s

- Travail à effectuer:
 - Inclure la librairie DHT dans le code
 - Créer un objet de type DHT
 - Lires les données du capteur
 - Afficher les données sur la console série



- Code du projet

```
1 // Ajout de la librairie DHT
2 #include <DHT.h>
3
4 // Definition de la pin data sur Arduino
5 #define DHTPIN 7
6
7 // Definition du type de capteur
8 #define DHTTYPE DHT11
9
10 // Creation d'un objet DHT11
11 DHT dht(DHTPIN, DHTTYPE);
12
13 void setup() {
14     // Initialisation du capteur
15     dht.begin();
16
17     //Initialisation de la liaison serie
18     Serial.begin(9600);
19 }
20
21 void loop() {
22     // Affichage sur la console serie
23     Serial.print("Temperature = ");
24     Serial.println(dht.readTemperature());
25     Serial.print(" Humidite = ");
26     Serial.println(dht.readHumidity());
27
28     delay(1000); // pause de 1 seconde
29 }
```



- Affichage des valeurs mesurées sur un écran LCD
 - Nous allons utiliser pour cela un afficheur LCD comportant 2 lignes de 16 caractères.



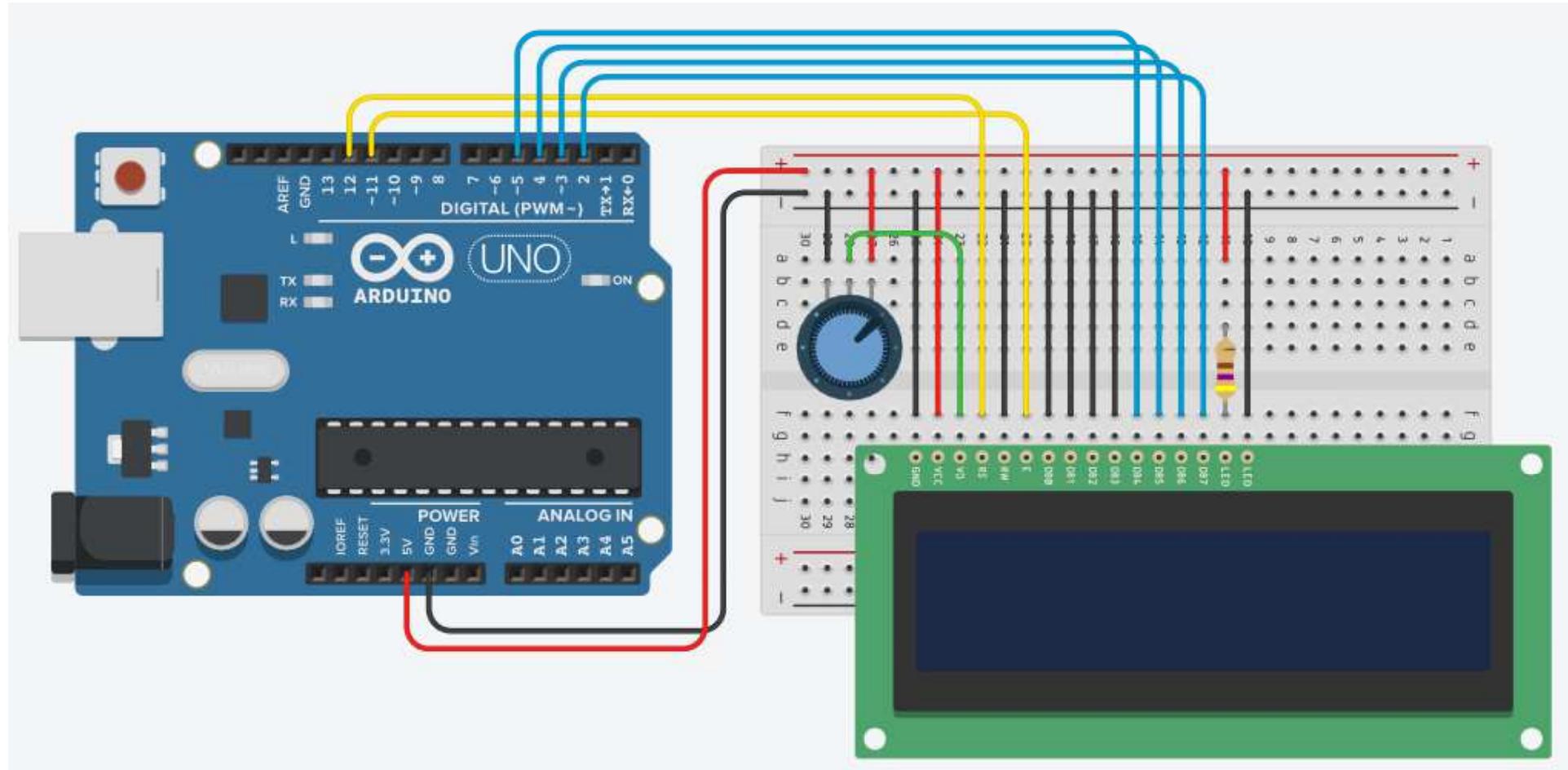


- Données d'entrée :
 - Capteur DHT 11
 - Ligne de data du capteur (pin 2) connectée à la pin 7 de l'Arduino
 - LCD connecté suivant le schéma (voir diapositive suivante)
 - Tempo : 1 s

- Travail à effectuer:
 - Inclure les librairies DHT et LiquidCristal dans le code
 - Créer un objet de type DHT
 - Créer un objet de type LCD
 - Lires les données du capteur DHT
 - Afficher les données sur le LCD



- Le câblage du LCD:





- Pour pouvoir afficher des données sur le LCD il faut d'abord inclure dans le code la librairie adéquate
 - **#include <LiquidCrystal.h>**
