

CY IUT – GEII Neuville

# Procédure d'installation et de mise en service

Projet « Drumpad »

Instrument de percussion numérique automatisé

Document rédigé par : « VRONSKYI Volodymyr »  
Version 1.1 – 11/02/2025

## Avant-propos

Le cahier des charges (CDC) du projet est rédigé par l'équipe projet sous la responsabilité du chef de projet. Ce document est à rédiger pendant la phase d'avant-projet pour décrire les objectifs du projet (le Quoi ?) et la façon de les mener à bien (le Comment ? et le Qui ?). Pour ce second point, un dossier organisationnel peut être adossé au CDC en phase de planification.

Plus précisément, le CDC est la traduction, sous forme écrite et claire, des besoins du client en termes de fonctions, de services, et de contraintes. S'il est rédigé par l'équipe exécutant le projet, cette dernière peut s'appuyer sur la fiche projet (dite parfois de pré-instruction) du client. Le CDC peut soit imposer une solution technique détaillée, soit décrire uniquement les fonctions souhaitées en laissant le choix de la solution à adopter.

Le CDC se décompose en plusieurs parties : le contexte du projet, l'analyse du besoin du client, le cahier des charges fonctionnel. Il peut être complété d'éléments détaillant les facteurs de risque et de succès du projet, ainsi que le budget le cas échéant.

## Table des matières

Avant-propos.....	1
Table des matières .....	<b>Error! Bookmark not defined.</b>
Introduction .....	4
1. Contexte du projet .....	4
1.1. Situation et description.....	4
1.2. Enjeux.....	<b>Error! Bookmark not defined.</b>
Enjeux techniques .....	<b>Error! Bookmark not defined.</b>
Enjeux pédagogiques.....	<b>Error! Bookmark not defined.</b>
Enjeux promotionnels .....	<b>Error! Bookmark not defined.</b>
Enjeux environnementaux .....	<b>Error! Bookmark not defined.</b>
Enjeux de collaboration.....	<b>Error! Bookmark not defined.</b>
1.3. Études déjà effectuées OU sur des sujets voisins ET suites prévues .....	<b>Error! Bookmark not defined.</b>
Projets antérieurs .....	<b>Error! Bookmark not defined.</b>
Projets en parallèle.....	<b>Error! Bookmark not defined.</b>
Améliorations et suites prévues .....	<b>Error! Bookmark not defined.</b>

1.4. Objectifs du projet .....	5
Objectifs techniques .....	Error! Bookmark not defined.
Objectifs pédagogiques .....	Error! Bookmark not defined.
Objectifs de livraison .....	Error! Bookmark not defined.
Objectifs de promotion .....	Error! Bookmark not defined.
1.5. Nature des prestations demandées .....	Error! Bookmark not defined.
Conception physique et électronique .....	Error! Bookmark not defined.
Réalisation d'un prototype fonctionnel .....	Error! Bookmark not defined.
Élaboration et réalisation des tests .....	Error! Bookmark not defined.
Maintenance du système .....	Error! Bookmark not defined.
Documentation complète du projet.....	Error! Bookmark not defined.
Présentation et démonstration .....	Error! Bookmark not defined.
1.6. Caractère de confidentialité.....	Error! Bookmark not defined.
2. Énoncé du besoin .....	6
3. Contraintes.....	Error! Bookmark not defined.
Contraintes techniques .....	Error! Bookmark not defined.
Contraintes environnementales.....	Error! Bookmark not defined.
Contraintes budgétaires .....	Error! Bookmark not defined.
Contraintes réglementaires.....	Error! Bookmark not defined.
Contraintes temporelles.....	Error! Bookmark not defined.
4. Description fonctionnelle technique.....	Error! Bookmark not defined.
4.1. Description fonctionnelle.....	Error! Bookmark not defined.
Fonctions principales (FP) .....	Error! Bookmark not defined.
Fonctions techniques (FT) .....	Error! Bookmark not defined.
Fonctions de contrainte (FC) .....	Error! Bookmark not defined.
Diagramme des interacteurs.....	Error! Bookmark not defined.
Diagramme FAST (Functional Analysis System Technique).....	Error! Bookmark not defined.
4.2. Description technique .....	Error! Bookmark not defined.
Composants matériels.....	Error! Bookmark not defined.
Logiciel et firmware .....	Error! Bookmark not defined.
Schéma structurel global.....	Error! Bookmark not defined.

