

MySensors

Documentation du projet

Club de Robotique et d'Electronique
Programmable de Ploemeur

3 septembre 2021
55 pages

- ▶ Document réalisé en L^AT_EX par Nicolas LE GUERROUÉ pour le Club de Robotique et d'Electronique Programmable de Ploemeur (CREPP).
Des remerciements à Régis PIDOUX, Guy DRAN, Florian MAUSY, Patrick ZÉO et Marcel RODRIGUEZ pour leur contribution à l'élaboration du contenu.
- ▶ Permission vous est donnée de copier, distribuer et/ou modifier ce document sous quelque forme et de quelque manière que ce soit.
- ▶ Version du 3 septembre 2021
- ▶ Taille de police : 11pt
- ☎ 06.20.88.75.12
- ✉ nicolasleguerroue@gmail.com
- ▶ **Dans la mesure du possible, évitez d'imprimer ce document si ce n'est pas nécessaire. Il est optimisé pour une visualisation sur un ordinateur et contient beaucoup d'images.**

I	Préparation du projet	4
1	Introduction	5
1.1	Présentation	5
1.2	Structure du projet	6
2	Programmation	7
2.1	Installation des bibliothèques	7
3	Programmation	10
3.1	Programmation de la Pro-Mini	10
3.2	Liste du matériel	10
3.3	Branchements	12
3.4	Téléversement	14
3.5	Programmation de la Nano	16
II	Montage des circuits imprimés	17
4	La passerelle	18
4.1	Rappels	18
4.2	Liste du matériel de la passerelle	18
4.3	Placement des composants	18
4.4	Mise en place du régulateur de tension	21
4.5	Rendus	22
5	La sonde	24
5.1	Rappels	24
5.2	Liste du matériel de la sonde	24
5.3	Placement des composants	25
5.4	Rendus	27
6	Vérification	28
6.1	Les court-circuit	28
6.2	Les sondes NRF24	29
III	Configuration des programmes Arduino	31
7	La sonde	32
8	Branchements du DHT	36

9 La passerelle	37
9.1 Envoi des programmes	38
 IV Configuration de Domoticz	 39
10 La passerelle	40
10.1 Ajout de la passerelle	40
10.2 Recherche des capteurs	41
10.3 Visualisation des données	43
 V Annexes	 45
A Installation	46
A.1 Installation de Domoticz sur Linux	46
B Pont diviseur	51
B.1 Démonstration	51
C Schéma passerelle	53
D Schéma sonde	54
E Questions	55

Première partie

Préparation du projet

Présentation

Ce document a pour but d'expliquer la mise en place d'une passerelle et d'une sonde MySensors.

Organigramme

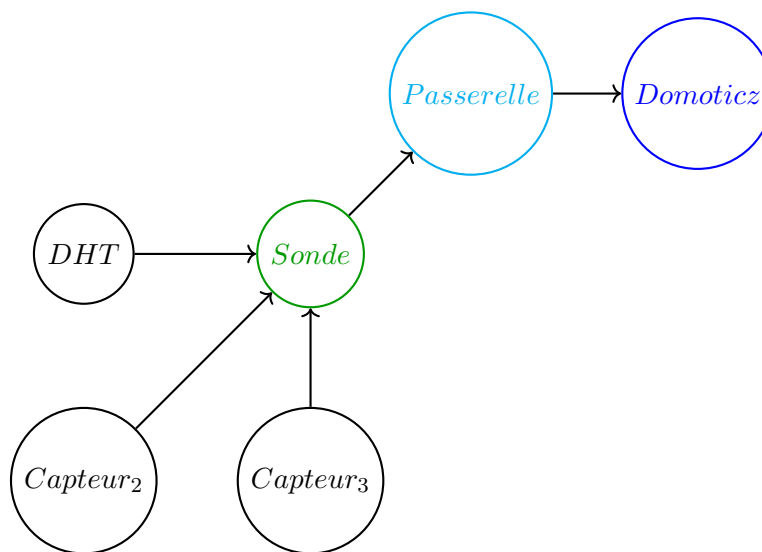


FIGURE 1.1 – Les différents composants du projet

Principe

Les capteurs vont être analysés par la sonde MySensors. Cette dernière enverra à distance les informations vers la passerelle qui se chargera d'envoyer les informations au serveur Domoticz via une liaison USB.

Une sonde représente un endroit physique, un lieu de mesure. Si vous souhaitez par la suite faire d'autres relevés dans un endroit différent, il suffira d'ajouter une sonde et de garder la passerelle. Chaque sonde est caractérisée par un identifiant de noeud (NODE_ID) et chaque capteur possède un identifiant enfant sur la sonde qui lui est rattachée (CHILD_ID)

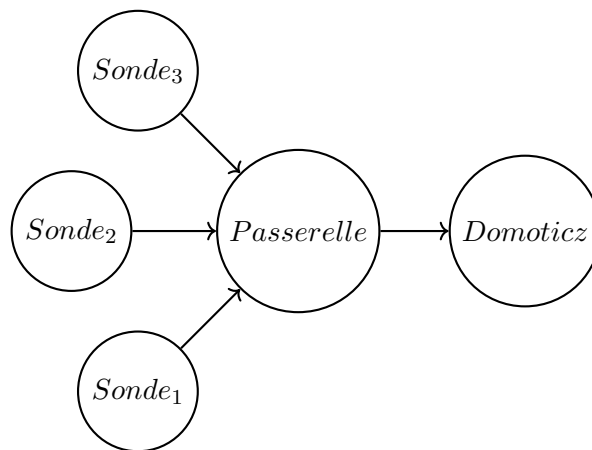


FIGURE 1.2 – Une extension possible

Structure du projet

Nous vous invitons à garder la structure suivante pour le projet :

Dans un dossier **DIR** Domoticz_Crepp, placez deux dossiers appelés **DIR** Sonde_MySensors et **DIR** Passerelle_MySensors

Ces deux derniers dossiers contiendront respectivement le programme de la sonde et de la passerelle.

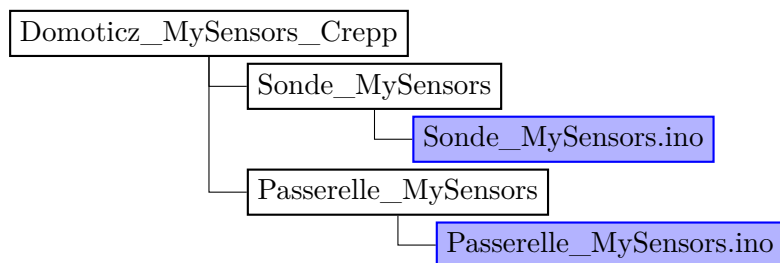


FIGURE 1.3 – Arborescence du projet

Occupons nous maintenant des bibliothèques Arduino.

Installation des bibliothèques

Lors de la première compilation d'un programme, il se peut que des bibliothèques soient manquantes. C'est ce que nous allons voir. Pour cela, ouvrez le programme de la sonde (**Sonde_MySensors.ino**) sans brancher de carte Arduino.

Ensuite, cliquez sur le bouton **Vérifier** (bouton de gauche) et patientez quelques secondes.

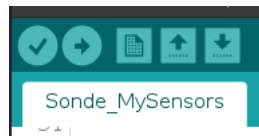


FIGURE 2.1 – Bouton de vérification

Si la bibliothèque MySensors est manquante, vous obtiendrez l'erreur suivante :

```
82 #include <MyConfig.h>
83 #include <MySensors.h>
84 #include <SPI.h>
85 #include <DHT.h>
86
```

```
MyConfig.h: No such file or directory
candidates: {}
Sonde_MySensors:82:10: fatal error: MyConfig.h: No such file or directory
  #include <MyConfig.h>
           ^~~~~~
compilation terminated.
exit status 1
MyConfig.h: No such file or directory
```

FIGURE 2.2 – La bibliothèque MySensors manquante

Pour installer la bibliothèque, il suffit d'aller dans **Croquis > Inclure une bibliothèque > Ajouter la bibliothèque .ZIP**

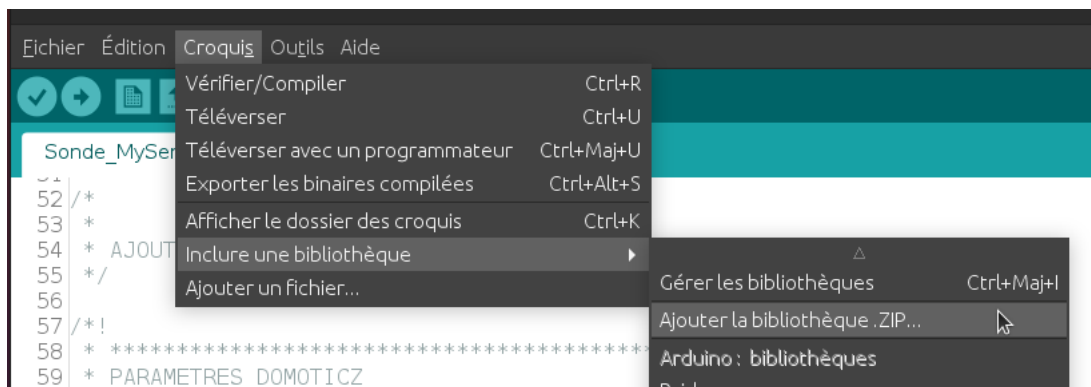


FIGURE 2.3 – Ajout d’une bibliothèque

Il ne reste qu’à trouver le fichier **Bibliothèque_MySensors.zip** et à faire **OK**

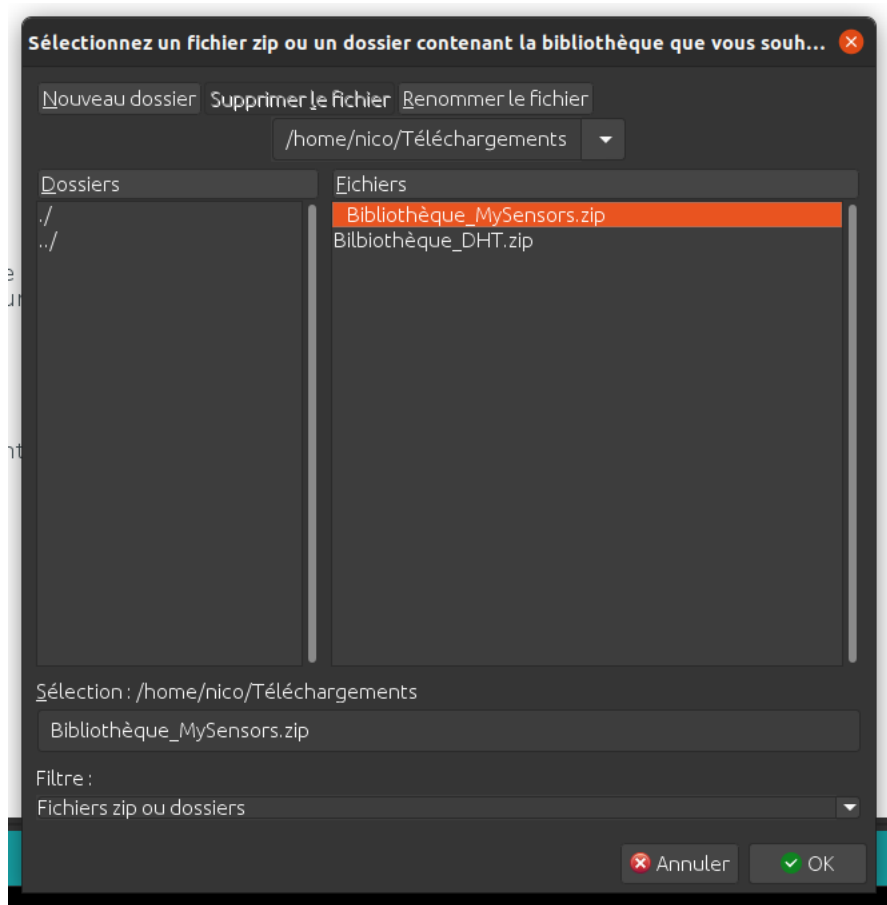


FIGURE 2.4 – Sélection du fichier ZIP

Un message de confirmation d’ajout est affichée en bas de la page du logiciel Arduino.

