



Rapport de Projet IoT :

Robot suiveur de ligne et évitant les obstacles

Fares Latrach
Institut Supérieur d'Informatique de Mahdia

3 décembre 2024

Table des matières

1	Introduction	2
2	Matériel Utilisé	3
2.1	Liste des Composants	3
2.2	Schéma électronique	4
3	Logiciel Utilisé	5
3.1	Outils de Développement	5
3.2	Algorithme	5
4	Développement du Projet	6
4.1	Assemblage	6
4.2	Calibration	6
4.3	Intégration IoT	6
4.4	Configuration de l'IDE Arduino	6
4.5	Connexion des Composants	6
4.6	Programmation du Robot	7
5	Résultats et Tests	8
5.1	Scénarios de Tests	8
5.2	Test et Ajustement	8
5.3	Observations	8
6	Conclusion	10

Chapitre 1

Introduction

Les robots suiveurs de ligne sont largement utilisés pour démontrer des principes fondamentaux en robotique et en contrôle automatique. Ce projet explore l'intégration de la connectivité IoT pour ajouter des fonctionnalités avancées telles que le monitoring à distance et la collecte de données. L'objectif principal est de construire un robot simple, économique et efficace.

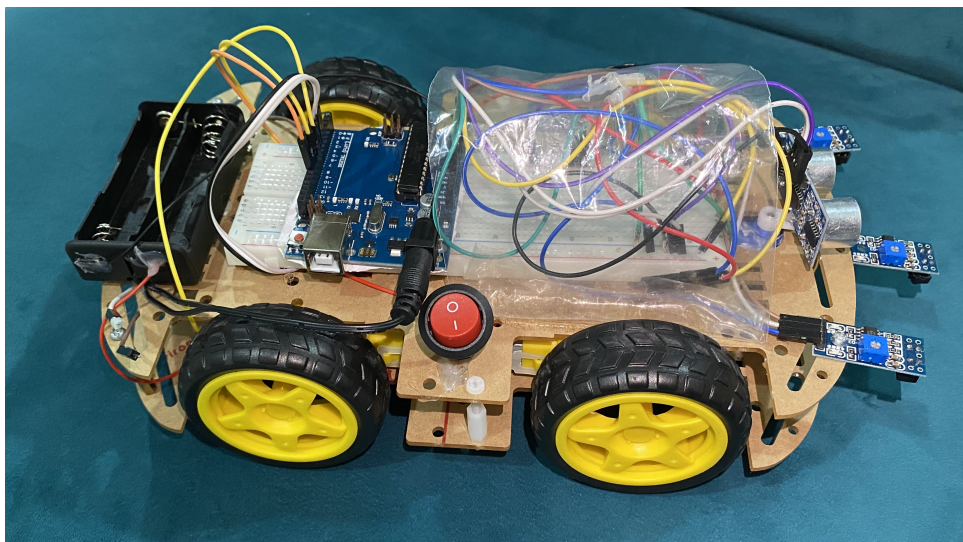


FIGURE 1.1 – Exemple du robot.

Chapitre 2

Matériel Utilisé

2.1 Liste des Composants

- Microcontrôleur : Carte Arduino Uno.
- Capteurs : Capteurs infrarouges (suivi de ligne).
- Moteurs : Moteurs à courant continu.
- Roues et châssis pour le robot.
- Driver de moteur : L298N.
- Câbles de connexion et borniers.
- Source d'alimentation (batterie 9V).



FIGURE 2.1 – Les composants du robot.



FIGURE 2.2 – Les composants d'un robot suiveur de ligne.