5. Lien inter-projets.....	<b>Error! Bookmark not defined.</b>
Interactions techniques.....	<b>Error! Bookmark not defined.</b>
Impact des retards.....	<b>Error! Bookmark not defined.</b>
6. Budget du projet .....	<b>Error! Bookmark not defined.</b>
Budget maximal autorisé.....	<b>Error! Bookmark not defined.</b>
Stratégies d’optimisation des coûts .....	<b>Error! Bookmark not defined.</b>
Justification des achats.....	<b>Error! Bookmark not defined.</b>

## Introduction

Ce document présente le cahier des charges du projet « Drumpad », un instrument de percussion numérique automatisé, développé dans le cadre de la SAE de 3<sup>ème</sup> année du BUT GEII (Génie Électrique et Informatique Industrielle) à l'IUT de Cergy-Pontoise. Ce projet a pour objectif de concevoir et réaliser un système de percussion électronique capable de jouer des sons mélodiques de manière manuelle, semi-automatique et automatique, tout en respectant des contraintes techniques, économiques et environnementales. Tout au long de ce document, le système, qui est l'objet principal de ce projet, est référencé « Drumpad ».

Le Drumpad s'inscrit dans le cadre des Journées Portes Ouvertes de l'IUT, où il servira de démonstrateur pour illustrer les compétences des étudiants en conception de systèmes électroniques complexes. Ce cahier des charges décrit les besoins du client, les fonctions techniques du système, ainsi que les contraintes liées au projet.

Ce document s'adresse principalement à l'équipe projet, aux enseignants encadrants, et au client représenté par l'IUT de Cergy-Pontoise, mais il pourra également être consulté par les parties extérieures au projet, étant donné que ce projet est réalisé dans le but éducatif, et tous les documents relatifs à ce projet seront disponibles en libre accès dans le répertoire du projet, cité dans les références. Le cahier des charges servira de référence tout au long du projet pour garantir la conformité des livrables aux attentes initiales.

## 1. Contexte du projet

### 1.1. Situation et description

Le projet « Drumpad » consiste à concevoir et réaliser un instrument de percussion numérique automatisé, capable de fonctionner en trois modes distincts : manuel, semi-automatique et automatique. Ce système s'inspire de l'Orchestrion, un dispositif permettant de reproduire de la musique de manière mécanique, mais en y intégrant des technologies modernes issues de l'électronique et de l'informatique industrielle.

Le Drumpad sera utilisé lors des Journées Portes Ouvertes de l'IUT de Cergy-Pontoise pour démontrer les compétences des étudiants en GEII. Il met en avant des concepts clés tels que la gestion des entrées/sorties (I/O), le traitement du signal audio, la programmation, et l'intégration de systèmes embarqués.

L'instrument sera composé de 16 touches configurables, permettant de jouer des sons sur une plage d'une octave complète. Il sera équipé d'un écran pour l'interface utilisateur, de haut-parleurs pour la reproduction sonore, et d'un microcontrôleur pour la gestion des fonctionnalités. Le système devra également être capable de fonctionner de manière autonome en énergie et, si possible, de se synchroniser avec une horloge externe pour une utilisation en groupe avec d'autres instruments.

## 1.2. Objectifs du projet

Le projet « Drumpad » a pour objectif principal de concevoir et réaliser un instrument de percussion numérique automatisé, capable de répondre aux attentes techniques et pédagogiques de l'IUT de Cergy-Pontoise.

Le Drumpad doit répondre aux attentes techniques du projet, pour ce faire il faut :

- **Concevoir un système polyvalent** : Le Drumpad doit fonctionner en trois modes distincts :
  - **Mode manuel** : Permettre à l'utilisateur de jouer des sons en temps réel via l'appui sur les touches.
  - **Mode semi-automatique** : Reproduire des séquences de sons préenregistrées.
  - **Mode automatique** : Jouer des sons en fonction de commandes externes (ex. synchronisation avec une horloge).
- **Garantir la qualité sonore** : Le système doit respecter des critères de justesse (fréquence des notes).
- **Garantir la rapidité de réponse** : Le système doit respecter des critères de temps de réponse (le temps entre l'action d'utilisateur et le retour du système).
- **Assurer l'autonomie énergétique** : Le Drumpad doit pouvoir fonctionner sur batterie et être alimenté par le secteur.