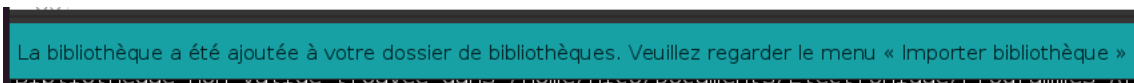


FIGURE 2.5 – La bibliothèque MySensors est ajoutée

On clique à nouveau sur le bouton **Vérifier** pour afficher les éventuelles erreurs.

Important

Dans certains cas, la bibliothèque **Adafruit_sensor** est manquante, il faut installer le fichier **Bibliothèque_Adafruit_sensor.zip** disponible en annexe.

On refait ensuite le bouton **Vérifier** et si la bibliothèque **DHT** n'est pas installé, on obtient de nouveau :

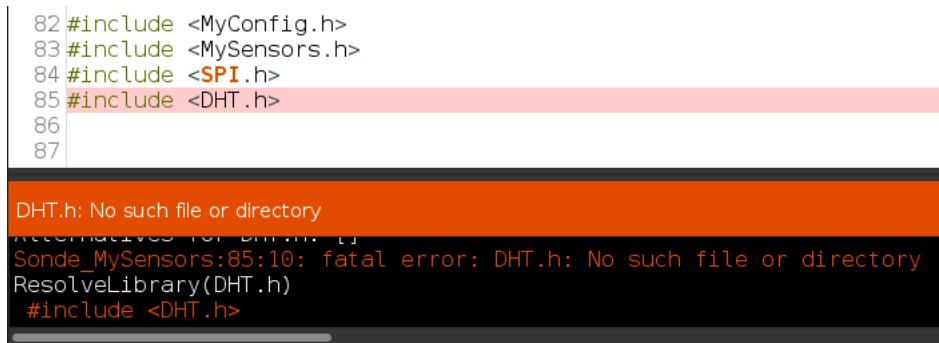


FIGURE 2.6 – La bibliothèque DHT manquante

On procède de la même façon, on va importer le fichier **Bibliothèque_DHT.zip** dans **Croquis > Inclure une bibliothèque > Ajouter la bibliothèque .ZIP**

Une fois toutes les bibliothèques installées, le message suivant apparaît :

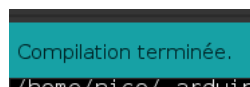


FIGURE 2.7 – La compilation est terminée

On va ensuite s'occuper de la vérification de la communication entre les cartes Arduino et l'ordinateur.

Programmation de la Pro-Mini

Programmer une carte Arduino pro-mini avec une carte Arduino évite d'acheter un module FTDI.

De plus, la carte Arduino Uno pourra être réutilisée pour d'autres projets.

L'objectif est de programmer la carte Pro-mini sur la sonde MySensors.

Liste du matériel

- 5 câbles Dupont mâles-femelles ¹



FIGURE 3.1 – Les câbles de connexion

- Une carte Arduino Pro-Mini



FIGURE 3.2 – La carte Arduino Pro-mini

1. Il est possible de faire des liaisons mâles-femelles avec des câbles mâles-mâles et femelles-femelles

► Une carte Arduino Uno

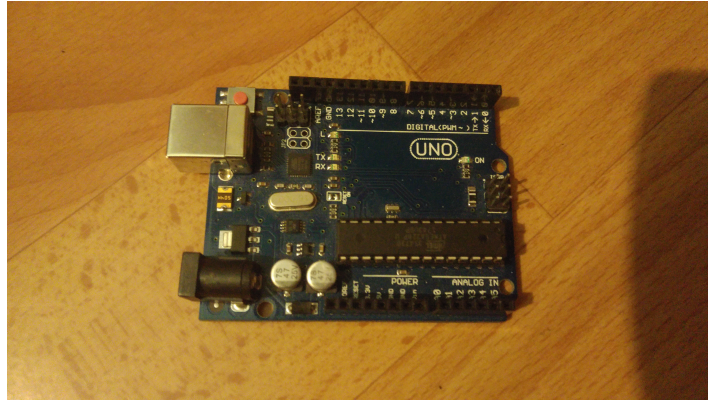


FIGURE 3.3 – La carte Arduino Uno

Important

Il faut retirer le microcontrôleur de la carte Arduino Uno pour pouvoir programmer la carte Pro-Mini

Pour le retirer, on prend un petit tournevis et on soulève délicatement la puce. On prendra le soin de repérer l'orientation de la puce sur la carte (méplat vers l'extérieur de la carte)

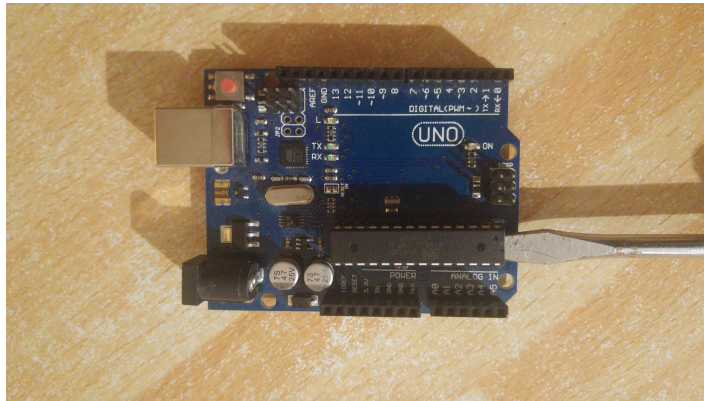


FIGURE 3.4 – On retire le microcontrôleur

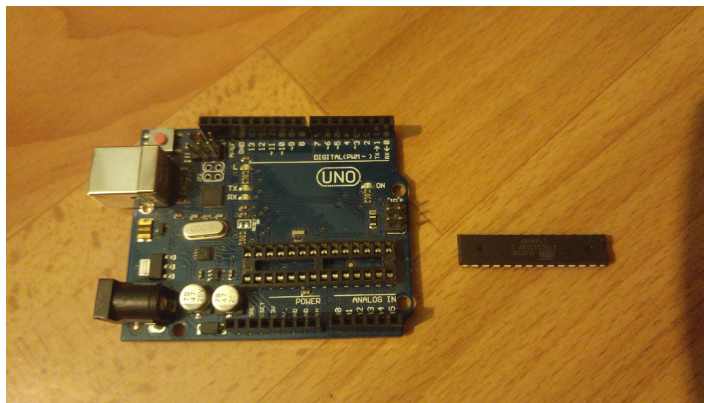


FIGURE 3.5 – La carte Arduino Uno sans son microcontrôleur

Branchements

Attention

La carte Arduino Pro-Mini doit être alimentée en **3.3V** et non en 5V !

La carte Arduino Pro-Mini ne doit pas être placée sur son support de sonde lorsque elle est en train d'être programmée !

Voici les connexions à faire pour programmer la Pro-Mini :

Le mot **PIN** `XXXX_UNO` représente une broche de la carte Arduino UNO et **PIN** `XXXX_PRO-MINI` représente une broche de la carte Arduino Pro-Mini. **XXXX** est l'indication du nom de la broche.

- ▶ **PIN** `RESET_UNO` vers **PIN** `RST_PRO-MINI`
- ▶ **PIN** `+3.3V_UNO` vers **PIN** `VCC_PRO-MINI`
- ▶ **PIN** `GND_UNO` vers **PIN** `GND_PRO-MINI`
- ▶ **PIN** `RX_UNO` vers **PIN** `RX_PRO-MINI`
- ▶ **PIN** `TX_UNO` vers **PIN** `TX_PRO-MINI`

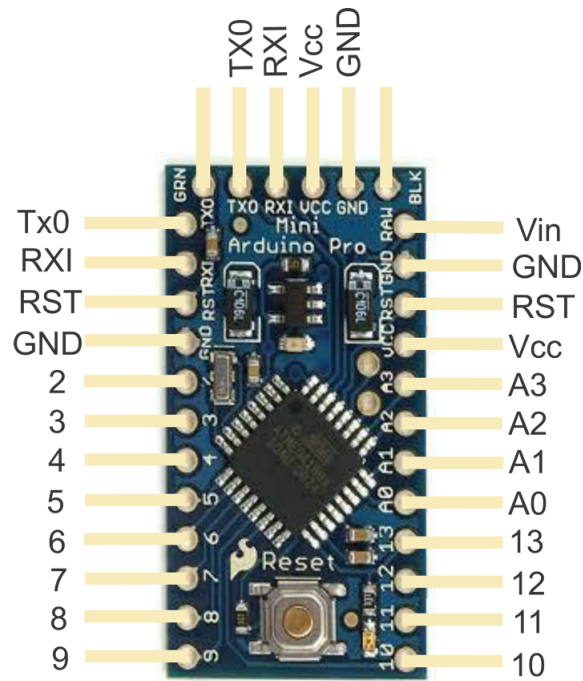


FIGURE 3.6 – Les broches du Pro-Mini

Remarque

Ici, la liaison série (**RX** et **TX**) n'est pas croisée, le **RX** de la carte Uno va sur le **RX** de la Pro-Mini, idem pour le TX

Vous pouvez ouvrir le programme Arduino que vous désirez charger² sur la carte Arduino Pro-Mini. Voici un programme minimal pour faire clignoter la LED du pro-mini. Ce programme est disponible en allant, dans le logiciel Arduino, dans la section **Fichiers** >

2. Vous pouvez charger le programme de clignotement de la LED pour l'exemple

Exemples > basics > Blink

```
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage  
    level)  
  delay(1000);                       // wait for a second  
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the  
    voltage LOW  
  delay(1000);                       // wait for a second  
}
```

Programme d'exemple Blink

Une fois le programme ouvert, voici les étapes pour compiler le programme.

Téléversement

- 1) Sélectionner la carte **Arduino Pro-mini** dans **Outils > Types de carte**

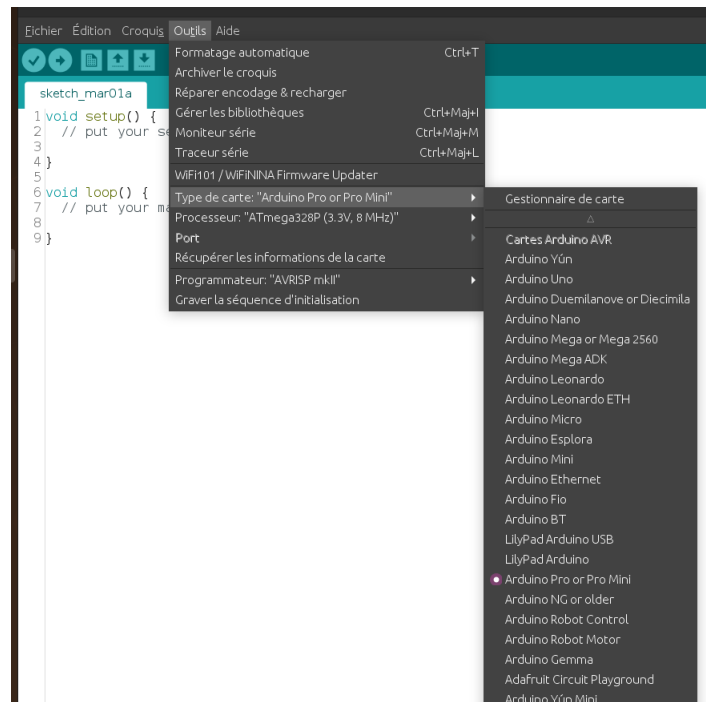


FIGURE 3.7 – Type de carte

- 2) Sélectionne le processeur **ATmega328P, 3.3V, 8Mhz** dans **Outils > Processeur**

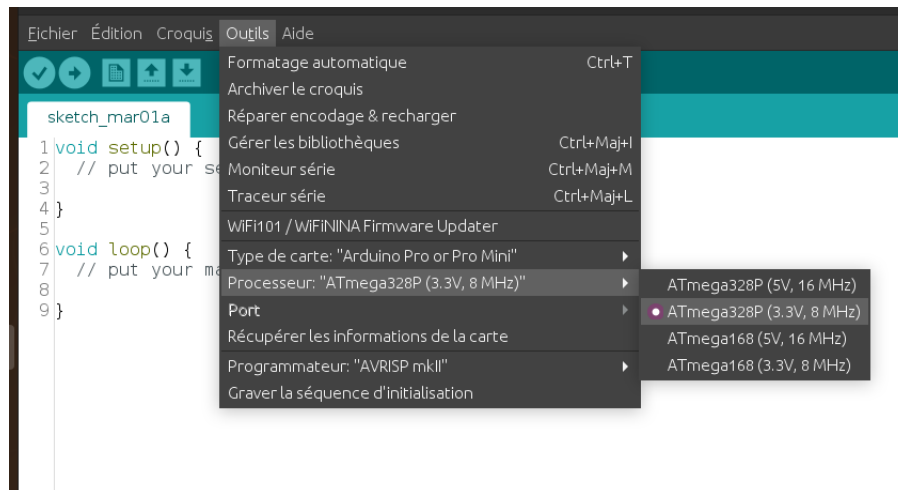


FIGURE 3.8 – Type de processeur

Avertissement

N'oubliez pas de sélectionner le port de communication de l'Arduino

Il ne vous reste plus qu'à cliquer sur le bouton de téléversement du programme.
La LED de la carte Pro-Mini devrait clignoter.

