



Mémoire de Master

Filière Électronique
Spécialité Electronique des systèmes embarqués

Présenté par :

Mehdid Mohamed

&

Amrane AbdelKahar

Systeme d'inventaire d'entrepôt intelligent

Proposé par : Mme D.Naceur & Mme H.Bougherira

Remerciements

Nous tenons tout d'abord à exprimer notre profonde gratitude à Dieu Tout-Puissant, qui nous a accordé la force, la patience et le courage nécessaires pour mener à bien ce travail.

Nous remercions chaleureusement notre promotrice, Madame Naceur Djamila, pour son encadrement attentif, sa disponibilité, sa bienveillance et ses conseils précieux tout au long de la préparation de ce mémoire.

Nos sincères remerciements s'adressent également à notre co-promotrice, Madame H. Bougherira et à l'ensemble des enseignants et du personnel administratif du département d'électronique, pour leur accompagnement pédagogique et leur soutien tout au long de notre parcours universitaire.

Nous remercions aussi les membres du jury, pour l'honneur qu'ils nous font en acceptant d'évaluer ce travail et pour le temps qu'ils y consacrent.

Enfin, nous adressons un remerciement particulier à tous les membres de la section « Instrumentation », ainsi qu'à nos camarades et amis de toute la faculté, pour leur soutien moral, leur entraide et les moments de partage durant ces années d'étude.

Dédicaces

Je dédie ce modeste travail à :

Ma chère Mère,

Mon cher Père,

Mes chères Sœurs,

Mes chers frères,

Ceux qui m'aiment,

Ceux que j'aime,

Mais également :

À nos pères,

À nos mères,

Les plus exquises et les plus ingénieuses des mères, À nos bienveillants frères et sœurs,

Nous tenons à exprimer toute notre reconnaissance à tous nos amis,

Mohamed Mehdi

El

Amrane Abd Kahar

تقترح هذه الأطروحة حلاً ذكياً لأتمتة إدارة مخزون المستودعات، استناداً إلى روبوت متنقل مستقل. يتمتع الروبوت بالقدرة على التنقل وفرز ونقل الأشياء وفقاً لتصنيفها، وهو متصل ارتباطاً وثيقاً بواجهة ويب مخصصة. تتيح لك هذه المنصة مراقبة أفعالك في الوقت الفعلي وضبط معلمات التشغيل وتجميع المعلومات الأساسية المتعلقة بالعمليات اللوجستية. ويمهد تكامل هذا النهج الطريق لإدارة حديثة وتفاعلية وفعالة للتدفقات المادية.

كلمات المفاتيح: MG90S ؛ HC-SR04 ؛ TCS3472 ؛ QTR-8A ؛ ESP32-S3 ؛ روبوت متنقل؛ مخزون ذكي
17HS4023؛

Résumé :

Ce mémoire propose une solution intelligente pour l'automatisation de la gestion des stocks en entrepôt, reposant sur un robot mobile autonome. Capable de naviguer, trier et transporter des objets selon leur classification, le robot est étroitement connecté à une interface web dédiée. Cette plateforme permet de superviser ses actions en temps réel, d'ajuster les paramètres de fonctionnement et de centraliser les informations essentielles liées aux opérations logistiques. L'intégration de cette approche ouvre la voie à une gestion moderne, interactive et efficace des flux physiques.

Mots clés : Robot mobile ; Inventaire intelligent ; ESP32-S3 ; QTR-8A ; TCS3472 ; HC-SR04 ; MG90S ; 17HS4023 ;

Abstract:

This thesis proposes an intelligent solution for automating warehouse inventory management, based on an autonomous mobile robot. Capable of navigating, sorting and transporting objects according to their classification, the robot is closely connected to a dedicated web interface. This platform allows you to monitor your actions in real time, adjust operating parameters and centralize essential information related to logistics operations. The integration of this approach paves the way for a modern, interactive and efficient management of physical flows.

Keywords: Robot mobile; Inventaire intelligent; ESP32-S3; QTR-8A; TCS3472; HC-SR04; MG90S; 17HS4023;

Liste des acronymes et abréviations

17HS4023 : Moteur pas à pas bipolaire pour les mouvements précis

CSS / CSS3 : Cascading Style Sheets – Feuilles de style en cascade utilisées pour la mise en forme des pages web

DC : Direct Current – Courant continu

ESP32-S3 : Microcontrôleur Wi-Fi et Bluetooth développé par Espressif Systems

HC-SR04 : Capteur à ultrasons utilisé pour la mesure de distances

HTML / HTML5 : HyperText Markup Language – Langage de balisage pour la création de pages web

HTTP : HyperText Transfer Protocol – Protocole de communication du web

I2C : Inter-Integrated Circuit – Bus de communication série permettant la connexion de plusieurs composants

IDE : Integrated Development Environment – Environnement de développement intégré

IoT : Internet of Things – Internet des objets

JS : JavaScript – Langage de programmation permettant de rendre les pages web interactives

L298N : Module de commande de moteurs à double pont en H

MG90S : Servomoteur miniature utilisé pour des mouvements mécaniques de faible amplitude

OTA : Over The Air – Mise à jour logicielle à distance

PCB : Printed Circuit Board – Carte à circuit imprimé

PID : Proportional–Integral–Derivative – Régulateur utilisé pour le contrôle précis des systèmes

PWM : Pulse Width Modulation – Modulation de largeur d’impulsion utilisée pour le contrôle des moteurs

QTR-8A : Capteur infrarouge de suivi de ligne comportant 8 phototransistors

RGB : Red Green Blue – Modèle de représentation des couleurs utilisé en détection colorimétrique

TCS3472 : Capteur de couleur RGB utilisé pour identifier les couleurs des objets

UART : Universal Asynchronous Receiver Transmitter – Interface de communication série asynchrone

UX/UI : User Experience / User Interface – Expérience utilisateur / Interface utilisateur

WebSocket : Protocole de communication bidirectionnelle utilisé dans les applications web en temps réel

Wi-Fi : Wireless Fidelity – Technologie de communication sans fil

Table des matières

Introduction Générale	1
Chapitre I : Concepts généraux.....	3
1.1 INTRODUCTION.....	3
1.2 DEFINITIONS.....	3
1.2.1 LA ROBOTIQUE :	3
1.2.2 LE ROBOT :	4
1.2.3 AUTOMATES :	4
1.3 HISTORIQUE	4
1.4 TACHES PRINCIPALES DU ROBOT.....	5
1.4.2 TRI ET CLASSEMENT PAR COULEUR :	6
1.4.3 TRANSPORT ET PLACEMENT AUTOMATISE :	6
1.4.4 SUPERVISION ET CONTROLE A DISTANCE :	6
1.4.5 GESTION DES ALEAS :	6
1.5 COMPOSANTS DU ROBOT	7
1.5.1 MICROCONTROLEUR ESP32-S3 :	7
1.5.2 L'ARDUINO NANO	9
1.5.3 MODULE DE CONTROLE DE MOTEURS L298N	11
1.5.4 MOTEURS DC 3,6V	12
1.5.5 CAPTEURS INFRAROUGES QTR-8A	14
1.5.6 CAPTEUR A ULTRASON RECOMMANDE (HC-SR04).....	15
1.5.7 BATTERIE LITHIUM ION	17
1.5.8 CAPTEUR DE COULEUR RGB TCS3472.....	19
1.5.9 STEPPER MOTORS 17HS4023	20
1.5.10 SERVOMOTEUR MG90S	21
1.6 INTEGRATION MECANIQUE ET STRUCTURELLE DU ROBOT	23
1.7 SITE WEB DE CONTROLE ET DE SUIVI DU ROBOT	23
1.8 CONCLUSION	24

Chapitre II : Conception et Réalisation du Robot Mobile Intelligent.....	26
2.1 INTRODUCTION.....	26
2.2 ARCHITECTURE GENERALE DU SYSTEME	26
2.2.1 FONCTIONNEMENT DU PROJET.....	26
2.2.2 SCHEMA BLOC FONCTIONNEL DU SYSTEME	27
2.3 ASSEMBLAGE MECANIQUE DU ROBOT	29
2.3.1 STRUCTURE DU CHASSIS DU ROBOT	30
2.3.2 FIXATION DES ROUES, MOTEURS ET COMPOSANTS	30
2.3.3 SYSTEME DE LEVAGE ET DE DEPOT DES OBJETS.....	34
2.4 STRUCTURE DU DEPOT ET DES ETAGERES	37
2.4.1 CONCEPTION DES ETAGERES	37
2.4.2 TRAÇAGE DE LA LIGNE NOIRE	38
2.4.3 FABRICATION DES COLIS	38
2.5 CONCEPTION DU CABLAGE ELECTRONIQUE	39
2.5.1 CONNEXIONS ELECTRIQUES ENTRE LES COMPOSANTS.....	39
2.5.2 SCHEMA DE CABLAGE GLOBAL.....	51
2.6 PROGRAMMATION	52
2.6.1 ENVIRONNEMENT DE DEVELOPPEMENT UTILISE	52
2.6.2 STRUCTURE DU CODE	56
2.6.3 SUIVI DE LIGNE AVEC QTR-8A	57
2.6.4 SYSTEME DE DETECTION DE COULEUR ET DE MANIPULATION AVEC UN MOTEUR PAS A PAS ET UN SERVOMOTEUR MG90S.....	60
2.7 CONCLUSION	63
Chapitre III : Interface Web, Résultats et Perspectives.....	64
3.1 INTRODUCTION	64
3.2 INTEGRATION IOT ET TABLEAU DE BORD WEB POUR LE CONTROLE ROBOTIQUE AVANCE	65
3.2.1 L'AVANTAGE STRATEGIQUE D'UN TABLEAU DE BORD WEB.....	65

3.2.2	ARCHITECTURE DU SYSTEME ET PILE TECHNOLOGIQUE.....	66
3.2.3	COMMUNICATION EN TEMPS REEL AVEC SOCKET.IO	67
3.3	RESULTATS OBTENUS	73
3.3.1	TESTS DE SUIVI DE LIGNE	73
3.3.2	TESTS DE DETECTION DES COULEURS	74
3.3.3	TESTS DE LEVAGE ET DEPOSE	75
3.3.4	INTERFACE WEB : DEMONSTRATION ET VALIDATION.....	77
3.4	DIFFICULTES RENCONTREES	79
3.4.1	PROBLEMES TECHNIQUES	79
3.4.2	SOLUTIONS APORTEES	79
3.5	PERSPECTIVES ET AMELIORATIONS FUTURES.....	80
3.5.1	ÉVOLUTIONS MATERIELLES	80
3.5.2	AMELIORATIONS LOGICIELLES	81
3.5.3	INTEGRATION DANS UN ENVIRONNEMENT INDUSTRIEL	81
3.6	CONCLUSION	81
	Conclusion Générale.....	82

Liste des figures

Figure 1.1 Parties commande- contrôle et opérative d'un robot	4
Figure 1.2 Chaîne de production robotisée.....	5
Figure 1.3 Microcontrôleur ESP32-S3	7
Figure 1.4 Arduino Nano.....	9
Figure 1.5 Contrôle de Moteurs L298N	11
Figure 1.6 Moteurs DC 3,6V	13
Figure 1.7 Compostant de QTR-8A	14
Figure 1.8 Compassant Capteur à Ultrasons Recommandé (HC-SR04).....	16
Figure 1.9 Détection d'obstacle par ultrason.....	16
Figure 1.10 Batterie lithium-ion rechargeable.....	18
Figure 1.11 Capteur de Couleur RGB TCS3472.....	19
Figure 1.12 Stepper Motors 17HS4023	20
Figure 1.13 Servo moteur MG90S	22
Figure 2.1 Schéma Bloc Fonctionnel du Robot Mobile Intelligent.....	28
Figure 2.2 Le châssis du robot en MDF	30
Figure 2.3 Robot Mobile à 3D.....	31
Figure 2.4 Robot Mobile à 4 Roues.....	31
Figure 2.5 Fixation du QTR-8A sur le Châssis du Robot	32
Figure 2.6 Installation du capteur TCS3472 pour la détection.....	33
Figure 2.7 Montage Interne du Robot : Arduino Nano, L298N, ESP32 S3 et Batterie 12V	33
Figure 2.8 Système de levage	34
Figure 2.9 Système de dépôt	34
Figure 2.10 Fixation du moteur pas à pas 17HS4023 dans la nacelle du robot	35
Figure 2.11 Fixation du moteur servo SG90 dans la nacelle du robot	36
Figure 2.12 Maps de système	37
Figure 2.13 Colis Classifiés par Couleur RGB pour le Tri Automatique	39
Figure 2.14 Arduino Nano -Pinout	41
Figure 2.15 QTR-8A-Pinout.....	41
Figure 2.16 Schéma de circuit électrique Module QTR-8A avec Arduino Nano	42
Figure 2.17 L298N moteur Driver Module pinout [15]	42
Figure 2.18 Schéma de circuit électrique Moteurs à courant continu (x4) avec Module L298N et Arduino Nano	44

Figure 2.19 Schéma de circuit électrique ultrasons HC-SR04 avec Arduino Nano.....	45
Figure 2.20 ESP32-S3 pinout	46
Figure 2.21 Schéma de circuit électrique Capteur TCS34725 avec ESP32-S3	47
Figure 2.22 Schéma de circuit électrique Servo moteur SG90 avec ESP32-S3.....	48
Figure 2.23 Schéma de circuit électrique Moteur pas à pas 17HS4023 avec Moteurs L298N et ESP32-S3.....	50
Figure 2.24 Schéma de câblage global du robot mobile intelligent	51
Figure 2.25 Interface Arduino IDE.....	54
Figure 2.26 Interface Arduino IDE expliquée.	55
Figure 2.27 Organigramme de Suivi de ligne avec QTR-8A.....	58
Figure 2.28 Organigramme de fonctionnement du système de détection de couleur et de commande des actionneurs.....	61
Figure 3.1 Architecture du système de haut niveau.....	67
Figure 3.2 Centre de contrôle RoboStock (Tableau de bord principal).....	69
Figure 3.3 Moniteur de robot.....	70
Figure 3.4 Analyse de la performance.....	71
Figure 3.5 Gestion des zones.....	72
Figure 3.6 Ajouter un robot	73
Figure 3.7 Trajectoire du robot lors du test de suivi de ligne.....	74
Figure 3.8 Détection correcte de la couleur par le capteur TCS34725	75
Figure 3.9 Séquence de levage et de dépose d'un objet par le robot.....	76
Figure 3.10 Capture d'écran de l'interface web affichant les données du robot.....	77
Figure 3.11 Exécution d'une mission complète par le robot (détection, transport et dépôt d'un objet).....	78

Liste des tableaux

Tableau 1.1 Caractéristiques techniques de Microcontrôleur ESP32-S3.....	8
Tableau 1.2 Caractéristiques techniques de l'Arduino Nano	10
Tableau 1.3 Caractéristiques techniques de Moteurs L298N.....	12
Tableau 1.4 Caractéristiques techniques du Moteurs DC 3,6V	13
Tableau 1.5 Caractéristiques techniques du capteur QTR-8A	15
Tableau 1.6 Caractéristiques techniques du Capteur à Ultrasons (HC-SR04).....	17
Tableau 1.7 Caractéristiques techniques du Batterie lithium-ion	18
Tableau 1.8 Caractéristiques techniques du Capteur de Couleur RGB TCS3472	20
Tableau 1.9 Caractéristiques techniques de Stepper Motors 17HS4023.....	21
Tableau 1.10 Caractéristiques techniques de Servomoteur MG90S.....	22
Tableau 2.1 Connexion du module QTR-8A à l'Arduino Nano (mode analogique	41
Tableau 2.2 Connexion des moteurs DC à l'Arduino Nano via le module L298N	43
Tableau 2.3 les broches entre le capteur HC-SR04 et l'ESP32-S.....	44
Tableau 2.4 Connexion d'un capteur I2C à l'ESP32-S3.....	46
Tableau 2.5 Connexion d'un moteur servo à l'ESP32-S3	48
Tableau 2.6 Connexion du moteur pas à pas 17HS4023 au module L298N	49
Tableau 2.7 Connexion des broches de commande entre l'ESP32-S3 et le L298N	49
Tableau 2.8 Connexions d'alimentation du circuit	50

Introduction générale

L'essor de l'industrie moderne a profondément transformé les modèles traditionnels de production et de logistique, en favorisant l'intégration des technologies avancées et des systèmes cyber-physiques. Ce changement structurel redéfinit les méthodes commerciales et ouvre la voie à une robotisation accrue, notamment dans les entrepôts, qui jouent un rôle central dans le contexte de la mondialisation des chaînes d'approvisionnement et de l'essor du commerce électronique.

Face aux défis actuels en matière d'efficacité, de réactivité et de contrôle, les solutions mécatroniques apparaissent comme des réponses innovantes de premier plan, promettant à la fois une grande précision opérationnelle et une adaptabilité aux flux logistiques en constante évolution.

Notre projet s'inscrit dans cette dynamique d'innovation. Il vise à concevoir et à mettre en œuvre un système intelligent et automatisé pour optimiser le stockage et les opérations logistiques au sein d'un centre de distribution.

Le robot développé représente une synergie entre la mécanique de précision, l'électronique embarquée et les algorithmes intelligents. Il repose sur une architecture modulaire, articulée autour d'un microcontrôleur ESP32-S3, doté de capacités de traitement avancées et d'une connectivité sans fil adaptée aux applications industrielles (Expressif Framework, 2022).

Le guidage du robot est assuré par un réseau de capteurs infrarouges QTR-8RC, permettant un suivi précis des lignes au sol. La commande des moteurs à courant continu est confiée à un module L298N, qui contrôle quatre moteurs assurant la propulsion et la direction.

L'autonomie énergétique est assurée par une batterie lithium 12 V, associée à un système de gestion optimisé pour les cycles de fonctionnement intensifs.

Sur le plan logiciel, une stratégie de navigation hybride est mise en place, combinant un contrôleur PID pour le suivi de ligne à des mécanismes de prise de décision adaptés aux points de bifurcation. Cette approche permet un ajustement dynamique du comportement du robot, garantissant à la fois une progression fluide sur les lignes droites et des mouvements précis entre les zones de chargement et de déchargement.

L'ajout d'un capteur de couleur permet en outre de classer les objets selon leur teinte, facilitant leur tri automatique vers les zones de stockage appropriées.

Une autre composante essentielle du système est son interface de supervision web, qui établit un lien entre la couche physique opérationnelle et les systèmes d'information de gestion. En exploitant les capacités Wi-Fi du microcontrôleur ESP32-S3, cette interface permet d'accéder en temps réel à des données telles que le niveau de la batterie, la position du robot ou encore le nombre d'objets traités. Elle offre également des fonctionnalités de commande à distance et de gestion de l'alimentation, assurant ainsi une flexibilité opérationnelle accrue.

Dans l'ensemble, ce projet met en lumière l'importance d'une approche intégrée, combinant à la fois la conception matérielle et les compétences logicielles, dans le contexte de la robotisation des processus logistiques. Il ouvre la voie à une mise en œuvre concrète des avancées technologiques de l'industrie 4.0 dans la gestion des flux physiques, et contribue à l'émergence des centres de distribution intelligents de nouvelle génération [1].

Ce mémoire est composé de trois chapitres :

- Le premier chapitre présente les concepts généraux de manière globale.
- Le deuxième chapitre est consacré à la conception et à la réalisation du robot mobile intelligent.
- Le troisième chapitre traite de l'interface web, des résultats obtenus et des perspectives envisagées.

Enfin, une conclusion générale vient clore ce mémoire.

Chapitre 1 : Concepts généraux

1.1 INTRODUCTION

Dans la scène mécanique d'aujourd'hui, freinée par la numérisation croissante des formulaires, la mécanisation des cadres de gestion des centres de distribution est devenue un arrangement de base pour répondre aux conditions préalables avancées en matière de vitesse, de précision et de diminution. Ce questionnement sur l'entreprise reprend cette dérive en proposant l'amélioration d'un robot astucieux capable de travailler de façon autonome dans un environnement contrôlé. Notre gadget combine la prise de route exacte selon un chemin prédéfini, la reconnaissance et le tri des objets conformes à leurs caractéristiques colorimétriques, ainsi qu'une interface d'observation plus lointaine. Ce premier chapitre vise à présenter les établissements de ce cadre mécanisé en mettant en évidence ses principales fonctionnalités, ses destinations opérationnelles et la conception des équipements choisis pour répondre à ces besoins particuliers. Il décrit également la technique reçue pour la détermination des diverses composantes, choix guidés par un ajustement idéal entre exécution spécialisée, qualité inébranlable et impératifs financiers - tout en ouvrant la voie aux avancées futures du cadre

1.2 Définitions

1.2.1 La robotique :

La robotique est l'ensemble des techniques pour automatisée une machine où un robot grâce à un système de commande intelligent a base d'un microprocesseur qui a le rôle de faire tous les calculs et la décision des opérations pour commander les actionneurs (moteur, servo moteur, vérin, vanne...) [6].

1.2.2 Le robot :

Un robot est un automate doté de capteurs et d'effecteurs lui donnant une capacité d'adaptation et de déplacement proche de l'autonomie. Un robot est un agent physique réalisant des tâches dans l'environnement dans lequel il évolue[6].