## 2. Énoncé du besoin

Le projet « Drumpad » répond à un besoin clair et précis exprimé par le client, représenté par l'IUT de Cergy-Pontoise. Ce besoin s'articule autour de la conception et de la réalisation d'un instrument automatisé, qui doit répondre aux contraintes définies par le client, et qui est destiné à être utilisé lors des Journées Portes Ouvertes de l'IUT.

Le besoin principal est de disposer d'un système capable de reproduire des sons de manière précise, couvrant une plage d'une octave complète<sup>1</sup> avec les demi-tons. Il doit fonctionner en 3 modes :

- **Mode manuel** : Permettre à l'utilisateur de jouer des sons en temps réel via l'appui sur des touches.
- **Mode semi-automatique** : Reproduire des séquences de sons préenregistrées.
- **Mode automatique** : Jouer des sons en fonction de commandes externes (ex. synchronisation avec une horloge ou d'autres instruments).

Le système doit être facile d'utilisation et comprendre une interface intuitive pour permettre la configuration.

De point de vue technique, le Drumpad doit répondre à plusieurs exigences :

- **Justesse des notes** : Les sons produits doivent être précis en termes de fréquence, avec une tolérance minimale d'erreur.
- **Faible latence** : Le temps de réponse entre l'appui sur une touche et la production du son doit être minimal pour une expérience utilisateur fluide.
- **Autonomie énergétique** : Le système doit pouvoir fonctionner sur batterie et être alimenté par le secteur, avec une autonomie suffisante pour les démonstrations.
- **Sécurité et CEM** : Le système doit respecter les normes de sécurité électrique, et de compatibilité électromagnétique (CEM) pour assurer une utilisation sécurisée, ainsi qu'éviter les interférences avec d'autres appareils.

Le Drumpad doit également disposer de :

- **Interface visuelle** : Un écran LCD doit afficher les informations nécessaires (ex. mode de fonctionnement, réglages).
- **Stockage de sons** : Le système doit pouvoir stocker des sons dans une mémoire interne ou externe pour les modes semi-automatique et automatique.
- **Synchronisation externe** : Le Drumpad doit pouvoir se synchroniser avec une horloge externe ou d'autres instruments pour une utilisation en groupe.

---

<sup>1</sup> Do, Do#, Ré, Ré#, Mi, Mi#, Fa, Sol, Sol#, La, La#, Si.

