



POLYTECH[®]
SORBONNE



SORBONNE
UNIVERSITÉ

POLYTECH SORBONNE UNIVERSITÉ

PROJET ELECTRONIQUE 2
SYSTÈME D'EMPRUNT D'OUTILS
RAPPORT

Gestion de prêt des outils

Élève(s) :

Papa Talla DIOUM
Daniel FERREIRA LARA
Nadir KAREM
David LAMOOT

Enseignant(s) :

Francis BRAS

20 janvier 2025

Table des matières

1	Introduction	2
2	Présentation du système	3
3	Bilan des entrées/sorties	4
4	Schématique et simulations	6
4.1	Schéma électronique	6
4.2	Simulation des résultats	6
5	Architecture du code	8
6	Validation et tests	9
6.1	Montage physique	9
6.2	Tests fonctionnels	9
6.3	Résumé des résultats	10
7	Limites et perspectives d'améliorations	10
8	Conclusion	11

1 Introduction

Ce projet a pour objectif de concevoir un système automatisé et sécurisé permettant de gérer le prêt de petits outils dans des espaces en libre accès, tout en garantissant leur retour en fin de séance. Les outils sont disposés dans des présentoirs équipés d'un système de verrouillage contrôlé par des servomoteurs. Voici les principales fonctionnalités :

- **Identification des utilisateurs** : réalisée via des badges RFID (type EM4102 – A6), ce qui permet de sécuriser l'accès au matériel.
- **Demande et restitution des outils** : en utilisant un bouton pour sélectionner le type d'outil (pince plate, pince coupante, tournevis, etc.). Le système identifie l'utilisateur et débloque le compartiment correspondant si l'autorisation est validée.
- **Suivi et enregistrement** : chaque emprunt ou restitution est enregistré dans un fichier de log, incluant l'identifiant du badge utilisé. Ce suivi garantit une traçabilité totale.
- **Détection d'anomalies** : en cas de problème (outil manquant ou erreur utilisateur), une option d'appel administrateur est prévue pour intervenir.
- **Interface administrateur** : l'administrateur peut consulter les logs, éditer la liste des utilisateurs autorisés ou résoudre des incidents à l'aide d'une liaison série.

Ce rapport présente les étapes de conception, de mise en œuvre, et de validation du système.

2 Présentation du système

Le système est composé des éléments suivants :

- **Badges RFID** : Les badges RFID utilisés sont de type EM4102 – A6, opérant à une fréquence de 125 kHz. Ils permettent une identification sécurisée des utilisateurs grâce à un protocole simple mais efficace.
- **Servomoteurs** : Ces dispositifs jouent un rôle crucial dans le verrouillage et le déverrouillage des compartiments contenant les outils. Chaque commande utilisateur actionne le servomoteur correspondant pour libérer ou sécuriser l'outil choisi.
- **Microcontrôleur STM32F103** : Ce microcontrôleur coordonne les interactions entre les composants du système. Il traite les données reçues des badges RFID, contrôle les servomoteurs et gère les communications UART avec l'administrateur.
- **Capteurs de position** : Bien que simples, ces capteurs garantissent une vérification fiable de la prise ou de la restitution des outils. Ils assurent que chaque opération est correctement validée.
- **Interface UART** : La communication UART permet à l'administrateur de consulter les logs, d'accéder à la liste des utilisateurs autorisés, ou de signaler des anomalies pour intervention immédiate.



FIGURE 1 – Badge RFID utilisé pour l'identification.

3 Bilan des entrées/sorties

Le tableau ci-dessous présente le bilan des entrées et sorties du STM32F103, incluant les périphériques utilisés, les signaux associés, et leur rôle dans le fonctionnement du système :

