République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Yahia Fares de Médéa

Faculté de Technologie

Département de Génie Électrique

Mémoire de Fin d'Études de Master

Présenté en vue de l'obtention du diplôme de Master

Filière : **Automatique**

Spécialité : **Automatique & Systèmes**

Thème

Système de Remplissage et Dosage de Précision pour Petits Volumes en Liquide

---

Préparé et présenté par :

M. Rafik METIDJI

M. Islam BENTEGRI

Proposé et dirigé par :

Prof. Mounir BOUHEDDA

Dr. Hamza BENYEZZA

Année universitaire 2024/2025

Travail préparé au sein du Laboratoire des Systèmes Électroniques Avancés (LSEA) de l'Université de Médéa, ALGÉRIE

Remerciements

Avant tout, nous remercions Dieu le Tout-puissant de nous avoir donné le courage, la volonté, la patience et la santé durant toutes ces années d'études et que grâce à lui que ce travail a pu être réalisé.

En second lieu, nous tenons à remercier notre encadrant Professeur Mounir BOUHEDDA et notre co-encadrant Docteur Hamza BENYEZZA, pour leur orientation et leur accompagnement.

Nous remercions Messieurs les membres de jury pour l'honneur qu'ils nous ont fait pour le jugement de ce travail.

Nous remercions aussi tous les enseignants du Département de Génie Électrique qui ont contribué à cette formation.

Sans oublier l'administration de l'université Dr. Yahia Fares de Médéa qui nous a ouvert ses portes et qui nous a donné l'opportunité d'arriver à ce niveau-là ainsi que la population de Médéa qui a été très aimable et accueillante.

Enfin nous tenons à exprimer nos remerciements à tous nos collègues de la promotion du Master Automatique & Systèmes.

Dédicace

Je dédie ce travail :

Louange à Dieu, le Tout-Puissant, le Clément et le Miséricordieux, pour sa guidance, sa force et sa patience tout au long de ce parcours.

À ma mère, ma grand-mère et mon grand-père, pour leur amour, leurs prières et leur présence constante.

À mon père, pour son soutien constant et ses paroles toujours rassurantes.

À mes sœurs , pour leur soutien et leurs encouragements.

À la famille Otsmane, à Faycel, Abdelghani et Soufiane, pour leur aide, leur confiance et leur appui moral.

À tous mes enseignants du Département de Génie Électrique de l'Université Dr. Yahia Fares de Médéa.

Enfin, à toutes les personnes qui m'ont soutenu de près ou de loin, et à tous les collègues de la promotion du Master Automatique & Systèmes.

Rafik Metidji

Dédicace

Louange à Dieu, Le Tout-Puissant, Le Clément et Le Miséricordieux, qui m'a guidé,
soutenu et accordé la force, la patience et la persévérance tout au long de ce parcours.

Je dédie ce mémoire à celle qui est la lumière de ma vie, ma mère bien-aimée, pour son amour inconditionnel, ses sacrifices silencieux et ses prières constantes.

À mon père, exemple de sagesse et de persévérance, pour sa confiance, ses conseils et son soutien indéfectible.

À mon frère et à mes deux sœurs, pour leur affection, leurs encouragements et leur présence rassurante dans les moments de doute.

À mes amis(frères), véritables piliers dans les moments difficiles, pour leur écoute,leur motivation et leur soutien constant.

À toutes celles et ceux qui, de près ou de loin, ont contribué à l'élaboration de ce mémoire, je vous exprime ma profonde gratitude.

Islam Bentegri

نظام تعبئة جرعات دقيقة لكميات صغيرة من السوائل

ملخص

يهدف هذا العمل إلى تصميم نظام آلي مدمج لتعبئة وقياس كميات صغيرة من السوائل (بين 1 و250 مل) بدقة عالية تبلغ 0.5%±. يعتمد النظام على مضخة تعمل بالتيار المستمر، ويوفر واجهة استخدام مرئية تتيح التحكم في البرمجة، التتبع، وتسجيل الوصفات. يضمن النظام السرعة والجودة في التعبئة، مع إمكانية التوصيل بأنظمة الإنتاج المختلفة لمراقبة العمل عن بعد.

الكلمات المفتاحية: IHM، الجرعات الدقيقة، تعبئة ذكية، مضخة محيطية، آلة تعبئة.

