

Projets de Fablab Semestre 1 (2024/2025)

Étude d'un Sismomètre TC1 et Introduction à Arduino

Écrit par : CONE Irem Su



TABLE DES MATIERES

MISE EN PLACE D'UN SYSTEME AVEC ARDUINO ET MQTT	2
Introduction.....	2
Matériels.....	2
Matériel Utilisé.....	2
Méthodes.....	2
Résultats	2
Conclusion	3
ETUDE ET DECOUVERTE D'UN SISMOMETRE TC1.....	4
Introduction.....	4
Matériel et Méthodes.....	4
Matériel Utilisé.....	4
Méthodes.....	4
Résultats	6
Conclusion	8
Améliorations à Apporter.....	8
Sources	8

MISE EN PLACE D'UN SYSTEME AVEC ARDUINO ET MQTT

Introduction

Dans cette expérience, nous avons exploré l'utilisation d'une carte Arduino pour collecter et transmettre des données en temps réel via un broker MQTT. L'objectif principal était d'apprendre à manipuler un microcontrôleur Arduino, à comprendre son fonctionnement, et à mettre en œuvre une communication en temps réel en utilisant des outils comme Python et le protocole MQTT. Ces compétences sont essentielles pour créer des objets connectés et visualiser des données de manière efficace.

Matériels

Matériel Utilisé

Une carte Arduino (UNO).
Des câbles pour le montage.
Un ordinateur pour programmer et connecter l'Arduino.
Un logiciel Arduino IDE pour télécharger les programmes.
Un logiciel Python avec la bibliothèque MQTT (pymqttusb).
Un broker MQTT (mqtt.univ-cotedazur.fr).
Un logiciel de visualisation comme MQTT Explorer.

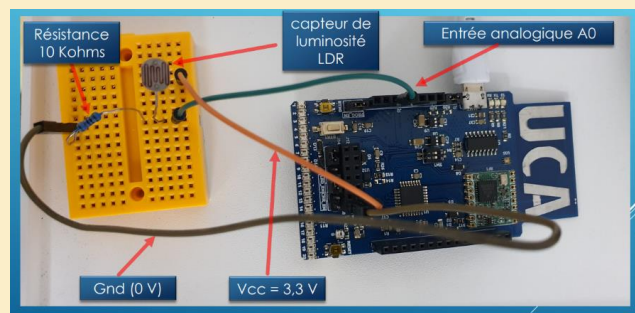


Figure 1 Carte Arduino avec capteur de luminosité

Méthodes

Connecter la carte Arduino à l'ordinateur via USB et réalisé un montage simple avec des câbles. Un programme pré écrit a été téléchargé sur la carte à l'aide de l'IDE Arduino.

Configurer un fichier JSON pour permettre à la carte Arduino de communiquer avec un serveur MQTT et manipuler les couler des LEDs à distance .

Collecter les données (intensité, couleur) via Python et visualisées en temps réel sur MQTT Explorer.

Résultats

Introduction à Arduino Nous avons commencé par nous familiariser avec la carte Arduino et ses différents composants, notamment les broches nécessaires pour connecter des fils et des LEDs. Après une introduction détaillée aux connexions et fonctionnalités de la carte, nous avons réalisé un montage avec des câbles pour relier l'Arduino à un ordinateur via un port USB. Ensuite, un programme préalablement écrit a été envoyé sur la carte Arduino à l'aide de l'IDE Arduino. Ce programme était conçu pour manipuler les cartes et les données de capteurs et nous a aussi aidé pour transmettre les résultats.

Pour vérifier si la carte fonctionnait correctement, nous avons observé les LEDs(Rouge,Vert,Bleu) intégrées : leur éclairage indiquait que le programme avait été téléchargé avec succès et que la carte fonctionnait.

Découverte du protocole MQTT Une fois la carte Arduino opérationnelle, nous avons étudié le fonctionnement d'un broker MQTT, un outil permettant de transmettre et de recevoir des messages entre des appareils connectés. À l'aide de Python, nous avons utilisé un fichier JSON préalablement configuré, contenant les informations nécessaires pour établir une communication avec le serveur MQTT (mqtt.univ-cotedazur.fr).

Pour surveiller et analyser les données, on a utilisé le logiciel MQTT Explorer, qui affiche en temps réel les messages envoyés par l'Arduino au serveur MQTT. Ce logiciel nous a permis de suivre les données collectées et de vérifier leur transmission correcte. Les commandes envoyées depuis l'IDE Arduino ont permis de nous contrôler précisément les couleurs de LEDs. Par exemple, nous avons changé la couleur de la LED RVB par le programme et observé des réactions immédiates dans les données qui sont affichées au serveur MQTT.

