



**ÉCOLE UNIVERSITAIRE
DE PHYSIQUE ET D'INGÉNIERIE**
Université Clermont Auvergne



UNIVERSITÉ CLERMONT AUVERGNE

École Universitaire de Physique et d'Ingénierie (EUPI)

Association RobotiClermont

Afficheur à Palettes

Dans le cadre des activités du club RobotiClermont

Présentation générale



Une idée de :

M. Wilson DUGUET

Organisation :

M. Yannis LOUMOUAMOU
M. Mohamed Dhia OUNALLY

Février 2025



Table des Matières

1	Introduction	3
2	Objectif du Projet	4
3	Version 1 : Prototype / Proof of Concept	4
3.1	Description	4
3.2	Fonctionnalités	4
3.3	Matériel nécessaire	4
3.4	Phases de développement	5
4	Version 2 : Afficheur Modulaire Évolué	5
4.1	Description	5
4.2	Fonctionnalités	5
4.3	Matériel nécessaire	6
4.4	Phases de développement	6
5	Contraintes Techniques	6
6	Livrables	6
7	Planning Prévisionnel	7
8	Équipe et Rôles	7
9	Inspiration : Projets en ligne	7

1 Introduction

L'**afficheur à palettes** est un projet proposé par le club de robotique **RobotiClermont**, visant à explorer divers domaines de la mécatronique : conception mécanique, électronique de pilotage et programmation. Le principe de fonctionnement repose sur des palettes articulées qui permettent d'afficher des caractères (lettres, chiffres, symboles) sous une forme *mécanique* et *dynamique*.



FIGURE 1 – Exemple de panneau d'affichage à palettes

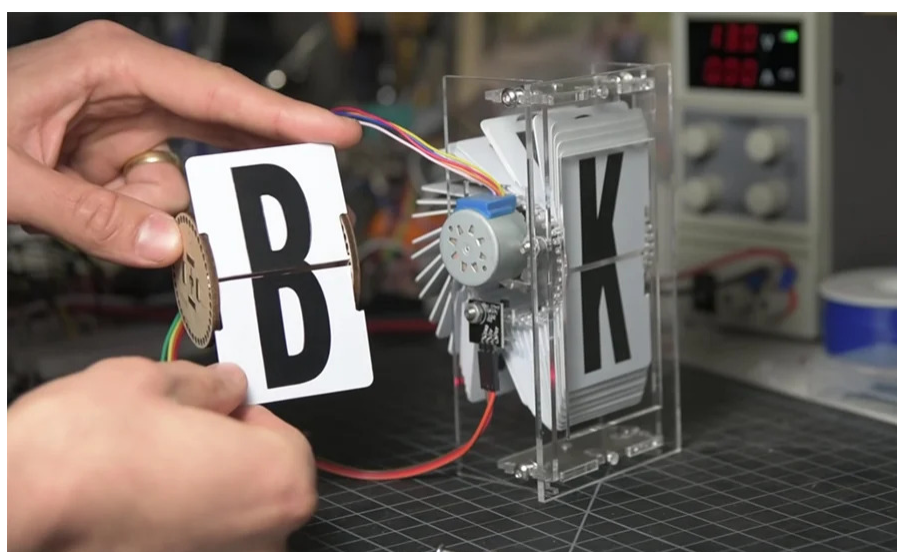


FIGURE 2 – Afficheur à palettes simple



2 Objectif du Projet

L'objectif principal est de **concevoir et réaliser** un bloc d'afficheur mécanique capable d'afficher divers caractères (26 lettres de l'alphabet, chiffres, symboles, motifs). Le tout sera **modulaire**, avec une structure imprimée en 3D et des composants électroniques standards pour :

- Créer un afficheur **compact et modulaire**.
- Afficher **lettres, chiffres, caractères spéciaux** et motifs.
- Exploiter des **composants accessibles** et une structure mécanique imprimée en 3D.
- Rendre le système **extensible** (hot-swappable) pour relier plusieurs blocs.
- Assurer la **communication et l'alimentation électrique** entre blocs.
- Développer un **algorithme de contrôle** pour gérer l'affichage et la communication entre blocs.

Le projet se décline en deux versions :

- **Version 1 : Prototype / Proof of Concept** – Afficheur simplifié pour tester le concept (lettres + espace vide).
- **Version 2 : Modulaire et Évolutive** – Système plus grand et adaptable avec fonctionnalités avancées.

3 Version 1 : Prototype / Proof of Concept

3.1 Description

Un **afficheur compact** simplifié, capable d'afficher les 26 lettres de l'alphabet et un espace vide. La mécanique repose sur des engrenages imprimés en 3D, avec une transmission de puissance basique.

3.2 Fonctionnalités

- Affiche **26 lettres** + un espace vide.
- Utilisation d'un **moteur pas à pas 28BYJ-48** pour piloter les caractères.
- Détection de position via un **capteur à effet Hall** et un aimant néodyme.
- Contrôle via une **carte Arduino Uno**.

3.3 Matériel nécessaire

- Imprimante 3D + bobines de fil de couleurs différentes.