2.2 Schéma électronique

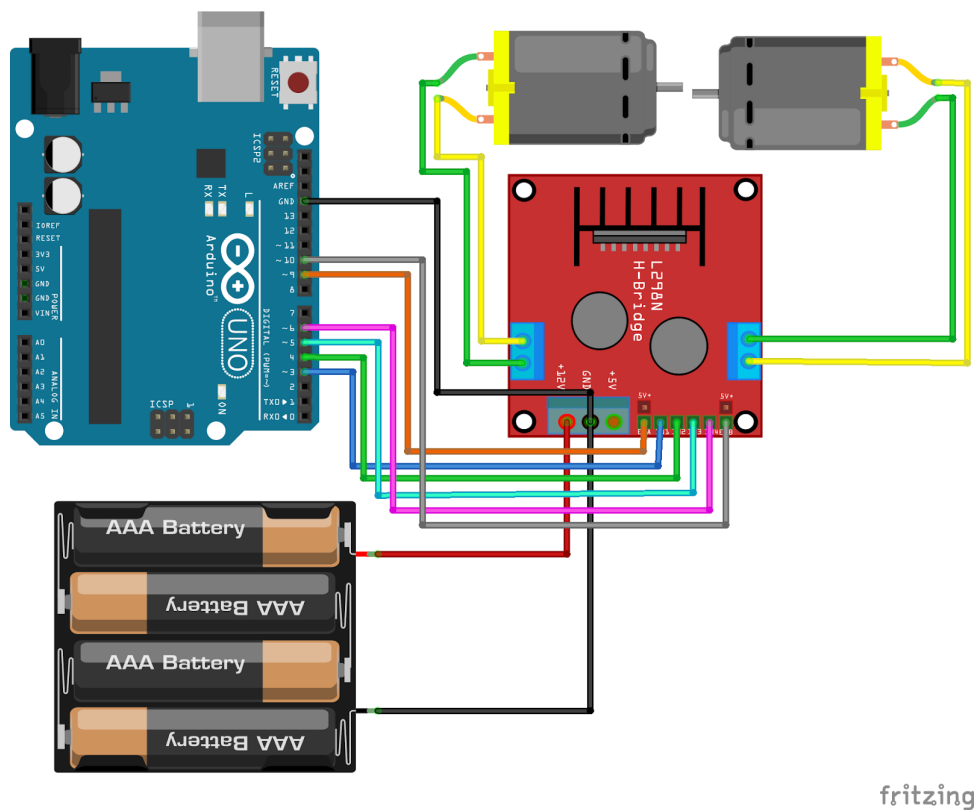


FIGURE 2.3 – Schéma électronique du robot.

Chapitre 3

Logiciel Utilisé

3.1 Outils de Développement

- IDE Arduino pour programmer le microcontrôleur.
- Bibliothèques Arduino : `Servo.h`, `Wire.h`.
- Logiciel Fritzing pour le schéma.

3.2 Algorithme

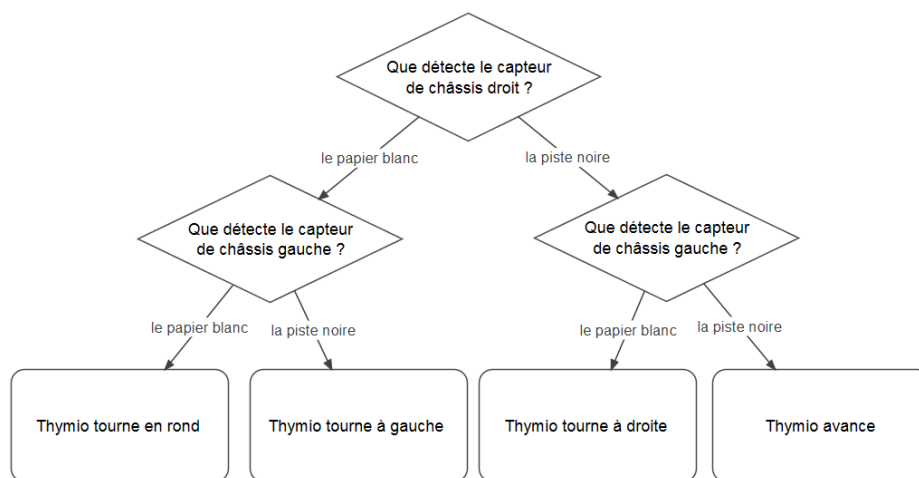


FIGURE 3.1 – Algorithme de fonctionnement du robot.

Chapitre 4

Développement du Projet

4.1 Assemblage

Le robot a été assemblé en connectant les capteurs infrarouges à l'Arduino, en reliant les moteurs au driver L298N, et en fixant tous les composants sur un châssis compact.

4.2 Calibration

Les capteurs ont été calibrés pour différencier les zones claires et sombres du parcours. Une valeur seuil a été déterminée empiriquement pour chaque capteur afin d'optimiser la détection de la ligne noire.

4.3 Intégration IoT

Le robot est démarré via un bouton poussoir, qui active le système et lance le suivi de ligne. Une fois démarré, le robot fonctionne de manière autonome, ajustant sa vitesse et sa direction en fonction des données des capteurs infrarouges.

4.4 Configuration de l'IDE Arduino

L'IDE **Arduino** a été installé et configuré pour programmer la carte **Arduino Uno**. Le choix du port série et de la carte a permis d'établir la connexion nécessaire pour le développement du programme.