Avec Python, nous avons automatisé la collecte des données et envoyé des commandes à l'Arduino. Le script a également facilité la diffusion des informations en temps réel. Par ailleurs, les données des capteurs, comme l'intensité lumineuse mesurée par un capteur dédié, ont été aussi publiées avec succès au serveur MQTT. Tous ces données, affichées dans une salle de MQTT Explorer, ont confirmé que la communication entre les différents éléments du système fonctionnait correctement.

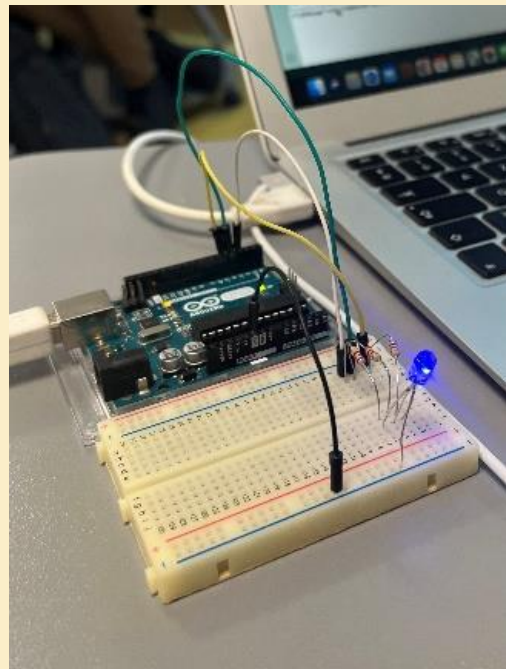


Figure 2 Carte Arduino avec LED RVB

Conclusion

Cette expérience nous a permis de développer des compétences pratiques en programmation, en manipulation de microcontrôleurs, et en communication via le protocole MQTT. Nous avons non seulement appris à configurer et tester un système Arduino, mais également à exploiter des outils pour collecter, visualiser, et transmettre des données en temps réel. Ces connaissances ouvrent la voie à des projets d'objets connectés et d'automatisation dans divers domaines.

ETUDE ET DECOUVERTE D'UN SISMOMETRE TC1

Introduction

Les sismomètres sont des instruments essentiels pour détecter les vibrations du sol causées par des phénomènes tels que les tremblements de terre. Ils jouent un rôle crucial dans la compréhension des processus géophysiques, notamment la propagation des ondes sismiques à travers les différentes couches de la Terre. Cette expérience vise à explorer les principes fondamentaux régissant le fonctionnement d'un sismomètre, en utilisant et en réparant le modèle TC1.

Le modèle TC1 est un appareil éducatif simple et peu coûteux, conçu pour enregistrer les ondes sismiques tout en permettant d'introduire des concepts de physique appliquée, tels que le mouvement harmonique, l'induction électromagnétique et l'amortissement magnétique. En outre, cette expérience inclut l'utilisation d'outils pour le traitement des données sismiques, tels que le logiciel Amaseis et des scripts Python développés pour recevoir, analyser et diffuser les données en temps réel via un broker MQTT.

L'objectif principal est d'associer théorie et pratique pour mieux appréhender les phénomènes géophysiques et les outils utilisés pour les étudier.

Matériel et Méthodes

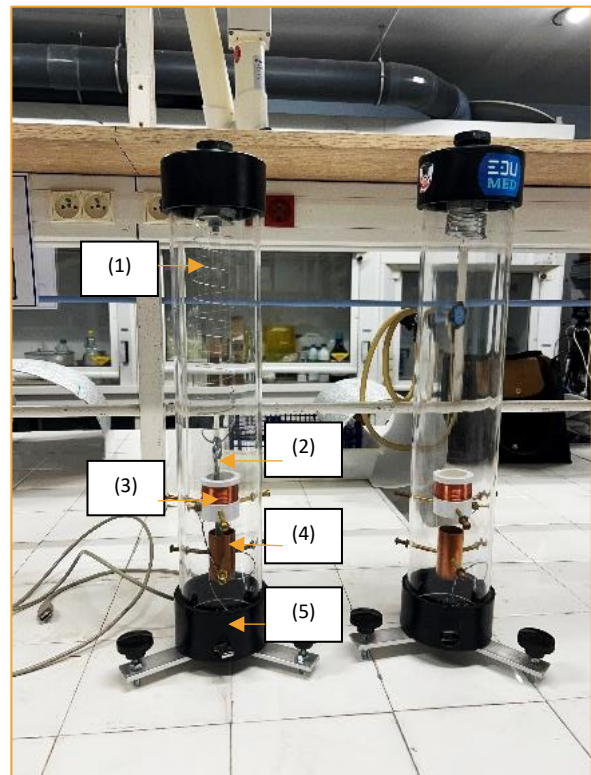
Matériel Utilisé

Un ressort (type Slinky). (1)
Un aimant en néodyme. (2)
Une bobine en cuivre (avec plusieurs spires de fil de cuivre fin). (3)
Un tube en cuivre pour l'amortissement magnétique. (4)
Une carte Arduino Uno.
Un convertisseur analogique-numérique (ADC).
Un logiciel de traitement des signaux (jAmaseis).
Un cadre en plastique transparent pour maintenir les composants.(5)