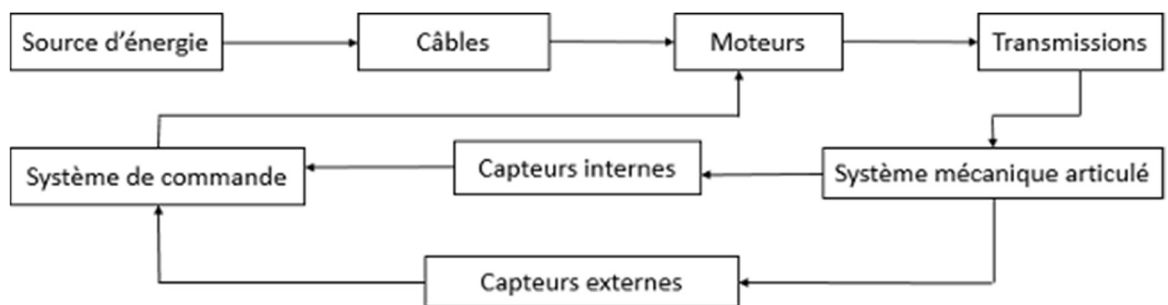


Figure 1.1 Parties commande- contrôle et opérative d'un robot

1.2.3 Automates :

Un automate est un dispositif se comportant de manière automatique, c'est-à-dire sans l'intervention d'un humain [6].

1.3 Historique

Histoire du robot industriel :

- **1920 : L'apparition du mot « robot » :**

Le mot « robot » trouve ses origines dans le dialecte tchèque, où son précurseur « robota » signifie « travail forcé ». Il a été introduit en 1920 par l'écrivain tchèque Karel Čapek dans sa pièce *Rossum's Universal Robots*.

- **1961 : Unimation, le premier robot mécanique :**

Descendant direct des télémanipulateurs développés pour les applications nucléaires, le

premier robot mécanique a été commercialisé en 1961 par la société américaine Unimation (devenue plus tard Stäubli Unimation), fondée par George Devol et Joseph Engelberger. Il a été utilisé pour la première fois sur les chaînes de montage de General Motors. Grâce à son bras articulé de 1,5 tonne, ce robot pouvait manipuler des pièces de fonderie pesant jusqu'à 150 kg.

- **1972 : Première chaîne de production entièrement robotisée :**

Nissan a lancé la première chaîne de production entièrement automatisée par des robots. Selon une étude menée par la Fédération Internationale de Robotique (IFR), 2,142 millions de robots ont été fabriqués entre les années 1960 et la fin de 2010. Les analystes estiment qu'aujourd'hui, entre 1 et 1,3 million de robots travaillent dans les usines à travers le monde.

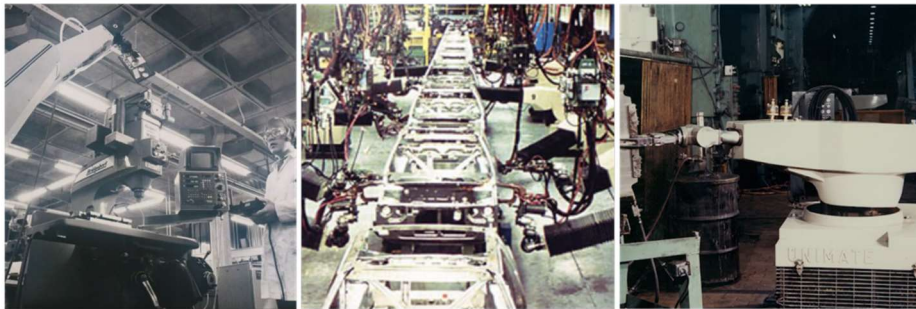


Figure 1.2 Chaîne de production robotisée

1.4 TACHES PRINCIPALES DU ROBOT

Le robot mobile intelligent développé dans le cadre de ce projet est conçu pour exécuter un ensemble de tâches spécifiques de manière autonome et efficace. Ces tâches sont directement liées à sa mission principale, qui consiste à identifier, transporter et classer des objets dans un environnement structuré tel qu'un entrepôt. Les fonctions décrites ci-dessous illustrent les différentes capacités du robot, depuis la navigation jusqu'à la gestion des imprévus.

1.4.1 Navigation Autonome Précise :

Le robot est conçu pour suivre un itinéraire bien défini au sol, généralement sous la forme d'un carré de 120 x 120 cm. Grâce aux capteurs infrarouges QTR-8RC, il peut maintenir son alignement avec la ligne directrice. Il est également capable de négocier les virages et de franchir les intersections avec une grande précision, garantissant une navigation fiable et sans déviation.

1.4.2 Tri et Classement par Couleur :

L'une des fonctions clés du robot est d'identifier la couleur des objets à transporter à l'aide du capteur de couleur (type RGB ou TCS3200). Une fois la couleur détectée, le robot l'associe à une zone de stockage spécifique. Ce tri automatisé permet d'optimiser le rangement selon un code couleur prédéfini.

1.4.3 Transport et Placement Automatisé :

Le robot assure le transport autonome des marchandises depuis leur point de départ jusqu'à l'emplacement correspondant à leur couleur. Il dépose les objets avec précision sur les étagères prévues à cet effet. De plus, il est capable de répéter plusieurs cycles de transport consécutifs sans commettre d'erreurs, en s'arrêtant avec exactitude aux points de dépôt.

1.4.4 Supervision et Contrôle à Distance :

Le système intègre une interface web permettant d'afficher en temps réel des informations essentielles telles que le niveau de la batterie, la position actuelle du robot ou encore le nombre d'objets traités. Cette interface offre également un mode de contrôle manuel à distance via WiFi, avec des fonctions comme le démarrage, l'arrêt ou le paramétrage du robot.

1.4.5 Gestion des Aléas :

Afin de garantir un fonctionnement robuste, le robot est capable de détecter et corriger automatiquement les écarts de trajectoire. Il peut également gérer les interruptions dues à une batterie faible ou à la présence d'un obstacle imprévu. En cas d'erreur, un mécanisme de redémarrage sécurisé permet de reprendre la mission sans risque de perturbation.

1.5 Composants Du Robot

Ce chapitre présente une analyse détaillée de chaque élément constitutif du système robotique, en mettant l'accent sur leurs caractéristiques techniques, leurs rôles spécifiques et leur intégration dans l'architecture globale.

1.5.1 Microcontrôleur ESP32-S3 :

a. Définition

L'ESP32-S3, fabriqué par Espressif Systems, pourrait bien être un microcontrôleur de contrôle moo connu pour son double centre de construction Xtensa LX7 et son support à proximité du Wi-Fi et du Bluetooth. Extrêmement apprécié pour ses capacités informatiques avancées et sa polyvalence, il est particulièrement bien adapté aux besoins des superviseurs implantés avancés, en particulier dans les gammes du Net of Things (IoT), l'indépendance mécanique flexible et la robotisation mécanique. Son intégration dans les modèles cyber-physiques de l'Industrie 4.0 lui permet de répondre aux demandes croissantes de préparation en temps réel, d'organisation dispersée et de gestion brillante des ressources[2].

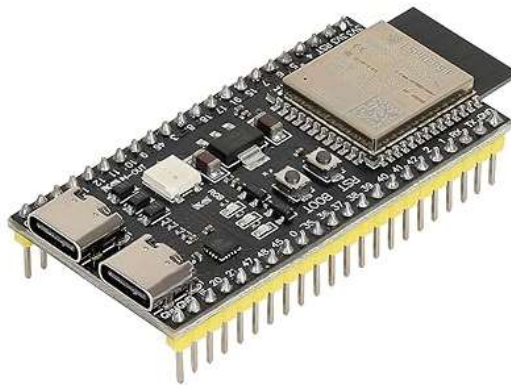


Figure 1.3 Microcontrôleur ESP32-S3

b. Fonction Principale

- Cerveau du système : Supervise le traitement des informations, le contrôle des périphériques et la communication sans fil.
- Coordination : Synchronise les capteurs, les actionneurs et l'interface web.

c. Spécifications Techniques

Paramètre	Valeur
Architecture	Dual-core 32-bit (LX7)
Fréquence d'horloge	Jusqu'à 240 MHz
Connectivité	Wi-Fi 4 (802.11 b/g/n) + Bluetooth 5
Entrées/Sorties	45 GPIO
Tension d'alimentation	3.0–3.6 V

Tableau 1.1 Caractéristiques techniques de Microcontrôleur ESP32-S3

d. Rôle dans le Projet

Le microcontrôleur joue un rôle central dans le fonctionnement du robot. Il lit les signaux analogiques fournis par les capteurs QTR-8A afin d'assurer un suivi précis de la ligne tracée au sol. Il peut également traiter les données issues du capteur de couleur, si ce dernier est intégré, pour permettre la reconnaissance et le tri des objets selon leur teinte.

1.5.2 L'Arduino Nano

a. Définition

L'Arduino Nano peut être une carte d'amélioration de microcontrôleur compacte ordinairement basée sur le ATmega328P. Il est prévu d'offrir les mêmes fonctionnalités que l'Arduino UNO, mais dans une organisation réduite, encourageant son intégration dans des projets où l'espace est limité [10].

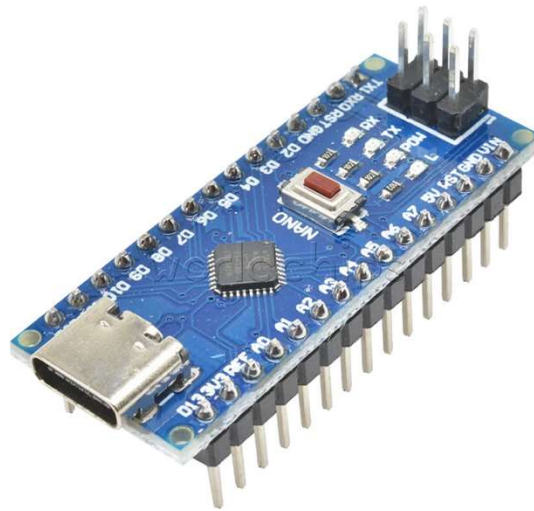


Figure 1.4 Arduino Nano

b. Fonction Principale

La plupart du travail de l'Arduino Nano est de servir l'unité centrale de manipulation dans des cadres insérés. Il permet d'examiner des capteurs, de contrôler des actionneurs (moteurs, Driven, transferts...), de communiquer avec d'autres modules par le biais de conventions telles que I2C, SPI ou UART, et d'exécuter un programme implanté.

c. Spécifications Techniques

Caractéristique	Valeur
Microcontrôleur	ATmega328P
Fréquence d'horloge	16 MHz
Mémoire Flash	32 Ko (dont 2 Ko utilisés par le
Broches numériques (E/S)	14 (dont 6 PWM)
Entrées analogiques	8
Interface USB	Mini-USB
Dimensions	45 x 18 mm

Tableau 1.2 Caractéristiques techniques de l'Arduino Nano

d. Rôle dans le Projet

Dans notre extension, l'Arduino Nano joue un rôle central en tant qu'unité de contrôle auxiliaire. Il est utilisé pour traiter des sous-tâches particulières du robot, telles que le contrôle de la structure de levage, le contrôle des moteurs auxiliaires ou l'examen des capteurs de base, afin de vider la plupart des microcontrôleurs (ESP32) de certaines opérations. Très oblige de son compacte estimation et compatibilité avec une large gamme de capteurs et modules, l'Arduino Nano permet d'amplifier facilement l'utilité du robot sans compliquer l'ingénierie générale. Il agit donc en tant que coprocesseurs, permettant une meilleure transmission des tâches et un fonctionnement plus fluide du cadre embarqué.

1.5.3 Module de Contrôle de Moteurs L298N

a. Définition

Le L298 est un circuit intégré robuste disponible en boîtiers MultiWatt-15 et PowerSO-20. Conçu comme un double pont en H, il permet de contrôler des charges inductives (moteurs à courant continu, moteurs pas-à-pas, relais) avec des courants et des tensions élevés, tout en acceptant des signaux logiques TTL standards [\[3\]](#).

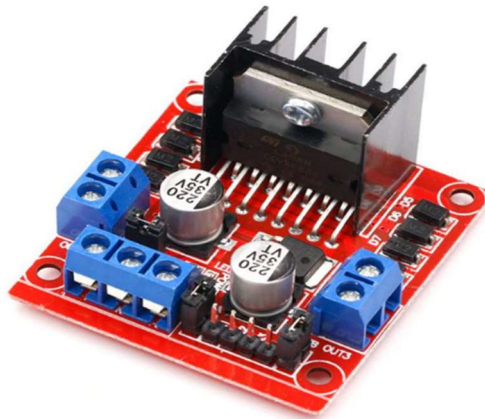


Figure 1.5 Contrôle de Moteurs L298N

b. Fonction Principale

- Contrôle bidirectionnel : Permet aux moteurs de pivoter dans les deux sens.
- Régulation de vitesse : Réalisée grâce à la modulation de largeur d'impulsion (PWM).
- Protection intégrée : Diodes de roue libre pour éviter les surtensions.

c. Spécifications Techniques

Parameter	Valeur
Tension alimentation	5 – 35 V
Courant max par Pont	2 A (4 A en pic)
Tension logique	5 V (compatible 3.3 V)

Fréquence PWM max	25 kHz
Pertes de puissance	Dissipation terminus moderate
Dimensions	43 x 27 mm (avec radiator)

Tableau 1.3 Caractéristiques techniques de Moteurs L298N

d. Rôle dans le Projet

Le système de commande des moteurs repose sur l'utilisation de deux modules L298N, ou d'un module double capable de piloter simultanément les quatre moteurs du robot. Cette configuration permet un contrôle indépendant des paires de moteurs gauche et droite, assurant ainsi une meilleure maniabilité du robot. De plus, les modules intègrent des diodes de protection qui sécurisent le circuit contre les courants inverses, évitant tout dommage aux composants électroniques.

1.5.4 Moteurs DC 3,6V

a. Définition

Les moteurs à courant continu (DC) 3,6 V sont des actionneurs électromécaniques largement utilisés dans les systèmes embarqués et la robotique mobile. Leur rôle principal est de convertir l'énergie électrique en mouvement rotatif, permettant ainsi la propulsion ou l'actionnement de mécanismes. Grâce à leur taille compacte, leur commande simple et leur faible consommation d'énergie, ils représentent une solution privilégiée pour les applications embarquées nécessitant à la fois efficacité et miniaturisation [4] .



Figure 1.6 Moteurs DC 3,6V

b. Fonction Principale

- Propulsion : Entraînement des roues pour le déplacement du robot.
- Contrôle de vitesse : Régulation via modulation PWM (rapport cyclique).

c. Spécifications Techniques

Parameter	Valeur
Tension nominal	3,6 V (functional 3-6V)
Couple	Foible (adapts aux robots lagers)

Tableau 1.4 Caractéristiques techniques du Moteurs DC 3,6V

d. Rôle dans le Projet

Le système de propulsion du robot repose sur deux moteurs dédiés à la traction, qui actionnent les roues motrices, et deux autres moteurs servant au contrôle directionnel ou jouant le rôle de roues folles selon la configuration. La vitesse des moteurs est modulée par le module L298N grâce à un signal PWM (modulation de largeur d'impulsion), ce qui permet d'ajuster la vitesse

en temps réel. Le sens de rotation des moteurs peut également être inversé, facilitant ainsi les manœuvres comme les demi-tours ou les corrections de trajectoire.

1.5.5 Capteurs Infrarouges QTR-8A

a. Définition

Le QTR-8A pourrait être une barre de 8 capteurs de réflectance infrarouge (IR), des phototransistors de coordination et des LED IR sur une seule carte de circuit imprimé. Conçu pour l'emplacement précis des lignes ou des repères au sol, il mesure les variétés de réflexion lumineuse (surfaces claires / sombres) et donne des rendements analogiques ou informatisés [4].

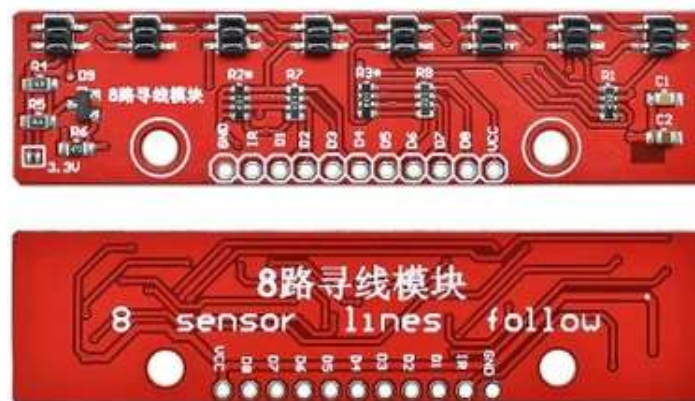


Figure 1.7 Compostant de QTR-8A

b. Fonction Principale

- Détection de lignes noires sur fond clair (ou inversement)
- Mesure analogique de la réflectance surface
- Navigation précise pour robots suiveurs de ligne

c. Spécifications Techniques

Paramètre	Valeur
Technologie	8 phototransistors IR + Leeds
Tension d'alimentation	3.3V - 5V DC
Plage de détection	0.3mm - 3mm (optimale)
Temps de réponse	< 1ms
Sortie	8 signaux analogiques (0-5V)
Dimensions	72mm x 10mm x 5mm
Consommation	100mA max (tous Leeds actifs)

Tableau 1.5 Caractéristiques techniques du capteur QTR-8A

d. Rôle dans le Projet

Le capteur QTR-8RC est utilisé pour assurer la détection précise de la ligne au sol grâce à l'activation de six capteurs centraux. Ces capteurs permettent au robot de suivre la trajectoire avec précision, tandis qu'un algorithme PID (Proportionnel, Intégral, Dérivatif) assure la correction en temps réel des écarts de direction. Le système est également capable d'identifier les croisements lorsque plusieurs capteurs sont activés simultanément, ce qui permet de déclencher automatiquement les routines de virage appropriées.

1.5.6 Capteur à Ultrasons Recommandé (HC-SR04)

a. Définition

Le capteur à ultrasons est un capteur qui fonctionne selon un principe similaire à celui d'un radar ou d'un sonar. Il émet un son à haute fréquence et calcule l'intervalle de temps entre

l'émission du signal et la réception de l'écho. Ainsi, un capteur à ultrasons peut être utilisé pour mesurer la distance [7].



Figure 1.8 Compassant Capteur à Ultrasons Recommandé (HC-SR04)

b. Fonction Principale

Le capteur ultrasonique fonctionne en fonction de la vitesse du son. Elle transmet des battements à haute fréquence qui, après réflexion d'une protestation, reviennent en retentissant. En mesurant le temps entre la sortie et la collecte, le capteur décide de la séparation. Un battement de 10 s déclenche un arrangement de 8 battements à 40 kHz. Le terme de l'indicateur sonore est

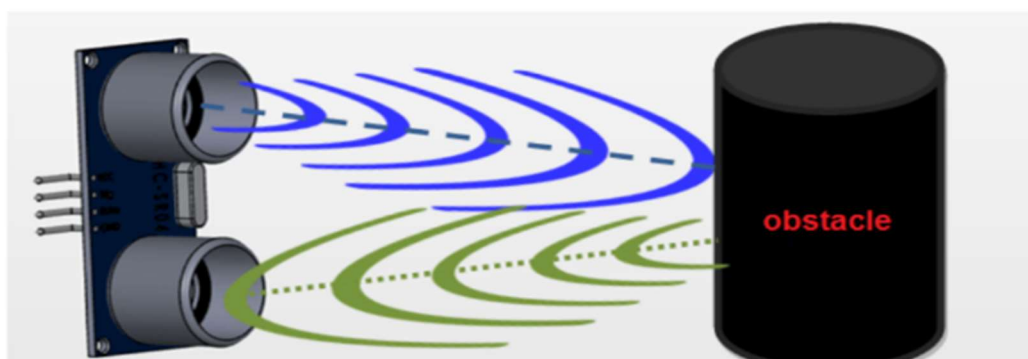


Figure 1.9 Détection d'obstacle par ultrason.

utilisé pour calculer la durée du trajet aller-retour. La plupart des matériaux, dans tous les cas de leur couleur ou simplicité, peuvent être reconnus [8] .

c. Spécifications Techniques

Paramètre	Valeur
Plage de détection	2 cm – 400 cm
Précision	± 3 mm
Dimensions	45 x 20 x 15 mm

Tableau 1.6 Caractéristiques techniques du Capteur à Ultrasons (HC-SR04)

1.5.7 Batterie lithium ion

a. Définition

Une batterie 18650 est une batterie lithium-ion qui peut servir d'alimentation pour des projets mobiles, le nom dérive des dimensions spécifiques de la batterie : 18 mm x 65 mm, pour l'échelle, c'est plus grand qu'une pile AA. La batterie 18650 a une tension de 3,7 V et a entre 2600 mAh et 3500 mAh (milli-ampères-heures), les 2 batteries utilisés dans ce projet ont une capacité de 2600mah chacune, ainsi qu'une tension de 3.7V, si on en assemble 3, on peut avoir 12 V environ, ce qui suffisant pour alimenter les 3 moteurs qui nécessitent une tension de 12V pour tourner en plein régime [9].



Figure 1.10 Batterie lithium-ion rechargeable

b. Fonction Principal

- Capacité et approvisionnement en énergie électrique polyvalente
- Alimentation de commande pour les cadres insérés et portatifs
- Cycles de charge/décharge monotones

c. Spécifications Techniques

Paramètre	Valeur Typique
Tension nominale	3.6V ou 3.7V
Tension de charge max	4.2V $\pm$ 0.05V
Plage température	-20°C à 60°C (opération)

Tableau 1.7 Caractéristiques techniques du Batterie lithium-ion

1.5.8 Capteur de Couleur RGB TCS3472

a. Définition

Le TCS3472 est un capteur de couleur numérique intégré qui détecte le rouge, le vert, le bleu et la lumière (transparente) à l'aide de photodiodes filtrées. Il communique généralement via l'interface I²C et permet de mesurer la couleur d'un objet en termes de composants RVB [12].

