



# Capteurs - Humidité

[Mise à jour le 31/1/2020]

## 1. Humidité de l'air



### 1.1 Généralités

- **Ressource** à consulter sur **Wikiversité** : [Détection et mesure d'éléments ou de molécules](#)

L'humidité relative de l'air, ou degré hygrométrique, couramment noté  $\phi$ , correspond au rapport de la pression partielle de la vapeur d'eau contenue dans l'air sur la pression de vapeur saturante (ou tension de vapeur) à la même température. Elle est donc une mesure du rapport entre le contenu en vapeur d'eau de l'air et sa capacité maximale à en contenir dans ces conditions. Ce rapport changera si on change la température ou la pression bien que l'humidité absolue de l'air n'ait pas changé. Elle est mesurée à l'aide d'un hygromètre. [Wikipédia](#)

### 1.2 HIH4030



- **Source** : [wiki](#)

Ce module, basé sur le capteur HIH4030, mesure l'humidité relative et délivre une tension en sortie qui peut être raccordée sur une entrée analogique d'un microcontrôleur.

- **Distributeur** : [Gotronic](#)
- **Caractéristiques**

- Alimentation: 5 Vcc (4 à 5,8 Vcc)
- Consommation: 200  $\mu$ A
- Plage de mesure: 0 à 100% HR
- Précision:  $\pm 3,5\%$  HR
- Temps de réponse: 5 s
- Sortie analogique linéaire
- Température de service:  $-40^{\circ}\text{C}$  à  $+85^{\circ}\text{C}$
- Dimensions: 19 x 7,6 mm

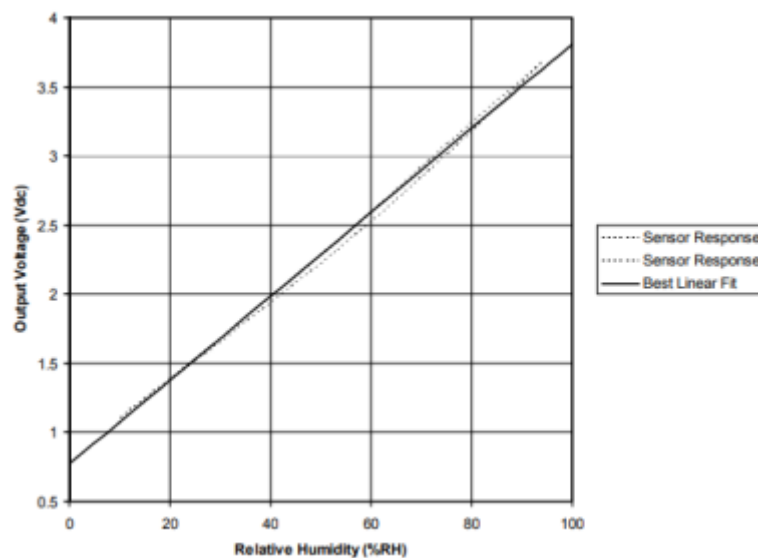


### • Documentation

- Fichier Acrobat Reader à télécharger [ici](#)

### • Modèles

- **Sans correction** de l'influence de la température :  $V_{OUT} = (V_{SUPPLY})(0.0062(\text{sensor RH}) + 0.16)$ , typical at  $25^{\circ}\text{C}$
- **Avec correction** :  $RH = (\text{Sensor RH}) / (1.0546 - 0.00216T)$ , T in  $^{\circ}\text{C}$



### • Aide pour la *simulation de la chaîne de mesure*

- Les équations de la chaîne de mesure sont téléchargeables [ici](#)
- Le modèle à simuler est téléchargeable [ici](#)



### • Programmation d'une carte Arduino Uno R3

- Bibliothèques à télécharger [ici](#) et à installer dans l'IDE.
  - Installation : dézipper le répertoire à télécharger sur [Github](#) et le copier dans `C:\Documents\Arduino\libraries`.
- Connexion à un shield [Tinkerkit v2](#) monté sur une Arduino Uno.



- Traitement à réaliser :  $RH=f(N)$
- Exemple 1 (sans correction de l'influence de la température ambiante)

### HIH4030.cpp

```
// Library Inclusions
#include <SparkFun_HIH4030.h>

// Analog IO Pin Connected to OUT
#define HIH4030_OUT A0

// Supply Voltage - Typically 5 V
#define HIH4030_SUPPLY 5

// Library Variables
HIH4030 sensorSpecs(HIH4030_OUT, HIH4030_SUPPLY);

void setup(void){
    Serial.begin(9600);           // Serial Port
}

void loop(void){
    Serial.print("Sensor Voltage = ");
    Serial.print(sensorSpecs.vout());
    Serial.println(" V");
    Serial.print("Relative Humidity = ");
    Serial.print(sensorSpecs.getSensorRH());
    Serial.println(" %");
}
```

```
Serial.println(""); // Return Space Between Readings  
delay(5000); // Slow Down Serial Output  
}
```

- *Exemple 2 : mesure de RH avec correction de l'influence de la température ambiante*

Arduino Examples → Examples from Custom Libraries → SparkFun\_HIH4030\_Library →  
HIH4030\_HumiditySensor\_Example.ino



Le répertoire du projet de l'exemple 1 pour l'IDE **VSCode** est téléchargeable [ici](#)



- **Programmation d'une carte FEZ avec l'IDE Visual Studio Community**

A venir

## 2. Humidité de la terre



### 2.1 Généralités

Il est pertinent de connaître la nature du sol avant de se lancer dans la plantation : sol drainé, sol humide ou sol sec.