Module	Nom_Signal / Label_Fil	In / Out	Périphérique (rôle signal)	Affectation Périphérique	Port x.y	Traitement par IT / scrutation	Variables liées	nom_fonctions liées	commentaires autres
RFID	RFID (Rx/Tx)	In	UART	USART1	PA9/PA10	IT	badge_id	read_badget()	Réception des données de la carte RFID pour l'identification utilisateur
USART3	UART	In/Out	USART3	USART3	PB10 / PB11	IT	log_data	handle_serial()	Communication série avec périphériques.
Bouton Gauche	Gauche	In	GPIO	GPIO	PC10	Scrutation	left_button_status	handle_left_button()	Sélection outil à gauche.
Bouton Droite	Droite	In	GPIO	GPIO	PC12	Scrutation	right_button_status	handle_right_button()	Sélection outil à droite.
Bouton Prêt	Prêt	In	GPIO	GPIO	PC5	Scrutation	ready_status	handle_ready()	Indique que l'utilisateur est prêt.
Clavier numérique (IHM)	Clavier	In	GPIO	GPIO	PC0 - PC7	Scrutation	tool_choice_x	selectionner_outil	Clavier 4x4 pour sélectionner l'outil. PC0 à PC3: Lignes; PC4 à PC7: Colonnes
Servomoteur de verrouillage	Servo_PWM	Out	Timer	TIM1	PA8	Scrutation	lock_status	set_lock_status()	Verrouillage et déverrouillage des outils
LED Indicatrice	LED	Out	GPIO	GPIO	PA7	Scrutation	led_status	set_led_status()	Notification de l'outil libéré ou position disponible
Présence outil	Presence Outil	In	GPIO	GPIO	PC4	Scrutation	tool_presence_status	check_tool_presence()	Vérifie si un outil est présent.
Bouton Admin (IHM)	Appel_admin	In	GPIO	GPIO	PB2	IT	admin_call	report_problem()	Signalement d'une anomalie
Bouton Demande (IHM)	Demande_bouton	In	GPIO	GPIO	PB3	IT	request_call	demand_handling()	Bouton pour demander un outil.
Bouton Retour (IHM)	Retour_bouton	In	GPIO	GPIO	PC8	IT	return_call	retourn_handling()	Bouton pour signaler la restitution de l'outil.
Liaison série	Connection_admin	In / Out	UART	USART2	PA2 / PA3	IT	log_access	connect_admin()	Pour l'administrateur : gérer les accès. Consultation des fichiers de log / Liste des badges
Mémoire non volatile	Memory	In/Out	SPI	SPI1	PA5, PA6, PA7	Scrutation	user_data	read_data(), write_data()	Enregistrement du dernier utilisateur emprunteur

FIGURE 2 – Bilan des entrées et sorties du STM32F103rb.

Ce bilan est essentiel pour comprendre comment chaque composant interagit avec le microcontrôleur. Par exemple :

- Les entrées, comme les signaux des badges RFID ou des boutons, permettent de capturer les actions de l'utilisateur.
- Les sorties, telles que les commandes pour les servomoteurs ou l'activation des LED, traduisent les décisions du microcontrôleur en actions physiques.
- Les communications bidirectionnelles, comme via l'interface UART, assurent la transmission des informations de diagnostic ou de configuration avec l'administrateur.

Ainsi, ce bilan structure la répartition des tâches entre le matériel et le logiciel, garantissant un fonctionnement cohérent et optimisé du système.

La figure suivante illustre l'attribution des broches sur le microcontrôleur STM32F103, mettant en évidence les connexions des composants principaux tels que les servomoteurs, les capteurs de position, et le lecteur RFID.