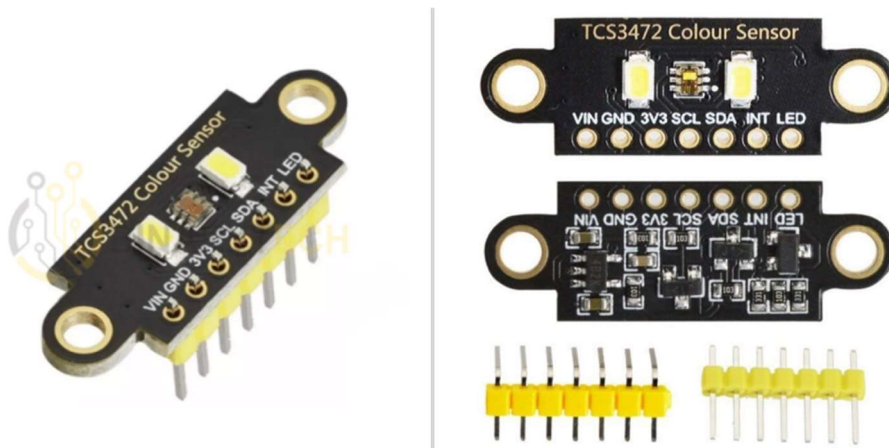


Figure 1.11 Capteur de Couleur RGB TCS3472

b. Fonction Principale

Le plus de travail du TCS3472 est d'identifier la couleur d'un objet trouvé avant lui en donnant les valeurs des composants ruddy, vert, bleu et lumière. Ces valeurs sont utilisées dans des applications telles que la reconnaissance des couleurs, le tri robotisé ou l'altération de l'éclairage.

c. Spécifications Techniques

Paramètre	Valeur
Interface	I ² C (jusqu'à 400 kHz)
Tension d'alimentation	2.7 V à 3.6 V
Plage spectrale	300 nm – 700 nm
Température de service	-40 °C à +85 °C

Tableau 1.8 Caractéristiques techniques du Capteur de Couleur RGB TCS3472

1.5.9 Stepper Motors 17HS4023

a. Définition

Le 17HS4023 est un moteur pas à pas bipolaire de type NEMA 17, largement utilisé dans des applications de précision telles que les imprimantes 3D, les robots et les machines CNC. Il est reconnu pour sa précision de positionnement, sa compacité et sa fiabilité dans les systèmes à commande contrôlée [13].



Figure 1.12 Stepper Motors 17HS4023

b. Fonction Principale

Le 17HS4023 convertit des impulsions électriques en mouvements mécaniques précis. Chaque impulsion correspond à un angle de rotation défini, permettant ainsi un contrôle très précis de la position, même sans système de rétroaction (contrôle en boucle ouverte).

c. Spécifications Techniques

Caractéristique	Valeur
Type	Bipolaire (2 phases)
Angle par pas	1.8° (200 pas/tr)
Courant nominal	1.7 A / phase
Poids	200 g (env.)

Tableau 1.9 Caractéristiques techniques de Stepper Motors 17HS4023

1.5.10 Servomoteur MG90S

a. Définition

Le servomoteur MG90S est un servomoteur à échelle miniaturisée avec des engrenages métalliques conçus pour offrir un couple plus élevé et une solidité supérieure aux servomoteurs en plastique, tout en maintenant une estimation compacte. Il est couramment utilisé dans la technologie mécanique et de démonstration des entreprises.[13]



Figure 1.13 Servo moteur MG90S

b. Fonction Principale

Le servomoteur MG90S est principalement utilisé pour un positionnement précis (virage d'un pivot à un point indiqué). Il peut être commandé en PWM pour déplacer un bras, une pince ou tout autre instrument mécanique.

c. Spécifications Techniques

Caractéristique	Valeur
Tension de fonctionnement	4.8V à 6.0V
Couple maximum	$\approx 2.2 \text{ kg cm}$ à 4.8V
Vitesse de rotation	0.1 s/60° à 4.8V
Poids	$\approx 13.4 \text{ g}$
Angle de rotation	180° (PWM)

Tableau 1.10 Caractéristiques techniques de Servomoteur MG90S

d. Rôle dans le Projet

Dans le cadre du développement du robot mobile intelligent, le MG90S est utilisé pour actionner le mécanisme de levage et de maintien. Il est particulièrement apprécié pour son

couple élevé et sa fiabilité mécanique. Ce servo moteur permet un contrôle précis des mouvements, indispensables pour manipuler efficacement des objets ou des produits.

e. Avantages par Rapport aux Alternatives

Le servomoteur MG90S se distingue par sa légèreté, son couple élevé et sa précision de contrôle, ce qui le rend idéal pour les applications embarquées et robotiques. Par rapport à des modèles plus grands ou moins précis, il offre un rapport poids/puissance avantageux, une réponse rapide et une faible consommation d'énergie, tout en maintenant un coût abordable [14].

1.6 Intégration mécanique et structurelle du robot

L'assemblage mécanique du robot constitue un élément essentiel de sa conception, garantissant à la fois solidité, précision et mobilité. Le châssis, principalement fabriqué en MDF, offre la robustesse nécessaire tout en étant suffisamment léger pour faciliter les déplacements du robot. Les capteurs de ligne et de couleur sont stratégiquement placés à l'avant afin d'assurer une détection précise des lignes au sol et des couleurs des emplacements (ou racks). Les moteurs, roues et composants électroniques sont solidement fixés au châssis afin d'assurer un fonctionnement fiable et de maintenir l'intégrité de l'ensemble du système.

Le robot intègre également un système de levage à ciseaux motorisé, lui permettant de soulever et de déposer des marchandises sur des rayonnages, ce qui renforce son autonomie et sa flexibilité dans un environnement industriel.

Ainsi, la conception mécanique du robot allie robustesse et modularité, tout en facilitant l'accès aux composants pour l'entretien et les éventuelles améliorations.

1.7 Site Web de contrôle et de suivi du robot

Dans le cadre du projet, un site web a été conçu pour permettre une interaction conviviale et centralisée avec le robot. Ce site web assure plusieurs fonctionnalités essentielles pour le bon fonctionnement et le suivi du robot mobile. Il permet notamment :

- De contrôler le robot à distance (démarrage, arrêt et gestion des mouvements),
- De visualiser la couleur du colis détectée par le capteur,
- De suivre la position du robot sur une carte en temps réel,
- Et de recevoir des notifications relatives à l'état du système (batterie, erreurs éventuelles, etc.).

Pour réaliser ce site web, nous avons utilisé les technologies **HTML**, **CSS** et **JavaScript** pour l'interface utilisateur, associées à un microcontrôleur ESP32-S3 WROOM côté robot pour la communication et le traitement des données. La communication entre le robot et le site web est assurée via un module Wi-Fi intégré à l'ESP32-S3, utilisant le protocole HTTP ou WebSocket pour échanger les informations. Cette approche permet au robot d'envoyer ses données en temps réel au serveur web, qui les affiche ensuite à l'utilisateur.

Ainsi, ce site web joue un rôle central dans le système global, en facilitant à la fois le contrôle et la surveillance du robot mobile.

1.8 Conclusion

Dans ce chapitre, nous avons tout d'abord posé les fondations du projet en présentant le cadre général, les motivations ainsi que les fonctionnalités principales attendues de notre robot intelligemment polyvalent. Ce robot a été conçu pour répondre aux exigences croissantes de l'industrie 4.0 en matière de robotisation, de précision et d'adaptabilité dans la gestion des stocks.

Nous avons également identifié les différents composants utilisés et précisé leur rôle spécifique dans la réalisation des tâches du robot, telles que le déplacement autonome, la détection des couleurs et la manipulation d'objets.

L'analyse fonctionnelle et matérielle développée dans ce chapitre permet une meilleure compréhension de l'architecture globale du système, ainsi que des contraintes techniques auxquelles nous avons été confrontés. Cette vue d'ensemble est essentielle pour aborder de manière structurée la suite du projet.

Dans le chapitre suivant, nous approfondirons la conception et la mise en œuvre du système embarqué, en détaillant l'architecture logicielle, le choix des algorithmes de contrôle et de navigation, ainsi que l'intégration des capteurs et actionneurs. Cette transition marque le passage de la phase de conception générale à l'étape technique du projet, où chaque choix de conception jouera un rôle déterminant dans le bon fonctionnement du robot.

Chapitre 2 : Conception et Réalisation du Robot Mobile Intelligent

2.1 Introduction

Ce chapitre est consacré à la conception et à la mise en œuvre du robot mobile. Un robot mobile est une structure programmée, capable de se déplacer de manière autonome dans un environnement donné afin d’y accomplir des tâches spécifiques, tout en interagissant en permanence avec son environnement.

Dans ce contexte, notre robot est conçu pour se déplacer de façon autonome dans un entrepôt, en suivant une ligne noire, en identifiant la couleur des marchandises, puis en les transportant et en les déposant à leur emplacement désigné.

Le chapitre débute par une présentation de l’architecture générale du robot, en détaillant l’organisation des modules matériels et logiciels. Il se poursuit avec la description de la structure mécanique, incluant la conception du châssis, la connexion des moteurs, des roues et des capteurs, ainsi que le système de levage et de dépose.

Une autre section est consacrée au câblage électronique, à l’explication des connexions entre composants, au schéma de câblage, et à la gestion de l’alimentation et du contrôle.

Enfin, le chapitre se conclut par une présentation de la programmation du microcontrôleur ESP32, en mettant l’accent sur les fonctionnalités clés telles que le suivi de ligne, la détection des couleurs, le contrôle des moteurs et la gestion des erreurs.

Ce chapitre fournit ainsi une vue d’ensemble détaillée de la réalisation technique du robot.

2.2 Architecture générale du système

2.2.1 Fonctionnement du projet

Le robot mobile intelligent conçu dans le cadre de ce projet repose sur une architecture modulaire et optimisée, visant à automatiser le tri et la gestion des marchandises dans un entrepôt. Au cœur du système se trouve le microcontrôleur ESP32-S3 WROOM, qui joue le rôle de centre de commande principal. Il assure la coordination des modules, la collecte et le traitement des données issues des

capteurs, la commande des moteurs, ainsi que la communication sans fil avec une interface web dédiée via le Wi-Fi. Un Arduino Nano complète ce système en tant que contrôleur secondaire, prenant en charge la lecture des capteurs de suivi de ligne QTR-8A et le contrôle des quatre moteurs à courant continu à l'aide d'un module L298N, permettant au robot de se déplacer avec précision.

Le système de tri s'appuie sur deux capteurs de couleur TCS34725 : le premier détecte la couleur des produits afin de les diriger vers l'étagère correspondante (rouge, jaune ou bleue), tandis que le second affiche cette couleur sur le côté du robot à l'aide d'un indicateur lumineux, facilitant ainsi la traçabilité visuelle. Pour le dépôt sur étagère, le robot est équipé d'un moteur pas à pas NEMA 17 (17HS4023), qui permet de lever ou d'abaisser les produits, et d'un servomoteur SG90 pour pousser ou déposer les marchandises avec précision. Ces deux moteurs sont chacun commandés par un module L298N dédié, garantissant un contrôle indépendant et efficace.

La sécurité de navigation est assurée par un capteur à ultrasons HC-SR04, qui permet au robot d'éviter les obstacles et de préserver les structures environnantes. L'alimentation du système repose désormais sur deux batteries de 12 V : la première alimente les moteurs DC, l'ESP32, l'Arduino Nano et les capteurs via des régulateurs de tension ; la seconde est exclusivement réservée à l'alimentation du moteur pas à pas et du servomoteur, assurant ainsi une puissance stable et évitant les chutes de tension lors des manœuvres mécaniques. L'interface web, conçue en HTML, CSS et JavaScript, permet à l'utilisateur de superviser en temps réel l'état du robot, de visualiser les colis détectés, de localiser le robot sur la carte, et de contrôler certaines fonctions à distance. Cette organisation matérielle rigoureuse garantit robustesse, autonomie énergétique et performance, rendant le système parfaitement adapté aux exigences de l'Industrie.

2.2.2 Schéma Bloc Fonctionnel du Système

Le schéma bloc fonctionnel du robot mobile intelligent représente l'architecture logique et modulaire du système, en mettant en évidence les différentes unités matérielles, leurs rôles respectifs, ainsi que les interactions fonctionnelles qui les relient. Cette représentation vise à illustrer de manière claire et synthétique le fonctionnement global du robot, sans entrer dans les détails du câblage physique.

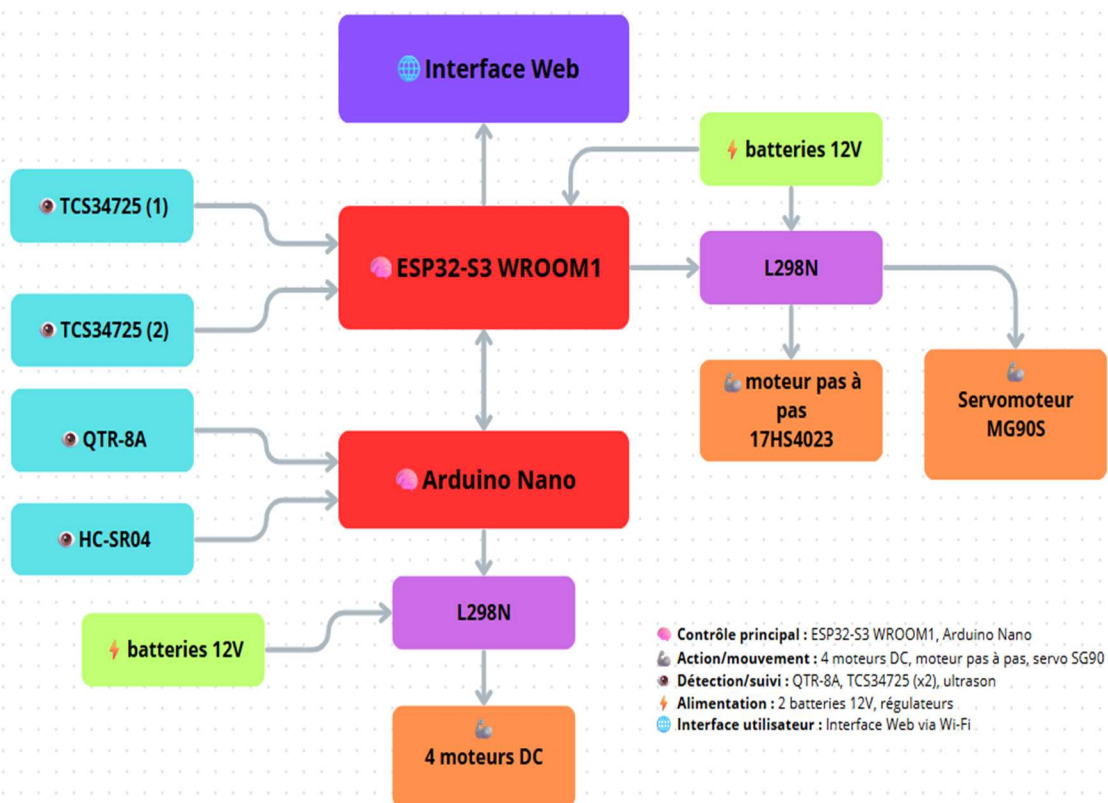


Figure 2.1 Schéma Bloc Fonctionnel du Robot Mobile Intelligent

Au cœur de l'architecture, on retrouve deux unités de contrôle :

- L'ESP32-S3 WROOM, qui agit comme unité centrale de commande, assure la coordination des capteurs intelligents, la gestion des actionneurs critiques (moteur pas à pas et servomoteur), ainsi que la communication Wi-Fi avec une interface web développée en HTML/CSS/JavaScript.
- L'Arduino Nano est utilisé comme contrôleur secondaire, principalement responsable de la gestion du suivi de ligne via le capteur QTR-8A et du pilotage des moteurs DC à travers un module L298N.

Les actionneurs du système incluent :

- Quatre moteurs à courant continu, commandés via l'Arduino Nano et un module L298N, qui assurent le déplacement du robot.
- Un moteur pas à pas (17HS4023) utilisé pour le levage vertical des colis.
- Un servomoteur SG90 utilisé pour le dépôt ou la poussée des produits sur les étagères.
- Ces deux derniers actionneurs sont commandés indépendamment par deux modules L298N distincts connectés à l'ESP32.

L'alimentation électrique est assurée par deux batteries de 12V :

- La première alimente les moteurs DC, les microcontrôleurs et les capteurs via des régulateurs de tension.
- La seconde est exclusivement dédiée à l'alimentation du moteur pas à pas et du servomoteur, garantissant ainsi une puissance stable et évitant toute chute de tension lors des mouvements mécaniques.

Enfin, la supervision et le contrôle à distance sont assurés par une interface web interactive qui permet de visualiser en temps réel les informations issues des capteurs (colis détectés, couleur, position...), d'envoyer des commandes, et de surveiller l'état global du robot. Cette interface communique en temps réel avec l'ESP32 via le Wi-Fi, utilisant des protocoles tels que HTTP ou WebSocket.

2.3 Assemblage mécanique du robot

2.3.1 Structure du châssis du robot

La structure du châssis est le cadre principal du robot astucieusement polyvalent, sécurisant les composants distinctifs tout en garantissant la qualité et la maniabilité nécessaires au développement au sein du centre de dispersion. La plupart de la base est en MDF, qui pourrait être un tissu léger et sécurisé qui fait la différence pour retenir les vibrations et renforcer le poids des appareils électroniques et mécaniques. Cette base adopte une forme rectangulaire avec des coins équilibrés pour garantir la sécurité des marches et maintenir un retrait vital lors de collisions contre les dissuadants. Il joint en outre des espaces de montage délibérément situés pour s'adapter aux modules distinctifs du robot, ainsi que des motifs définis dans le cadre de rainures et de crevasses permettant d'envisager une section de câble et une ventilation progressive des composants électroniques.[22]

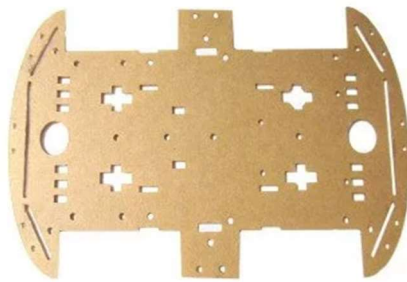


Figure 2.2 Le châssis du robot en MDF

2.3.2 Fixation des roues, moteurs et composants

Le calage des roues, des moteurs et des différents composants du robot a été esquissé pour assurer la solidité, une qualité sans faille et la facilité de soutien. Les quatre roues élastiques jaunes, montées sur des bords en plastique résistant, sont solidement reliées au châssis par des supports en métal vissés sous la plupart des bases. Ces appuis garantissent une disposition

exacte des roues, ce qui permet au robot de se déplacer facilement le long de la ligne sombre sur le sol.

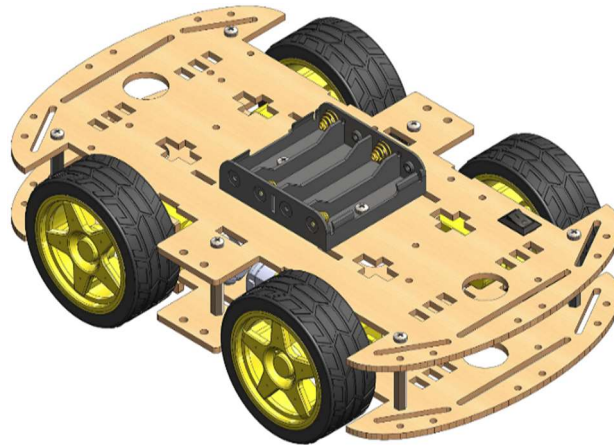


Figure 2.3 Robot Mobile à 3D

Chaque roue est entraînée par un moteur à courant coordonné (CC), qui est monté sous la base du châssis en utilisant des vis et des entretoises. Ces verrous vibrent le tuyau et garantissent un couple suffisant pour que le robot se déplace, en effet lors du transport des produits.



Figure 2.4 Robot Mobile à 4 Roues

Les capteurs QTR-8A sont situés à l'avant du robot, légèrement en dessous du châssis, pour garantir une détection précise des lignes sombres. Ils sont vissés directement sur le châssis donné pour cette raison, ce qui permet de modifier simplement la hauteur et l'inclinaison pour optimiser l'exécution.



Figure 2.5 Fixation du QTR-8A sur le Châssis du Robot

Dans notre conception, deux capteurs TCS3472 ont été installés : le premier est dédié à la détection de la couleur des étagères, tandis que le second est utilisé pour analyser et mémoriser la couleur de chaque produit à transporter, assurant ainsi une correspondance précise entre le produit et sa destination.

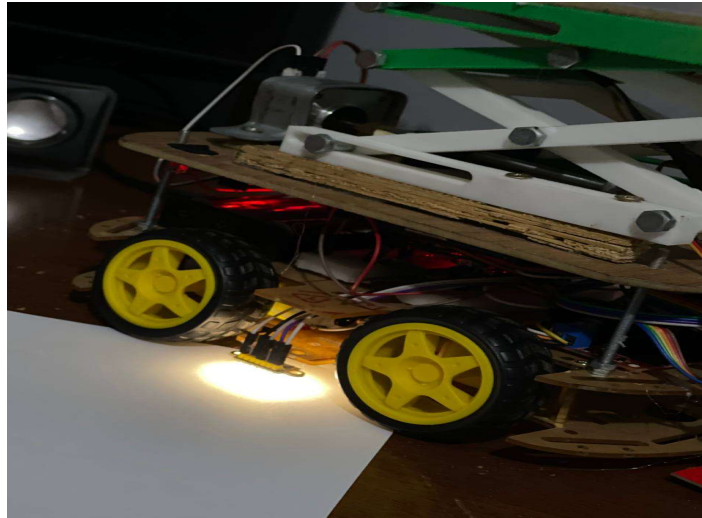


Figure 2.6 Installation du capteur TCS3472 pour la détection

Les modules électroniques, incluant l'ESP32-S3 WROOM, l'Arduino Nano, le pilote moteur L298N sont montés sur le châssis en utilisant des vis et des entretoises. Cette action garantit que les composants sont légitimement ventilés, que le câblage est propre et que les obstructions entre les signaux électriques sont évitées. Les batteries sont placées trop en toute sécurité sur la base du châssis, garantissant un grand ajustement du robot et encourageant leur remplacement lorsqu'il est essentiel.

