

# Arrosage automatique

Projet d'informatique

06.06.2021

HES-SO Valais/Wallis  
Haute école ingénierie  
Filière Energies et techniques environnementales  
8122 - Informatique

Auteurs : Tchavo FRUND,  
Valentin SANDOZ

Professeurs : P. Ferrez,  
P. Roduit

# Table des matières

Table des matières.....	1
1 Introduction.....	1
2 Conception.....	1
3 Acquisition des données .....	3
3.1 Mesure d'humidité du sol .....	3
3.1.1 Fonctionnement des capteurs .....	3
3.1.2 Compréhension des capteurs.....	3
3.1.3 Problèmes rencontrés.....	4
3.1.4 Traitement des données .....	4
3.2 Mesure d'humidité et de température de l'air .....	5
3.3 Mesure de la température du sol .....	5
3.4 Données météorologiques.....	5
4 Automatisation de l'arrosage.....	5
4.1 Modes d'arrosage .....	5
4.2 Conditions d'arrosage automatique .....	5
4.3 Boutons .....	6
4.4 LEDs d'information .....	6
4.5 Electrovanne.....	6
4.6 Dashboard.....	6
5 Conclusion.....	7
Annexe A - Schématique de l'installation .....	0

## 1 Introduction

Dans le cadre de ce projet de fin d'année, il nous a été demandé de réaliser un système de collecte de donnée nous permettant d'utiliser les différents programmes que nous avons appris à utiliser au cours de ces deux semestres.

Nous avons donc décidé de créer un système d'arrosage automatique, prenant en compte certaines données mesurées en temps réel comme l'humidité ou encore la température. Toutes les données mesurées ainsi que des graphes seront visible en temps réel sur une dashboard.

## 2 Conception

Comme il n'était pas possible d'installer notre système à l'école, nous l'avons mis en place dans le jardin de Valentin, à Saint-Légier. La platebande concernée comporte des herbes aromatiques (persil,

coriandre, sauge), diverses fleurs et des artichauts. Le choix s'est porté principalement par sa proximité au cagibi, permettant des courtes distances en le Raspberry pi protégé des intempéries et les capteurs.

Il nous a d'abord fallu rassembler tous les éléments nécessaires soit :

- Plusieurs capteurs d'humidité, le premier qui se trouvera juste en dessous de la surface du sol et un deuxième à cinquante centimètres plus bas. Cette différence de profondeur nous permettra de voir à quelle vitesse l'eau s'infiltre dans la terre, mais aussi avoir une mesure d'humidité plus précise sachant que celle en surface a tendance à varier très rapidement.
- Une sonde DS18B20 qui mesure la température du sol
- Un capteur de température AHT10 mesurant la température et l'humidité de l'
- Une électrovanne 12V que nous allons relier au robinet d'arrosage du jardin de Valentin et qui sera piloté par le Raspberry. Cette vanne s'enclenchera/s'éteindra en fonction de l'humidité et de s'il fait jour ou non, mais sera aussi activable manuellement si besoin.
- Pour finir, plusieurs petits éléments comme des LEDs, des relais, des cartes de montage rapide, etc..

Le schéma du circuit est disponible en Annexe A - . Nous l'avons beaucoup utilisé une fois le boîtier installé pour que les liens entre le Raspberry pi et la programmation soient justes.

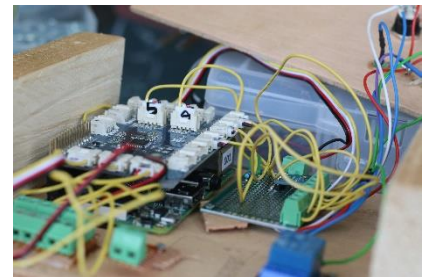
Nous avons ensuite monté le circuit et le boîtier ainsi que mis en place le tuyau et les capteurs. Pour ce faire, nous avons soudé et relié les différents éléments sur des petites plaques de montage rapide, que nous avons ensuite fixé à l'intérieur d'un boîtier en bois avec le Raspberry. Nous avons relié le boîtier aux capteurs et à la vanne comme le montre les images ci-dessous :