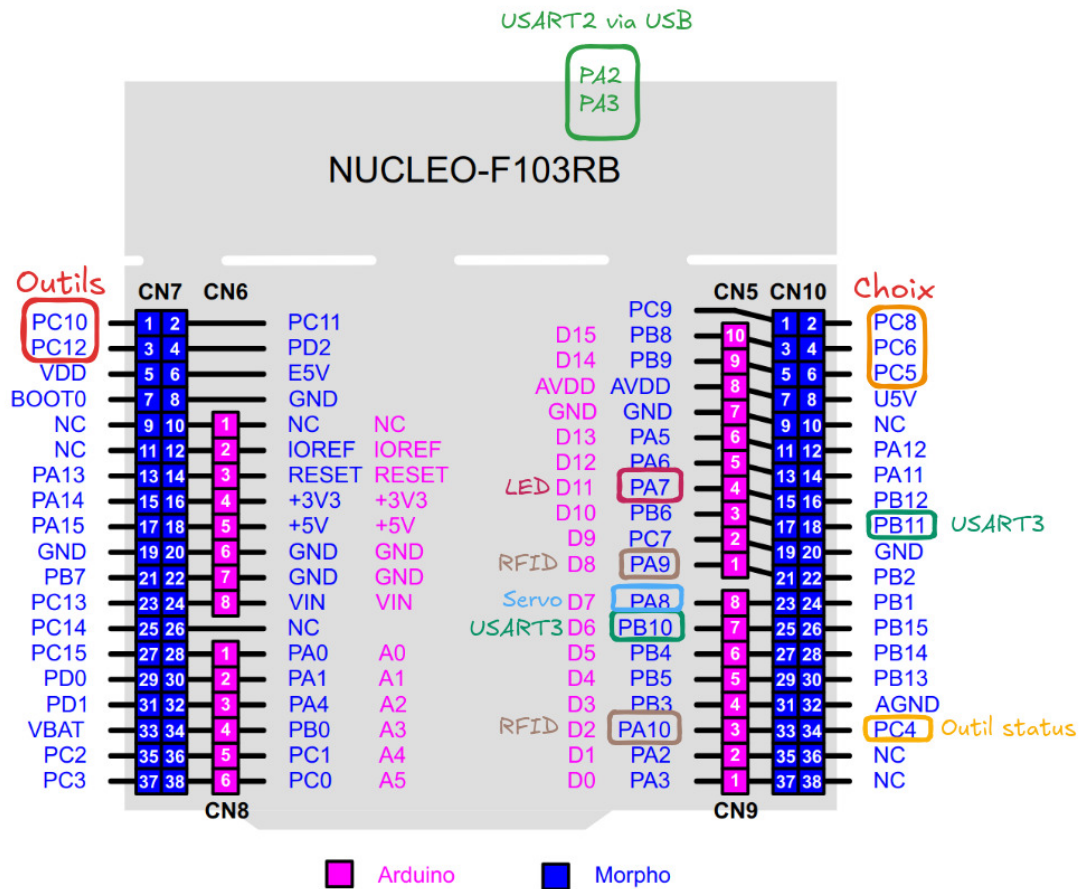


FIGURE 3 – Attribution des broches sur la NUCLEO-F103RB.

4 Schématique et simulations

4.1 Schéma électronique

Le système repose sur une architecture modulaire permettant une interaction fluide entre les composants. Voici le schéma principal :

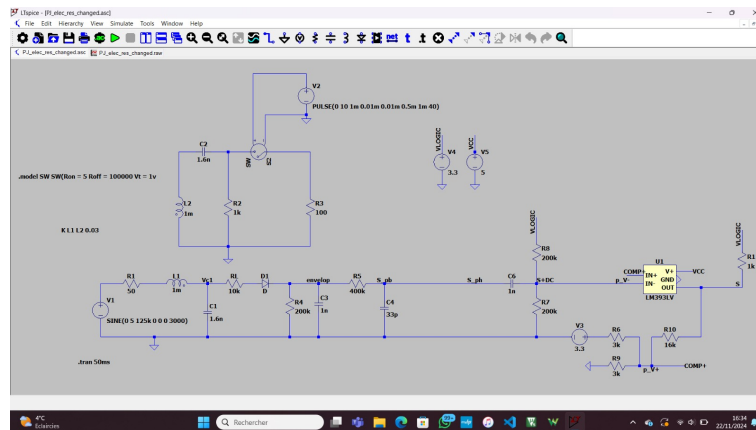


FIGURE 4 – Schéma électronique du système sur LTSpice.

4.2 Simulation des résultats

Les résultats de simulation montrent les différentes étapes du traitement du signal au sein du système.

Signal initial à VC1 : Le premier graphique illustre le signal modulé capté, affichant des oscillations importantes correspondant au signal brut reçu. Ce signal doit être transformé pour en extraire l'information utile.

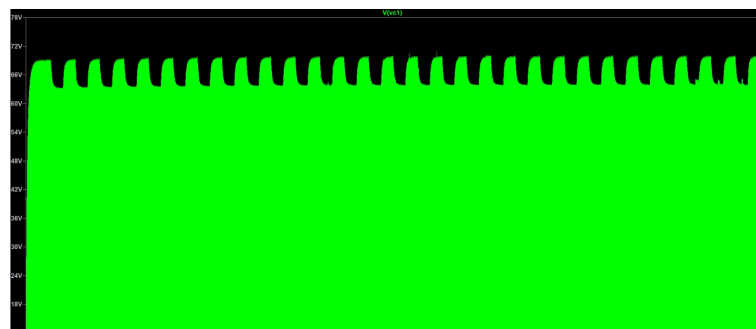


FIGURE 5 – Signal initial capté à VC1.

Détection de l'enveloppe : Après passage par le détecteur d'enveloppe (incluant un filtre passe-bas), le signal est lissé pour extraire l'enveloppe des oscillations. L'enveloppe représente l'information utile à décoder.