## 3. Mise en œuvre électrique

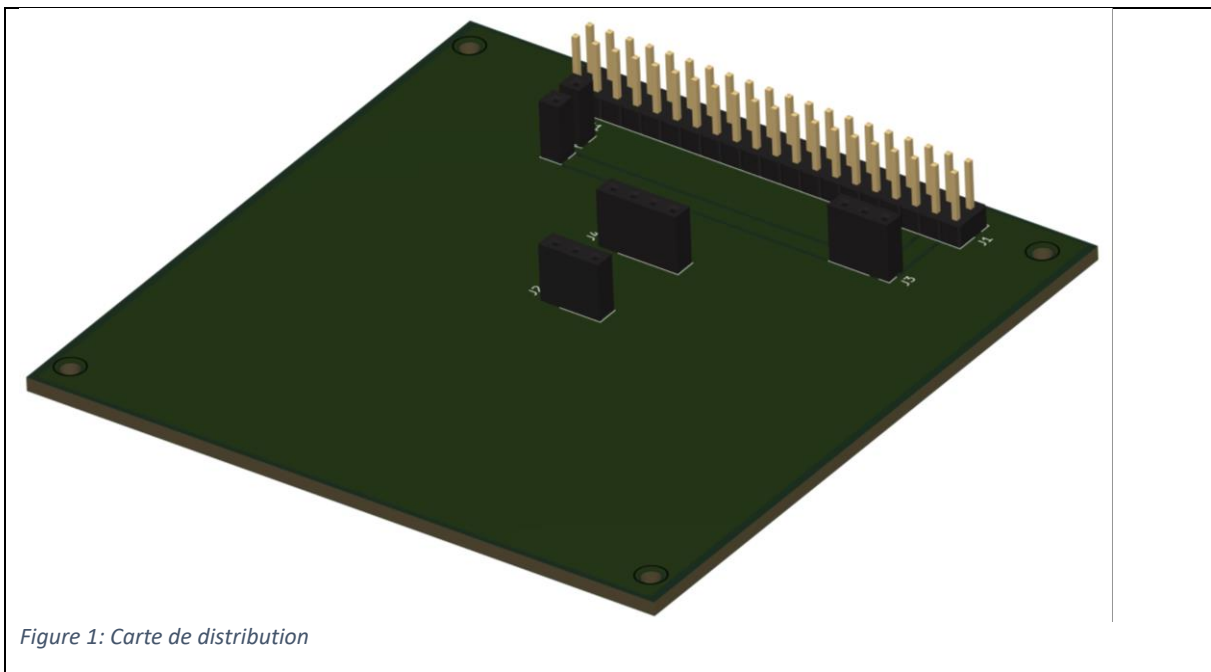
### 3.1. Précautions

- Couper l'alimentation avant tout branchement / débranchement.
- Vérifier le sens de branchement des modules avant la mise en tension.
- Vérifier la nappe de la Raspberry Pi 4, les deux côtes doivent être orientés dans le même sens, sans que la nappe tourne.
- Utiliser des câbles de section adaptés à 5A pour éviter tout dégât.
- Vérifier la présence de dissipateurs de chaleur dans tous les emplacements prévus.

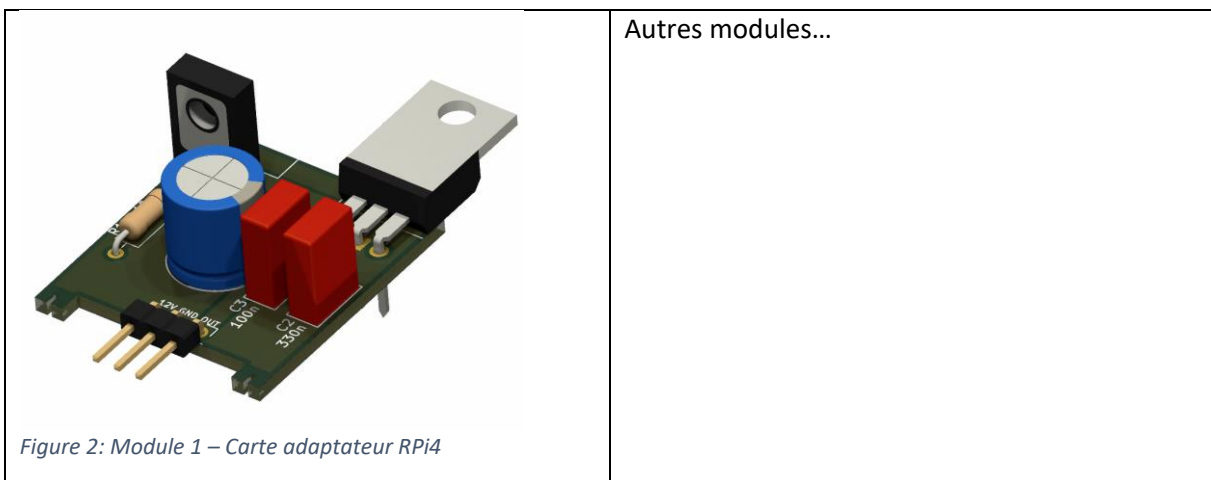
### 3.2. Installation et câblage

Il est conseillé de brancher les modules sur la carte de distribution avant de l'installer dans le boîtier, commencer donc par prendre la carte de distribution et les cartes modules.

- Prendre la carte de distribution, représentée sur la Figure 1 ci-dessous :



- Brancher les modules, représentés sur les Figure 2, Figure x, ... ci-dessous, dans les emplacements prévus sur la carte de distribution.



- Installer la carte de distribution dans le boîtier et la visser avec les vis prévues pour cet effet.



- Prendre la Raspberry Pi 4, l'installer dans le boîtier et la visser.
- Installer le Jack DC dans l'emplacement prévu, et connecter les bornes positive et négative dans les sockets correspondants sur la carte de distribution.
- Installer la batterie dans l'emplacement prévu et connecter les bornes positive et négative dans les sockets correspondants sur la carte de distribution.
- Installer et visser les haut-parleurs.
- Installer et visser les ventilateurs pour avoir le flux d'air vers l'intérieur du boîtier.
- Connecter la Raspberry Pi 4 à la carte de distribution avec une nappe prévue à cet effet.
- Installer la matrice de boutons dans l'emplacement prévu.
- Installer l'écran dans le couvercle et s'assurer qu'il soit bien vissé.
- Fermer le couvercle et le visser au boîtier.