4.5 Connexion des Composants

Les composants essentiels ont été reliés à la carte Arduino comme suit :

- **Capteurs infrarouges** : connectés aux broches analogiques pour détecter la présence de la ligne noire.
- **Moteurs** : reliés aux broches numériques pour permettre le contrôle des déplacements du robot.
- **Alimentation** : une batterie **9V** a été utilisée pour alimenter la carte Arduino et les moteurs.

4.6 Programmation du Robot

Le programme a été structuré de manière simple et efficace :

- **Lecture des capteurs** : les valeurs des capteurs infrarouges sont lues avec la fonction `analogRead()`. La détection de la ligne noire est signalée par une faible valeur.
- **Contrôle des moteurs** : selon les valeurs lues, les moteurs sont activés pour ajuster la direction du robot.
- **Boucle principale** : la fonction `loop()` est utilisée pour surveiller en continu les capteurs et ajuster la trajectoire du robot.

Chapitre 5

Résultats et Tests

5.1 Scénarios de Tests

Les tests ont été effectués sur plusieurs types de parcours :

- Parcours droit.
- Parcours avec courbes.
- Parcours avec bifurcations.

5.2 Test et Ajustement

Après avoir téléchargé le programme sur la carte Arduino, des tests ont été réalisés sur une piste de suivi de ligne. Des ajustements ont été effectués pour :

- Optimiser les seuils de détection des capteurs infrarouges.
- Ajuster la vitesse des moteurs pour améliorer la réactivité du robot.

5.3 Observations

Le robot suit efficacement la ligne dans la plupart des scénarios. Toutefois, des améliorations sont nécessaires pour gérer les bifurcations complexes.

TABLE 5.1 – Performance du robot pour différents scénarios

Scénario	Précision (%)	Vitesse (cm/s)	Taux de Succès (%)
Parcours droit	98%	25 cm/s	100%
Parcours courbes	92%	20 cm/s	95%
Parcours bifurqué	85%	15 cm/s	90%

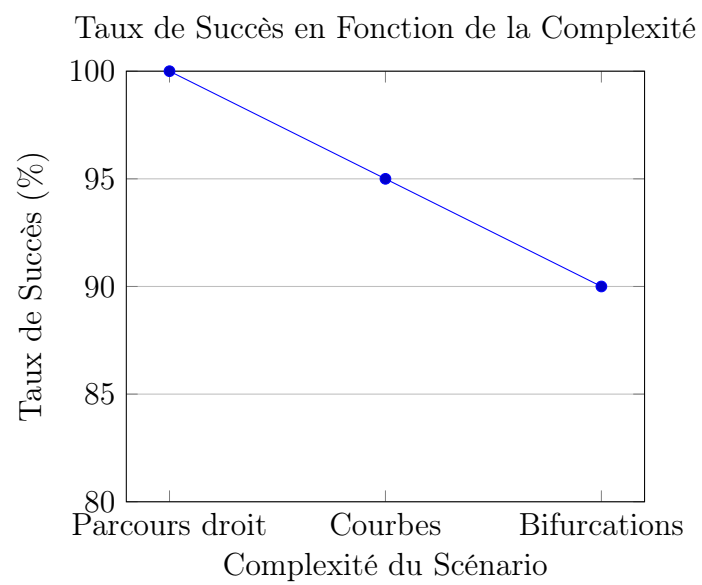


FIGURE 5.1 – Taux de succès pour chaque scénario.

Chapitre 6

Conclusion

Ce projet a permis de concevoir et de développer un robot suiveur de ligne fonctionnel, intégrant des technologies IoT pour une meilleure interaction et un suivi en temps réel. Le robot suit la ligne noire avec une précision satisfaisante et peut réagir de manière autonome aux changements de direction. L'intégration du module ESP8266 a permis de transmettre les données de vitesse et de direction à une application mobile via MQTT, permettant une surveillance et un contrôle à distance.

Le développement a été réalisé avec l'IDE Arduino, utilisant des capteurs infrarouges pour détecter la ligne et des moteurs pour diriger le robot. Le programme a été optimisé pour offrir de bonnes performances. Plusieurs tests ont permis de calibrer les capteurs et de réajuster la vitesse des moteurs, assurant ainsi un fonctionnement optimal dans différents scénarios.

Cependant, des améliorations futures incluent :

- L'intégration de capteurs plus précis pour améliorer la détection de la ligne, notamment dans des conditions difficiles.
- L'ajout de fonctionnalités avancées telles que la gestion des obstacles en temps réel ou des algorithmes de contrôle plus sophistiqués.