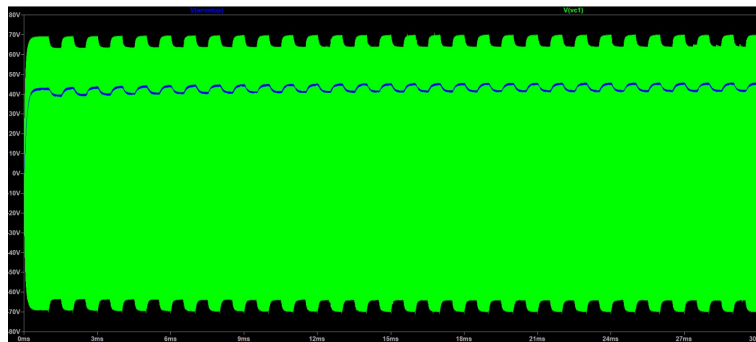


FIGURE 6 – Signal lissé avec l’enveloppe détectée en bleu.

Signal à l’entrée du comparateur : Enfin, le signal traité est préparé pour être comparé à un seuil par le comparateur. Cela permet de convertir l’information analogique en une sortie numérique claire, prête à être exploitée par le microcontrôleur.

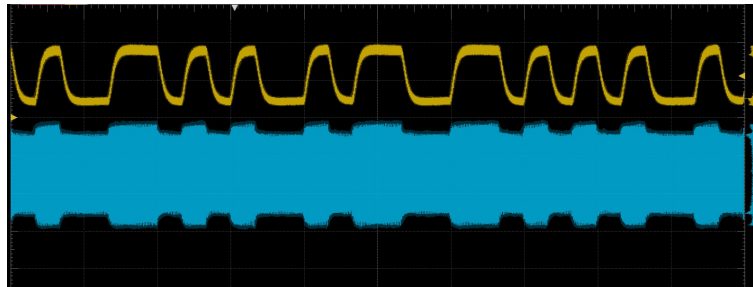


FIGURE 7 – Signal à l’entrée du comparateur.

Ces étapes illustrent comment le système convertit un signal brut en une donnée exploitable, en passant par des traitements intermédiaires tels que le filtrage, la détection d’enveloppe et la conversion numérique.

5 Architecture du code

Le code est divisé en modules pour une meilleure gestion et évolutivité :

- **‘main.c’** : Point d’entrée, gère le traitement principal, y compris le décodage des badges RFID et le contrôle des servomoteurs.
- **‘servo.c’** : Module dédié au contrôle précis des servomoteurs via PWM.
- **‘timer.c’** : Fournit la configuration des timers pour les interruptions et le PWM.
- **‘uart.c’** : Gère la communication série pour le diagnostic et les journaux.
- **‘manchester.c’** : Implémente le décodage Manchester pour lire les trames RFID.
- **‘logs.c’** : Gestion des Logs.
- **‘uart.c’** : Gestion des deux communications USART2 et USART3.
- **‘SPI.c’** : Gestion des tous les ports pour les SPI, principalement pour le ADXL.

En bref, nous avons divisé le code en bibliothèques génériques qui permettent de configurer plus librement tous les composants, de modifier les ports et d’ajouter des composants au système de manière plus abstraite.

Nous gérons l’interface HMI et le système de verrouillage du moteur dans le STM32, qui communique avec l’esp pour envoyer des logs et demander des vérifications de l’identité du badge. esp agit comme un intermédiaire entre le cloud et la machine, en appelant l’api qui tourne sur un serveur et en envoyant des notifications à l’administrateur par email/notification pop-up. L’api Back gère les données dans la base de données. Enfin, l’administrateur peut consulter les données et activer/désactiver les badges à partir du site web.