Information

Ici, nous avons chargé un programme de test, par la suite, il conviendra de charger le programme **Sonde_MySensors.ino**.

Cette étape de chargement de programme sera nécessaire à chaque modification du code de la sonde.

Programmation de la Nano

La carte Nano étant reliée à l'ordinateur par un câble USB, sa programmation sera plus aisée. On alimente la carte via l'ordinateur, on sélectionne le type de carte (**Type de carte > Arduino Nano**), le type de processeur (**Outils > Processeur > Old bootloader**), le port puis on téléverse le programme désiré.

Deuxième partie

Montage des circuits imprimés

Rappels

La passerelle reçoit sur sa barrette connecteurs femelles un **Arduino Nano** et est reliée par une nappe à 8 conducteurs à un **module transmetteur NRF24**.

L'Arduino Nano est relié au Raspberry Pi de votre plateforme Domoticz par un câble USB (transmission d'infos et alimentation 5v) et alimentera la passerelle en 5v.

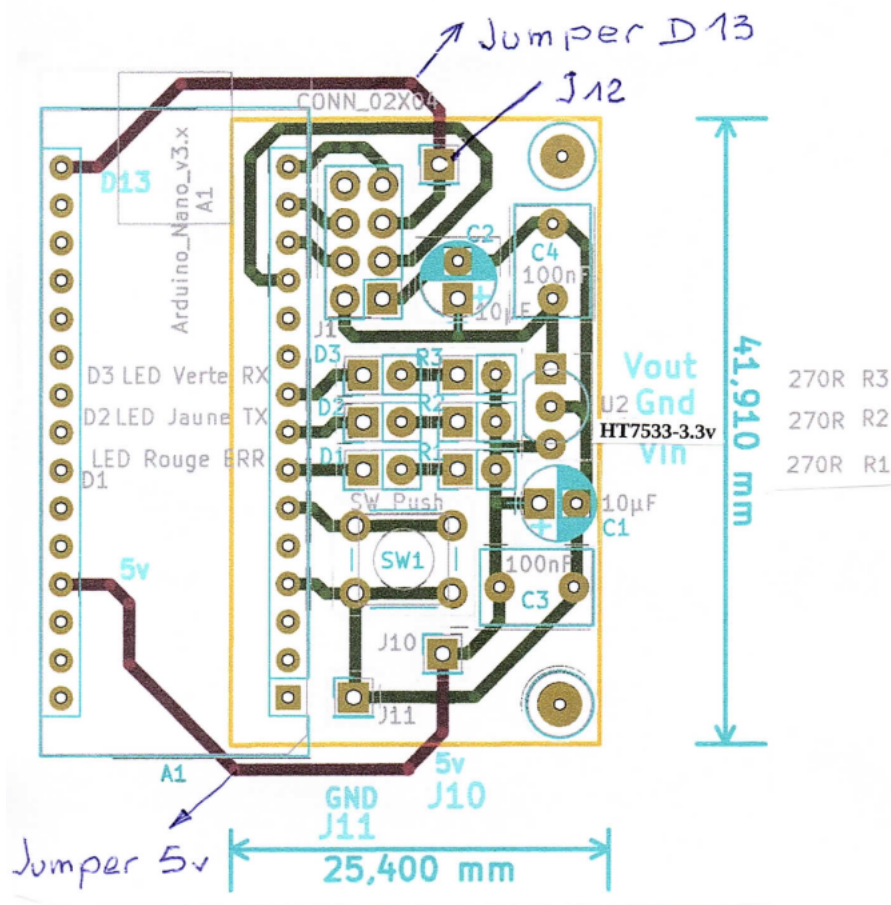
Le module transmetteur NRF24 assure la liaison radio avec les sondes.

Liste du matériel de la passerelle

- 3 Led
- 3 résistances de $270\ \Omega$
- 2 condensateurs électrolytiques ($100\ \mu F$, $16\ V$ – cylindriques noirs)
- 2 condensateurs céramiques monolithiques ($100nF$, $50\ V$ – couleur jaune foncé)
- 1 régulateur 3.3v HT7533-1
- 1 module transmetteur NRF24
- 1 bouton poussoir
- 1 circuit imprimé
- 1 barrette connecteurs femelle (déjà montée sur le circuit imprimé)
- 1 nappe 8 conducteurs (dont l'un porte un liseré rouge),
- 2 jumpers

Placement des composants

Vue de dessus du circuit



Remarques :

* Le circuit imprimé est entouré en jaune.
 * La partie entourée en vert représente l'Arduino Nano. N'est pas concernée pour le moment.

* Le régulateur de tension HT7533-3.3v remplace le LE33-3.3v schématiquement la partie plate se situe vers l'extérieur du circuit imprimé (voir Conseils pour l'implantation du HT7533).

FIGURE 4.1 – Circuit imprimé vu de dessus, coté composants

Étapes

1. Souder la barrette déjà en place

Remarque

Il faudra faire très attention au placement de la carte Arduino nano par la suite : La broche D13 de la Nano doit être impérativement dans le trou le plus avancé (coté nappe de fils).

Un décalage de 1 trou lors du placement de la carte Nano dans sa barrette pourra endommager la carte Nano et le module RF24!

2. Souder les 8 brins de la nappe en respectant les consignes suivantes :

- ▶ Séparer les 8 conducteurs sur 2 cm environ.
- ▶ Chaque conducteur étant multibrins, s'assurer qu'ils sont bien torsadés puis étamer.

- Respecter l'ordre de soudage -> Fil au liseré rouge en 1 puis conducteurs suivants en 2,3 etc.

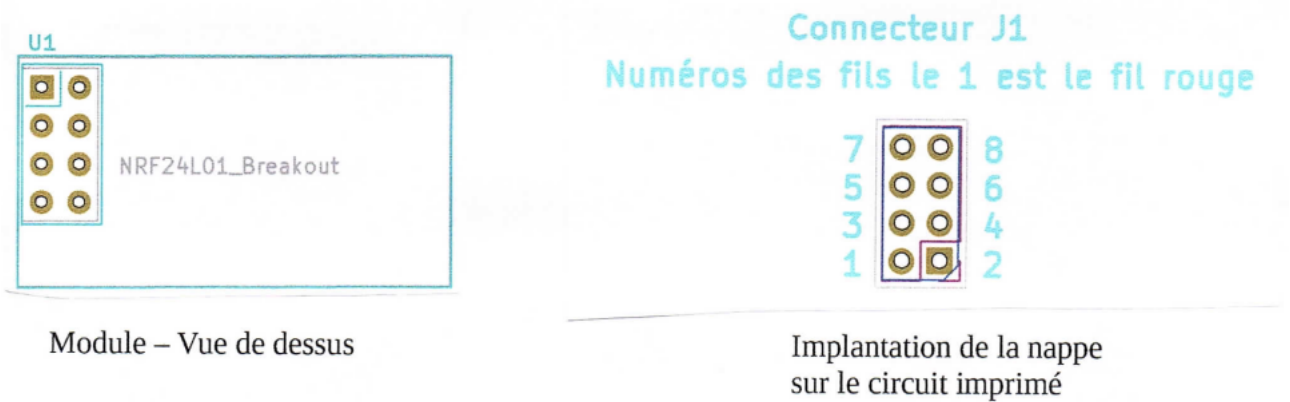


FIGURE 4.2 – Emplacement de la nappe

Remarque

Ne pas se tromper sur la soudure de la nappe
En gardant la vue de la Figure 3.1, le câble rouge de la nappe est en bas à gauche, le n°2 est en bas à droite, etc...

Puis dans l'ordre que vous souhaitez :

- Led rouge en D1 – Led jaune en D2 – Led verte en D3 (sont polarisées, patte longue au + , patte courte, méplat sur la led au -),
- Les 3 résistances sont à souder en R1, R2 et R3
- Les 2 condensateurs électrolytiques sont à souder en C1 et C2 (sont polarisées, le corps du condo est noir avec une bande grisée, la patte de ce côté est le -),
- Les 2 condensateurs céramiques sont à souder en C3 et C4
- Le bouton poussoir en SW1
- Le jumper D13 (Arduino Nano) en J12
- Le jumper 5v (Arduino Nano) en J10
- Le régulateur 3.3v HT7533 est à souder suivant les conseils de la section suivante

Mise en place du régulateur de tension

Comme vous l'avez sûrement remarqué, l'implantation précédente correspond à un régulateur LE33 et non à un HT7533

Si vous voulez utiliser un HT7533, il faut adapter le brochage du HT7533 au circuit.

- Vin : Entrée 5v
- GND : la masse
- Vout : sortie 3.3V

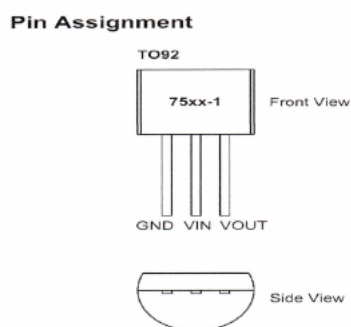


FIGURE 4.3 – Broches du HT7533

Adaptation des broches du HT7533 au schéma de la passerelle

Il faut que les broches **GND**, **Vin** et **Vout** rentrent dans les mêmes broches que celle du schéma de la passerelle. Même si les broches ne sont pas dans le même ordre, c'est assez simple à faire en tordant les broches du HT7533 avec une petite pince plate.

Sur le HT7533, sans que les broches se touchent, **on tord Vin vers l'avant, Vout vers l'arrière et on ramène GND au milieu.**