Méthodes

Le système Le ressort est fixé au cadre et supporte une masse (l'aimant en néodyme). Une bobine de cuivre est positionnée autour de l'aimant pour détecter son mouvement relatif. Un deuxième aimant suspendu dans un tube de cuivre assure l'amortissement magnétique en générant des courants de Foucault lorsque l'oscillateur est perturbé.

Acquisition des données Les signaux générés par la bobine sont amplifiés et convertis en numérique à l'aide de la carte Arduino. Ces données sont ensuite filtrées numériquement pour éliminer les bruits et améliorer la qualité du signal.



Analyse des données Les données sont visualisées sur un logiciel comme Amaseis, permettant d'observer les ondes sismiques et de les analyser.

Cadre Théorique

Oscillateur harmonique amorti

Le mouvement de la masse suspendue est décrit par l'équation du mouvement harmonique amorti :

$$m\ddot{x} + b\dot{x} + kx = 0 \quad (1)$$

où m est la masse de l'aimant, b est le coefficient d'amortissement, k est la constante de raideur du ressort, x est le déplacement. Cette équation montre comment les forces de rappel et d'amortissement influencent le mouvement.

Induction électromagnétique

Quand un aimant se déplace à l'intérieur d'une bobine, une force électromotrice (fem) est induite selon la loi de Faraday :

$$\mathcal{E} = -N \frac{d\Phi_B}{dt} \quad (2)$$

où N est le nombre de spires de la bobine, Φ_B est le flux magnétique à travers la bobine, donné par :

$$\Phi_B = B \cdot A \cdot \cos \theta \quad (3)$$

avec B le champ magnétique, A la surface de la bobine, et θ l'angle entre le champ et la bobine.

Courants de Foucault et amortissement magnétique

L'amortissement magnétique est obtenu grâce aux courants de Foucault générés dans le tube en cuivre par le mouvement de l'aimant secondaire. Ces courants produisent une force opposée proportionnelle à la vitesse :

$$F_{\text{eddy}} = -k_{\text{eddy}}v \quad (4)$$

où k_{eddy} est une constante dépendant des propriétés du cuivre.

Propagation des ondes sismiques

Les ondes sismiques enregistrées obéissent à l'équation de la vitesse des ondes :

$$v = \sqrt{\frac{E}{\rho}} \quad (5)$$

où v est la vitesse des ondes, E le module élastique du milieu, ρ la densité.

Les temps d'arrivée des ondes primaires (P) et secondaires (S) permettent de calculer la distance à l'épicentre :

$$t = \frac{d}{v} \quad (6)$$

où t est le temps de parcours et d la distance.

Principe de Fonctionnement

Le sismomètre TC1 fonctionne en utilisant un **oscillateur harmonique amorti (1)** pour détecter les mouvements du sol causés par les ondes sismiques. L'appareil se compose d'un ressort et d'une masse (un aimant) suspendue dans un cadre en plastique. Lorsque le sol bouge, le cadre suit ce mouvement, mais la masse reste en place à cause de son inertie. Cela entraîne un déplacement relatif entre l'aimant et une bobine de cuivre fixée au cadre.

Ce déplacement génère une **force électromotrice induite (2)** dans la bobine, selon la loi de Faraday. Pour éviter les oscillations prolongées, le système est équipé d'un **amortissement magnétique (4)**. Un deuxième aimant suspendu dans un tube de cuivre génère des courants de Foucault lorsque l'oscillateur bouge, ce qui produit une force opposée proportionnelle à la vitesse de l'aimant, réduisant ainsi les vibrations. Les signaux produits par la bobine sont ensuite amplifiés et convertis en numérique à l'aide d'un convertisseur analogique-numérique (ADC). Ces données sont traitées avec des filtres numériques pour réduire le bruit et améliorer la qualité du signal. Enfin, le signal est affiché sous forme de sismogramme dans un programme, permettant d'analyser les ondes sismiques enregistrées.