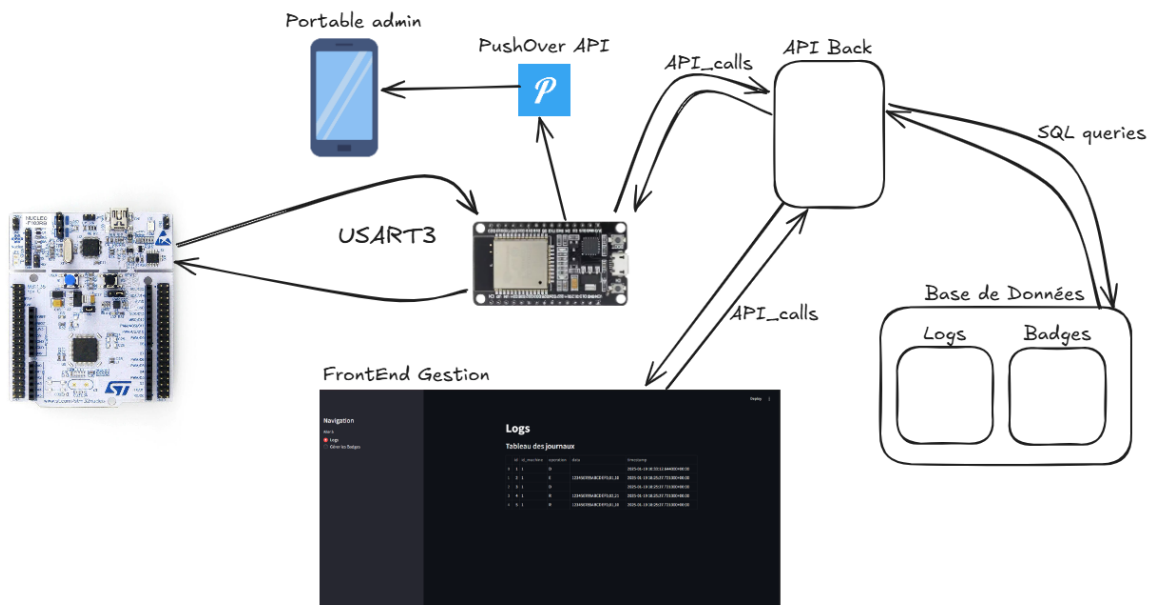


FIGURE 8 – Architecture

Malheureusement, l’intégration complète n’a pas été réalisée parce que nous ne disposons pas du système de détection des badges, qui fait partie de l’interface HMI, de sorte que la validation de la valeur du badge n’a pas été correctement testée de bout en bout.

6 Validation et tests

6.1 Montage physique

Le montage final inclut les connexions essentielles pour démontrer les fonctionnalités principales du système. Les composants suivants ont été intégrés :

- Un microcontrôleur STM32F103 pour la gestion centrale.
- Un ESP32 pour transmettre les données à une base de données en ligne.
- Des boutons poussoirs divisés en deux groupes :
 - Ceux de gauche pour sélectionner les outils à emprunter ou à rendre.
 - Ceux de droite pour verrouiller, déverrouiller, et signaler une demande d'assistance.
- Un servomoteur pour contrôler l'accès aux compartiments des outils.
- Une interface UART pour afficher en temps réel les journaux des actions sur un ordinateur (visualisable figure 8).

Le montage physique est illustré ci-dessous :

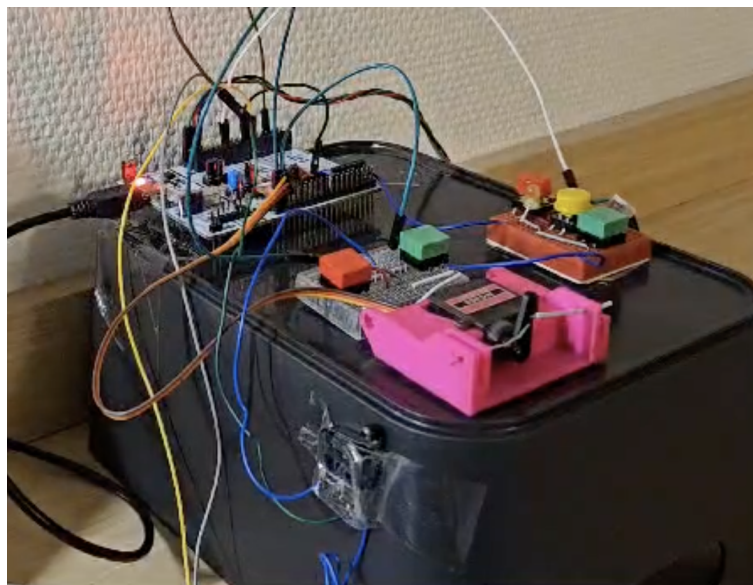


FIGURE 9 – Montage physique du système.