### 3.3. Mise en route des sous-systèmes

Les sous-systèmes sont alimentés simultanément avec l'intégralité du système, ils n'ont donc pas besoin d'être mis en route séparément.

## 4. Mise en œuvre informatique

Cette section regroupe l'ensemble des étapes nécessaires pour installer, configurer et exécuter un programme sur une Raspberry PI dans le cadre de notre Drumpad. Elle comprend l'installation des outils de développement, la gestion des bibliothèques requises, la compilation du code source, ainsi que le déploiement et l'exécution du programme.

### 4.1. Installation et configuration de la Raspberry Pi

Avant de commencer, assurez-vous d'avoir le matériel suivant :

- **Raspberry Pi 4** (ou modèle compatible)
- **Carte microSD (minimum 16 Go, Classe 10 recommandée)**
- **Alimentation 5V/3A pour la Raspberry Pi**
- **Câble HDMI et moniteur (optionnel, mais utile pour la configuration initiale)**
- **Clavier et souris USB**
- **Connexion Internet (Wi-Fi ou Ethernet)**

### 4.2. Installation de Raspberry Pi OS

Voici un tutoriel d'installation de l'OS Raspberry sur la carte microSD :

1. Téléchargez **Raspberry Pi Imager** depuis le site officiel.
2. Insérez la carte microSD dans votre ordinateur.
3. Lancez Raspberry Pi Imager et sélectionnez :

**Système d'exploitation** : Raspberry Pi OS (32-bit recommandé).

**Stockage** : Choisissez votre carte microSD.

**Paramètres avancés** (optionnel) : Configurez le Wi-Fi, activez SSH.

4. Cliquez sur "Écrire" et attendez la fin de l'installation.

### 4.3. Premier Démarrage

1. Insérez la carte microSD dans la Raspberry Pi.
2. Branchez l'alimentation pour allumer la Raspberry Pi.
3. Configurez les paramètres régionaux et connectez-vous au Wi-Fi.
4. Ouvrez un terminal et mettez à jour le système :

```
sudo apt update && sudo apt upgrade -y
```

### 4.4. Activation des Interfaces

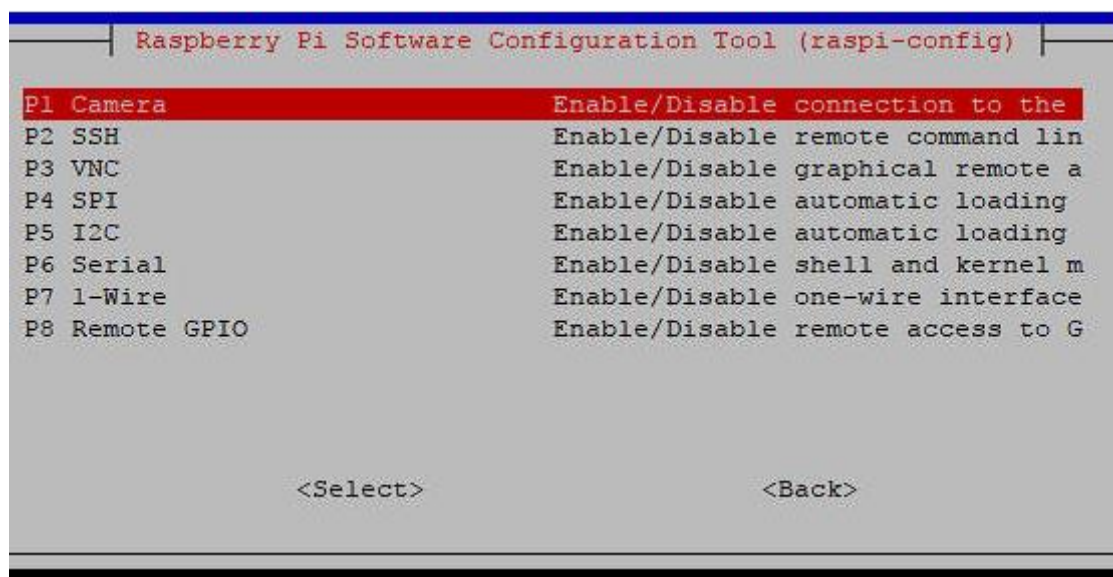
Nous devons activer certaines interfaces utiles pour notre projet.