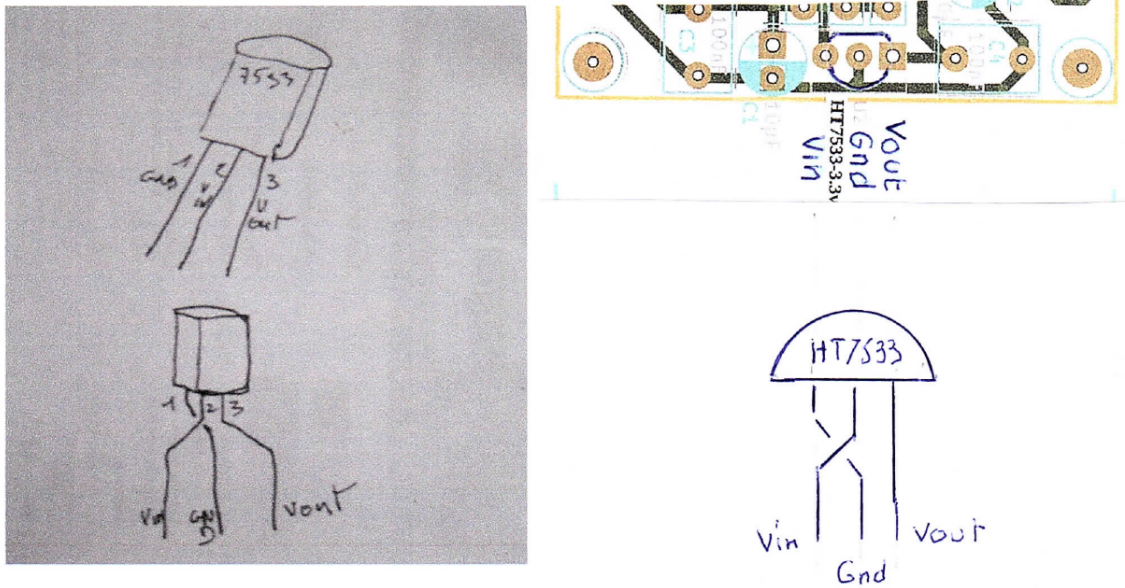


FIGURE 4.4 – Insertion du HT7533

Rendus

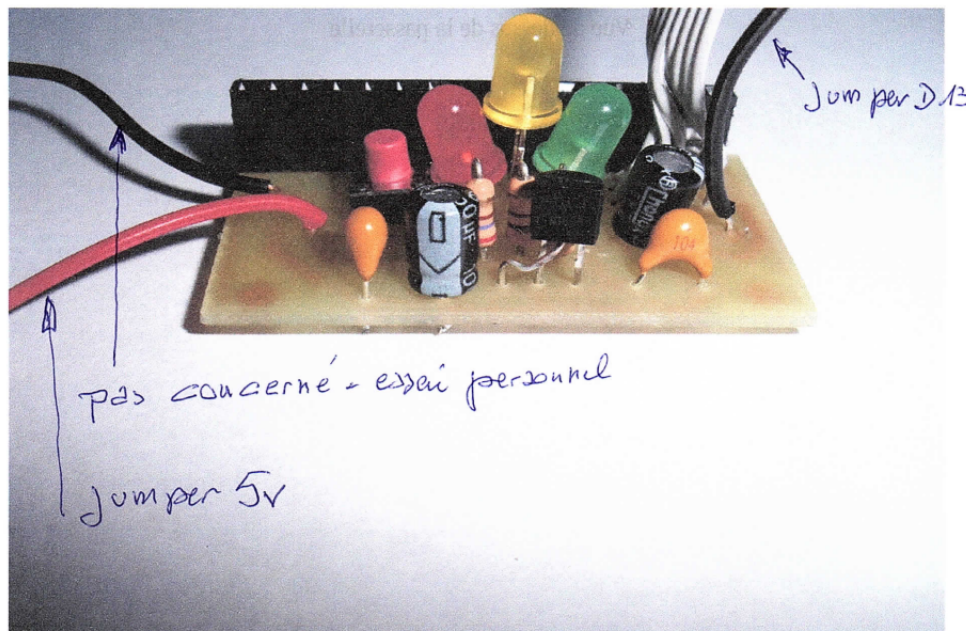


FIGURE 4.5 – Vue de côté

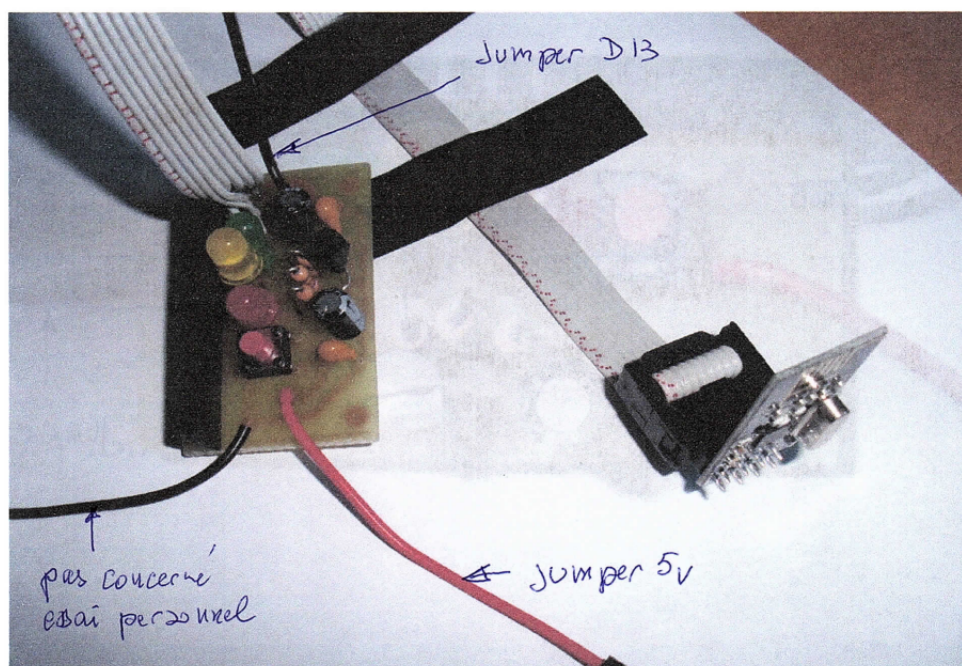


FIGURE 4.6 – Vue de la passerelle et de l'émetteur

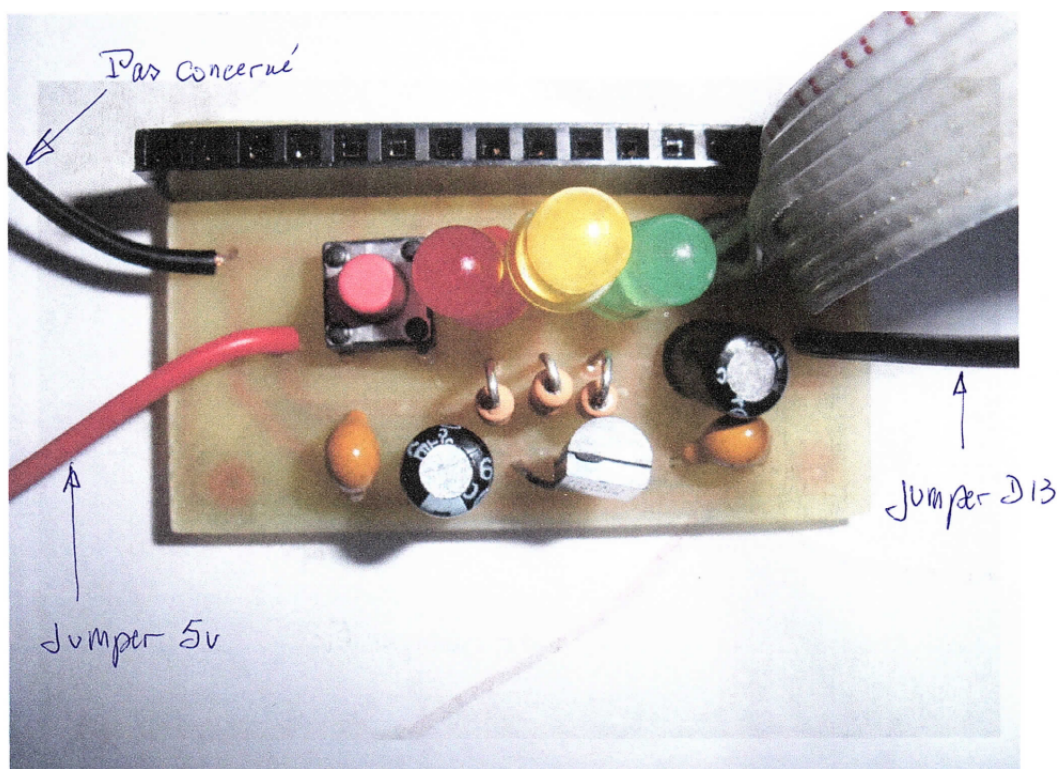


FIGURE 4.7 – Vue de dessus

Rappels

La sonde reçoit sur sa barrette connecteurs femelles un **Arduino Pro-Mini** et est reliée par une nappe à 8 conducteurs à un **module transmetteur NRF24**. Le module transmetteur NRF24 assure la liaison radio avec la passerelle.

Liste du matériel de la sonde

- 1 pcb
- 2 barrettes mâle-femelle 12 plots
- 1 barrette mâle-femelle 3 plots
- 1 barrette mâle-femelle 2 plots
- 3 barrettes mâle-mâle 5 plots
- 1 barrette mâle-mâle 4 plots
- 1 barrette mâle-mâle 3 plots
- 3 barrettes mâle-mâle 2 plots
- 1 barrette mâle-mâle 1 plot
- 2 condensateurs céramiques 100 nF
- 2 condensateurs chimiques 10 μ F
- 1 résistance 1/4 w 330 K Ω
- 1 résistance 1/4 w 1M Ω
- 1 régulateur HT7533
- 1 longueur de fil d'acier pour shunts
- 1 longueur de fil isolé pour shunt
- 1 carte Pro mini
- 1 émetteur NRF24
- 1 nappe 8 conducteurs

Placement des composants

Vue de dessus du circuit

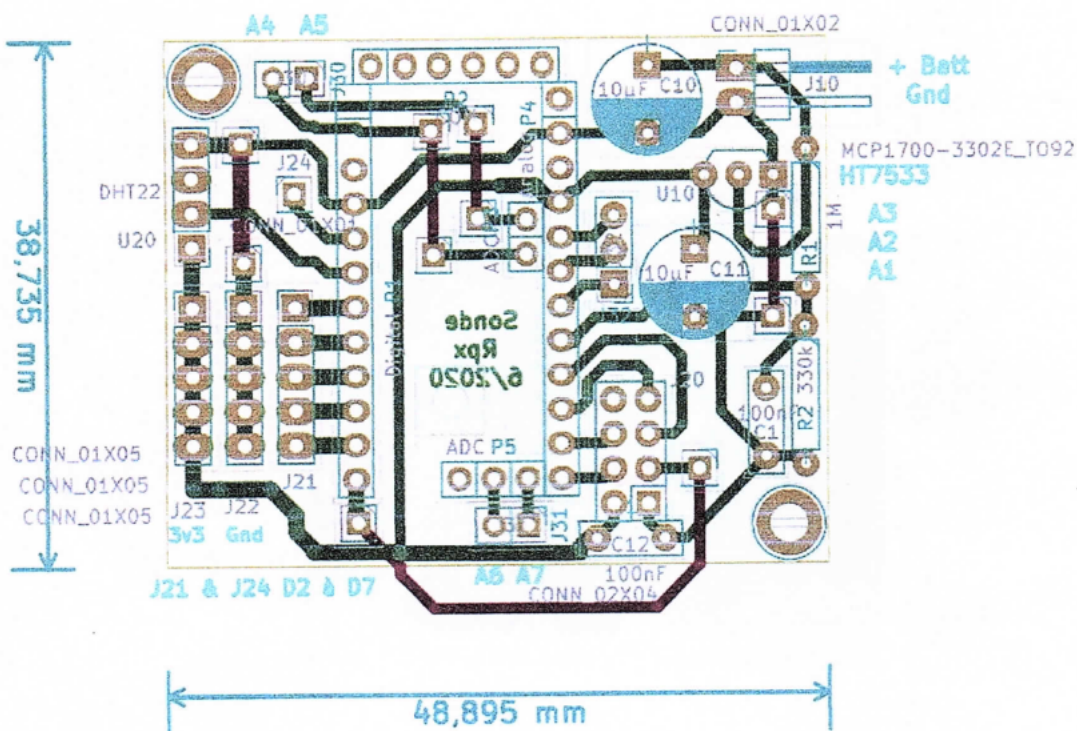


FIGURE 5.1 – Circuit imprimé vu de dessus, coté composants

Étapes

1. Souder les shunts. Voir schéma général et la photo platine sonde shunts
2. Souder les 8 conducteurs de la nappe en suivant les mêmes instructions que pour le montage de la passerelle

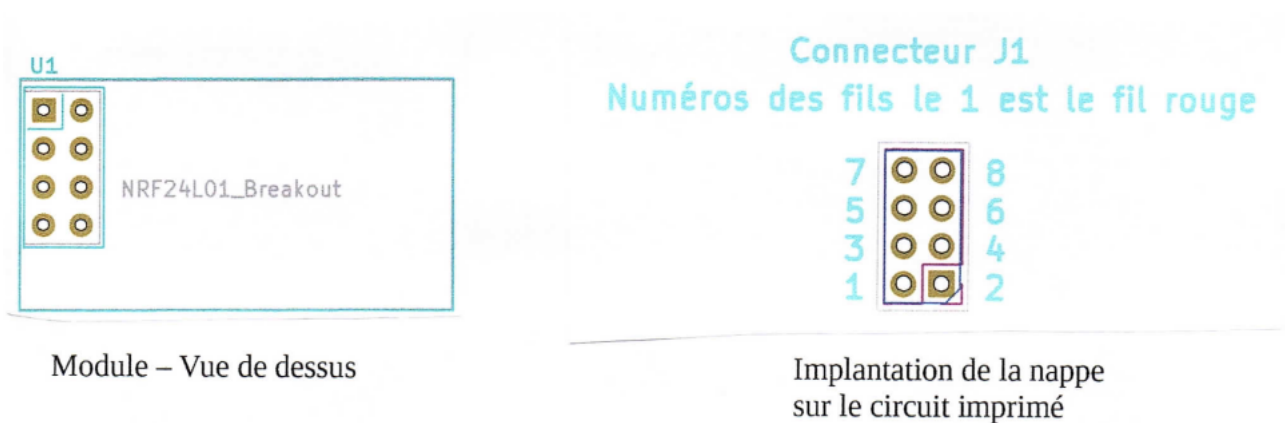


FIGURE 5.2 – Emplacement de la nappe

Puis dans l'ordre que vous souhaitez :

- Les 2 condensateurs électrolytiques sont à souder en C1 et C2 (**sont polarisées**, le corps du condensateur est noir avec une bande grisée, la patte de ce côté est le -),
- Les 2 condensateurs céramiques sont à souder en C3 et C4
- Le régulateur 3.3v HT7533
A la différence de son implantation sur la passerelle MySensor, ici il ne faut pas croiser les pattes, implantez le module tel quel en respectant son positionnement sur le pcb grâce au méplat
- Les 2 condensateurs céramiques de 100 nF
- Les 2 résistances de 330 k Ω et de 1 M Ω (Elles ont le même aspect extérieur vérifier à l'ohmmètre avant la pose)
- les 2 condensateurs chimiques de 10 μ F (Le moins est du côté grisé sur le corps du condensateur)
- Souder les connecteurs mâle-mâle sur la carte Pro mini, ils sont livrés dans la pochette. Attention à ne pas trop chauffer les points de soudure.
- Faire de même avec la carte Arduino Nano nécessaire à la passerelle.