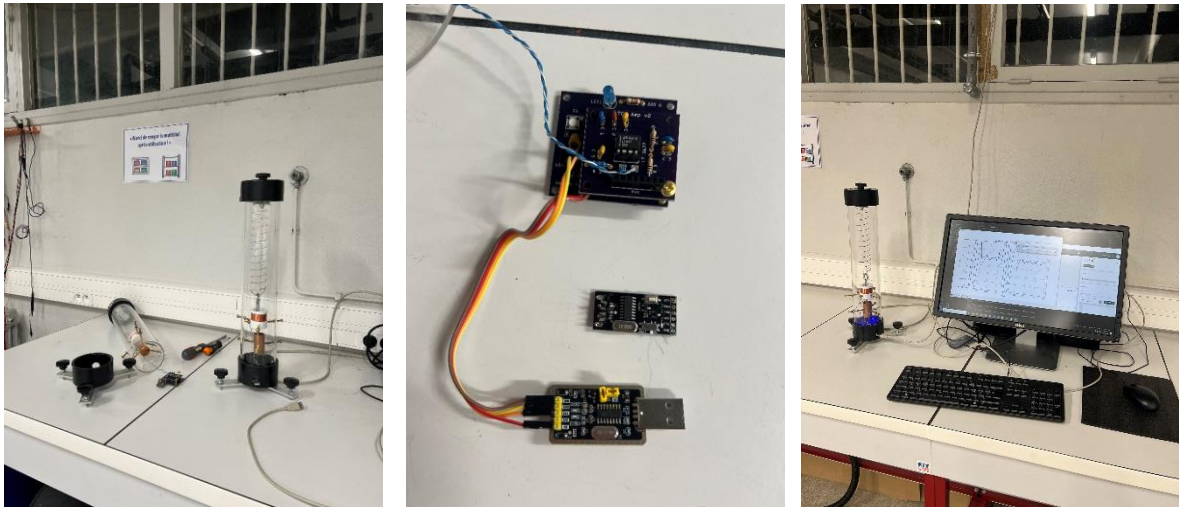
Résultats

Lors de cette étude, on avait à disposition deux sismomètres TC1 : l'un fonctionnel et l'autre défectueux. En comparant les deux appareils, on a identifié que le port connecteur USB du sismomètre défectueux était cassé. Cette panne empêchait la transmission des données et le fonctionnement de l'appareil. Le problème a été résolu en remplaçant le port USB, connecté à la carte Arduino. Cette réparation, qui nécessite une grande précision en raison de la fragilité des connexions, a été réalisée avec délicatement et a permis de rétablir le bon fonctionnement du sismomètre !

Une fois réparés, les deux sismomètres ont été testés pour confirmer leur bon état. Lors des tests, on a observé que deux LEDs sur la carte Arduino recevant les données étaient allumées, ce qui nous a permis de nous assurer que les appareils fonctionnaient correctement. Les signaux ont été collectés premièrement à l'aide du logiciel Amaseis, permettant d'observer les données sismiques enregistrées par les appareils. Par la suite, on a utilisé un code Python développé par notre enseignant M. Orlandi, pour recevoir et traiter les données de manière plus flexible et automatisée.