Les différents végétaux n'ont pas les mêmes besoins en eau. Certains préfèrent un sol humide tandis que d'autres évoluent plus harmonieusement dans un sol riche en sable. [ooreka](#)

### 2.2 Moisture Sensor



- **Source** : [wiki](#)

Ce module capteur d'humidité compatible Grove permet de connaître la concentration d'eau dans la terre par exemple. Le capteur délivre une valeur analogique en fonction de la teneur en eau. Applications: contrôle d'humidité pour plante, détecteur d'eau etc.

- **Distributeur** : [Gotronic](#)

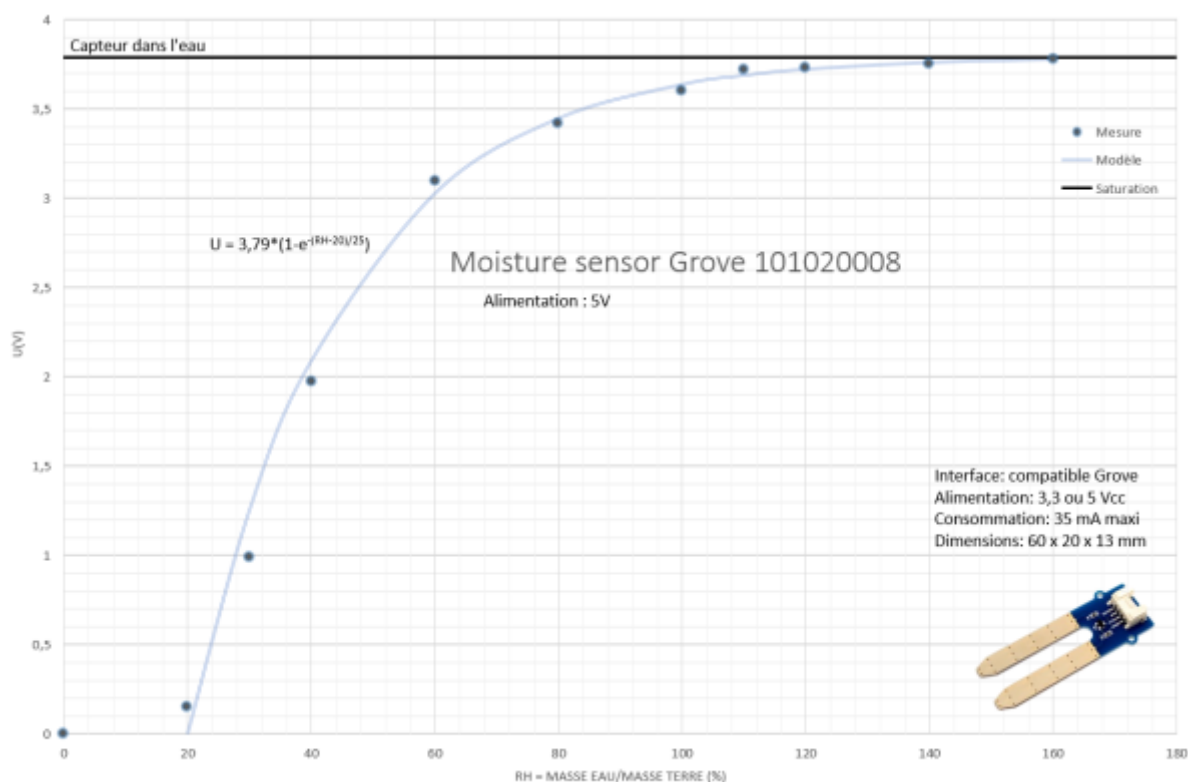
- **Caractéristiques**

- Alimentation: 3,3 ou 5 Vcc
- Consommation: 35 mA maxi
- Dimensions: 60 x 20 x 13 mm



- **Modèle**

- Fichier Excel à télécharger [ici](#)