Rendus

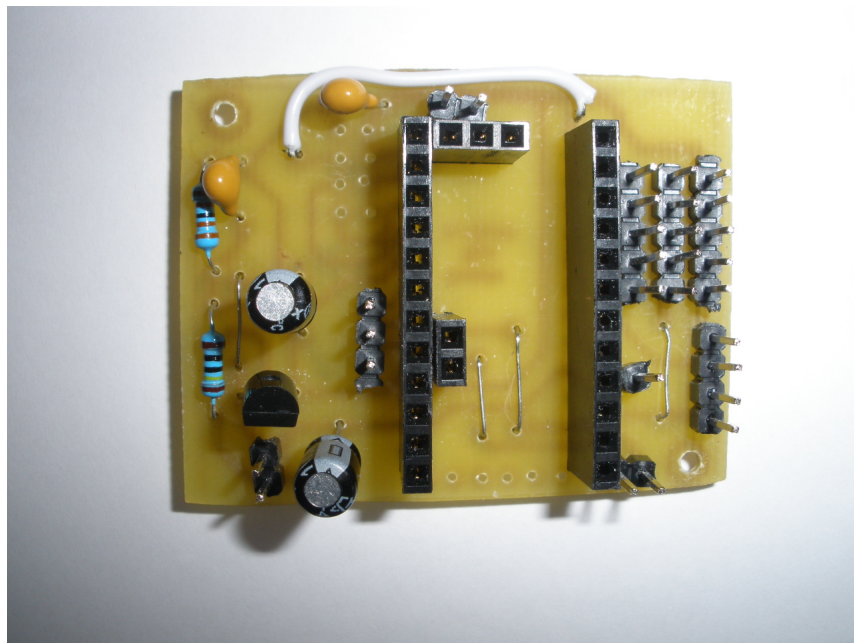


FIGURE 5.3 – Sonde vue de dessus sans la nappe

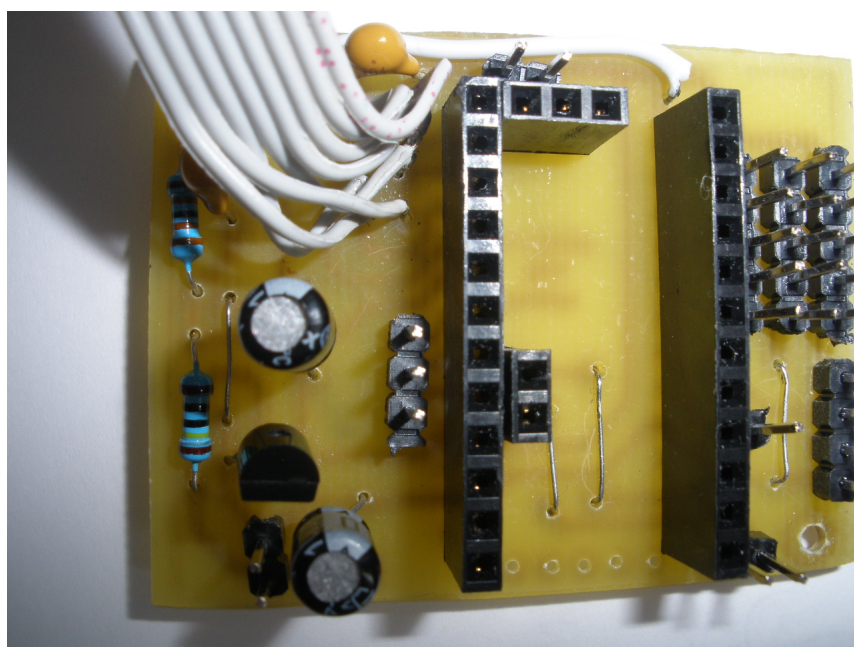


FIGURE 5.4 – Sonde vue de dessus avec la nappe

Les court-circuit

Le multimètre

Le meilleur allié contre les courts circuit est le multimètre.
Sur les rangées de barrettes, nous allons regarder la résistance entre deux broches voisines.
Si la résistance est infinie (**Un 1 affiché sur l'écran**), il n'y a pas de court-circuit et si elle tend vers 0, il y a un risque.

6.1.1.1 Réglage

On règle le multimètre en mode **Ohmmètre**, c'est à dire avec le fil noir sur **COM**, le rouge sur Ω et le curseur réglé sur la résistance la plus élevée de l'appareil.

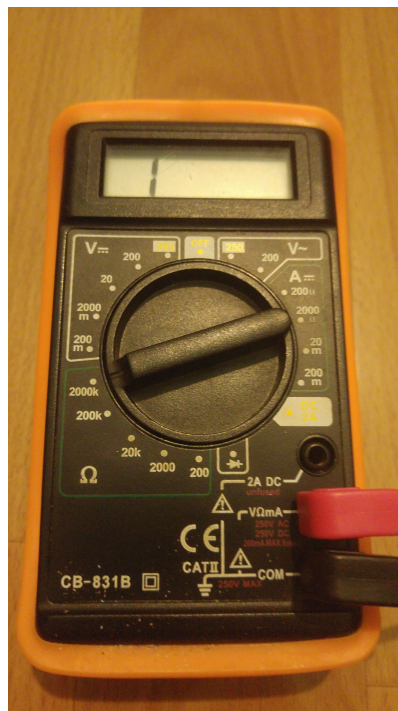


FIGURE 6.1 – un multimètre bien réglé

On regarde la résistance entre les broches 1 et 2 par exemple pour commencer puis ensuite entre la broche 2 et 3, etc...

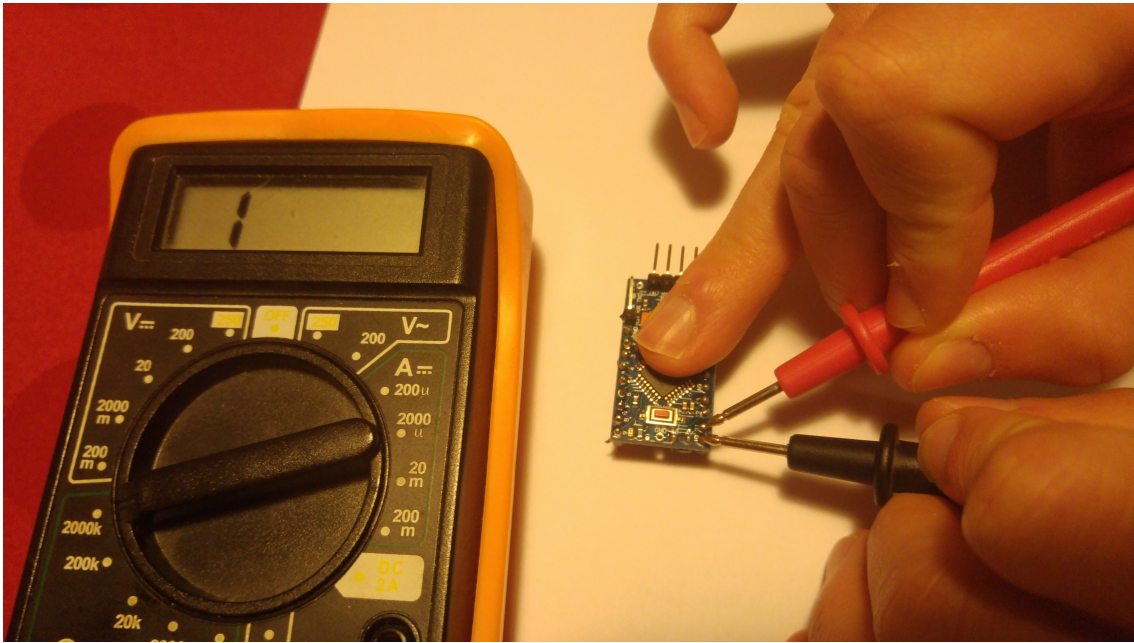


FIGURE 6.2 – Une vérification

L'alimentation

Le problème le plus grave peut survenir si un court circuit a lieu entre la broche +VCC¹ et la masse.

Il convient donc de trouver ces deux broches (VCC et GND) et de regarder la valeur de la résistance entre ces deux broches. Cette valeur doit être infinie (**1 sur l'afficheur**)

Les sondes NRF24

Les sondes NRF24 viennent s'insérer dans la nappe de fils (8 brins). Il ne faut pas se tromper de sens sous peine de détruire le module NRF24 lors de sa mise sous tension.

Pour cela, il faut que le coté avec les broches 1 et deux du NRF24 (repéré avec le carré blanc sur la broche 1) soit du même coté que le fils rouge de la nappe.

1. Alimentation positive, ici +3.3V pour la sonde et +5V pour la passerelle

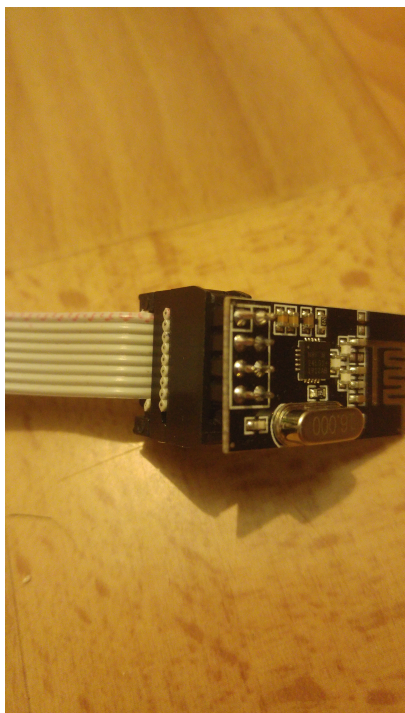


FIGURE 6.3 – Insertion du NRF24 dans son connecteur

Une fois que la connexion électrique est exacte, on peut alimenter le montage et vérifier la tension au bornes de la sonde NRF24. Si une tension inférieure à $3.2V$ ou supérieure à $3.4V$ apparaît, coupez l'alimentation et reprenez les vérifications.

Puis procédez de la même manière pour la sonde.

Troisième partie

Configuration des programmes

Arduino

Nous allons configurer le programme de la sonde (**Sonde_MySensors.ino**) pour l'envoi des données des capteurs.

Paramétrage du NRF24

Pour éviter les interférences entre les modules NRF24 dans une même salle, nous allons sélectionner un canal de communication pour chaque personne.

Un canal correspond à une fréquence précise d'émission et de réception pour le module NRF24.

Voici le tableaux des canaux attribué aux personnes :

Prénom	CANAL_NRF24
André P.	84
Florian M.	85
Guy D.	86
Marcel R.	87
Michel T.	88
Nicolas L.G.	89
Patrice G.	90
Patrick P.	91
Patrick Z.	92
Philippe C.	93
Pierre G.	94
Yvon	95

FIGURE 7.1 – Répartition des canaux pour les utilisateurs

Voici un extrait du code **Sonde_MySensors.ino** :

```
/*!  
 * *****  
 * PARAMETRES NRF24 (exemple canal 86)  
 * Légende : (*) = A changer pour chaque personne  
 * *****  
 */  
//Mode debug activé  
#define MY_DEBUG  
  
// Enable and select radio type attached  
#define MY_RADIO_RF24  
  
#ifndef MY_RF24_PA_LEVEL  
#define MY_RF24_PA_LEVEL RF24_PA_MAX
```

```

#endif

#ifndef MY_RF24_CHANNEL
#define MY_RF24_CHANNEL 86 //(*) A CHANGER
#endif

#ifndef MY_RF24_DATARATE
#define MY_RF24_DATARATE RF24_250KBPS
#endif

```

Paramétrage du NRF24

Paramétrage de Domoticz

Comme vue dans le chapitre de présentation, chaque sonde possède un identifiant et chaque capteur rattaché à la sonde possède lui aussi un identifiant.