*Figure 3 - Electrovanne*



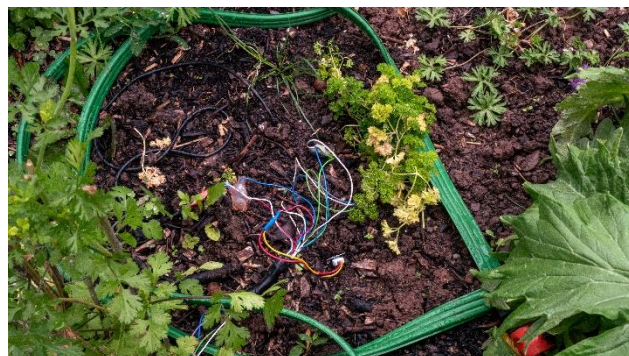
*Figure 2 - Boîtier de contrôle fermé*



*Figure 1 - Boîtier ouvert*



*Figure 5 - Capteur capacitif installé dans la terre*



*Figure 4 - Vue de l'arrosage par le haut*

## 3 Acquisition des données

### 3.1 Mesure d'humidité du sol

Nous avons utilisé deux types de capteurs : un capteur résistif (un modèle fourni par la HES et un modèle possédé par Valentin) et un capteur capacitif (possédé par Valentin).

#### 3.1.1 Fonctionnement des capteurs

Le capteur **résistif** mesure la résistance du sol, ainsi si le sol est humide, la résistance diminue et il y a plus de courant qui pourra circuler. Le principal défaut de ce capteur est que la résistance est mesurée entre deux électrodes qui vont se corroder au fil de son utilisation. Dans notre cas, deux semaines après le début des mesures, le capteur s'est retrouvé complètement corrodé et affichait continuellement une valeur aux alentours de 1024, indiquant une humidité minimale.

Le capteur **capacitif** mesure la permittivité du sol. Il agit comme un condensateur, sauf que le diélectrique. L'eau a une permittivité relative de 78,5 (Wikipedia.org, 2021), l'air d'environ 1 et le sol sec entre 8 et 14 (Baumer SAS, 2021). Ainsi plus sol sera humide, plus sa permittivité augmente.

#### 3.1.2 Compréhension des capteurs

Tous les capteurs sont branchés au Raspberry Pi sur les ports analogiques. Afin de déterminer la correspondance entre les valeurs analogiques acquises sur la Raspberry Pi (par le biais du convertisseur analogique-digital 10bits du GrovePi) et le taux d'humidité réel, nous avons pris deux références : la valeur retournée lorsque le capteur est sec et dans l'air, et la valeur lorsque le capteur est plongé dans l'eau. Pour les capteurs capacitifs, nous avons aussi mesuré lorsque le sol est détrempe pour nous assurer de la valeur retournée par le capteur.

Capteur	Abréviation utilisée	Air (sec)	Eau (humide)	Sol détrempe
Résistif <i>Grove Moisture Sensor 1.1</i>	Grove	0	600	-
Résistif fourni par Valentin	Chinois	1023	340	-
Capacitif <i>Capacitive Soil Moisture Sensor v1.2</i>	Cap1	550	295	292
	Cap2	485	292	315

Ces valeurs analogiques brutes n'étant pas utilisables, nous avons décidé de les convertir en pourcentage d'humidité relative du sol. 0% correspond à un sol totalement sec et 100% correspond à la valeur du capteur trempé dans l'eau.

Il existe deux cas de conversion : soit plus la terre est humide, plus la valeur analogique est élevée ; soit plus la terre est sèche, plus la valeur analogique est élevée.

1<sup>er</sup> cas (pour le cap1, cap2 et chinois) :

$$H_R = 1 - \frac{\text{input} - \text{hum}_{\max}}{\text{hum}_{\min} - \text{hum}_{\max}}$$

2<sup>ème</sup> cas (pour le grove) :

$$H_R = \frac{\text{input} - \text{hum}_{\min}}{\text{hum}_{\max} - \text{hum}_{\min}}$$



### 3.1.3 Problèmes rencontrés

#### Corrosion

Le principal problème des capteurs résistifs est la corrosion des deux électrodes. Les capteurs étant de qualité médiocre, ils sont fortement soumis à cette contrainte. Le capteur grove avait déjà été utilisé dans un projet précédent était déjà très corrodé. Nous avons décidé de ne pas l'utiliser pour les mesures.

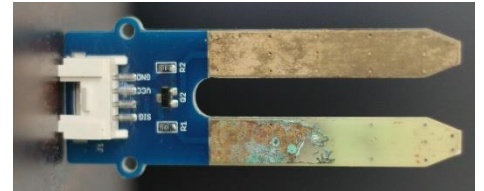


Figure 6 - Capteur Grove déjà corrodé

Le capteur chinois nous a aussi posé plusieurs problèmes : Il était difficile de comprendre les données qu'il transmettait. On peut observer sur la Figure 7 que la courbe verte, représentant le capteur chinois, ne donne pas une tendance lisse exploitable. On peut surtout remarquer qu'à partir du 25 mai, la courbe se stabilise vers 1024. Nous avons retiré le capteur de terre et il s'est avéré qu'il était complètement corrodé et inexploitable.