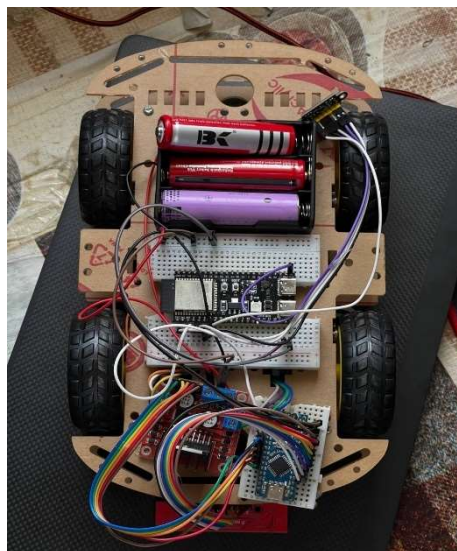


Figure 2.7 Montage Interne du Robot : Arduino Nano, L298N, ESP32 S3 et Batterie 12V

Cette organisation mécanique garantit la vigueur et la qualité inébranlable de la charpente, tout en permettant un accès simple aux composants pour l'entretien et les révisions envisageables.

2.3.3 Système de levage et de dépôt des objets

Le cadre de levage et de largage est une partie fondamentale du robot portable, lui permettant de transporter et de placer les produits dans la bonne position sur les rayonnages. Ce cadre est basé sur un élévateur à ciseaux, choisi pour sa vigueur et sa compacité, ce qui le rend particulièrement approprié pour un robot portable travaillant dans un petit espace tel qu'un centre de distribution.



Figure 2.8 Système de levage

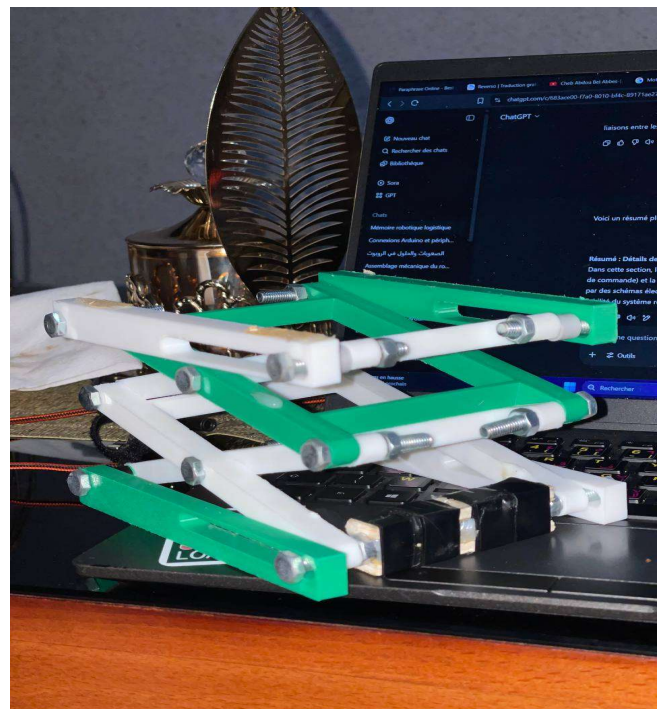


Figure 2.9 Système de dépôt

Logiciel 3D :

Pour concevoir les modules 3D du robot, on a utilisé le logiciel Blender, un outil open-source puissant dédié à la modélisation, l'animation et le rendu 3D. Grâce à ses fonctionnalités avancées et sa grande flexibilité, Blender nous a permis de créer des représentations détaillées et réalistes des différentes parties du robot. Cette modélisation nous a aidé à anticiper les contraintes mécaniques et à affiner le design global avant la phase de fabrication. Les images et rendus présentés dans ce mémoire sont le fruit de ce travail de modélisation.

La nacelle est entraînée par un moteur pas à pas 17HS4023, qui garantit un développement en ligne droite dynamique et précis très apprécié à sa commande de pas. Ce moteur est actionné par un Moteurs L298N, permettant de contrôler avec précision la montée et la descente de la nacelle en fonction de l'éclairage obtenu du microcontrôleur. Avec ce cadre, le robot peut élever la marchandise à la taille requise et les mettre sur le support de comparaison.

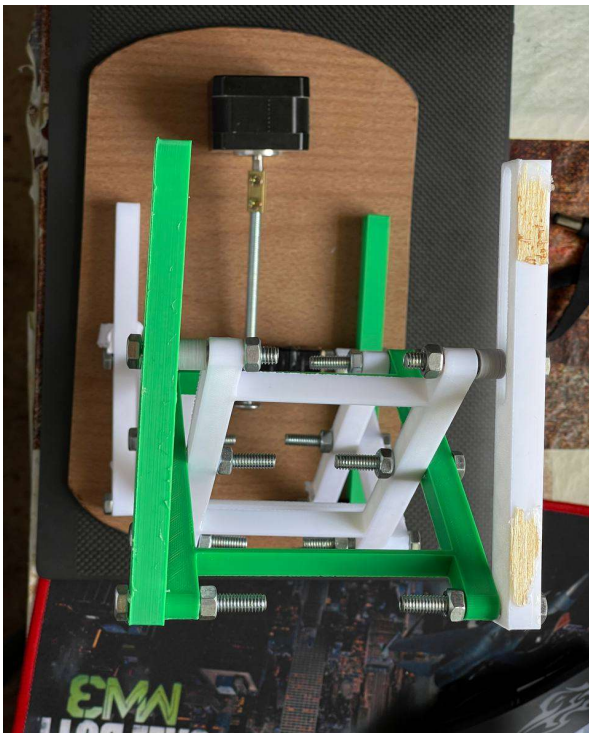


Figure 2.10 Fixation du moteur pas à pas 17HS4023 dans la nacelle du robot

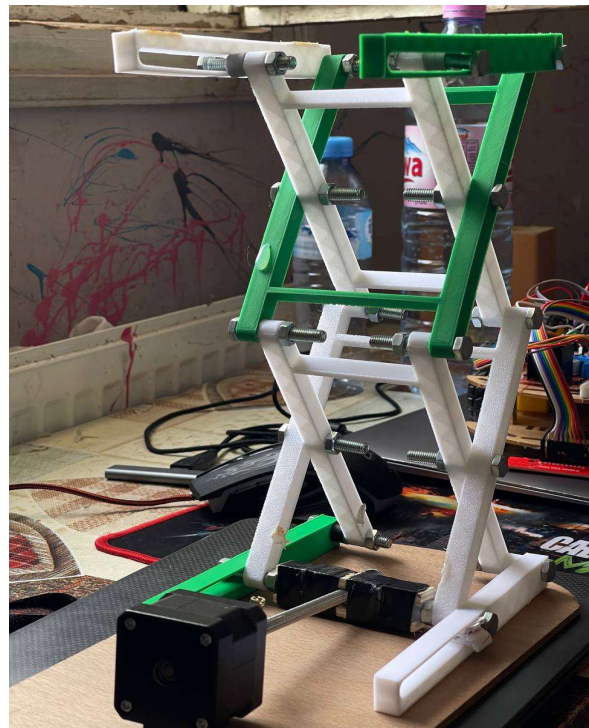


Figure 2.11 Fixation du moteur pas à pas 17HS4023 dans la nacelle du robot

Pour effectuer le véritable témoignage, un moteur servo SG90 est coordonné dans le cadre. le servomoteur SG90 travaille la roue d'adaptation, qui incline facilement le faisceau dans sa zone de visée. La roue d'adaptation donne un développement de révolution contrôlé qui pousse le faisceau vers l'avant, tandis que le rail guide et stabilise la cargaison au milieu de ce mouvement. Une fois le faisceau transporté, l'adaptation retourne à sa position unique très liée au contrôle du servomoteur, permettant au robot de poursuivre sa mission.



Figure 2.12 Fixation du moteur servo SG90
dans la nacelle du robot

De cette façon, ce cadre de levage et de stockage combine la précision du moteur pas à pas avec l'adaptabilité du servomoteur, permettant au robot de manipuler efficacement les marchandises depuis leur levage jusqu'à leur magasin sécurisé. Ce choix technologique assure un fonctionnement solide et précis adapté aux exigences d'une réserve robotisée.

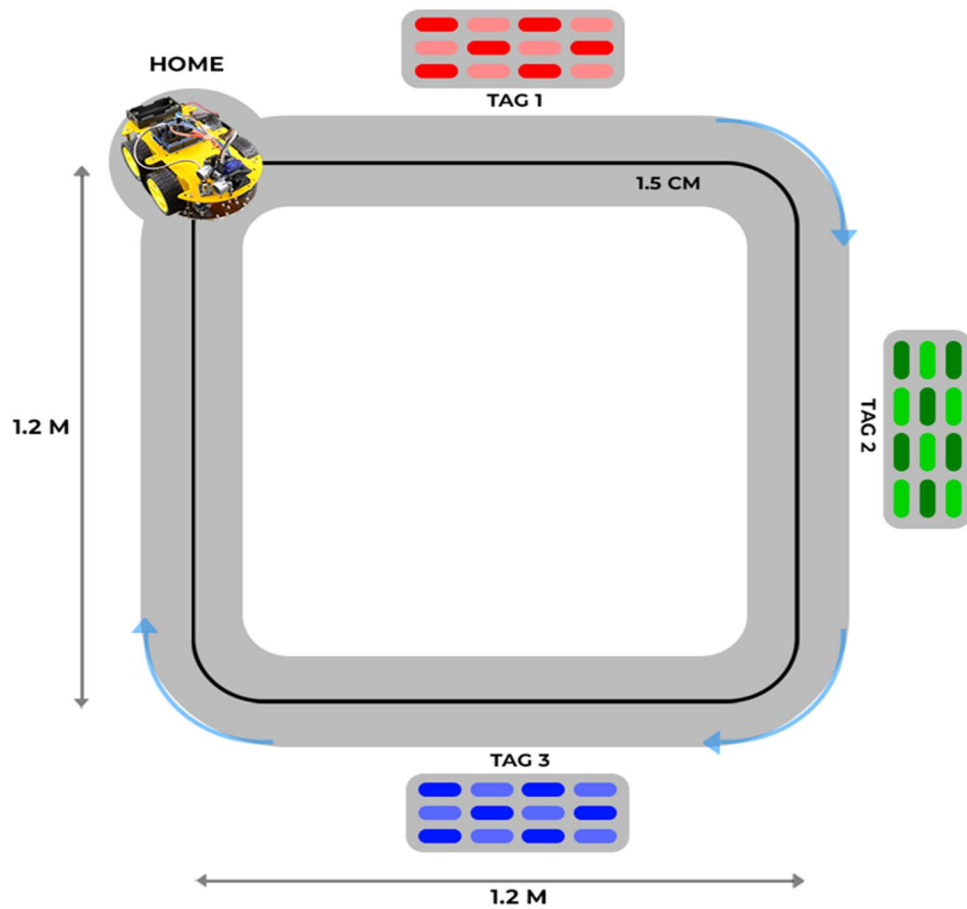


Figure 2.12 Maps de système

2.4 Structure du dépôt et des étagères

Pour assurer le dépôt efficace des colis transportés par le robot, une structure de dépôt à trois étagères a été conçue, adaptée à la manipulation robotisée. Cette structure joue un rôle essentiel dans le processus global de tri et de stockage.

2.4.1 Conception des étagères

Afin de permettre un fonctionnement autonome du robot pour le dépôt des colis, un système à trois étagères a été conçu et fabriqué. Chaque étagère correspond à une catégorie de couleur et facilite le tri

automatique. La structure comprend donc trois étagères superposées, chacune différenciée par une couleur spécifique et disposant de dimensions adaptées.

Les dimensions précises sont les suivantes :

- Largeur (plateau) : 15 cm
- Profondeur (plateau) : 14 cm
- Hauteur entre chaque étage : 9 cm
- Hauteur des pieds avant : 20 cm
- Hauteur des pieds arrière : 10 cm

Chaque étage est peint d'une couleur distinctive afin de permettre une reconnaissance aisée par le robot grâce à son capteur de couleur TCS3472 :

- Rouge pour l'étagère 1
- Bleu pour l'étagère 2
- Vert pour l'étagère 3

Ces couleurs ont été choisies pour assurer une différenciation visuelle optimale.

2.4.2 Traçage de la ligne noire

Pour guider le robot jusqu'à la position de dépôt des colis, une ligne noire a été tracée sur le sol. Cette ligne a été réalisée en utilisant du ruban adhésif noir (de type électricien) afin d'assurer un contraste maximal avec le sol clair et une bonne durabilité dans le temps. La ligne noire joue un rôle crucial en servant de guide de trajectoire pour le robot lors du processus de dépose des colis, grâce au module suiveur de ligne (capteur QTR-8A).

2.4.3 Fabrication des colis

Les colis simulés, utilisés pour les tests et la démonstration, sont fabriqués en carton léger (charton) puis peints dans les mêmes couleurs que les étagères correspondantes (rouge, bleu, vert).

Chaque colis présente des dimensions uniformes d'environ 8 cm x 8 cm x 8 cm, afin de garantir une manipulation cohérente et fiable par le système de préhension du robot.

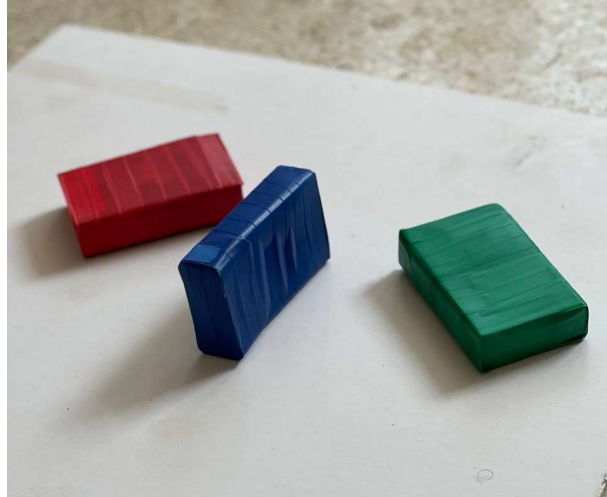


Figure 2.13 Colis Classifiés par Couleur RGB pour le Tri Automatique

Ce système à trois étagères et la ligne noire au sol constituent un élément essentiel de l'architecture du dépôt automatisé. Les couleurs distinctives (rouge, bleu, vert) permettent un tri automatique et un placement correct des colis par le robot, tandis que la ligne noire assure une navigation fluide et sécurisée du robot jusqu'au point de dépôt.

2.5 Conception du câblage électronique

Le câblage électronique du robot polyvalent est une étape de base qui garantit l'interconnexion entre les modules distinctifs (ESP32, capteurs, moteurs, etc.). Cette partie comporte des éléments subtils de l'organisation des associations électriques, du graphe de câblage et de l'administration de commande du robot.

2.5.1 Connexions électriques entre les composants

Dans ce domaine, nous détaillons les associations électriques requises pour que le robot astucieusement portable fonctionne correctement. L'objectif est de garantir une communication efficace entre les divers modules et capteurs, tout en garantissant la sécurité du cadre et une qualité sans faille.

Composants principaux à connecter

Le robot comprend plusieurs sous-ensembles électroniques interconnectés :

- Microcontrôleur Arduino Nano : unité centrale de traitement des signaux.
- Capteur de suivi de ligne (QTR-8A) : détection des trajectoires au sol.
- Capteur de détection de présence (PIR HC-SR501) : détection des mouvements.
- Capteur de couleurs TCS34725 : identification des couleurs d'étagères pour le tri.
- Module de pilotage de moteurs (L298N) : contrôle des moteurs à courant continu.
- Moteurs DC (x4) : propulsion du robot.
- Servomoteur MG90S : levage ou dépôt des marchandises.
- Moteur pas à pas 17HS4023 : rotation précise pour certaines tâches mécaniques.
- Batterie principale (12 V) : alimentation générale du système.

a. Bloc 1 : Module QTR-8A avec Arduino Nano

Le module QTR-8A est un capteur de suivi de ligne constitué de huit capteurs infrarouges qui permettent la détection précise d'une ligne sur le sol. Dans ce projet, six capteurs de ce module sont utilisés pour assurer un suivi fiable de la trajectoire. Chaque capteur renvoie un signal analogique capté sur une broche dédiée de l'Arduino Nano, facilitant ainsi la lecture individuelle et le traitement des données. Le module est alimenté en 5V et ses LEDs infrarouges sont activées par une broche spécifique (LEDON) pour garantir une détection optimale.

La structure de câblage pin par pin est présentée dans le tableau ci-dessous :

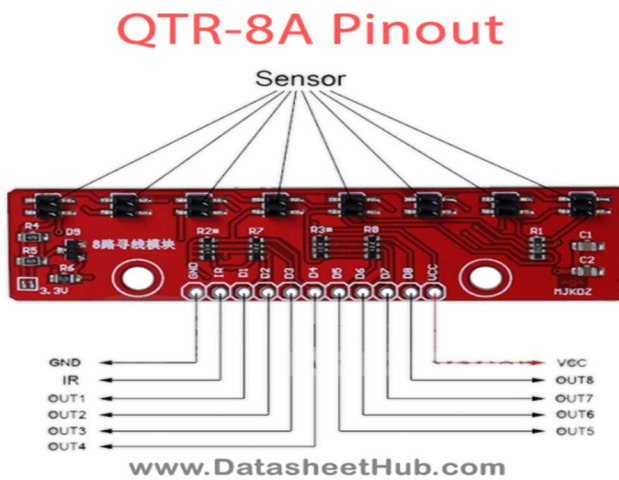


Figure 2.14 QTR-8A-Pinout

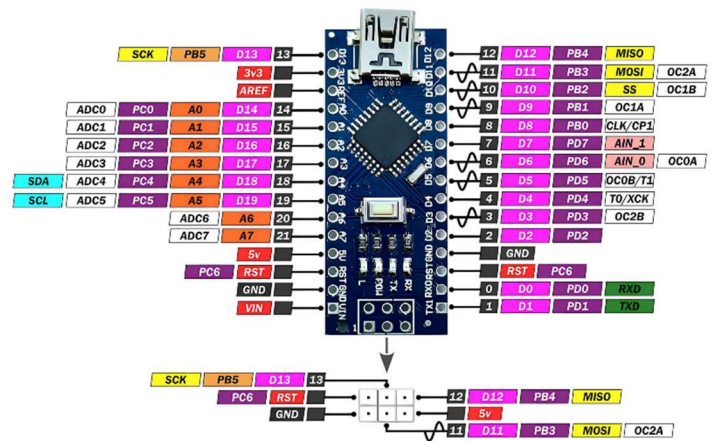


Figure 2.15 Arduino Nano -Pinout

Module QTR-8A	Arduino Nano	Description
VCC	5V	Alimentation électrique du module
GND	GND	Masse commune
OUT0 à OUT7	A0 à A7	Sorties analogiques des capteurs (8 capteurs utilisés)

Tableau 2.1 Connexion du module QTR-8A à l'Arduino Nano (mode analogique)

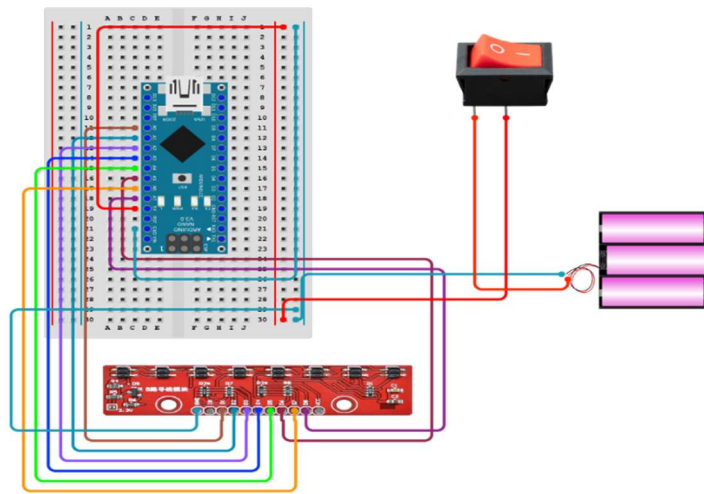


Figure 2.16 Schéma de circuit électrique Module QTR-8A avec Arduino Nano

b. Moteurs à courant continu (x4) avec Module L298N et Arduino Nano

Pour assurer le déplacement et la manœuvrabilité du robot, quatre moteurs à courant continu sont pilotés à l'aide du module driver L298N, qui agit comme une interface de puissance entre l'Arduino Nano et les moteurs. Ce module permet de contrôler la direction et la vitesse de chaque moteur indépendamment grâce aux signaux envoyés par l'Arduino.