Pour éviter toute confusion, nous allons également attribuer un identifiant à notre sonde et à nos capteurs, ces identifiants sont définis pour chaque personne dans le tableau suivant :

Prénom	MY_NODE_ID	ID batterie	ID température	ID Humidité
André P.	10	11	12	13
Florian M.	20	21	22	23
Guy D.	30	31	32	33
Marcel R.	40	41	42	43
Michel T.	50	51	52	53
Nicolas L.G.	60	61	62	63
Patrice G.	70	71	72	73
Patrick P.	80	81	82	83
Patrick Z.	90	91	92	93
Philippe C.	100	101	102	103
Pierre G.	110	111	112	113
Yvon G.	120	121	122	123

FIGURE 7.2 – Répartition des identifiants pour les utilisateurs

```

/#!/
* *****
* PARAMETRES DOMOTICZ (Exemple avec l'ID 30)
* *****
* Légende : (*) = A changer pour chaque personne
*/
#define MY_NODE_ID 30 //Noeud dans Domoticz (*)

```



```
//Voir le tableau
#define CHILD_ID_BATT 31      //(*) Identifiant Domoticz pour le niveau de
    batterie
#define CHILD_ID_TEMP 32     //(*) Identifiant Domoticz pour la température
#define CHILD_ID_HUM 33      //(*) Identifiant Domoticz pour l'humidité
```

Paramétrage de Domoticz

ici, il nous reste à définir le temps entre deux envois de données par la sonde.

```
//Temps entre deux envois de données
static const uint64_t UPDATE_INTERVAL = 10000;
```

Temps de mise à jour

Ensuite, on peut éventuellement changer le nombre de mesure au bout duquel la sonde envoie les données même si elles n'ont pas changés

```
//Nombre au bout duquel la sonde envoie les données même si elles n'ont
pas changés
static const uint8_t FORCE_UPDATE_N_READS = 10;
```

Nombre de lectures forcée

Paramétrage du DHT

Il faut préciser si le capteur est un DHT11 ou DHT22 et indiquer la broche du capteur

```
/*!
 * *****
 * DHT22/DHT11
 * *****
 */
//Broche de données du DHT22 (ou DHT11)
#define DHT_DATA_PIN 3

#define DHT_TYPE DHT22 //use DHT11 or DHT22

//définir une valeur si le capteur à un offset permanent par rapport à la
température réelle
#define SENSOR_TEMP_OFFSET 0

float lastTemp;    //Dernière valeur de température lue
float lastHum;     //dernière valeur d'humidité lue
```

Paramétrage du DHT

Paramétrage de la mesure de la batterie

Il suffit de renseigner les valeurs des résistances (en Ω) formant le pont diviseur de tension. Dans mon cas, les résistances ont une valeur de $330\text{ k}\Omega$ et $1\text{ M}\Omega$

Le calcul pour afficher la tension réel de la batterie est détaillé en annexe.

```
/*!  
 * *****  
 * BATTERIE  
 * *****  
 */  
//Partie pour mesure Batterie  
int BATTERY_SENSE_PIN = A0; //ne pas toucher  
  
float oldBatteryV = 0; //Sauvegarde du niveau de batterie  
  
const float r1_value = 1000000.0; //Valeur de la résistance la plus proche  
    de l'alimentation de la batterie  
const float r2_value = 330000.0; //Valeur de la résistance la plus proche  
    de la masse
```

Paramétrage du multimètre

Branchements du DHT

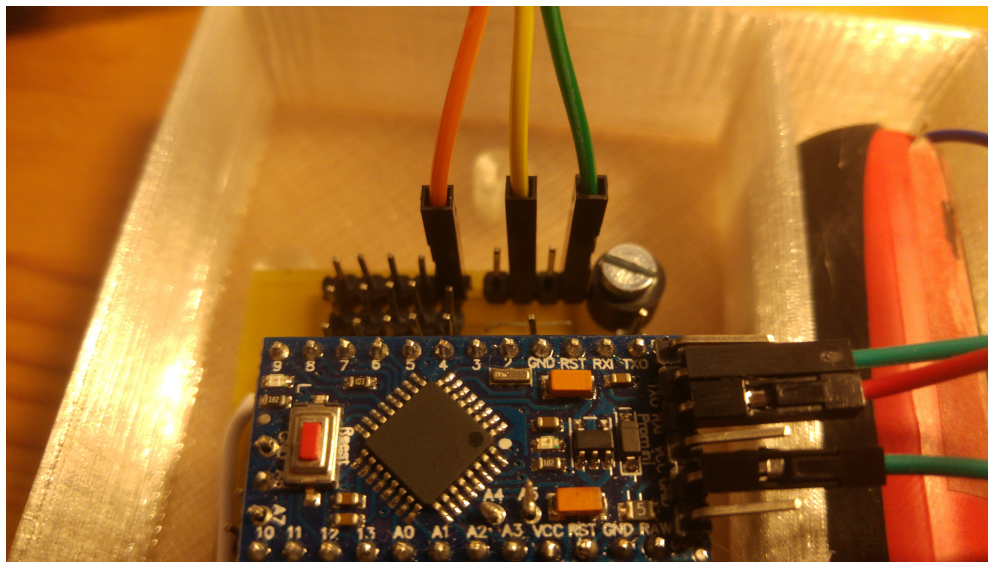


FIGURE 8.1 – Branchement du DHT sur la sonde

Voici les connexions pour brancher le DHT :

- ▶ **PIN GND_DHT** est représenté par le câble verts (droite)
- ▶ **PIN VCC_DHT** est représenté par le câble orange (gauche)
- ▶ **PIN OUT_FHT** est représenté par le câble jaune (D3) (centre)

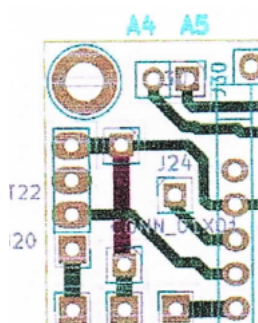


FIGURE 8.2 – Schéma du circuit imprimé

Nous allons configurer le programme de la passerelle(**Passerelle_MySensors.ino**)pour la réception des données.

Paramétrage du NRF24

Il faut mettre le même canal de communication que pour la sonde.
Un module recevant sur le canal 84 ne pourra pas recevoir des données en provenance d'un canal 83 ou 85.

Pour rappel, voici le tableau des canaux :

Prénom	CANAL_NRF24
André P.	84
Florian M.	85
Guy D.	86
Marcel R.	87
Michel T.	88
Nicolas L.G.	89
Patrice G.	90
Patrick P.	91
Patrick Z.	92
Philippe C.	93
Pierre G.	94
Yvon	95

FIGURE 9.1 – Répartition des canaux pour les utilisateurs

Voici un extrait du code **Passerelle_MySensors.ino** :

```

/*!
 * *****
 * PARAMETRES NRF24 (exemple canal 86)
 * Légende : (*) = A changer pour chaque personne
 * *****
 */
#ifndef MY_RF24_CHANNEL
#define MY_RF24_CHANNEL    86           //Le canal doit être le même que celui
    de la sonde
#endif

```

Paramétrage du NRF24

Envoi des programmes

Une fois les deux programmes modifiés avec les bonnes valeurs, il ne vous reste plus qu'à envoyer le programme de la sonde sur la carte Pro-Mini et celui de la passerelle sur la carte Nano.

Toutes les informations pour programmer les deux cartes sont disponibles au chapitre 2 (**Programmation**).

Lorsque les deux cartes sont programmées, occupons nous maintenant de Domoticz.

Quatrième partie

Configuration de Domoticz

Une fois que la passerelle est fonctionnelle, nous allons configurer Domoticz pour que la plateforme reçoive les données en provenance de la passerelle.

Ajout de la passerelle

Tout d'abord, allez dans la section **Configuration > Matériel**

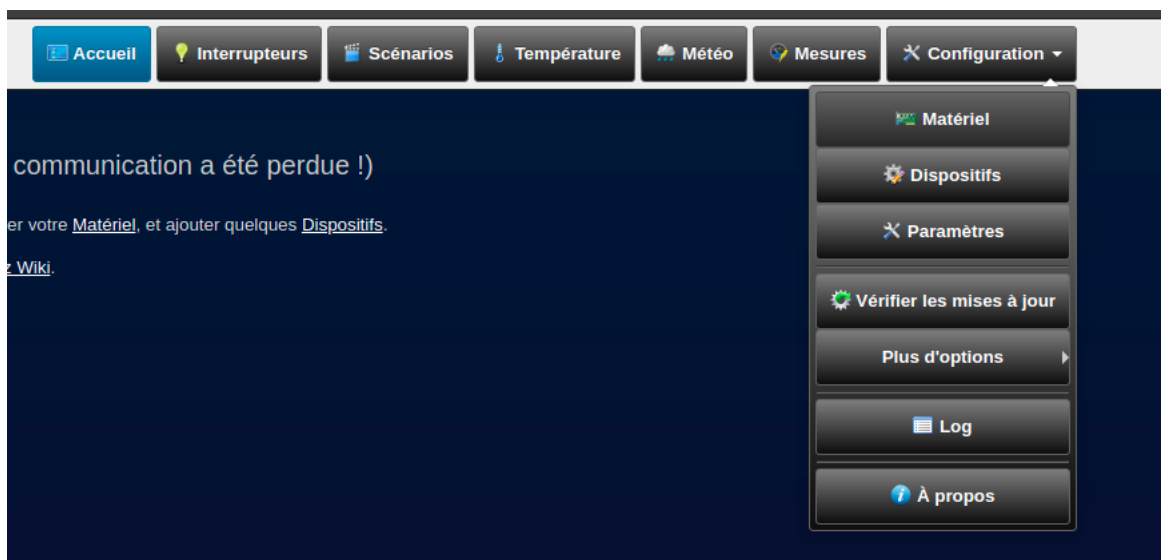


FIGURE 10.1 – Emplacement du matériel

Ensuite, saisissez les informations suivantes :

The image shows the configuration form for a gateway in Domoticz. The form has the following fields: 'Activé' with a green toggle switch; 'Nom' with a text input field containing 'Gateway'; 'Type' with a dropdown menu showing 'MySensors Gateway USB'; 'Délai d'inactivité' with a dropdown menu showing 'Désactivé' and a warning message in red: 'Le périphérique sera redémarré si aucune donnée n'est reçue pendant ce délai. N'activez pas cette option pour les périphériques ne recevant pas de données !'; 'Port série' with a dropdown menu showing '/dev/serial/by-id/usb-1a86_USB2.0-Ser_if00-port0'; and 'Vitesse de transmission' with a dropdown menu showing '115200'. At the bottom, there is a blue 'Ajouter' button.

FIGURE 10.2 – Paramétrage de la passerelle

Le port série sélectionné sera celui où est raccordé la passerelle en liaison USB. Il ne faut pas prendre les noms simplifiés des ports USB (*COM_XXX*) mais le nom le plus complet. Pour plus de simplicité, veuillez déconnecter tous les autres périphériques du Raspberry-Pi

Recherche des capteurs

Visualisons les données en provenance de la sonde en allant dans **Configuration > Matériel**

L'ensemble de vos dispositif apparaît. En cas de liste trop longue, saisissez **Gateway** dans la barre de recherche.

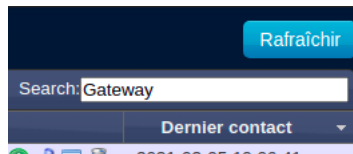


FIGURE 10.3 – Recherche de la passerelle

Remarque

Si le dispositif n'apparaît pas immédiatement, patientez quelques instants.