- 1 moteur pas à pas 28BYJ-48 + driver ULN2003.
- 1 capteur à effet Hall + aimant néodyme.
- 1 carte Arduino Uno.
- Fils, breadboard et composants de base.

3.4 Phases de développement

1. **Présentation du projet** : Explication du principe aux membres.
2. **Conception mécanique** : Modélisation 3D (CATIA V5 ou équivalent), impression 3D des engrenages.
3. **Développement du code Arduino** : Contrôle du moteur pas à pas, lecture du capteur à effet Hall.
4. **Tests et validation** : Test du code sur le moteur seul, puis intégration au dispositif mécanique.
5. **Assemblage final** : Montage des composants et essais fonctionnels.
6. **Finalisation** : Impression de la façade et réglages finaux de l'afficheur.

4 Version 2 : Afficheur Modulaire Évolué

4.1 Description

Une version plus **grande et modulaire**, qui peut afficher lettres, chiffres, caractères spéciaux et motifs. Les blocs d'affichage peuvent être connectés/déconnectés à la volée (hot-swappable) et communiquent entre eux.

4.2 Fonctionnalités

- Affiche **lettres, chiffres, caractères spéciaux et motifs**.
- Moteurs pas à pas plus puissants pour une meilleure robustesse.
- **Modularité** : blocs connectables à la volée.
- Transmission de puissance (12 V ou 24 V) + communication (Rx, Tx) via connecteurs Pogo.
- **Détection automatique** des blocs adjacents.
- Connexion WiFi via un **bloc maître** (ESP8266 ou ESP32) pour contrôle à distance.

4.3 Matériel nécessaire

- Moteurs pas à pas plus puissants + drivers associés.
- Connecteurs Pogo pour la modularité.
- Capteurs à effet Hall et aimants pour la détection de position.
- Carte Arduino ou microcontrôleur plus puissant pour gérer la communication.
- Module WiFi (ex. ESP8266 ou ESP32).
- Alimentation 12 V ou 24 V.

4.4 Phases de développement

1. **Conception mécanique** : Adapter la structure pour la modularité et les moteurs plus puissants.
2. **Développement du protocole de communication** : Transmissions de données, détection des blocs adjacents.
3. **Programmation du bloc maître** : Intégration WiFi + algorithme de contrôle global.
4. **Tests de modularité** : Vérifier la connexion et la communication de plusieurs blocs.
5. **Optimisation de l’affichage** : Gérer l’affichage global en fonction du nombre de blocs.
6. **Validation finale** : Tests et ajustements finaux.

5 Contraintes Techniques

- **Compatibilité des matériaux** : Les pièces imprimées doivent supporter les contraintes mécaniques.
- **Alimentation électrique** : Les blocs fonctionnent avec une alimentation 12 V ou 24 V, gérant la puissance efficacement.
- **Communication fiable** : Le protocole doit être robuste et rapide.
- **Modularité** : Les connecteurs Pogo doivent assurer une liaison stable et durable.

6 Livrables

- Un **prototype fonctionnel** de la Version 1.
- Une **Version 2 modulaire** avec au moins 3 blocs interconnectés.
- Documentation technique (schémas, code source, instructions d’assemblage).
- Manuel d’utilisation et de maintenance.



7 Planning Prévisionnel

- **Semaines 1-2** : Conception mécanique et impression 3D.
- **Semaines 2-3** : Développement du code Arduino et tests unitaires.
- **Semaines 3-4** : Assemblage et validation du prototype.
- **Sur toute la durée (1-4)** : Documentation et mise à jour du cahier des charges.

8 Équipe et Rôles

- **Chef de projet** : Coordination, suivi des échéances.
- **Ingénieur mécanique** : Conception 3D, impression des pièces.
- **Ingénieur électronique** : Circuits, capteurs, contrôle moteurs.
- **Développeur logiciel** : Code Arduino, protocoles de communication.
- **Testeur** : Vérification des fonctionnalités et performances.

9 Inspiration : Projets en ligne

Voici quelques ressources utiles pour s'inspirer :

- **Split-Flap Display par Scott Bezek** : Projet open-source avec fichiers 3D et code Arduino.
<https://scottbez1.github.io/splitflap/>
- **Instructables : Split-Flap Display** : Tutoriel détaillé pour construire un afficheur à palettes.
<https://www.instructables.com/Split-Flap-Display/>
- **Projet Split-Flap avec ESP8266** : Afficheur connecté (WiFi).
<https://github.com/scottbez1/splitflap>
- **Vidéo de démonstration DIY** : Exemple visuel de fonctionnement.
https://youtu.be/UAQJJAQsg_g?si=SSXFXy-9FWzp1V5i
- **Thingiverse : Modèles 3D pour split-flap** :
<https://www.thingiverse.com/thing:2820599>
- **Documentation Arduino** :
<https://docs.arduino.cc/>



Conclusion

Ce document définit les **objectifs**, les **fonctionnalités**, les **contraintes** et les **étapes** du projet d'afficheur à palettes modulaire. Il servira de **référence** pour guider le développement et garantir la réussite du projet. L'équipe de **RobotiClermont** se tient à disposition pour tout conseil ou soutien technique.