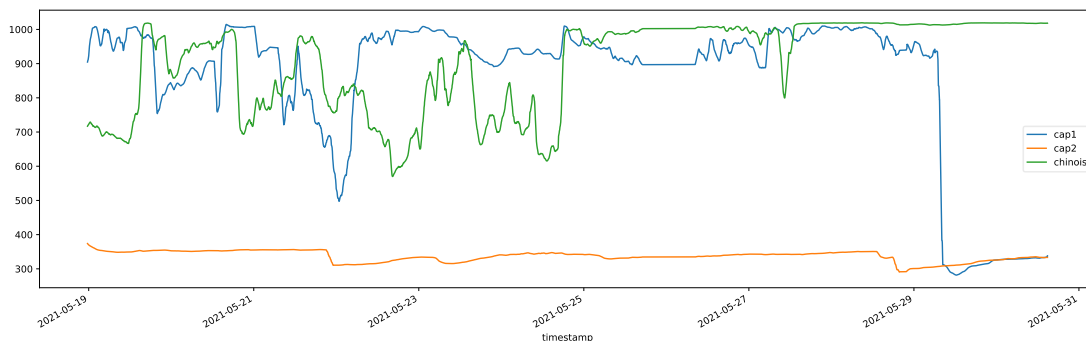


Figure 7 - Données analogiques des capteurs d'humidité du sol

#### Capacitif enterré

Le capteur cap1 (en bleu) était installé dans la terre à 30cm de profondeur. Ses mesures étaient, elles aussi difficile à exploiter. Nous avons décidé de le déterrer et de l'installer en surface, ses résultats ont rapidement été semblables à ceux du cap2 (en orange).

### 3.1.4 Traitement des données

Le convertisseur analogique/digital renvoie parfois des valeurs impossibles pour sa conception sur 10bits (plus grandes que 1023). Parfois il arrive aussi qu'une donnée isolée soit beaucoup plus grande ou petite que ce qu'elle devrait être.

Pour la création des graphiques d'analyse, nous avons supprimé toutes les données qui représentaient un écart de plus de 20 avec la suivante, ainsi que toutes les données plus grandes que 1024.

Nous avons ensuite effectué une moyenne mobile sur 60 données afin de supprimer les petites variations et avoir seulement la tendance de mesure.

Dans l'arrosage en fonctionnement, nous n'avons pas pu utiliser pandas pour effectuer cette moyenne mobile. Nous avons fait un tableau retenant les 20 dernières mesures. Les mesures sont prises toutes les deux minutes et à chaque nouvelle mesure, l'arrosage compare la moyenne de ces 20 mesures. Cela permet d'éviter que l'arrosage s'enclenche de courts instants à cause d'une petite variation de

l'humidité. Cela permet aussi d'éviter que l'arrosage s'éteigne tout de suite après s'être enclenché (les capteurs étant à la surface, l'humidité mesurée monte très rapidement).

### 3.2 Mesure d'humidité et de température de l'air

Un capteur (AHT10) mesurant la température et l'humidité de l'air a été installé dans le boîtier de commande. Il est connecté en I<sup>2</sup>C au Raspberry Pi. Nous avons utilisé la librairie AHT10 créé par Alexandr Lenetc<sup>1</sup>.

### 3.3 Mesure de la température du sol

Nous avons enterré un capteur DS18B20 à 5cm de profondeur dans la terre proche des mesures d'humidité.

### 3.4 Données météorologiques

Afin d'obtenir les prévisions météorologiques et en particulier les prévisions de pluie, nous avons utilisé le plan gratuit de l'API d'OpenWeatherMap. Les données ont été récupérées par la librairie pyOWM<sup>2</sup> permettant un lien facile entre l'API et Python.

L'api permet de récolter les prévisions de pluie 3h par 3h, la fonction les somme jour par jour, puis retourne soit un tableau des de la quantité de pluie annoncée jour par jour, soit la pluie annoncée pour le jour même et le lendemain, soit la quantité de pluie annoncée pour les 5 prochains jours.

## 4 Automatisation de l'arrosage

La plus grande partie du projet a constitué à automatiser l'arrosage. Nous avons opté pour une programmation orientée objet et modulable pour facilement ajouter/modifier des fonctionnalités.