Idx	Nom	Activé	Type	Adresse	Port	Délai d'inactivité
2	Gateway	Oui	MySensors Gateway USB Version: ? Configuration		/dev/serial/by-id/usb-1a86_USB2.0-Ser_if00-port0	Désactivé

Affichage de 1 à 1 sur 1 entrée(s)

Première Précédente 1 Suivante Dernière

FIGURE 10.4 – La passerelle est détectée

Nous allons ensuite vérifier que les capteurs de la sonde envoient bien les données. Pour cela, cliquez sur **Configuration**. La page suivante apparaît :

Dispositif: Gateway

Nœuds

Afficher 25 entrées
Recherche :

NodeID	Name	Sketch Name	Version	Childs	Last Seen
0	Unknown	Unknown	1.0	6	2021-05-13 21:05:49
1	Unknown	Unknown	1.0	4	-
25	Unknown	Unknown	1.0	1	-
30	Unknown	Unknown	1.0	3	2021-05-13 21:09:31
55	Unknown	Unknown	1.0	1	-

Affichage de 1 à 5 sur 5 entrée(s)
Première
Précédente
1
Suivante
Dernière

Modifier
Supprimer
Rafraichir

Enfant

Afficher 25 entrées
Recherche :

ChildID	Type	Name	Values	Ack	Ack Timeout	Last Seen
Aucune donnée disponible dans la table						

Affichage de 0 à 0 sur 0 entrée
Première
Précédente
Suivante
Dernière

Modifier
Supprimer
Rafraichir

FIGURE 10.5 – Page de la Gateway

Pour visualisez les valeurs des capteurs, il faut sélectionner la passerelle avec l’ID de la sonde (ici, 30).

Nœuds

Afficher 25 entrées

NodeID	Name
0	Unknown
1	Unknown
25	Unknown
30	Unknown
55	Unknown

Affichage de 1 à 5 sur 5 entrée(s)

FIGURE 10.6 – Sélection de la passerelle

En cliquant dessus, on voit que la partie **Enfants** est mise à jour et contient les 3 capteurs avec les ID définis dans le programme de la sonde (31,32 et 33 en ce qui me concerne)

Enfant

Afficher 25 entrées

ChildID	Type	Name	Values	Ack	Ack Timeout	Last Seen
31	S_UNKNOWN		#1. V_VOLTAGE (4.10323)	true	1200	2021-05-13 21:12:12
32	S_UNKNOWN		#2. V_TEMP (21.8)	true	1200	2021-05-13 21:12:12
33	S_UNKNOWN		#3. V_HUM (57)	true	1200	2021-05-13 21:12:12

Affichage de 1 à 3 sur 3 entrée(s)

Première Précédente 1 Suivante Dernière

FIGURE 10.7 – Visualisation des enfants

Visualisation des données

Maintenant que nous savons que la sonde envoie les bonnes données, nous allons ajouter les capteurs dans les dispositifs. Pour cela, allez dans **Configuration > Dispositifs**, les 3 capteurs de la sonde (Tension batterie, humidité et température) apparaissent dans la liste. Si vous ne les trouvez pas, vous pouvez nettoyer la page des capteurs en sélectionnant les capteurs non-utilisés et en le mettant à la poubelle.

Show 25 entries

Idx	Matériel	ID	Unit	Nom	Type	Sous-type	Données	Dernier contact
1	Gateway	00001E1F	1	Voltage	General	Voltage	4.103 V	2021-05-13 21:26:01
2	Gateway	1E21	1	Hum	Humidity	LaCrosse TX3	Humidity 57 %	2021-05-13 21:26:01
3	Gateway	1E20	32	Temp	Temp	LaCrosse TX3	21.7 C	2021-05-13 21:26:01

Showing 1 to 3 of 3 entries

First Previous 1 Next Last

FIGURE 10.8 – Visualisation des capteurs

Les capteurs apparaissent sous les 3 noms suivants :

Nom
Voltage
Hum
Temp

FIGURE 10.9 – Nom des capteurs

Pour ajouter un dispositif, il suffit de cliquer sur la flèche verte et de choisir le nom du dispositif.

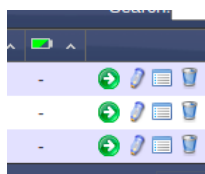


FIGURE 10.10 – Ajout des dispositifs

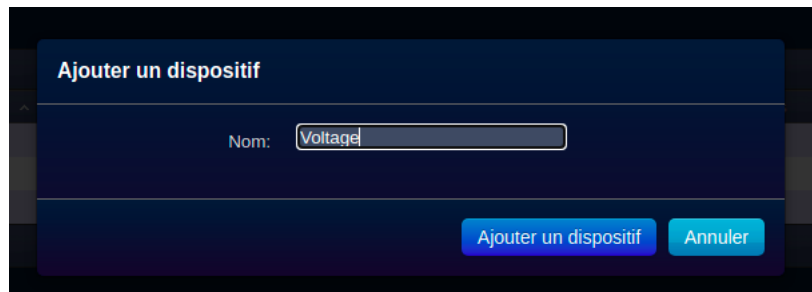


FIGURE 10.11 – Ajout des dispositifs - Sélection du nom

Il suffit de cliquer dans le menu **Mesures**



FIGURE 10.12 – Mesures

Et apparaît la tension de la batterie.

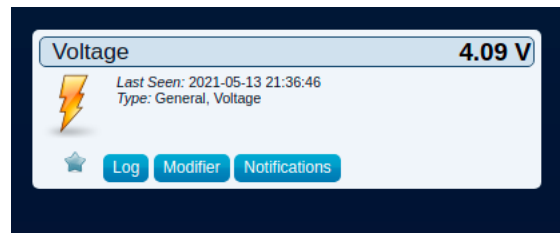


FIGURE 10.13 – tension de la batterie

On procède de même pour l'humidité et la température, les dispositifs seront mis dans l'onglet **Température**.

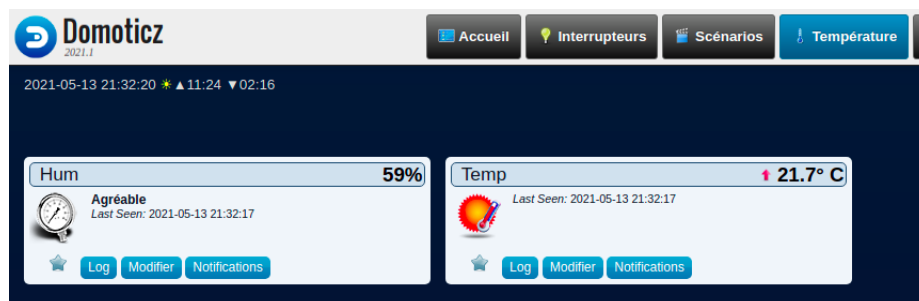


FIGURE 10.14 – Mesures de l'humidité et de la température

Pour visualiser les données, il suffit de cliquer sur le bouton **logs**

Cinquième partie

Annexes

Installation de Domoticz sur Linux

Veuillez ouvrir un terminal puis saisir les commandes suivantes :

```
sudo apt-get -y install cmake make gcc g++ libssl-dev git libcurl4-openssl-  
dev libusb-dev python3-dev curl zlib1g-dev zlib1g
```

Installation de domoticz

Puis lancez le script d'installation avec la commande suivante

```
sudo curl -L https://install.domoticz.com | bash
```

Installation de domoticz

Le terminal devrait afficher un contenu similaire :

```
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current  
00 16569 100 16569    0     0    35555    0 --:--:-- --:--:-- --:--:-- 35479  
::  
:: Script called with non-root privileges. The Domoticz installs server packages and configures  
:: system networking, it requires elevated rights. Please check the contents of the script for  
:: any concerns with this requirement. Please be sure to download this script from a trusted source.  
::  
:: Detecting the presence of the sudo utility for continuation of this install...  
:: Utility sudo located.  
::  
:: You are root.  
ash: ligne 100 : test: /etc/os-release:20.04 : nombre entier attendu comme expression  
:: Verifying free disk space...  
::  
:: Checking apt-get for upgraded packages.... done!  
::  
:: There are 108 updates available for your system!  
:: We recommend you run 'apt-get upgrade' after installing Domoticz!  
::  
:: Checking for apt-utils... installed!  
:: Checking for whiptail... installed!  
:: Checking for git... installed!  
:: Checking for curl... installed!  
:: Checking for unzip... installed!  
:: Checking for wget... installed!  
:: Checking for sudo... installed!  
:: Checking for cron... installed!  
:: Checking for libudev-dev...  
::
```

FIGURE A.1 – Vérification des bibliothèques

Ensuite, une interface utilisateur se lance dans le terminal :

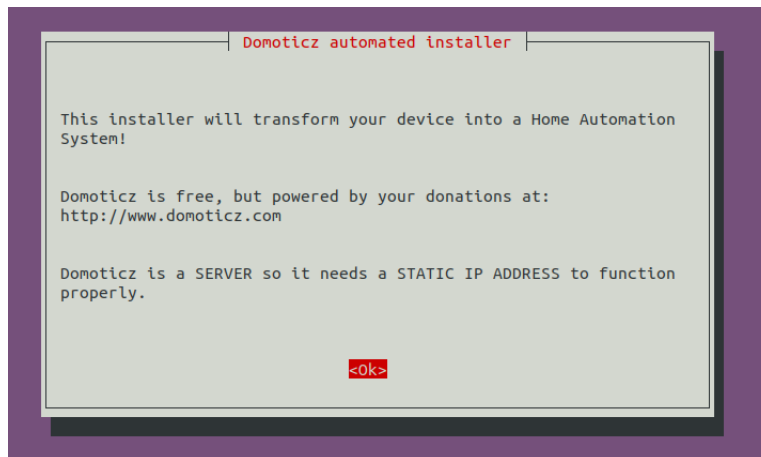


FIGURE A.2 – Présentation de Domoticz

Il faut saisir la touche **KEY ENTREE** pour afficher la fenêtre suivante. Une deuxième fenêtre apparaît. Veuillez sélectionner le service HTTP (par défaut) puis **KEY ENTREE**

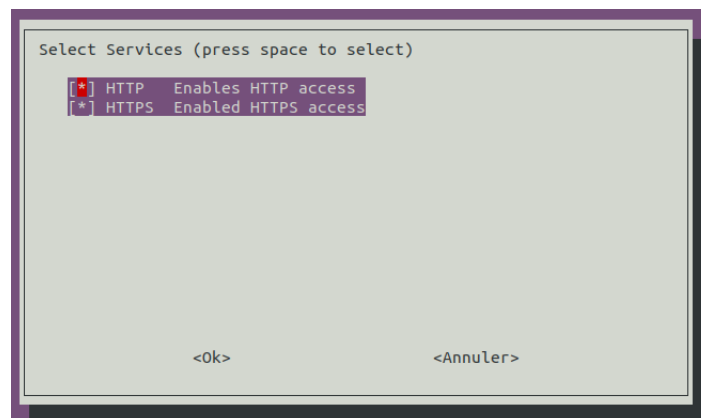


FIGURE A.3 – Choix du protocole par défaut

Nous allons ensuite choisir le port 8080 pour communiquer sur le réseau (par défaut : 8080)

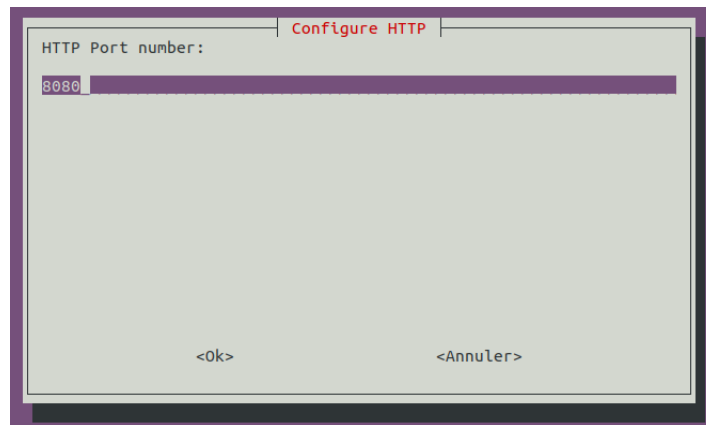


FIGURE A.4 – Choix du port

Nous utiliserons le port 443 (HTTPS) pour un protocole plus sécurisé.