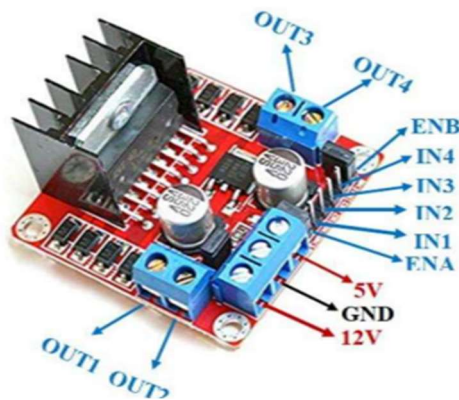


Figure 2.17 L298N moteur Driver Module pinout [15]

Le tableau suivant détaille les connexions entre le module L298N, les moteurs et l'Arduino Nano :

Composant	Arduino Nano	Fonction
IN1, IN2 (moteur droit)	D5, D7	Contrôle du sens de rotation moteur droit
ENA (moteur droit)	D6	Contrôle de la vitesse moteur droit (PWM)
IN3, IN4 (moteur gauche)	D8, D4	Contrôle du sens de rotation moteur gauche
ENB (moteur gauche)	D3	Contrôle de la vitesse moteur gauche (PWM)

Tableau 2.10 Connexion des moteurs DC à l'Arduino Nano via le module L298N

Pour une meilleure compréhension de la connexion entre les moteurs à courant continu, le module L298N et l'Arduino Nano, le schéma électrique suivant illustre clairement le câblage des différentes broches. Ce diagramme facilite la visualisation des liens entre les composants et assure une compréhension précise de l'architecture matérielle utilisée dans ce projet

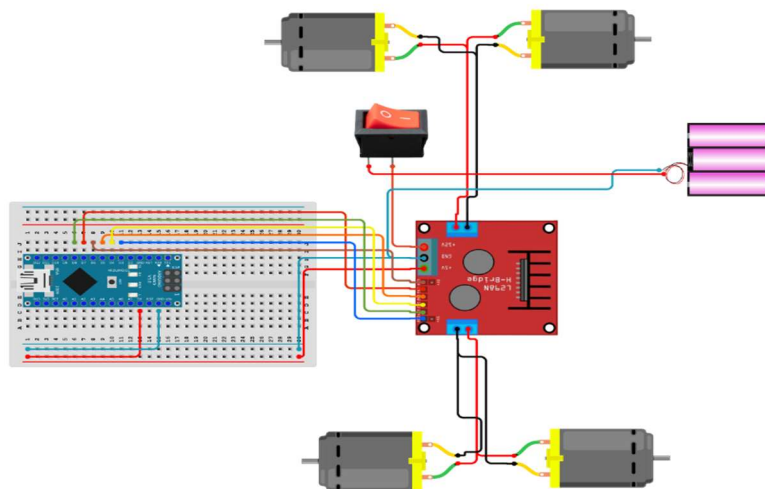


Figure 2.18 Schéma de circuit électrique Moteurs à courant continu (x4) avec Module L298N et Arduino Nano

c. Capteur à ultrasons HC-SR04 avec ESP32-S3

Le tableau suivant détaille les connexions broche par broche :

Broche HC-SR04	Broche ESP32-S3	Fonction
VCC	5V	Alimentation du capteur
GND	GND	Masse commune
TRIG	GPIO12	Émission du signal ultrasonique (Trigger)
ECHO	GPIO13	Réception du signal réfléchi (Echo)

Tableau 2.11 les broches entre le capteur HC-SR04 et l'ESP32-S

Pour compléter faciliter la compréhension de cette interface matérielle, le schéma électrique ci-dessous illustre de manière claire et précise les connexions entre le capteur HC-SR04 et l'Arduino Nano. Ce diagramme visuel permet d'avoir une vue d'ensemble du câblage et assure une meilleure lisibilité de la structure matérielle mise en place dans le cadre de ce projet.

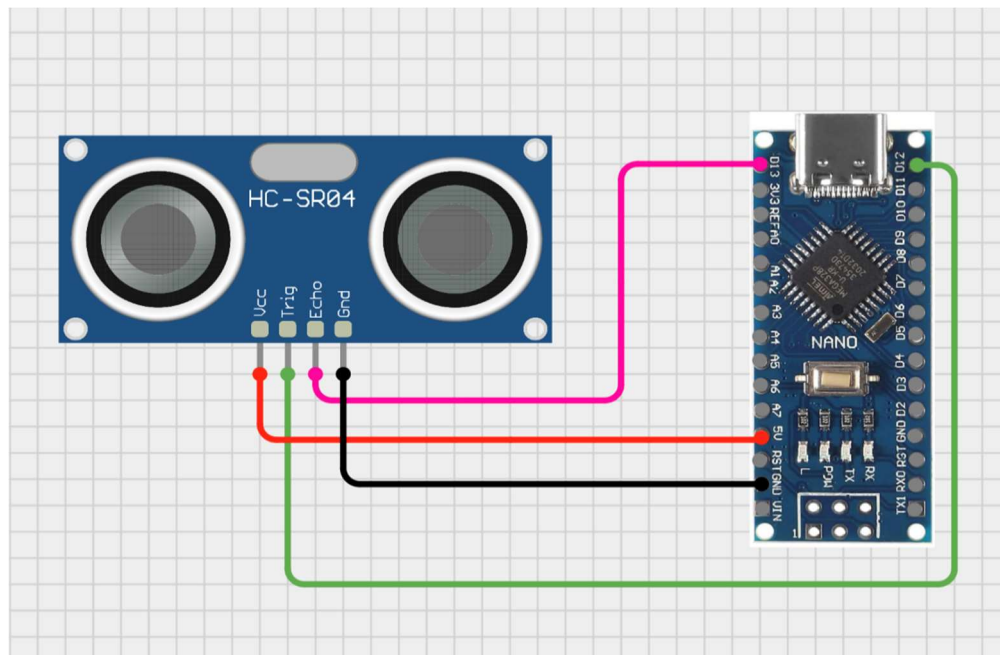


Figure 2.19 Schéma de circuit électrique ultrasons HC-SR04 avec Arduino Nano

d. Capteur de Couleur TCS34725 ↔ ESP32-S3

Le capteur de couleur TCS34725 est un composant indispensable pour la détection et la classification des objets selon leur couleur, ce qui permet au robot de prendre des décisions précises pour le tri ou le positionnement des marchandises. L'alimentation du capteur est assurée via la broche VIN de l'ESP32-S3, connectée à 3.3V, tandis que la broche GND assure une référence de masse commune. Les broches SDA et SCL du capteur, reliées respectivement aux broches GPIO 06 et GPIO 07 de l'ESP32-S3, assurent la communication I2C avec le microcontrôleur. La broche INT, optionnelle, n'a pas été utilisée dans ce projet.

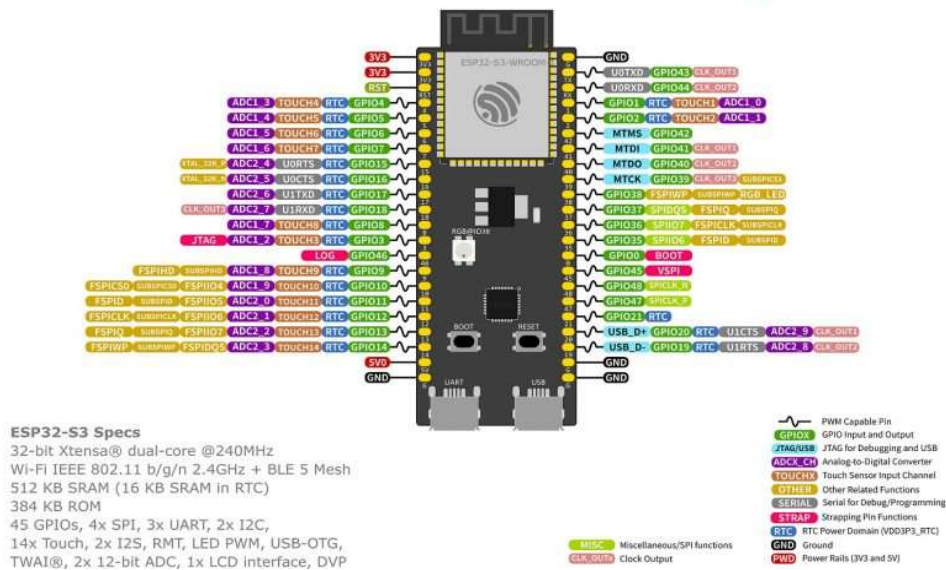


Figure 2.20 ESP32-S3 pinout

Le tableau suivant détaille les connexions broche par broche :

Composant	ESP32-S3	Description
VIN	3.3V	Alimentation du capteur
GND	GND	Masse commune
SDA	GPIO 06	Ligne de données I2C
SCL	GPIO 05	Ligne d'horloge I2C

Tableau 2.12 Connexion d'un capteur I2C à l'ESP32-S3

Afin de renforcer la clarté et la compréhension des interconnexions entre le capteur de couleur TCS34725 et l'ESP32-S3, un schéma électrique détaillé est fourni ci-dessous. Ce schéma illustre visuellement l'architecture du câblage utilisé, permettant de mieux

appréhender la configuration matérielle et de faciliter la reproduction de cette interface pour toute implémentation future.

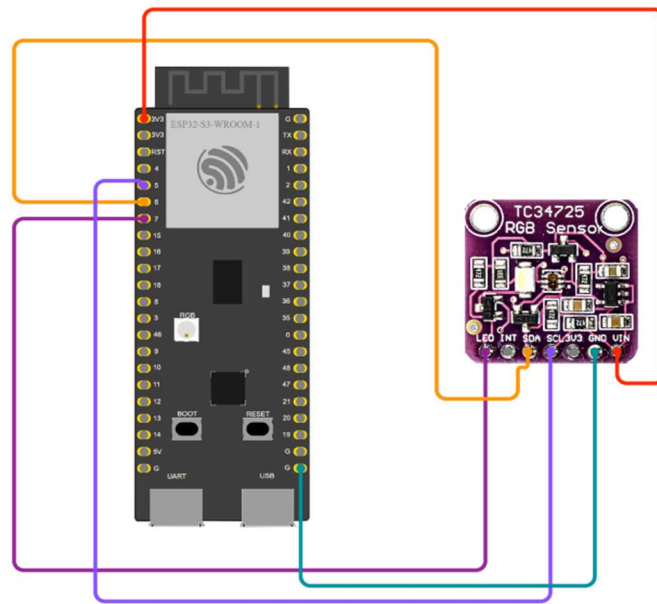


Figure 2.21 Schéma de circuit électrique Capteur TCS34725 et ESP32-S3

e. **Servo moteur SG90 avec ESP32-S3**

Le tableau suivant détaille les connexions broche par broche :

Composant	ESP32-S3	Description
VCC	5V	Alimentation du servo
GND	GND	Masse commune

Signal (Orange)	GPIO 23	Signal PWM pour le contrôle de l'angle du servo
-----------------	---------	---

Tableau 2.13 Connexion d'un moteur servo à l'ESP32-S3

Pour compléter cette clarification spécialisée et décrire les associations entre le servomoteur SG90 et l'ESP32-S3, un tableau électrique détaillé est donné ci-dessous. Ce schéma donne une représentation visuelle claire et exacte du câblage en place, encourageant la compréhension et la génération de cette interface d'équipement comme partie de l'entreprise.

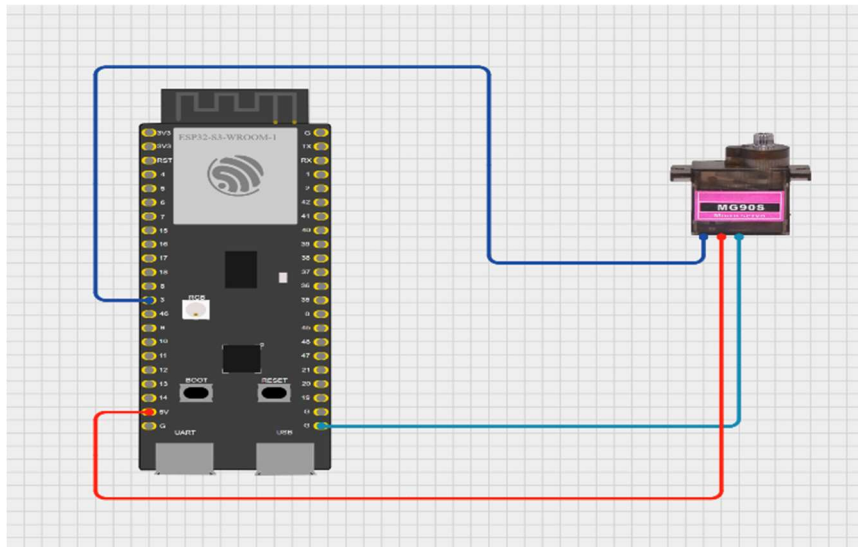


Figure 2.22 Schéma de circuit électrique Servo moteur SG90 avec ESP32-S3

Cette structure matérielle m'a permis d'intégrer des mouvements fluides et précis du bras mécanique, contribuant à la performance globale du robot. Le contrôle du servo à travers le microcontrôleur a été essentiel pour automatiser les tâches de manutention et améliorer l'efficacité de l'ensemble du système robotique.

f. Moteur pas à pas 17HS4023 avec Moteurs L298N et ESP32-S3

Le moteur pas à pas 17HS4023 est utilisé pour réaliser des mouvements précis et contrôler les déplacements de manière séquentielle et contrôlée, indispensable pour les opérations de tri et de placement des objets. Dans cette configuration, le moteur est piloté par un Moteurs L298N, qui reçoit les signaux de commande de l'ESP32-S3.

Les tableau suivants détaille les connexions broche par broche :

- Câblage entre le moteur pas à pas 17HS4023 et le L298N

Fils du moteur	L298N (Sortie)
Bobine A fil 1	OUT1
Bobine A fil 2	OUT2
Bobine B fil 1	OUT3
Bobine B fil 2	OUT4

Tableau 2.14 Connexion du moteur pas à pas 17HS4023 au module L298N

- Câblage entre le L298N et l'ESP32-S3

L298N (Entrée)	ESP32-S3 (GPIO)	Fonction
IN1	GPIO 8	Contrôle bobine A-1
IN2	GPIO 9	Contrôle bobine A-2
IN3	GPIO 10	Contrôle bobine B-1
IN4	GPIO 11	Contrôle bobine B-2

Tableau 2.15 Connexion des broches de commande entre l'ESP32-S3 et le L298N

- **Alimentation**

Élément	Connexion	Détail
VCC (L298N)	Batterie 12V (borne +)	Alimente le moteur
GND (L298N)	Batterie 12V (borne -) + GND ESP32	Masse commune pour tout le système

Tableau 2.16 Connexions d'alimentation du circuit

Pour illustrer cette architecture de câblage et faciliter la compréhension, le schéma électrique ci-dessous présente les interconnexions détaillées entre le moteur pas à pas 17HS4023, le Moteurs L298N et l'ESP32-S3.

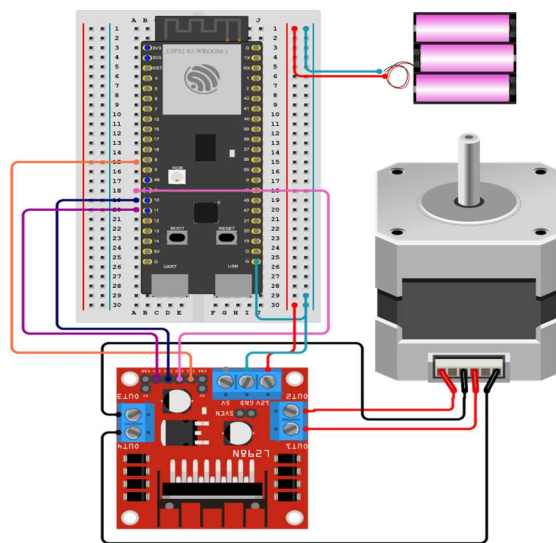


Figure 2.23 Schéma de circuit électrique Moteur pas à pas 17HS4023 avec Moteurs L298N et ESP32-S3

2.5.2 Schéma de câblage global

Afin de synthétiser l'ensemble des connexions matérielles utilisées dans le projet, un schéma de câblage global a été conçu. Ce schéma regroupe l'ensemble des blocs décrits précédemment (QTR-8A, capteur de couleur TCS34725, moteurs à courant continu via L298N, servo moteur SG90, moteur pas à pas 17HS4023 via L298N, capteur à ultrasons HC-SR04) et leurs interconnexions avec les microcontrôleurs (Arduino Nano et ESP32-S3).

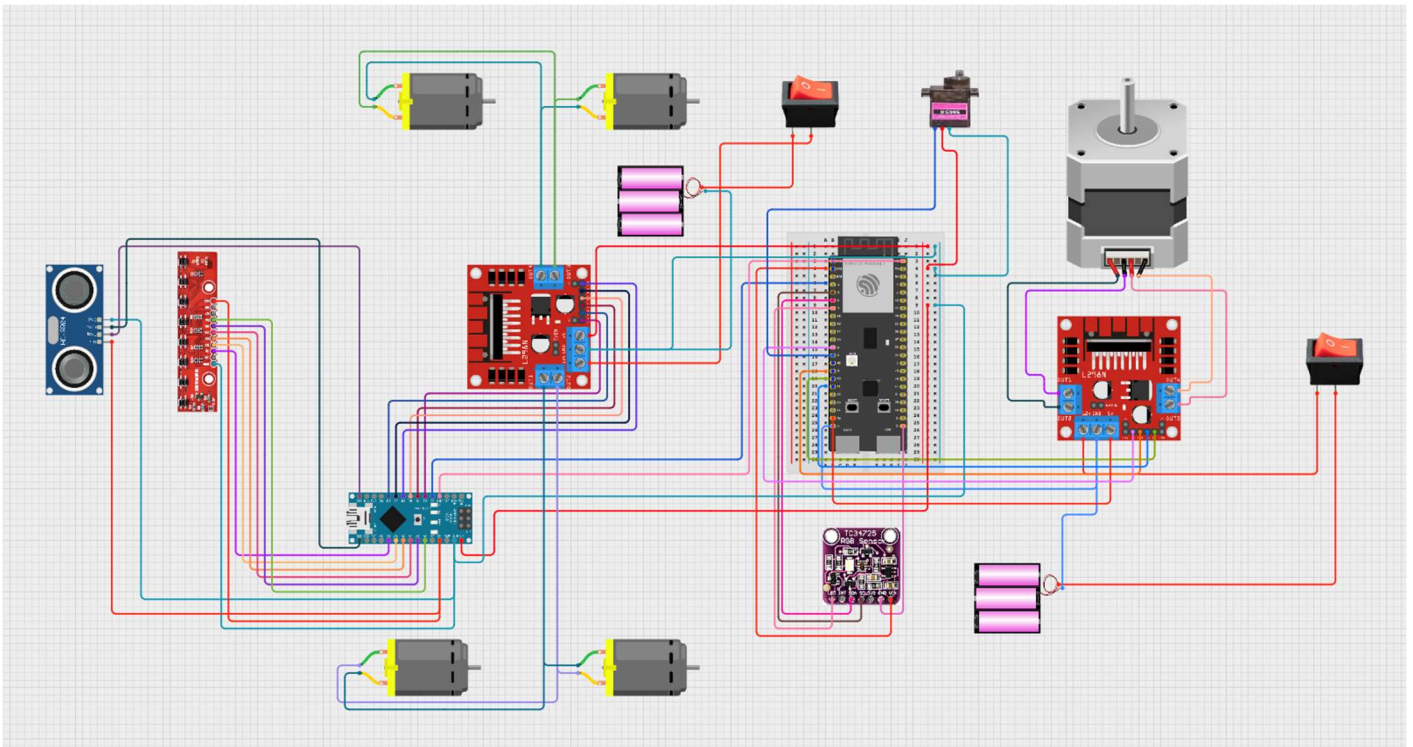


Figure 2.24 Schéma de câblage global du robot mobile intelligent

Ce schéma de câblage global offre une vision d'ensemble claire et précise de l'architecture matérielle du système. Il permet de visualiser la répartition des composants, l'alimentation, les lignes de commande et les broches de contrôle, garantissant ainsi une intégration cohérente et fiable du système embarqué. Il a été réalisé à l'aide du logiciel Circuit Designer afin de faciliter la compréhension et la reproduction du montage pour les besoins d'implémentation et de maintenance du projet.

2.6 Programmation

La programmation du microcontrôleur ESP32-S3 constitue une étape cruciale dans la conception de notre robot mobile intelligent. Cette phase permet d'intégrer et de coordonner les différents capteurs, actionneurs et modules de communication, garantissant ainsi le bon fonctionnement et l'autonomie du système. Grâce à une approche modulaire et structurée, le code développé permet au robot d'exécuter ses principales tâches, notamment le suivi de ligne, la détection des couleurs, la commande des moteurs, ainsi que la prise de décision en fonction des données collectées par les différents capteurs. Cette partie détaille l'environnement de développement utilisé, la structure du code, ainsi que l'implémentation des principales fonctionnalités logicielles.

2.6.1 Environnement de développement utilisé

Pour réaliser la programmation et les tests de notre robot, nous avons choisi des outils de développement adaptés à notre besoin de flexibilité et de fiabilité :

a. **Arduino IDE**



Arduino IDE est le programme qui vous permet de programmer des fiches Arduino, ou d'autres cartes de microcontrôleur, cet environnement d'amélioration des coordonnées (EDI) est né en même temps que la carte Arduino, il est également un avancement de câblage et Poignée (d'autres dialectes composés sous licences libres depuis peu de temps

la création d'Arduino), le dialecte utilisé est exceptionnellement comparable aux dialectes C et C++ auxquels sont incluses les capacités des nombreuses bibliothèques.

Il est écrit en Java (un langage de programmation multiplateforme), ce qui permet un portage facile et une interface quasiment identique, quel que soit le système d'exploitation.

L'IDE d'Arduino permet de regrouper dans le même outil les programmes nécessaires au pilotage de la carte, il comprend également un éditeur de texte, un débogueur/compilateur, une interface permettant de gérer les ports COM et le type de carte, de plus, il fournit de nombreux exemples, installe automatiquement les drivers les plus courants, et assure ensuite la communication avec l'Arduino (téléversement et moniteur série).[16]

Cet Arduino IDE simplifie la programmation pour les cartes d'Arduino ou autre, il ressemble à n'importe quel autre IDE de programmation, c'est-à-dire, il contient un éditeur pour coder et compiler, de plus, il inclut de différents éléments d'aide pour l'utilisateur qui vont au-delà des propres guides simples, mais aussi il a des exemples de code pour différentes fonctionnalités et appareils.