L'arrosage est composé de trois modules :

- Une partie qui gère le matériel;
- Une partie qui récupère les données météo ;
- Une dernière partie qui effectue la communication avec la Dashboard.

### 4.1 Modes d'arrosage

L'arrosage peut être dans deux états différents : automatique ou forcé. L'utilisateur peut choisir activer les états avec les deux boutons situés sur l'interface de l'arrosage.

### 4.2 Conditions d'arrosage automatique

Lorsque l'arrosage est en mode automatique, plusieurs facteurs sont observés par l'arrosage pour déterminer s'il est nécessaire d'arroser le jardin ou non :

- L'humidité du sol ;
- L'heure de la journée ;
- La quantité de pluie annoncée pour le lendemain.

---

<sup>1</sup> <https://github.com/momoru-kun/AHT10>

<sup>2</sup> <https://github.com/csparpa/pyowm>

L'arrosage s'allumera quand :

- L'humidité est 66%
- Ce n'est pas la pleine journée (de 18h00 à 08h00)
- Il annonce moins de 10mm de pluie le jour suivant

Si l'humidité est à 25%, l'arrosage passera en mode critique : peu importe l'heure de la journée ou la pluie annoncée, l'irrigation des plantes se fera de toute manière.

L'arrosage est allumé, jusqu'à avoir un taux d'humidité de 80%, à cette valeur, l'arrosage s'éteint automatiquement.

### 4.3 Boutons

Deux boutons sont installés sur le boîtier de commande. Celui à droite avec le sigle on/off sert à allumer ou éteindre la minuterie de l'arrosage. Celui à gauche sert à forcer l'allumage de l'arrosage. Ils contiennent une LED intégrée qui est allumée en correspondance avec l'état de l'arrosage (à gauche si l'arrosage est en mode forcé, à droite si l'arrosage est en mode automatique).

Il a fallu souder deux résistance pull-up sur les boutons, sans cela les valeurs lues étaient aléatoires.

### 4.4 LEDs d'information

Deux leds sont disposées en dessus des boutons. La LED verte est allumée dès que le système de contrôle est alimenté. La LED bleue s'allume quand l'arrosage est enclenché

### 4.5 Electrovanne

L'électrovanne sert à contrôler l'arrivée de l'eau dans la platebande. Elle doit être alimentée en 12v CC pour s'ouvrir. Un relais a été utilisé pour la contrôler.

### 4.6 Dashboard

Afin de rendre l'arrosage le plus interactif possible, nous avons souhaité créer un site où il est possible de visualiser les données d'humidité, les températures mesurées et l'état de l'arrosage s'il est allumé ou non.

Nous avons utilisé la plateforme Ubidots STEM<sup>3</sup> qui permet de facilement créer une interface pour visualiser des données et interagir avec. Cependant, nous avons été confrontés à quelques limitations : la plateforme limite à 4 000 données envoyées par jour. Nous avons dû limiter l'envoi d'informations toutes les 3 minutes (6 données sont envoyées à chaque fois : l'humidité du sol, la température du sol et de l'air et des états si l'arrosage est allumé, est sur la minuterie ou est en mode « forcé »).

Le dashboard est accessible sous ce lien : <http://e2-watering.vsandoz.ch/>

---

<sup>3</sup> <https://github.com/ubidots/ubidots-python/> (API)  
<https://ubidots.com/> (Site)

## 5 Conclusion

Nous avons réussi à créer un arrosage qui s'active selon les précipitations prévues, l'humidité du sol et l'heure de la journée. Il est aussi possible de le contrôler manuellement et de visualiser son état sur une dashboard, ce qui permet une utilisation facile par l'utilisateur.

Pour rendre l'expérience encore plus agréable, il faudrait rendre l'arrosage contrôlable directement depuis la dashboard, ce qui n'a pas encore été implémenté.

Le projet a été effectué en printemps et il y a eu beaucoup de précipitations dans la période de test. Il est difficile de dire actuellement si les valeurs cibles d'humidités sont les valeurs correspondant à une irrigation adéquate. Des réglages devront être fait pendant l'été.

Il serait aussi intéressant d'étendre l'arrosage à d'autres platebandes. Cela nécessiterait d'installer des capteurs sans fils fonctionnant sur batterie et panneau solaire dans le jardin.

Le projet est disponible sur github à l'adresse : [https://github.com/lomys/E2\\_watering](https://github.com/lomys/E2_watering)



## Annexe A - Schématique de l'installation