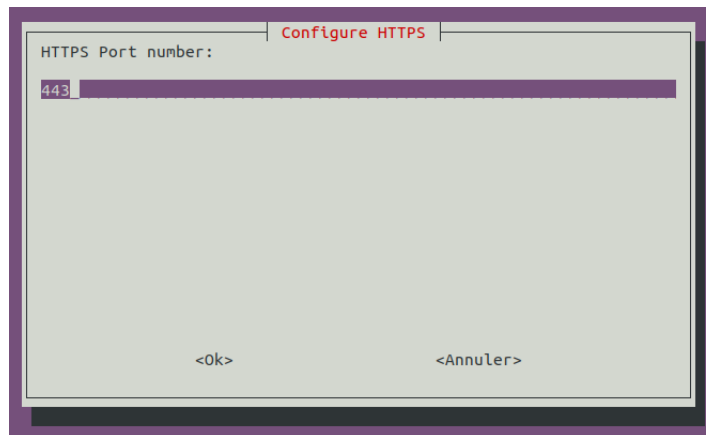


FIGURE A.5 – Protocole HTTPS

Il ne vous reste plus qu'à choisir l'emplacement du logiciel Domoticz.
Par défaut, Domoticz le place dans vos documents personnels (/home/nom_utilisateur)

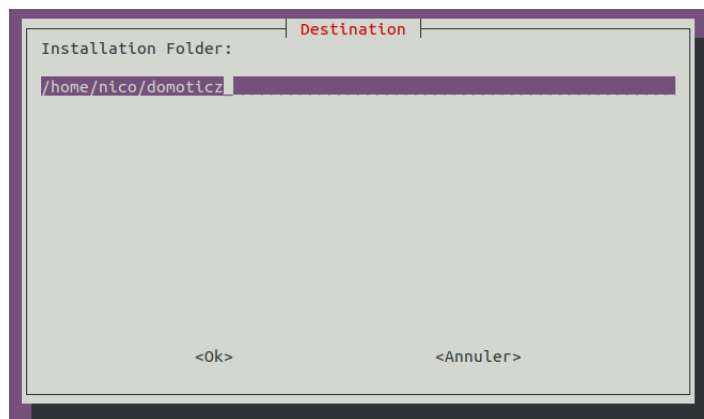


FIGURE A.6 – Emplacement des fichiers Domoticz

Il ne vous reste plus qu'à valider l'installation :

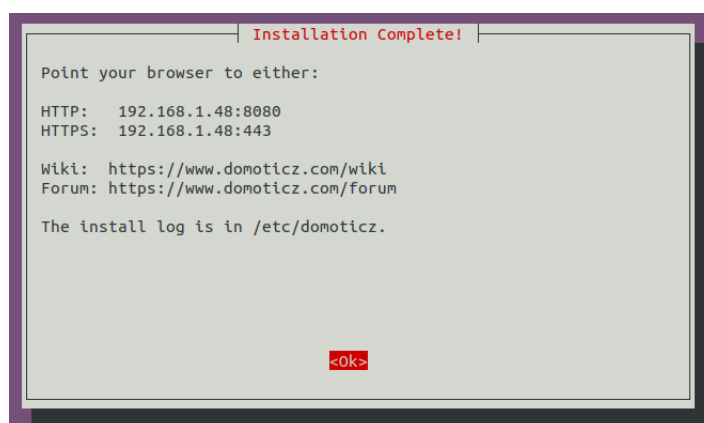


FIGURE A.7 – Validation de l'installation

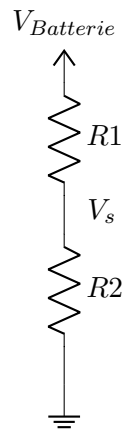
Puis dans votre navigateur internet, saisir :

localhost:8080

Lancement de Domoticz

Démonstration

Soit le pont diviseur de tension formé par les deux résistances R_1 et R_2 .
 Le pont est alimenté avec la tension de la batterie.
 La tension de sortie V_s va en entrée de la carte Arduino. (A0)



La tension en sortie d'un pont diviseur de tension vaut :

$$V_s = V_{Batterie} \cdot \frac{R_2}{R_1 + R_2}$$

La résolution du Convertisseur Analogique Numérique de l'Arduino est de 10 bits, c'est à dire que la CAN va donner une valeur N_{CAN} comprise entre 0 et $2^{10} - 1$, c'est à dire entre 0 et 1023.

Ainsi, si le CAN affiche une valeur de 1023, cela veut dire que $V_s = 3.3V$ et si $N_{CAN} = 512$, la tension V_s vaut environ 1.65 V.

La tension V_s est obtenue par la relation suivante :

$$V_s = N_{CAN} \cdot \frac{\text{Tension de référence}}{\text{Résolution CAN}} = N_{CAN} \cdot \frac{3.3}{2^{10}}$$

La tension de référence est généralement la tension de fonctionnement du microcontrôleur, c'est à dire ici 3.3V.

On obtient donc la formule liant $V_{Batterie}$ et la valeur N_{CAN} :

$$V_{Batterie} = \frac{R_1 + R_2}{R_2} \cdot N_{CAN} \cdot \frac{3.3}{2^{10}}$$

Cette formule sera utilisé pour déterminer la tension réelle de la batterie en fonction du Convertisseur Analogique Numérique.

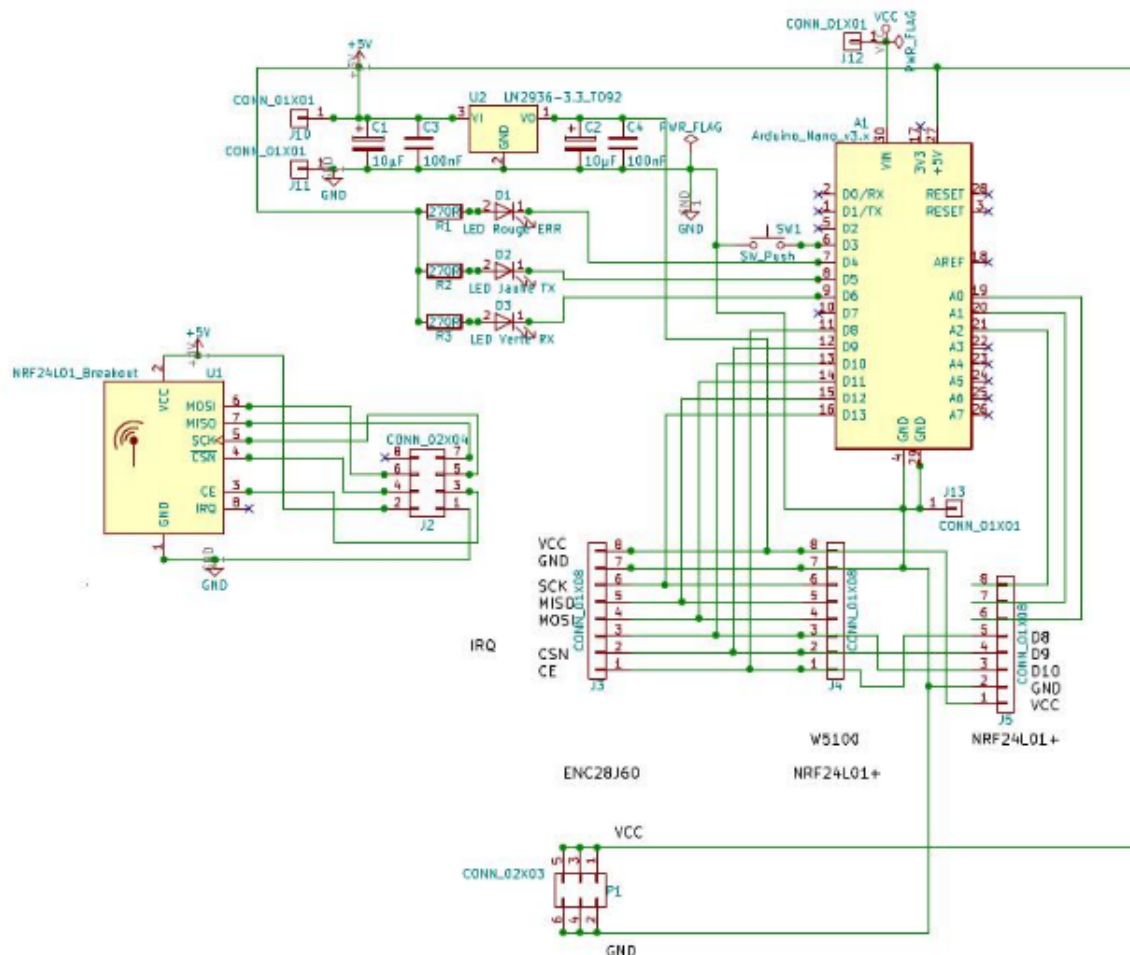


FIGURE C.1 – Schéma de la passerelle

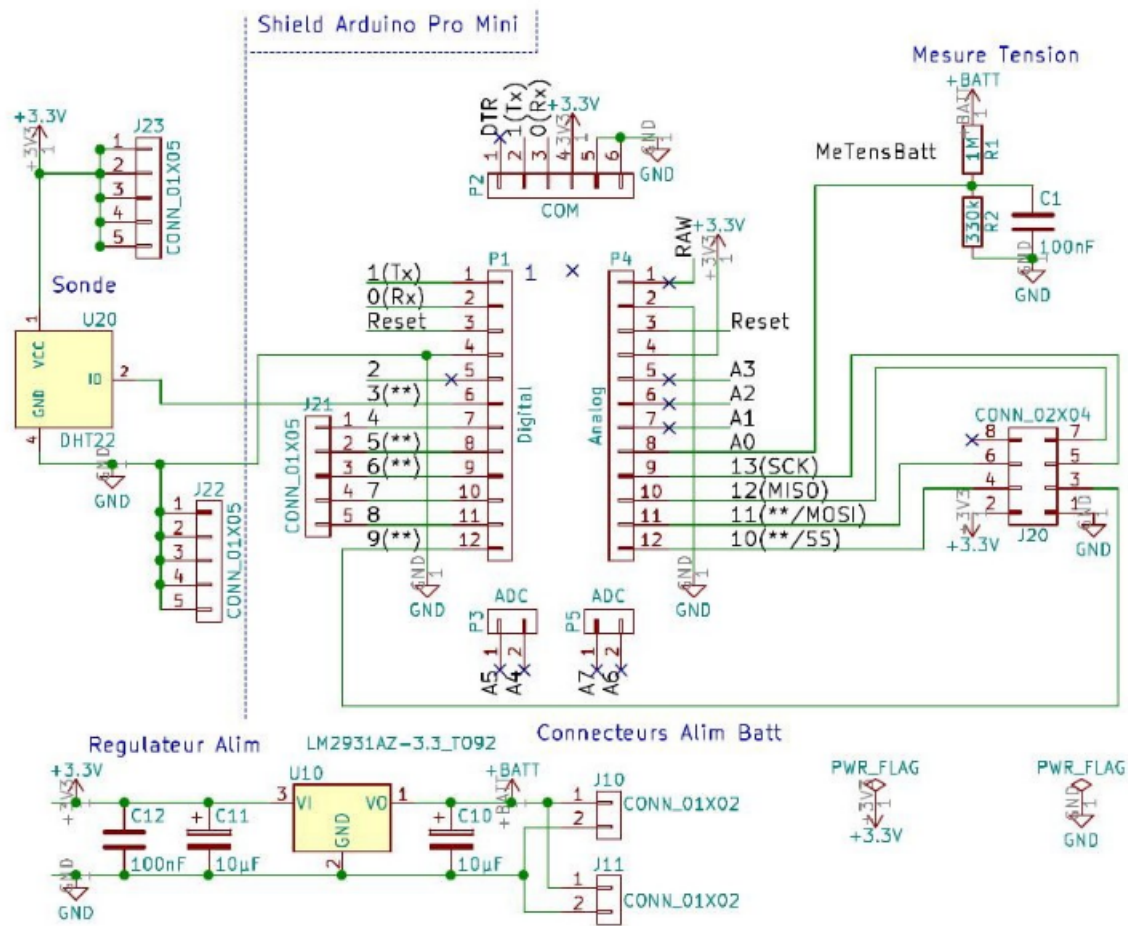


FIGURE D.1 – Schéma de la sonde

Question 1. *Pourquoi ma passerelle n'est pas détectée sur Domoticz ?*

>>> **1.** *Une erreur fréquence est de sélectionner le mauvais port lors de la configuration de la passerelle dans Domoticz.*

Question 2. *Pourquoi la passerelle allume sa LED rouge ?*

>>> **2.** *La LED rouge veut dire que des erreurs de communication sont survenues entre la passerelle et la sonde. Vérifier les branchements de la sonde.*