Il est pensé pour que n'importe qui puisse commencer à s'en servir avec des connaissances de base, dans tous les cas il existe de milliers de tutoriels sur internet pour apprendre programmer pour les débutants[17]

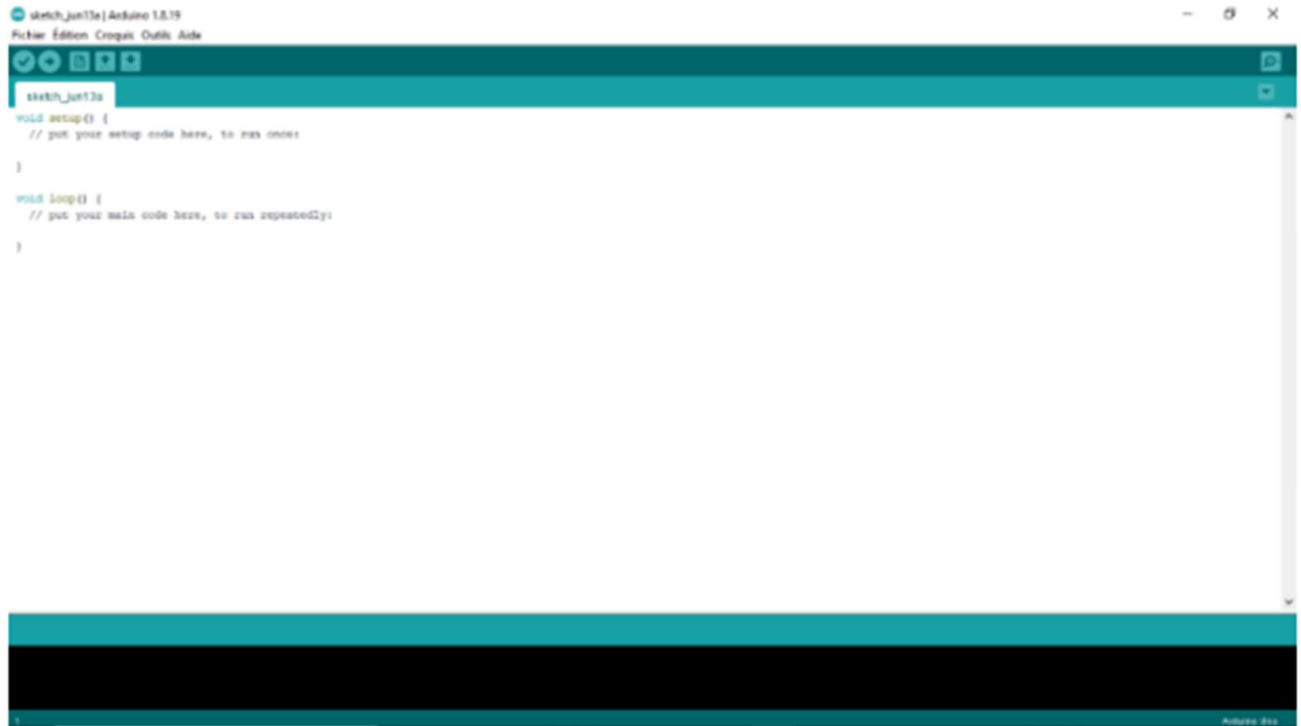


Figure 2.25 Interface Arduino IDE

Structure d'un programme Arduino :

Un programme Arduino est constitué de trois parties (fonctions) :

Partie 1 : C'est la partie où l'on déclare les variables, on affecte les broches, et aussi on appelle les bibliothèques.

Partie 2 : C'est l'étape d'initialisation (Setup), où l'on configure les broches en entrée ou en sortie, mais aussi pour initialiser les différents objets (comme le moniteur série,

Partie 3 : C'est le programme principal (Loop), cette fonction est répétée indéfiniment.

Comme pour n'importe quel autre logiciel de programmation, on peut rajouter de commentaires, ce qui est vivement conseillé, deux types de commentaires sont possibles

-Un commentaire après une instruction débute par « // ».

-Un commentaire sous forme de paragraphe commence par « /* » et se termine par «

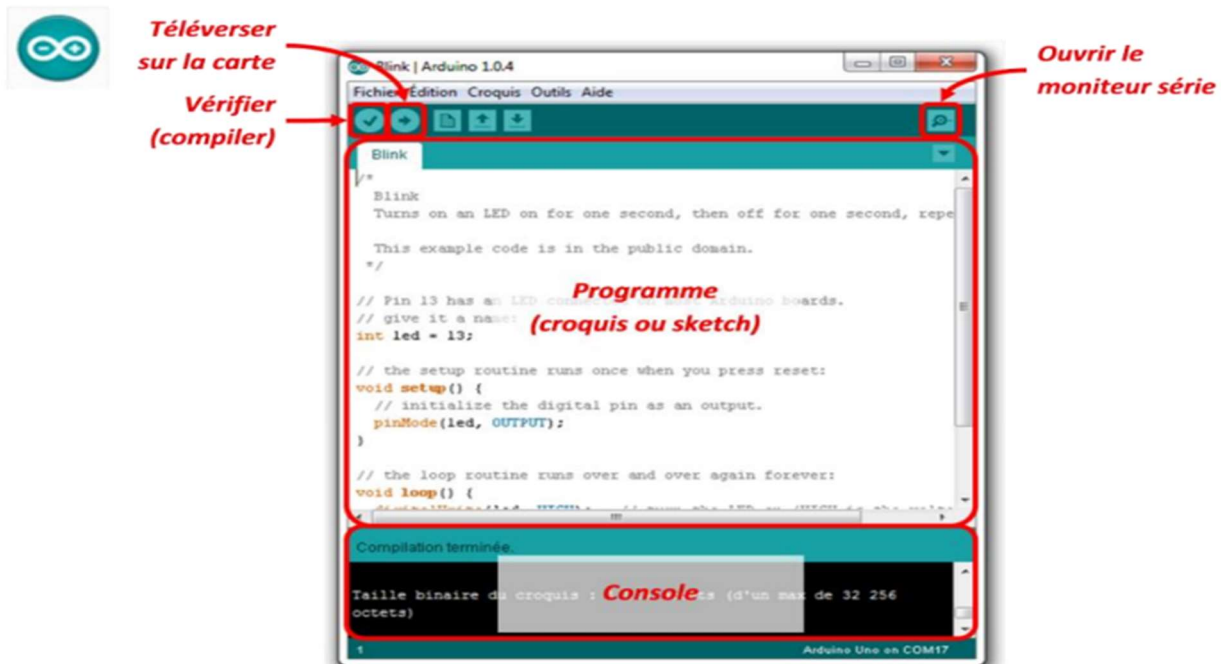


Figure 2.26 Interface Arduino IDE expliquée.

L'IDE d'Arduino possède aussi une aide en ligne où toutes les instructions, structures du langage Arduino sont décrites et accompagnées d'un exemple, pour accéder à cette aide, il faut cliquer sur Aide puis Référence. [18]

L'IDE d'Arduino permet :

- D'éditer un programme : des croquis (sketch en Anglais), les programmes sont écrits en langage C.
- De compiler ce programme dans le langage « machine » de l'Arduino, la compilation est une traduction du langage C vers le langage du microcontrôleur.

- D'afficher dans la console les informations sur le déroulement de la compilation et d'afficher les messages d'erreur.
- De communiquer avec la carte Arduino grâce au terminal (ou moniteur série). pendant le fonctionnement du programme en mémoire sur l'Arduino, il peut communiquer avec l'ordinateur tant que la connexion est active (câble USB) [19]

L'un des avantages de l'EDI Arduino réside dans le fait qu'il est piloté par la communauté, Arduino qui dispose d'un forum très actif où les utilisateurs peuvent partager leurs créations avec d'autres programmeurs et obtenir des commentaires et des conseils de dépannage, les utilisateurs les plus avancés peuvent également délivrer leurs propres conseils, les utilisateurs ont aussi la possibilité de télécharger leur code directement sur les forums ce qui est extrêmement pratique. [20]

2.6.2 Structure du code

La structure du code développé pour le microcontrôleur **ESP32-S3** a été conçue de manière modulaire et hiérarchique afin de faciliter la lisibilité, la maintenance et les évolutions futures. Le code est principalement écrit en langage C/C++ et organisé sous forme de plusieurs fichiers et fonctions spécifiques à chaque tâche principale du robot.

Le fichier principal (main.cpp) contient la fonction d'initialisation (setup()) et la boucle principale (loop()) qui orchestre le fonctionnement global du robot. Chaque fonctionnalité est ensuite déléguée à des fichiers ou des sections de code distincts :

- qtr_sensor.cpp : gère la lecture des capteurs de suivi de ligne QTR-8A et le calcul de la déviation par rapport au tracé.
- color_sensor.cpp : traite les données provenant du capteur de couleurs TCS34725 et prend les décisions en fonction de la couleur détectée.
- motor_control.cpp : pilote les moteurs à courant continu via le module L298N ainsi que les moteurs supplémentaires (servo SG90 et moteur pas à pas).

- `ultrasonic_sensor.cpp` : assure la détection d'obstacles à l'aide du capteur HC-SR04 pour éviter les collisions éventuelles.

Cette organisation favorise une gestion claire des tâches logicielles, permettant ainsi un développement évolutif et un débogage efficace. Chaque module est testé individuellement avant intégration afin de garantir la stabilité du système global

2.6.3 Suivi de ligne avec QTR-8A

Ce robot autonome est conçu pour suivre une ligne noire tracée sur le sol, tout en intégrant des fonctions d'évitement d'obstacles et de communication sans fil. Il repose sur plusieurs capteurs et composants électroniques qui assurent la détection, le traitement des données, la prise de décision et l'exécution des mouvements.

Composants principaux utilisés :

- 6 capteurs infrarouge (de A1 à A6) : utilisés pour détecter la ligne noire sur le sol.
- 2 moteurs à courant continu : assurent la propulsion du robot via un contrôleur de moteur.
- 1 capteur à ultrasons : détecte les obstacles situés devant le robot.
- 1 pin d'interruption (pin 2) : reçoit des signaux externes permettant l'arrêt du robot à distance.
- Module de communication série : pour la transmission ou la réception de données.

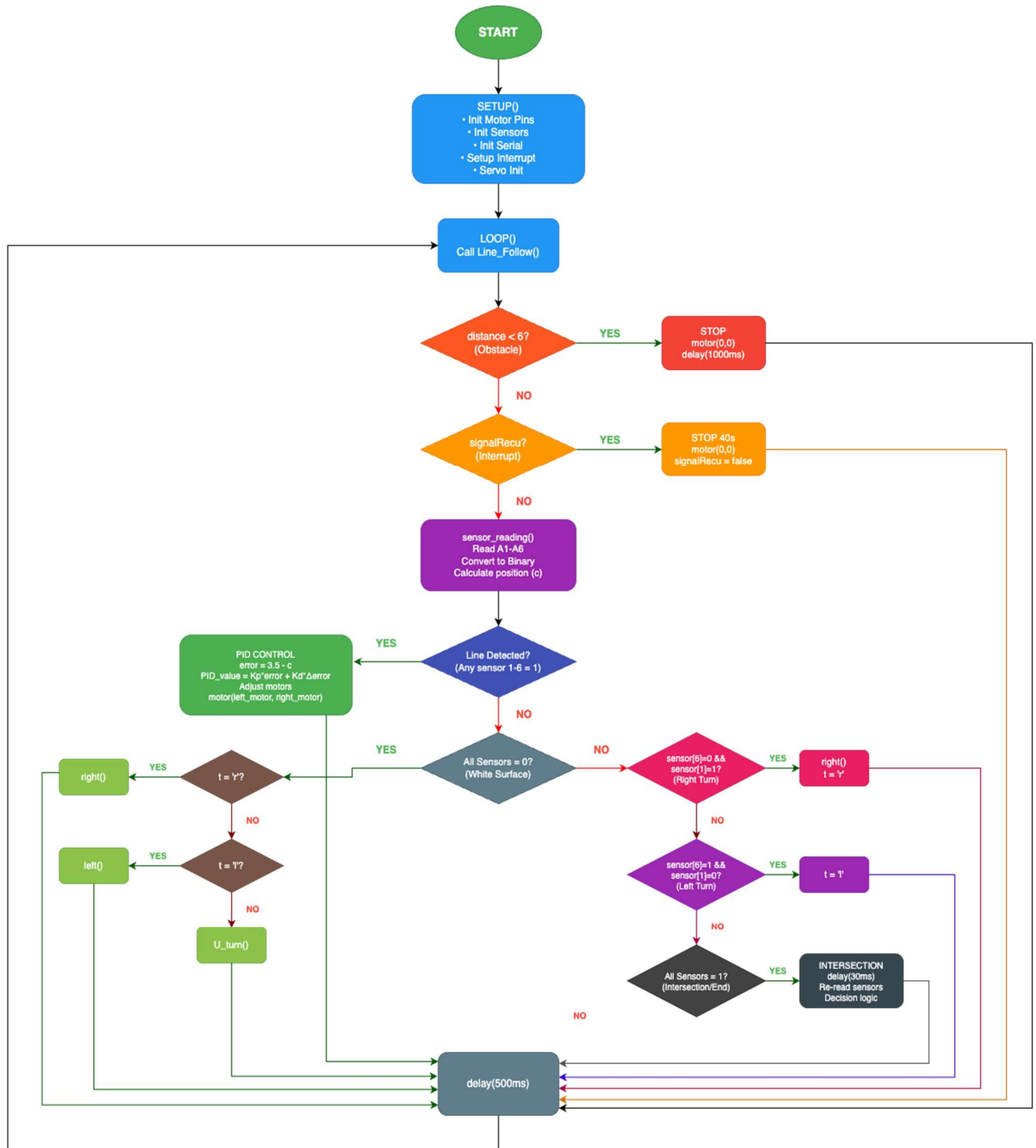


Figure 2.27 Organigramme de Suivi de ligne avec QTR-8A

a. **Initialisation (fonction setup)**

- Configuration des broches (capteurs, moteurs, etc.).
- Activation de l'interruption matérielle sur la broche 2.
- Initialisation de la communication série pour le diagnostic ou le contrôle.

b. **Boucle principale (fonction loop)**

Appelle en continu la fonction principale `Line_Follow()` qui gère la navigation. Le microcontrôleur entre ensuite dans une boucle principale dans laquelle est appelée la fonction `Line_Follow()`, responsable de la logique complète de suivi de ligne.

c. **Fonction `Line_Follow()` – Suivi de la ligne**

- Si un obstacle est détecté à une distance inférieure à 6 cm, le robot s'arrête pendant 1 seconde.
- Si un signal d'arrêt est détecté via l'interruption, le robot s'arrête pendant 40 secondes.

d. **Analyse de la ligne :**

- Lecture des six capteurs infrarouge.
- Conversion des valeurs lues en binaire : 1 pour noir (ligne détectée), 0 pour blanc (aucune ligne)
- Calcul de la position de la ligne pour déterminer l'erreur de trajectoire.

e. **Comportements selon les cas : l'erreur**

- Ligne détectée : Le robot utilise un contrôleur PID (Proportionnel – Dérivé) pour corriger sa trajectoire en ajustant la vitesse des deux moteurs.
- Tous les capteurs sur blanc : Signifie que le robot a perdu la ligne ; il effectue un virage dans la dernière direction mémorisée.

- Virage à droite détecté : Le robot exécute une manœuvre spécifique de rotation à droite pour retrouver la ligne
- Virage à gauche détecté : La direction est mémorisée pour guider le comportement futur.
- Intersection ou fin de ligne : Le robot s'arrête et attend qu'une nouvelle ligne soit détectée pour reprendre le suivi.

f. **Fonctions de mouvement dédiées**

- `right()` : Rotation vers la droite jusqu'à la détection de la ligne.
- `left()` : Rotation vers la gauche jusqu'à retrouver la ligne.
- `U_turn()` : Demi-tour à 180°.

g. **Contrôle PID**

- Erreur = position réelle – position idéale (centrée).
- Le PID ajuste dynamiquement la vitesse des moteurs gauche et droit pour minimiser cette erreur et maintenir le robot aligné.

2.6.4 **Système de détection de couleur et de manipulation avec un moteur pas à pas et un servomoteur MG90S**

Ce programme pilote un système automatisé capable de reconnaître des couleurs à l'aide d'un capteur, puis de déclencher une séquence mécanique adaptée selon la couleur détectée.

Composants principaux :

- Capteur de couleur TCS34725 : permet d'identifier les couleurs, notamment le rouge et le vert.
- Moteur pas-à-pas (4 fils) : assure les mouvements verticaux (montée et descente).
- Servomoteur : effectue une action mécanique, comme pousser ou relâcher un objet.

- Sortie numérique (pin OUT) : utilisée pour envoyer un signal électrique externe.

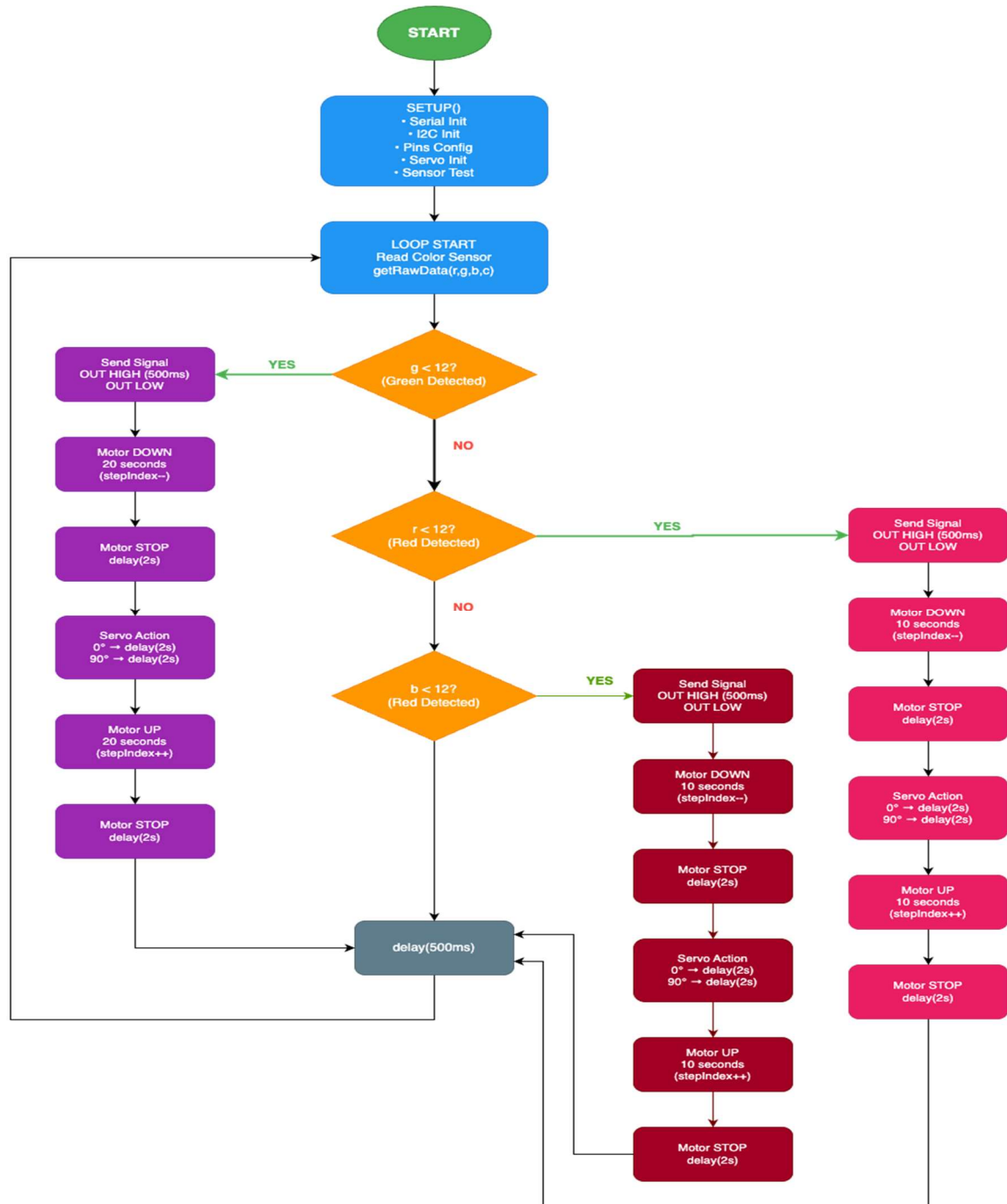


Figure 2.28 Organigramme de fonctionnement du système de détection de couleur et de commande des actionneurs

a. **Phase d'initialisation**

- Configuration de la communication série et du bus I2C.
- Initialisation du capteur de couleur et du servomoteur.
- Mise en position neutre du servomoteur à 90°.

b. **Boucle principale**

- Lecture continue des composantes RVB depuis le capteur.
- Affichage des valeurs sur le moniteur série.

c. **Réaction en fonction de la couleur détectée**

Si la couleur verte est détectée (valeur de G < 11) :

- Envoi d'un signal de 500 ms sur la sortie numérique.
- Descente du mécanisme : activation du moteur pas-à-pas en sens inverse pendant 20 secondes.
- Pause de 2 secondes.
- Action du servomoteur : rotation à 0°, attente de 2 secondes, retour à 90°, nouvelle attente de 2 secondes.
- Montée : activation du moteur pas-à-pas en sens normal pendant 20 secondes.
- Pause finale de 2 secondes.

Si la couleur rouge est détectée (valeur de R < 11 ; valeur de B < 11) :

- La même séquence est exécutée, mais avec des durées réduites à 10 secondes pour les mouvements du moteur pas-à-pas.

d. **Contrôle du moteur pas-à-pas**

- Le moteur fonctionne selon une séquence de 8 phases pour garantir un déplacement fluide.
- Le sens de rotation est contrôlé en modifiant l'index de la séquence (incrémentement ou décrémentement).
- La vitesse de rotation est réglée par un délai de 800 microsecondes entre chaque phase.