1. Ouvrez la configuration Raspberry Pi :

```
sudo raspi-config
```

2. Activez les options suivantes :

- o **Interfacing Options** → Activer I2S (si un DAC externe est utilisé).
- o **Interfacing Options** → Activer SPI et I2C (si besoin d'un ADC externe).



3. Redémarrez la Raspberry Pi :

`sudo reboot`

Une fois la **Raspberry Pi configurée**, il est essentiel d'installer les bibliothèques requises pour contrôler les ports **GPIO** et assurer le bon fonctionnement du programme.

#### 4.5. Installation des bibliothèques via le terminal

Ouvrez un **terminal** et exécutez les commandes suivantes :

1. Mettre à jour la liste des paquets

```
sudo apt update && sudo apt install -y gcc
```

2. Installer FFmpeg (pour l'encodage et le traitement audio/video)

```
sudo apt install -y ffmpeg
```

3. Installer WiringPi (pour le contrôle des GPIO)

```
sudo apt install -y wiringpi
```

4. Installer libsndfile (pour la manipulation de fichiers audios)

```
sudo apt install -y libsndfile1-dev
```

#### 4.6. Vérification de l'installation des bibliothèques (étape optionnelle)

Après l'installation, vous pouvez vérifier si les bibliothèques sont bien installées avec :

5. Vérifier la version de WiringPi avec la commande suivante :

```
gpio -v
```

6. Vérifier la présence de FFmpeg avec la commande suivante :

```
ffmpeg -version
```

7. Vérifier la présence de libsndfile avec la commande suivante :

```
pkg-config --modversion sndfile
```

Si une erreur survient, assurez-vous que votre Raspberry Pi est bien connectée à Internet et que vous avez les **droits administrateur** (sudo).

## 4.7. Chargement des programmes et bibliothèques

Pour charger et exécuter un programme en langage C sur la Raspberry Pi, suivez les étapes ci-dessous

1. Installation d'un éditeur de code
  - a. Installez un éditeur de texte adapté au développement, tel que **Visual Studio Code (VSCode)**, **Nano**, ou **Vim**.
  - b. Assurez-vous que votre fichier source est bien enregistré avec l'extension **.c** (exemple : programme.c).
2. Compilation et exécution du programme en C
  - a. Contrairement aux systèmes où un compilateur externe est nécessaire, la Raspberry Pi dispose généralement du compilateur **GCC** (GNU Compiler Collection), permettant de compiler et d'exécuter le programme directement depuis le terminal.

## 4.8. Commandes Bash pour compiler et exécuter un fichier .c sur Raspberry Pi :

1. Se déplacer dans le répertoire contenant le fichier source :

```
cd desktop/SAE/code
```

2. Compiler le fichier source avec GCC avec la commande suivante :

```
gcc -o main main.c -wiringPi -lsndfile
```

3. Vérifier l'exécutable a bien été généré avec la commande suivante :

```
ls -l main
```

4. Exécuter le programme compilé avec la commande suivante :

```
sudo ./main
```

⚠ Assurez-vous que votre Raspberry Pi a bien **GCC** installé. Si ce n'est pas le cas, installez-le avec la commande :

```
sudo apt update && sudo apt install -y gcc
```

## 5. Mise en œuvre globale

### 5.1. Checklist de mise en œuvre

Avant la mise en service, vérifier les points suivants.

Electronique :

Vérifier les branchements de :

### Projet Drumpad

- La batterie et son niveau de charge
- La Raspberry
- L'écran
- L'amplificateur audio
- La carte d'alimentation de la Raspberry
- La carte d'alimentation générale
- La carte de détection d'appui
- Les ventilateurs
- Port Jack DC

### Informatique

Vérifier les l'installations des éléments suivants :

- OS Raspberry
- Compilateur
- Librairie FFMPEG
- Librairie WiringPi
- Activation des interfaces de communication

### 5.2. Notice d'utilisation

Pour utiliser le DrumpPass :

- Allumer le système
- Dans un terminale lancer les commandes suivantes :
  - `cd desktop/SAE/code`
  - `Gcc -o wavesound wavesound.c -lwiringPi -lsndfile`
  - `Sudo ./wavesound`
- Frapper dans le capteur