- Il faut ensuite créer l'objet LCD en définissant les pin Arduino utilisées :
 - **LiquidCrystal lcd(12, 11, 5, 4, 3, 2); //RS E D4, D5, D6, D7**
- Il faudra ensuite dans le setup définir la structure de l'afficheur :
 - **lcd.begin(16, 2); //nb caractères, nb lignes**
- Instruction pour positionner le curseur d'écriture
 - **lcd.setCursor(0,0); // colonne, ligne**
- Instruction pour écrire
 - **lcd.print("Bonjour");**



Projet 7 : Thermomètre hygromètre



- Code complet du projet

```
1 // Ajout de la librairie DHT
2 #include <DHT.h>
3
4 // Ajout de la bibliothèque LCD
5 #include <LiquidCrystal.h>
6
7 // Définition de la pin data sur Arduino
8 #define DHTPIN 7
9
10 // Définition du type de capteur
11 #define DHTTYPE DHT11
12
13 // Création d'un objet DHT11
14 DHT dht(DHTPIN, DHTTYPE);
15
16 //Création de l'objet LCD et on définit les Pins utilisés
17 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
18
19 void setup() {
20     // Initialisation du capteur
21     dht.begin();
22
23     //on initialise l'écran LCD
24     lcd.begin(16, 2);
25 }
```

```
27 void loop() {
28     // Affichage sur le LCD
29     lcd.clear(); // on efface l'affichage précédent
30     lcd.setCursor(0,0); //on place le curseur en haut à gauche
31     lcd.print("Temp = ");
32     lcd.print(dht.readTemperature(), 1); // 1 décimale
33     lcd.print((char)0xB2);
34     lcd.print("C");
35     lcd.setCursor(0,1); //on place le curseur sur la 2eme ligne
36     lcd.print("Humi = ");
37     lcd.print(dht.readHumidity(), 0); // pas de décimale
38     lcd.print("%");
39
40     delay(1000); // pause de 1 seconde
41 }
```



Merci pour votre attention

Avez-vous des questions ?