2.7 Conclusion :

Dans ce chapitre, nous avons exposé de manière détaillée le processus de conception et de réalisation du robot mobile intelligent. À travers la définition de l'architecture générale, l'intégration des modules électroniques et mécaniques, ainsi que la structuration du câblage, chaque composant a été sélectionné et agencé de manière cohérente afin d'assurer la fiabilité, la précision et l'autonomie du système. Le microcontrôleur ESP32, les capteurs (QTR-8A pour le suivi de ligne, TCS3472 pour la reconnaissance des couleurs, et capteurs à ultrasons pour la détection d'obstacles), les moteurs et les actionneurs ont été mis en œuvre selon une logique modulaire. Par ailleurs, le développement du programme embarqué a permis d'assurer les principales fonctionnalités du robot, notamment le déplacement autonome, la détection intelligente de l'environnement et la prise de décision. Les contraintes techniques rencontrées au cours du prototypage ont été analysées avec rigueur, donnant lieu à des ajustements matériels et logiciels efficaces. Ainsi, cette phase de conception constitue une base solide pour le développement des fonctionnalités de supervision à distance via une interface web, qui feront l'objet du chapitre suivant, en lien avec l'analyse des résultats obtenus et les perspectives d'amélioration du système.

Chapitre 3 : Interface Web, Résultats et Perspectives

3.1 Introduction

Ce troisième chapitre prolonge le travail amorcé dans le chapitre 2, où ont été présentés en détail la conception, l'équipement et la programmation du robot intelligemment polyvalent. Après avoir décrit la mise en œuvre matérielle et logicielle du microcontrôleur ESP32-S3, ce chapitre s'intéresse à l'interface web ainsi qu'à la phase de validation expérimentale du système.

L'intégration d'une interface web constitue une étape essentielle pour assurer une supervision avancée et un contrôle à distance du robot. Cette interface permet de visualiser en temps réel plusieurs paramètres clés du système, tels que l'état de la batterie, le nombre d'articles traités, ou encore le contrôle à distance du robot. Elle permet également de valider le bon fonctionnement global de l'architecture précédemment présentée.

Dans une première partie, nous détaillons les objectifs assignés à cette interface web, ainsi que les technologies retenues pour sa mise en œuvre. Nous décrivons ensuite les principales fonctionnalités développées et leur intégration avec les modules matériels et logiciels présentés dans le chapitre précédent.

La seconde partie de ce chapitre est consacrée aux résultats expérimentaux, obtenus à travers différents tests : suivi de ligne, détection de couleur, levage et dépôt de produits, ainsi que la visualisation via l'interface web. Ces résultats permettent d'évaluer les performances globales du système et de valider la pertinence des choix techniques décrits au chapitre 2.

Enfin, nous présentons les difficultés rencontrées lors de la phase de développement et de test, ainsi que les solutions mises en œuvre pour les surmonter. Le chapitre se conclut par une réflexion

sur les perspectives d'amélioration, qu'elles concernent la structure matérielle, le code logiciel, ou encore une éventuelle évolution de l'intégration mécanique.

3.2 Intégration IoT et tableau de bord web pour le contrôle robotique avancé

L'évolution de la robotique a été de plus en plus étroitement liée à l'Internet des objets (IoT), permettant un nouveau paradigme de surveillance et de contrôle à distance. Pour notre robot d'entrepôt autonome, « RoboStock », le développement d'un tableau de bord sophistiqué basé sur le web n'est pas simplement une interface mais une composante essentielle de son efficacité opérationnelle et de sa gestion. L'objectif principal est de fournir un environnement fluide, manière en temps réel et intuitive d'interagir avec le robot, de surveiller ses performances et de gérer ses tâches dans l'environnement de l'entrepôt. En exploitant les technologies web modernes, nous pouvons créer un outil puissant qui améliore les capacités du robot et fournit des informations précieuses sur la dynamique opérationnelle de l'entrepôt.

3.2.1 L'avantage stratégique d'un tableau de bord Web

Un tableau de bord web dédié offre de nombreux avantages par rapport aux systèmes de contrôle traditionnels et localisés. Ces avantages sont essentiels pour un environnement dynamique comme un entrepôt : Vérification en temps réel, permettant au client d'obtenir des données de base telles que le niveau de la batterie et l'état du capteur, pour garantir l'état du robot et une qualité constante.

- **Contrôle Surveillance et contrôle en temps réel** : Le tableau de bord fournit une vue d'ensemble instantanée de l'état du robot, y compris son emplacement, le niveau de la batterie, la tâche actuelle et les lectures des capteurs. Il permet également une intervention manuelle immédiate, comme mettre en pause une tâche ou contrôler les actionneurs du robot directement depuis l'interface web.
- **Gestion centralisée** : Pour les entrepôts déployant plusieurs robots, un seul tableau de bord peut gérer l'ensemble de la flotte. Cela simplifie l'allocation des tâches, la surveillance et la maintenance, conduisant à un fonctionnement plus coordonné et efficace.

- **Data-Driven Insights** : Le tableau de bord sert de hub pour collecter et visualiser les données de performance. En analysant des métriques comme les colis traités par heure, le temps de disponibilité du robot et les taux d'erreur, nous pouvons identifier les goulets d'étranglement, optimiser les flux de travail et prendre des décisions éclairées pour améliorer la productivité globale.
- **Accessibilité et évolutivité** : Étant basé sur le web, le tableau de bord peut être accessible depuis n'importe quel appareil avec un navigateur, offrant une flexibilité aux gestionnaires d'entrepôt et aux techniciens. L'architecture est conçue pour être évolutive, permettant l'ajout de nouveaux robots et fonctionnalités avec un minimum de frais généraux.
- **Maintenance et diagnostics améliorés** : Les journaux du système et les outils de diagnostic matériel disponibles sur le tableau de bord (comme indiqué dans la section « Robot Monitor ») permettent une maintenance proactive et un dépannage plus rapide, réduisant ainsi les temps d'arrêt.

3.2.2 Architecture du système et pile technologique

Pour atteindre le niveau d'interactivité et de performance en temps réel souhaité, nous avons sélectionné une stack technologique robuste et moderne. L'architecture est composée de trois couches principales : le robot (matériel), le serveur (backend) et le tableau de bord web (frontend).

- **Frontend (Web Dashboard)** : L'interface utilisateur est développée à l'aide de Nuxt.js, un framework puissant construit sur Vue.js. Nuxt.js a été choisi pour ses capacités en matière de rendu côté serveur (SSR), qui améliore les temps de chargement des pages initiales et le référencement, ainsi que son architecture modulaire, qui simplifie le développement d'une application complexe comme celle-ci.
- **Backend (Server)** : Un serveur Node.js agit comme intermédiaire entre le robot et le tableau de bord web. Il héberge l'API, gère les connexions WebSocket et gère la logique métier.
- **Protocoles de communication** : La liaison en temps réel entre le serveur et le tableau de bord Nuxt.js est établie à l'aide de Socket.IO. Pour la communication avec des services tiers

où le déclenchement de workflows externes, des webhooks sont utilisés. Le robot lui-même communique avec le serveur via une connexion Wi-Fi, généralement en utilisant le protocole MQTT ou une connexion WebSocket directe, envoyant des données de télémétrie et recevant des commandes.



Figure 3.1 Architecture du système de haut niveau

3.2.3 Communication en temps réel avec Socket.IO

L'interface, au cœur de l'expérience interactive, est propulsée par Socket.IO, une bibliothèque JavaScript permettant une communication bidirectionnelle, en temps réel, reposant sur un système d'événements.

Lorsqu'un opérateur clique sur un bouton du tableau de bord—par exemple, "Open Gripper"—l'application Nuxt.js émet un événement `control:gripper` via Socket.IO vers le serveur Node.js. Le serveur écoute cet événement et le traduit en une commande spécifique qui est ensuite envoyée au robot. À l'inverse, le robot envoie continuellement des mises à jour d'état (par exemple, niveau de batterie, position, données du capteur) au serveur. Le serveur traite ces données et les diffuse à tous les clients de tableau de bord connectés en utilisant Socket.IO. L'application Nuxt.js écoute ces événements entrants et met à jour dynamiquement l'interface utilisateur pour refléter l'état actuel du robot sans nécessiter de rafraîchissement de la page. Cela crée une expérience utilisateur fluide et réactive, essentielle pour un fonctionnement à distance efficace.

3.2.4 Caractéristiques et fonctionnalités du tableau de bord

Le tableau de bord RoboStock est organisé en plusieurs sections dédiées, chacune conçue pour aborder un aspect spécifique de la gestion de l'automatisation d'entrepôt.

a. **Centre de contrôle RoboStock (Tableau de bord principal)**

C'est la vue opérationnelle principale. Elle fournit des informations en un coup d'œil et un contrôle direct sur un robot sélectionné. Les modules clés incluent :

- **État du système** : Données de télémétrie en direct, y compris le pourcentage de batterie, la force du signal, le total des colis traités et la distance parcourue.
- **Cartographie de l'entrepôt** : Une visualisation 2D en temps réel de la position du robot dans la disposition de l'entrepôt, y compris son chemin et ses repères clés.
- **Contrôle du servo-moteur** : Boutons pour le contrôle manuel direct sur le bras et la pince du robot.
- **File d'attente de tâches et de paquets en cours** : affiche la tâche active du robot, sa progression et la file d'attente des tâches à venir.
- **Détection de points de repère** : Montre la progression du robot le long d'un chemin prédéfini, en cochant les jonctions et les destinations au fur et à mesure qu'il les traverse.

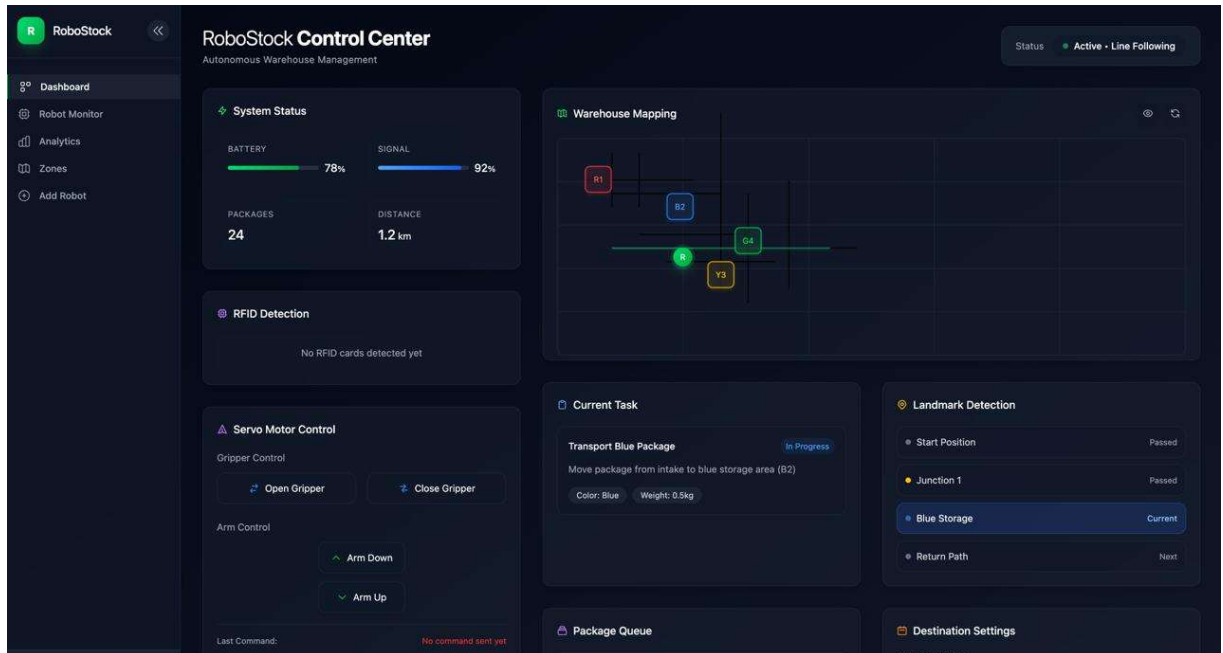


Figure 3.2 Centre de contrôle RoboStock (Tableau de bord principal)

b. Moniteur de robot

Cette section est pour le diagnostic et la surveillance de l'état :

- Schéma directeur du matériel : Une représentation graphique des principaux composants du robot (par exemple, microcontrôleur ESP32, servos, capteurs) et de leur état de connexion.
- Journaux du système : Un journal horodaté de tous les événements majeurs du système et du matériel, ce qui est inestimable pour le débogage.
- Utilisation des ressources : Graphiques en direct montrant l'utilisation du CPU, la consommation de mémoire et la consommation d'énergie.

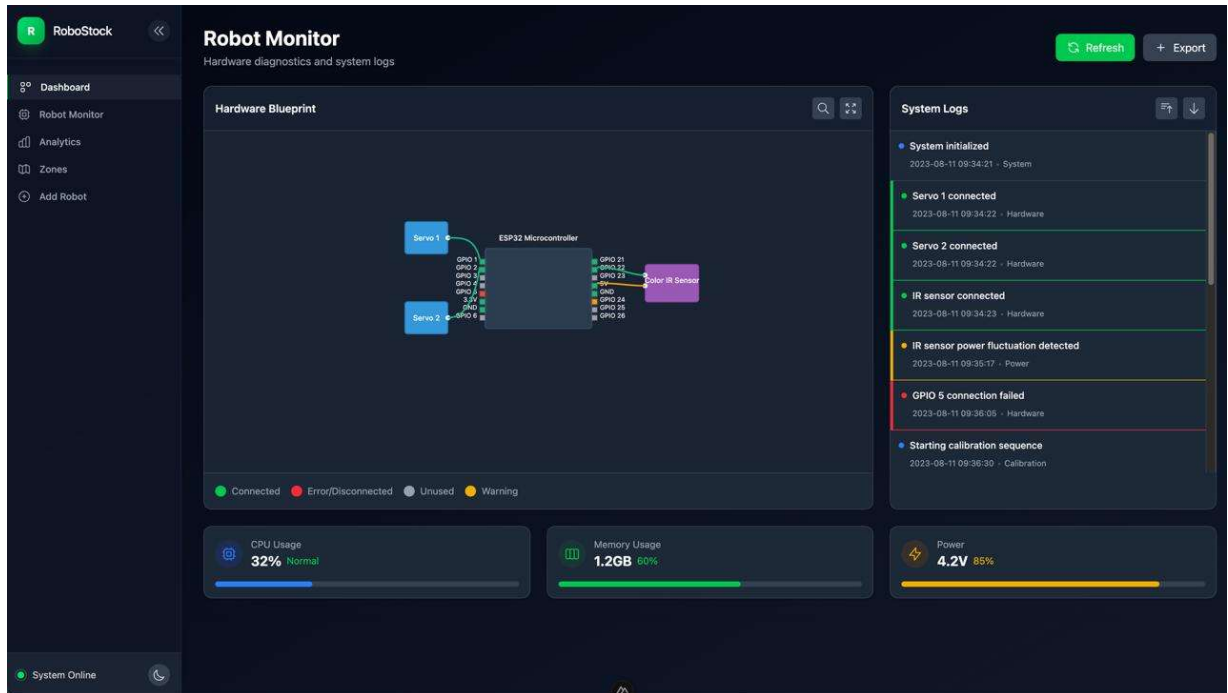


Figure 3.3 Moniteur de robot

c. Analyse de la performance

Ce tableau de bord offre un aperçu général de l'efficacité de l'ensemble des opérations.

- Indicateurs clés de performance (KPI) : Mesures telles que le nombre total de paquets traités aujourd'hui, le nombre de robots actifs, le taux d'efficacité global et le temps de disponibilité du système.
- Diagramme des colis traités : Un graphique en barres visualisant le nombre de colis traités au fil du temps.
- Statut de la flotte de robots : Une liste de tous les robots dans la flotte et leur statut actuel (par exemple, Actif, Chargement, Maintenance).

- Alertes du système : Un journal des alertes importantes à l'échelle du système et leur statut de résolution.

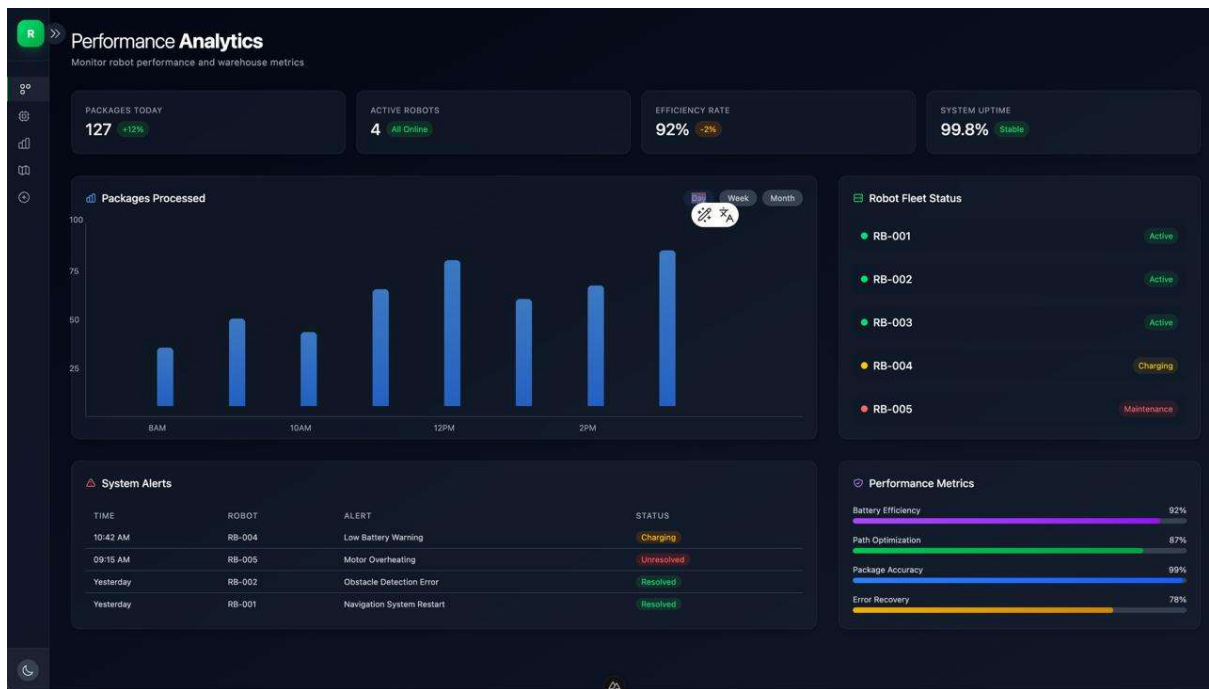


Figure 3.4 Analyse de la performance

d. Gestion des zones

Pour les entrepôts divisés en différentes zones opérationnelles, cette vue permet la micro-gestion.

- Métriques spécifiques à la zone : visualisez les données de performance filtrées par une zone spécifique (par exemple, Débit, Efficacité).
- Carte de zone : Une carte ciblée de la zone sélectionnée, montrant les positions de tous les robots qui y sont affectés.

- Affectation des robots : Une liste des robots actuellement actifs dans la zone, avec des options pour afficher les détails de chacun

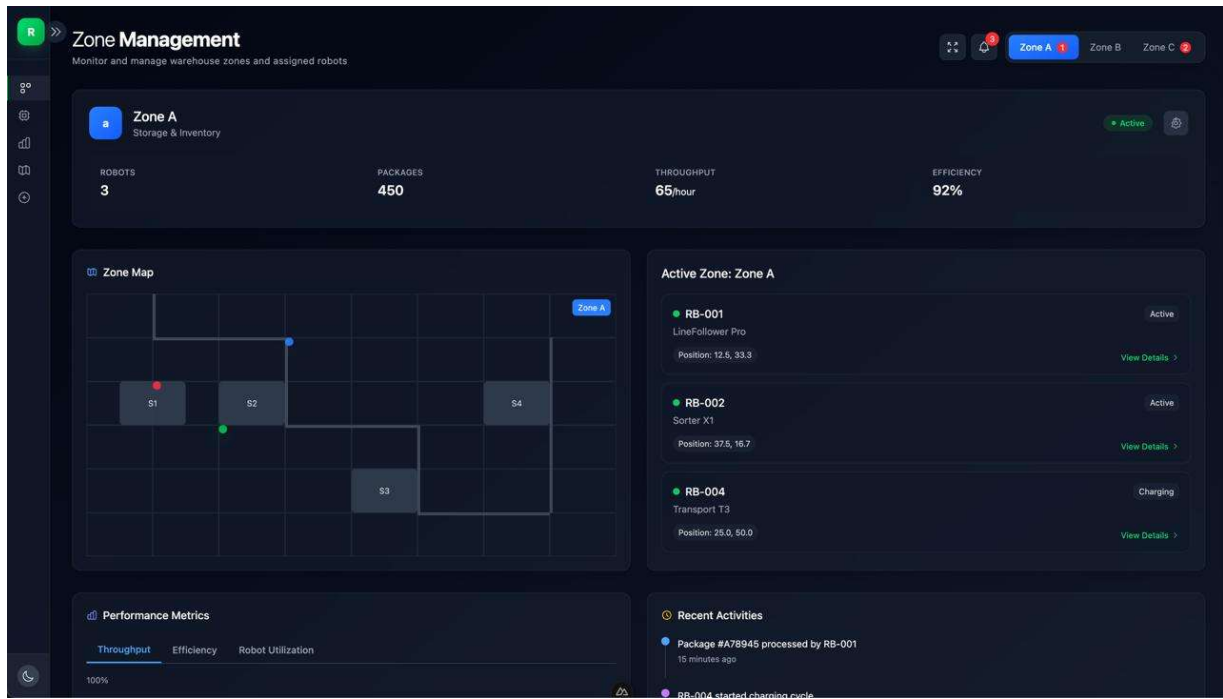


Figure 3.5 Gestion des zones

e. Ajouter un robot

Pour maintenir l'évolutivité, le tableau de bord comprend un formulaire simple pour provisionner et déployer de nouveaux robots dans le système, en configurant leur ID, modèle, capacités et paramètres d'exploitation

Add Robot
Configure and deploy a new robot to your warehouse

Basic Information

Robot ID: RB-006

Model Name: LineFollower Pro

Type: Transport

Firmware Version: v2.1.4

Capabilities

- ☒ Color Detection
- ☒ Autonomous Navigation
- ☒ Obstacle Avoidance
- ☐ Load Sensing

Operating Parameters

Max Speed (m/s): 1.5

Max Load (kg): 50

Battery Life (hours): 8

Operating Zone: All Zones

Additional Notes

Any additional details about this robot...