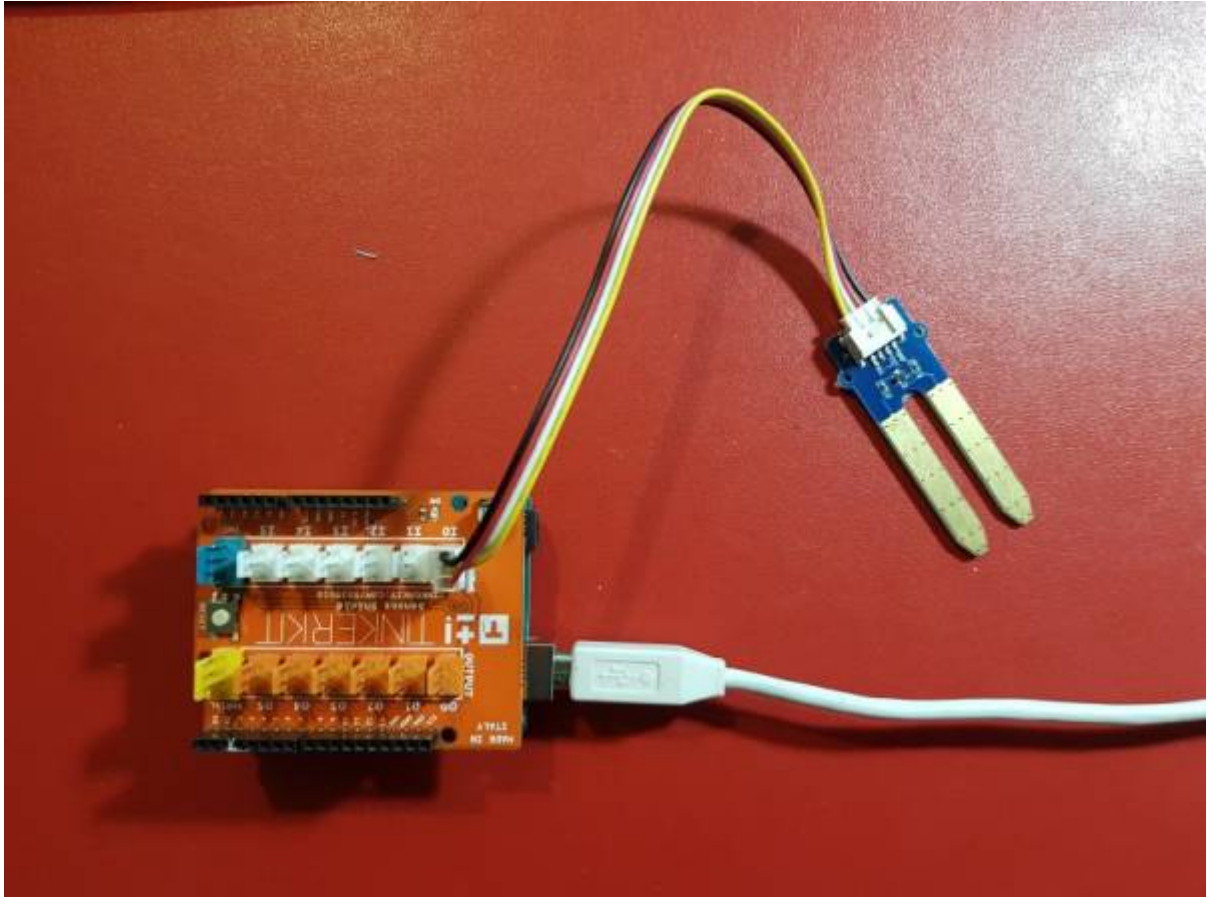
- **Aide pour la *simulation de la chaîne de mesure***


- Les équations de la chaîne de mesure sont téléchargeables [ici](#)
- Le modèle à simuler est téléchargeable [ici](#)



- **Programmation d'une carte Arduino Uno R3**

- Bibliothèques à installer dans l'IDE : aucune
- Connexion à un shield [Tinkerkit v2](#).



- Traitement à réaliser :  $RH=f(N)$
- Un premier exemple 

[moisture.cpp](#)

```
// -----  
// Capteur : humidité terre  
// Arduino : Uno  
// Date : 3/2019  
// Fichier : moisture.ino  
// Auteur : Philippe Mariano  
// -----  
#include <math.h>  
#include <stdio.h>  
  
// Variables  
// remarque : float codé sur 4 octets  
float k = 776;
```

```
int N = 0; // image de l'humidité

void setup()
{
    Serial.begin(9600);
}

void loop()
{
    // Lire
    N = analogRead(A0);
    // Traiter
    float RH = 20 - 25 * log(1 - N / k); // RH=f(N)
    // Ecrire
    Serial.print("RH");
    Serial.print(N);
    Serial.print("=");
    Serial.print(RH);
    Serial.print("\n");
    delay(2000);
}
```



Le projet pour l'IDE **VSCode** de l'exemple ci-dessus est téléchargeable [ici](#)

From:  
<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:  
<http://webge.fr/dokuwiki/doku.php?id=materiels:capteurs:humidite:humidite&rev=1656579306>

Last update: **2022/06/30 10:55**