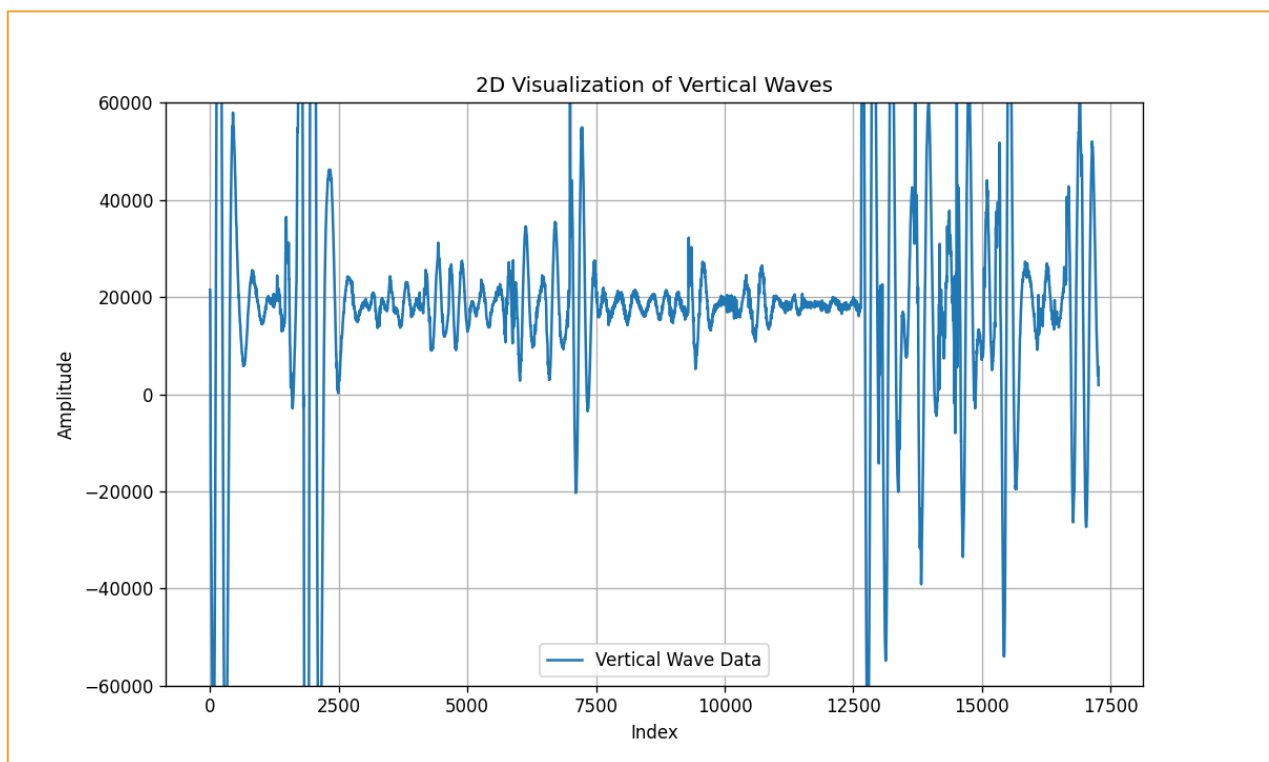
Ce code a ensuite été amélioré pour permettre la publication des données en temps réel via un broker MQTT, rendant les informations accessibles sur différentes plateformes. Cela a permis d'ajouter une dimension dynamique au projet, avec la possibilité de surveiller et de partager les enregistrements en

direct. En résumé, cette expérience a démontré l'efficacité des réparations effectuées et la capacité des sismomètres TC1 à collecter, traiter et diffuser des données de manière fiable.



Le graphique ci-dessous présente un sismogramme, une visualisation des ondes verticales enregistrées par le sismomètre TC1. L'axe horizontal (index) représente le temps en secondes, de 0 à environ 17 500 secondes, tandis que l'axe vertical montre l'amplitude des ondes sismiques, avec des valeurs comprises entre -60 000 et +60 000, représentant l'intensité des mouvements du sol.

Les pics importants, comme ceux observés autour de 2 500 secondes ou 12 500 secondes, indiquent des perturbations significatives, potentiellement dues à des vibrations extérieures. Par exemple, à 2 500 secondes, l'amplitude atteint environ +40 000, ce qui indique un mouvement vertical marqué du sol à ce moment précis.



Ce sismogramme met en évidence la capacité du sismomètre à détecter avec précision les variations des vibrations verticales au fil du temps

Conclusion

Cette expérience nous a permis de découvrir le fonctionnement d'un sismomètre TC1 et d'explorer les principes liés à la création d'un sismogramme. Elle a offert d'acquérir des compétences pratiques en identifiant et en réparant un problème courant lié au port USB de l'appareil. Grâce à Amaseis et au codage en Python, nous avons appris à visualiser et diffuser des données sismiques en temps réel. Le projet nous a initiés aux bases de la programmation, à l'utilisation d'un broker MQTT, et à l'analyse des phénomènes sismiques.

En résumé, cette étude a renforcé nos connaissances en électronique, en programmation et en physique appliquée, tout en nous offrant une compréhension pratique des sismomètres et de leur rôle dans l'étude des tremblements de terre. Cette intervention a permis non seulement de remettre en état le sismomètre, mais aussi de mieux comprendre les aspects techniques de ces équipements.

Améliorations à Apporter

On a commencé à créer un site web pour afficher les données, disponible sur [github/iremsucone](https://github.com/iremsucone). Cependant, ce website n'est pas encore terminé : il n'a pas été publié sur Internet et la réception des données MQTT n'a pas encore été intégrée au code HTML.

De plus, une interprétation plus approfondie du sismogramme est nécessaire afin de mieux comprendre le fonctionnement détaillé de la machine. Enfin, il serait intéressant de tester le sismomètre sur une période prolongée pour évaluer sa fiabilité et son efficacité dans des conditions d'utilisation à long terme.

Sources

https://www.researchgate.net/publication/256422299_Teaching_Geophysics_with_a_Vertical-Component_Seismometer