Figure 3.6 Ajouter un robot

3.3 Résultats obtenus

Dans cette section, nous présentons les résultats expérimentaux obtenus suite à la mise en œuvre du robot mobile intelligent. Des tests ont été réalisés pour valider le bon fonctionnement des principales fonctionnalités du système, à savoir : le suivi de ligne, la détection des couleurs, le système de levage et de dépose, ainsi que l'interface web de supervision. Chaque fonctionnalité a été soumise à plusieurs scénarios de test pour en évaluer la fiabilité, la précision et la réactivité.

3.3.1 Tests de suivi de ligne

Le test de suivi de ligne a consisté à faire évoluer le robot sur un parcours rectangulaire de 120 cm x 120 cm comportant des lignes droites, des virages à angle droit et des bifurcations. Le capteur infrarouge QTR-8A a été utilisé pour détecter la ligne noire tracée sur un fond blanc. Plusieurs essais ont été effectués à différentes vitesses afin de tester la stabilité et la précision du déplacement du robot.

Résultats obtenus :

Le robot a réussi à suivre la ligne avec une efficacité satisfaisante. À vitesse modérée, le taux de réussite était supérieur à 90 %, avec un maintien stable de la trajectoire. À vitesse élevée, quelques déviations ont été observées dans les virages serrés, nécessitant une révision des seuils du capteur. Aucun arrêt ou blocage n'a été constaté lors des tests, ce qui démontre une bonne intégration du module QTR-8A.



Figure 3.7 Trajectoire du robot lors du test de suivi de ligne

3.3.2 Tests de détection des couleurs

Pour évaluer la détection de couleurs, plusieurs objets de teinte distincte (rouge, vert, bleu) ont été placés successivement devant le capteur TCS34725. Les mesures RVB ont été analysées pour vérifier si elles correspondaient correctement aux seuils prédéfinis dans le programme. Les tests ont été effectués sous lumière ambiante, sans lumière directe pour éviter les interférences.

Résultats obtenus :

Le capteur a identifié correctement les couleurs dans 95 % des cas. La couleur verte a été détectée avec le plus haut taux de fiabilité, suivie du rouge, tandis que la reconnaissance du bleu présentait quelques confusions avec des teintes proches comme le marron. Les tests ont également confirmé que l'intensité lumineuse ambiante pouvait influencer légèrement la précision des valeurs. Le système reste cependant suffisamment fiable pour assurer le tri automatique.

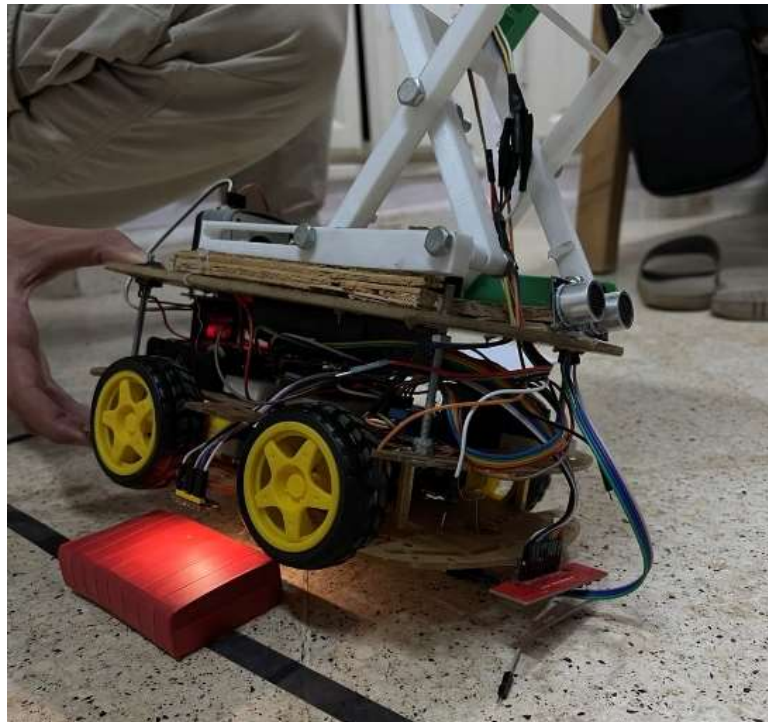


Figure 3.8 Détection correcte de la couleur par le capteur TCS34725

3.3.3 Tests de levage et dépose

Le système de levage, basé sur un moteur à courant continu, a été testé pour sa capacité à soulever et déposer des objets standards de 200 à 300 grammes. L'objet est d'abord détecté, puis saisi, transporté, et enfin relâché dans une zone définie. Le test a été répété plusieurs fois afin d'évaluer la fiabilité mécanique et la précision de positionnement.

Résultats obtenus :

Le système a fonctionné de manière satisfaisante dans 90 % des essais. Le levage s'effectuait sans à-coups, et le dépôt était généralement précis. Les rares échecs étaient liés à un mauvais alignement de l'objet au départ ou à un glissement pendant le transport. Aucune surcharge moteur n'a été observée. Le mécanisme peut donc être jugé robuste pour les charges prévues dans le cadre de ce projet.



Figure 3.9 Séquence de levage et de dépose d'un objet par le robot

3.3.4 Interface web : démonstration et validation

L'interface web a été testée via différents terminaux (PC, smartphone Android) en se connectant à l'adresse IP locale de l'ESP32. L'objectif était de vérifier l'accessibilité de l'interface, l'affichage dynamique des données (batterie, couleur détectée, compteur de marchandises) ainsi que la stabilité de la communication Wi-Fi. Des tests ont aussi été menés sur plusieurs navigateurs pour évaluer la compatibilité.

Résultats obtenus :

L'interface s'est montrée accessible, intuitive et fluide. Les données étaient mises à jour en temps réel sans latence perceptible. Les tests ont révélé une bonne compatibilité avec les navigateurs Chrome, Firefox et Edge. Aucune perte de connexion n'a été constatée durant les sessions de test. Cette interface apporte une réelle valeur ajoutée en matière de supervision du système.

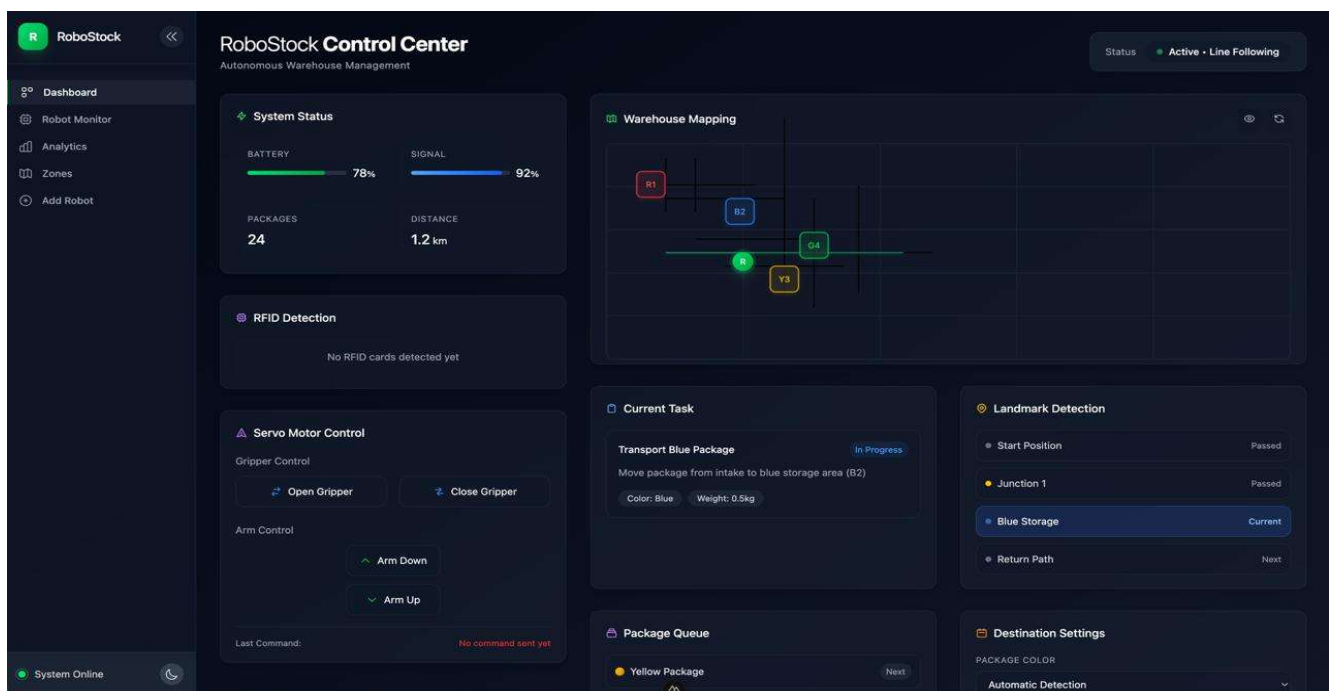


Figure 3.10 Capture d'écran de l'interface web
affichant les données du robot

3.3.5 Test du système complet

Ce test a consisté à évaluer le comportement global du robot dans un scénario représentatif d'une mission réelle de tri dans un entrepôt. Le robot devait suivre un parcours défini en ligne noire, détecter un objet coloré placé sur le trajet, l'identifier selon sa couleur, le lever, le transporter, puis le déposer dans une zone correspondant à cette couleur. L'ensemble du processus a été supervisé via l'interface web, permettant de suivre l'état du robot (batterie, couleur détectée, nombre d'objets traités). L'objectif principal de ce test était de valider l'interopérabilité des modules matériels et logiciels dans une séquence complète de fonctionnement autonome.

Résultats obtenus :

Le robot a accompli avec succès la mission dans 4 cycles consécutifs sans intervention manuelle. Il a suivi la ligne avec précision, identifié correctement la couleur des objets (vert et rouge), puis exécuté les opérations de levage et de dépose avec fluidité. L'interface web a permis de visualiser en temps réel les valeurs de batterie et les objets comptés, sans retard ni erreur de transmission. Aucune collision ni perte d'objet n'a été enregistrée. Le test valide donc la coordination efficace de tous les modules du système (capteurs, moteurs, traitement embarqué, interface) et prouve que le robot est capable de fonctionner de manière autonome dans son environnement cible.



Figure 3.11 Exécution d'une mission complète par le robot (détection, transport et dépôt d'un objet)

3.4 Difficultés rencontrées

Le développement du robot mobile intelligent a impliqué l'intégration de divers composants matériels et logiciels, ce qui a entraîné plusieurs difficultés techniques. Ces obstacles, bien que classiques dans un projet multidisciplinaire, ont nécessité des ajustements ciblés pour assurer un fonctionnement fiable du système dans des conditions réelles.

3.4.1 Problèmes techniques

Plusieurs défis techniques ont été identifiés au cours de la phase de mise en œuvre :

- Le capteur QTR-8A s'est montré sensible aux variations de luminosité ambiante, ce qui a perturbé la précision du suivi de ligne sur certaines zones du parcours. Des soupçons de défaillance matérielle du capteur ont également été relevés, compliquant les phases de calibration.
- Le capteur de couleur TCS34725 a, quant à lui, présenté des difficultés liées aux reflets et aux changements d'éclairage, entraînant des erreurs de détection et des confusions lors du tri des objets par couleur.
- Sur le plan mécanique, la conception du système de levage imprimé en 3D a posé plusieurs problèmes : manque d'espace pour le moteur, frictions entre pièces articulées et déséquilibre général du robot sous charge, menaçant sa stabilité.
- L'alimentation initiale du robot (batterie 12 V unique) s'est révélée insuffisante pour alimenter l'ensemble des moteurs et composants, causant des baisses de tension, des redémarrages intempestifs, et une autonomie limitée.

3.4.2 Solutions apportées

Pour surmonter ces problèmes, plusieurs solutions techniques ont été mises en œuvre :

- Le capteur QTR-8A a été recalibré avec précision, accompagné d'un filtrage logiciel pour minimiser l'influence de la lumière ambiante. Un remplacement préventif du capteur a été effectué, et des protections électroniques ont été ajoutées pour renforcer sa fiabilité.
- La structure du bras de levage a été repensée, avec l'ajout de supports de renfort, la réduction des frottements mécaniques et le repositionnement du moteur pour assurer un meilleur fonctionnement. Un moteur plus puissant avec réducteur a été sélectionné pour garantir la force nécessaire au levage, tout en maintenant la stabilité du châssis.
- un module d'alimentation secondaire a été ajouté pour séparer l'alimentation des moteurs et de la logique. Des batteries lithium-ion de plus grande capacité ont été envisagées pour améliorer l'autonomie et garantir une alimentation stable.

Grâce à ces ajustements, la stabilité et la performance du robot ont été considérablement améliorées, permettant d'atteindre les objectifs du projet tout en consolidant une démarche rigoureuse en conception de systèmes embarqués.

3.5 Perspectives et améliorations futures

Au regard des performances actuelles du robot mobile intelligent et des limites constatées lors des phases de test, plusieurs perspectives d'amélioration ont été identifiées pour optimiser ses fonctionnalités et garantir son intégration efficace dans un environnement industriel réel.

3.5.1 Évolutions matérielles

Afin de renforcer la robustesse et la fiabilité mécanique du système, il est envisagé d'optimiser la structure du robot en utilisant des matériaux plus légers et plus résistants, ainsi qu'en renforçant le système de levage à vis sans fin par un moteur plus puissant et plus stable. Des batteries de plus grande capacité et des moteurs à rendement énergétique élevé seraient également intégrés pour prolonger l'autonomie et améliorer les performances globales.

3.5.2 Améliorations logicielles

Sur le plan logiciel, il serait pertinent d'améliorer les algorithmes de suivi de ligne et de détection des couleurs, notamment en intégrant des techniques de filtrage adaptatif et de compensation de la lumière ambiante. Une optimisation de la communication entre l'ESP32 et l'interface web serait également envisagée afin de réduire les délais de réponse et d'assurer une meilleure interactivité. Enfin, l'ajout de fonctionnalités de diagnostic automatique permettrait de détecter et corriger rapidement les anomalies du système.

3.5.3 Intégration dans un environnement industriel

Dans la perspective d'une industrialisation, il conviendrait d'adapter le robot aux normes de sécurité et aux contraintes logistiques des entrepôts (gestion des stocks, intégration avec les ERP). Le développement d'interfaces de communication industrielles (Ethernet, RS-485, etc.) et la mise en place d'un protocole d'intégration standardisé faciliteraient son adoption dans un cadre professionnel. Des phases de tests en conditions réelles et la formation des opérateurs seraient également planifiées afin de valider l'efficacité et la fiabilité du robot en environnement industriel.

3.6 Conclusion

Nous avons présenté en détail l'interface réseau développée pour le robot intelligemment mobile, ainsi que les résultats obtenus au cours des différentes phases de test.

L'intégration de cette interface web a significativement renforcé les capacités du système en apportant des fonctionnalités clés telles que le suivi du niveau de batterie, le contrôle du chargement, et la commande à distance – des éléments essentiels pour une gestion efficace dans un environnement logistique.

La phase expérimentale a permis de valider plusieurs aspects du fonctionnement du robot : suivi de ligne, détection de couleurs, ainsi que les opérations de levage et de déchargement. Ces essais ont mis en évidence la fiabilité globale du système, tout en révélant certains obstacles techniques qui ont été identifiés et analysés.

Parmi les problèmes rencontrés, on peut citer la sensibilité des capteurs aux variations d'éclairage ambiant, les contraintes mécaniques liées à l'intégration du système de levage, ainsi que la gestion de l'alimentation électrique. Des solutions ont été mises en place pour renforcer la stabilité et la robustesse de l'ensemble du système.

Enfin, les perspectives d'évolution du matériel et des logiciels, ainsi que leur intégration dans un environnement industriel, ouvrent la voie à des améliorations majeures et à une valorisation concrète du robot intelligemment polyvalent.

Ce chapitre constitue ainsi une étape clé dans la consolidation des résultats et dans la planification des futures phases d'optimisation du projet.

Conclusion générale

À travers cette étude, nous avons présenté la conception, la réalisation et l'exploitation d'un système intelligent de gestion de stock dans un centre de distribution, reposant sur un robot mobile autonome. Ce robot a été conçu pour identifier les objets, les transporter et les classer automatiquement en fonction de la couleur des emplacements de stockage.

Pour atteindre ces objectifs, nous avons combiné des éléments matériels (capteurs, moteurs, châssis, système de levage, etc.) avec des solutions logicielles (code embarqué, interface web), afin de répondre aux exigences croissantes d'automatisation dans un environnement logistique.

Nos expérimentations ont montré que le robot est capable de :

- Suivre une ligne noire avec précision,
- Reconnaître les couleurs des objets et des emplacements à l'aide du capteur TCS3472,
- Éviter les obstacles à l'aide du capteur à ultrasons,
- Manipuler les objets (levage, transport, dépôt) de manière efficace.

De plus, une interface web a été développée pour permettre :

- Un contrôle à distance du robot,
- Le suivi de l'état de la batterie,
- Le comptage des produits traités.

L'ensemble du système a été testé dans un environnement de centre de distribution simulé, démontrant ainsi la fiabilité et le potentiel de notre solution.

Bien que certaines améliorations soient encore envisageables – notamment pour renforcer la robustesse, la vitesse ou la capacité à manipuler des objets plus complexes – le prototype développé constitue une base solide pour de futures applications dans le domaine de la robotisation des entrepôts.

Enfin, ce projet nous a permis de mettre en pratique nos compétences en systèmes embarqués, programmation, et mécatronique, tout en nous confrontant à des problématiques concrètes de l'industrie.

Bibliographie

- [1] Kagermann, H., Wahlster, W., & Helbig, J. (2013). Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Final report of the Industrie 4.0 Working Group. Retrieved from <https://www.acatech.de>.
- [2] Lee, J., Bagheri, B., & Kao, H.-A. (2015). *A cyber-physical systems architecture for industry 4.0-based manufacturing systems*. Manufacturing Letters, 3, 18–23.
<https://doi.org/10.1016/j.mfglet.2014.12.001>.
- [3] Siciliano, B., & Khatib, O. (2016). *Springer Handbook of Robotics* (2nd ed.). Springer.
<https://doi.org/10.1007/978-3-319-32552-1>
- [4] Lee, J., Bagheri, B., & Kao, H. A. (2015). A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. Manufacturing Letters, 3, 18–23.
<https://doi.org/10.1016/j.mfglet.2014.12.001>
- [5] Capteur à ultrasons, microsonic, (consulté le 20/07/2022), disponible sur :
<https://www.microsonic.de/fr/support/capteurs-%C3%A0-ultrasons/principe.html>.
- [6] Hocine TAKHI Conception et réalisation d'un robot mobile à base d'arduino Université Amar Telidji - Instrumentation 2014
- [7] <https://eskimon.fr/tuto-arduino-602-un-moteur-qui-a-de-la-t%C3%A0te-le-servomoteur>.
- [8] <https://www.universalis.fr/encyclopedie/infrarouge>.
- [9] Batterie 18650, Fenixlightning, (consulté le 12/9/2022), disponible sur :

<https://www.fenixlighting.com/blogs/news/the-ultimate-guide-to-the-18650-battery>

[10] Arduino, "Arduino Nano," Arduino Official Website. [En ligne]. Disponible sur: <https://www.arduino.cc/en/Guide/ArduinoNano>. [Consulté le 28 mai 2025]

[12] AMS. (2012). TCS3472 Color Light-to-Digital Converter. Retrieved from https://ams.com/documents/20143/36005/TCS3472_DS000390_2-00.pdf

[13] TowerPro, "MG90S Micro Servo Technical Datasheet," [Online]. Available: <https://www.towerpro.com.tw/product/mg90s-2/>

[14] TowerPro, "MG90S Micro Servo Technical Datasheet," [Online]. Available: <https://www.towerpro.com.tw/product/mg90s-2/>

[15] Espesson-Vergeat, B. (2021). L'intelligence artificielle et la blockchain au service de la sécurisation logistique des produits dans le secteur pharmaceutique. Droit, Santé et Société, 8(2), 50-54

[16] Arduino IDE, editions-eni, (consulté le 11/9/2022), disponible sur : <https://www.editionseni.fr/open/mediabook.aspx?idR=45f8874f5e46660935b41d4120bdc22>

[17] Elies Guzman, Arduino IDE, Malavida, (consulté le 13/9/2022), disponible sur : <https://arduino.fr.malavida.com/windows/#gref>

[18] Arduino IDE, WiX, (consulté le 13/9/2022), disponible sur : <https://pecquery.wixsite.com/arduino-passion/ide-arduino>

[19] Arduino IDE, arduino blaise pascal, (consulté le 15/9/2022), disponible sur : <https://arduino.blaisepascal.fr/presentation/logiciel/>

[20] Arduino IDE, softonic, (consulté le 16/9/2022), disponible sur : <https://arduino-ide.fr.softonic.com/?ex=DINS-276.0>

[21] Châssis du robot portable, imprimé en 3D, (consulté le 11/06/2025), disponible sur : <https://monrobotportable.com/chassis-3D>