6.2 Tests fonctionnels

Nous avons réalisé plusieurs tests pour valider les différentes fonctionnalités du système. Les tests incluait :

- **Contrôle des servomoteurs** : Les boutons poussoirs permettent de contrôler précisément le mouvement du servomoteur pour verrouiller ou déverrouiller les compartiments contenant les outils. Ces mouvements sont synchronisés avec les journaux affichés via l'interface UART.
- **Sélection des outils** : Les boutons de gauche permettent de choisir l'outil à emprunter ou à rendre. Chaque sélection est correctement enregistrée et affichée dans les journaux.
- **Demande d'assistance** : Un bouton dédié permet de signaler une demande d'aide, qui est immédiatement logée et envoyée à la base de données via l'ESP32.

- **Enregistrement et transmission des données :** Toutes les actions (verrouillage, sélection, demande d'assistance) sont transmises à une base de données en ligne accessible via une API, renforçant ainsi la traçabilité et l'administration à distance.

Les journaux affichés sur l'ordinateur, via l'interface UART, permettent de suivre chaque action en temps réel, offrant une visibilité complète sur les opérations effectuées. Cette démonstration, visible dans la vidéo, met en évidence la capacité du système à gérer efficacement les interactions utilisateur et à transmettre les données à distance.

6.3 Résumé des résultats

Les tests réalisés ont confirmé la faisabilité technique des fonctionnalités suivantes :

- Contrôle des compartiments via servomoteur.
- Gestion interactive par boutons poussoirs.
- Enregistrement et transmission des données en temps réel.

Cependant, des tests supplémentaires seront nécessaires pour valider l'intégration complète des capteurs et garantir la robustesse du système dans des conditions réelles.

7 Limites et perspectives d'améliorations

À l'issue de ce projet, nous avons identifié plusieurs limites de notre système ainsi que des pistes d'amélioration. Ces réflexions nous ont permis de concevoir des solutions potentielles pour enrichir le système dans le futur. Toutefois, en raison des contraintes de temps, ces idées n'ont pas pu être mises en œuvre dans cette version. Voici une liste des améliorations envisagées :

- **Optimisation du décodage RFID pour des conditions bruyantes :** Les systèmes RFID sont particulièrement sensibles aux interférences électromagnétiques et aux environnements bruyants. Une optimisation pourrait inclure l'implémentation de filtres numériques ou analogiques pour nettoyer le signal reçu, ainsi que l'utilisation d'algorithmes plus robustes pour le décodage des données, garantissant ainsi une lecture fiable même en présence de perturbations.
- **Tests supplémentaires sur la robustesse mécanique des serrures :** Les servomoteurs et mécanismes de verrouillage doivent être capables de résister à des utilisations fréquentes et à des manipulations potentiellement brutales. Des tests prolongés sur la durabilité des composants mécaniques et des simulations d'usure permettront d'identifier et de renforcer les points faibles du système.

8 Conclusion

Ce projet nous a permis de concevoir et de mettre en œuvre un système automatisé pour le prêt sécurisé des outils. Bien que tous les éléments prévus n'aient pas pu être intégrés dans une version finale, nous avons réalisé un prototype fonctionnel démontrant les concepts clés du projet. Le système permet notamment :

- Verrouillage et déverrouillage des compartiments contenant les outils, grâce au contrôle précis des servomoteurs.
- Sélection des outils à emprunter ou à rendre via des boutons interactifs.
- Signalement d'assistance en cas de problème, activé par un bouton dédié.
- Suivi des actions avec des journaux affichés en temps réel sur un ordinateur via UART, permettant une traçabilité accrue.
- Connexion à une base de données en ligne grâce à un ESP32, offrant une gestion administrative à distance par une interface dédiée.

Malgré ces réussites, nous avons identifié des axes d'amélioration :

- Tester l'ensemble du système avec tous les modules connectés pour garantir sa robustesse.
- Optimiser les performances RFID pour réduire les interférences dans des environnements complexes.
- Renforcer les mécanismes de verrouillage pour une utilisation intensive.

Ce projet a représenté une excellente opportunité d'appliquer des compétences techniques variées, telles que l'électronique, la programmation embarquée, et la gestion de projet. Il constitue une base solide pour des développements futurs, visant à rendre le système modulaire, évolutif et prêt à être utilisé dans des environnements réels.

Références

Repo GitHub : [Gestionnaire Emprunt Outils](#)
Démonstration vidéo 1 : [Démonstration vidéo](#)
Démonstration vidéo 2 : [Démonstration vidéo](#)