Rapport technique – Développement et contrôle d’un bras robotique

**1. Objectifs**

* Étudier et améliorer un modèle 3D de bras robotique.
* Programmer le contrôle fluide des servomoteurs via Arduino.
* Concevoir une application PC en langage C pour piloter le bras (interface série/Bluetooth).
* Développer un boîtier de contrôle physique Bluetooth avec boutons et potentiomètres.
* Mettre en place un suivi de projet clair et documenter toutes les étapes.

**2. Analyse et conception mécanique**

**Étude des pièces à imprimer**

* Fichiers STL des pièces par plateau d’impression.
* Points faibles détectés : jeu mécanique, fixation instable des servos.

**Matériaux utilisés**

* PLA choisi pour sa facilité d'impression et PETG pour sa solidité.
* Impression avec une épaisseur de couche de 0,2 mm et 15 % d’infill.

**3. Composants**

**Carte Arduino UNO R3**

* Microcontrôleur utilisé pour piloter les servos et gérer les communications.
* Fiche technique : [Arduino UNO R3](https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf)

**Servomoteurs**

* **MG995** : couple élevé, utilisé pour la base et les articulations principales.
  + [Datasheet MG995](https://www.electronicoscaldas.com/datasheet/MG995_Tower-Pro.pdf)
* **MG90S** : plus petit et rapide, utilisé pour les pinces.
  + [Datasheet MG90S](https://www.electronicoscaldas.com/datasheet/MG90S_Tower-Pro.pdf)

**Modules Bluetooth**

* **HC-05** : utilisé côté Arduino du boîtier en mode maître.
  + [Datasheet HC-05](https://components101.com/sites/default/files/component_datasheet/HC-05%20Datasheet.pdf)
* **HC-06** : utilisé pour l’Arduino du robot (mode esclave uniquement).
  + [Datasheet HC-06](https://www.alldatasheet.fr/datasheet-pdf/view/1179032/ETC1/HC-06.html)

**4. Câblage et Schéma Électrique**

**Connexions de base**

* Servomoteurs connectés aux broches PWM de l’Arduino.
* HC-05 connecté via RX/TX (avec diviseur de tension pour RX).

Une image contenant texte, Ingénierie électronique, Appareils électroniques, câble

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, diagramme, Police

Le contenu généré par l’IA peut être incorrect.

**Programmation Arduino**

* Utilisation de la bibliothèque Servo.h.
* Traitement des commandes Bluetooth en temps réel.
* Communication série interprétée sous forme de commandes : S N°SERVO ANGLE (0 à 180°)

**5. Application PC en Langage C**

* Interface SLD2 en C.
* Communication via port série (USB ou Bluetooth).
* Fonctionnalités :
  + Contrôle manuel avec les curseurs
  + Possibilité d’importer/exporter/sauvegarder et réinitialiser les positions.

**6. Boîtier de Contrôle Bluetooth**

* Boîtier physique imprimé avec emplacements pour :
  + 6 potentiomètres logarithmiques
  + 1 potentiomètre linéaire
  + 4 boutons
* Communication Bluetooth

**7. Tests Réalisés**

* **Tests mécaniques** : résistance des supports, amplitude max.
* **Tests de communication** : stabilité Bluetooth, portée, latence.
* **Tests logiciels** : exécution de commandes simultanées.

**8. Problèmes rencontrés**

* **Interférences RX/TX** entre Arduino et Bluetooth → solution : delay + diviseur de tension.
* **Surchauffe des moteurs** → ajout de pauses dans les séquences.
* **Limite de courant USB** → alimentation externe 5V 2A pour les moteurs

**9. Innovations à intégrer**

* Base de reconnaissance vocale via micro.
* Ajout d’une caméra

**10. Suivi de projet**

Utilisation de GitHub

**11. Conclusion**

Ce projet nous a permis de maîtriser l’ensemble de la chaîne de développement mécatronique : modélisation 3D, programmation embarquée, développement logiciel, communication Bluetooth, gestion d’un projet technique et rédaction d’un rapport structuré.