Precision Filling and Dosing System for Small Liquid Volumes

Abstract

This project focuses on the design of an automated precision filling and dosing system for small liquid volumes (1–250 ml) with an accuracy of ±0.5%. The machine uses a DC Water Pump 365 (6–24V), real-time adaptive control, and a visual HMI interface for intuitive operation and recipe management. It ensures high-speed filling while preserving product quality. The system supports modular integration and remote monitoring.

Key words : Automation, Filling machine, HMI, Peristaltic pump, Precision dosing.

Système de Remplissage et Dosage de Précision pour Petits Volumes en Liquide

Résumé

Ce travail présente la réalisation d'un système automatisé de dosage et de remplissage de précision pour de petits volumes liquides (1 à 250 ml) avec une tolérance de ±0,5%. Équipé d'une Pompe à eau en courant continu DC 365 (6–24V), d'un système de contrôle adaptatif et d'une IHM intuitive, le dispositif permet une programmation simple, la gestion de recettes, et l'export des données de production. Il s'adapte à plusieurs types de contenants et offre une connectivité pour une intégration industrielle.

Mots-clés : Automatisation, Dosage de précision, IHM, Pompe péristaltique, Remplisseuse.

Table des matières

Table des figures

Liste des tableaux

List of Abbreviations

General introduction **1**

I État de l'Art et Technologies Existantes **3**

I.1 Introduction . . . 3

I.2 PRÉSENTATION DES TECHNIQUES DE REMPLISSAGE . . . 3

I.2.1 Remplissage à Niveau . . . 3

I.2.2 Remplissage débitmétrique . . . 6

I.2.3 Remplissage Volumétrique . . . 8

I.2.4 Remplissage pondéral . . . 11

I.3 ANALYSES DES SYSTÈMES AUTOMATISÉS . . . 12

I.3.1 Système automatisé . . . 12

I.3.2 Schéma d'un système automatisé . . . 13

I.3.3 Chaîne d'information . . . 13

I.3.4 Chaîne d'énergie . . . 14

I.4 BESOINS SPÉCIFIQUES DES PETITES ENTREPRISES . . . 14

I.4.1 Flexibilité et modularité des équipements . . . 14

I.4.2 Automatisation pour petites entreprises . . . 15

I.4.3 Solutions compactes et polyvalentes . . . 15

I.5 CONCLUSION . . . 15

II Matériels et softwares utilisés **16**

II.1 Introduction . . . 16

II.2 CARTE DE COMMANDE ARDUINO 16
II.2.1 Une Carte Microcontrôleur Polyvalente 17
II.2.2 Pourquoi Arduino Uno ? 18
II.2.3 Microcontrôleur : ATmega328P 19
II.2.4 Présentation physique de l'ATmega328P 21
II.2.5 **Partie Logicielle** 23
II.3 LES CAPTEURS, LES ACTIONNEURS ET LES MODULES DE TRANSMISSION 32
II.3.1 DC WATER PUMP 365 (6–24V) 32
II.3.2 Relais 5V à 2 canaux 33
II.3.3 débitmètre YF-S201 35
II.3.4 Capteur de niveau sans contact XKC Y25 T12V 37
II.3.5 Capteurs Infrarouges (IR) 38
II.4 CONCLUSION 39

III Réalisation Pratique du Système **40**
III.1 INTRODUCTION 40
III.2 CAHIER DES CHARGES 40
III.3 FONCTIONNEMENT GÉNÉRAL DE LA MACHINE DE REMPLISSAGE PROPOSÉE 41
III.3.1 Supervision et interface utilisateur 43
III.3.2 Exemple de scénario réel 43
III.4 PRINCIPE DE MONTAGE DES DIFFÉRENTS COMPOSANTS 44
III.4.1 Capteurs de niveau (XKC Y25 T12V) 44
III.4.2 Capteur de présence bouteille (IR) 45
III.4.3 Pompes à liquide (DC WATER PUMP 365) 46
III.4.4 Module relais 2 canaux 47
III.4.5 Montage et câblage des débitmètres 48
III.4.6 Montage complet du système 48
III.4.7 Illustrations des circuits réels et montage final de la machine 49
III.5 PROGRAMME DE CONTRÔLE DU REMPLISSAGE (ARDUINO) 51
III.5.1 Algorithme du mode automatique 52
III.5.2 Organigramme du système de contrôle 52
III.5.3 Déclaration des constantes et des variables 54
III.5.4 Initialisation 54

III.5.5 Boucle principale 55
III.5.6 Fonction `fillBottle` 56
III.5.7 Fonction `showMenu` 58
III.5.8 Gestion des erreurs et sécurité 59
III.5.9 Communication avec l'IHM 59
III.6 INTERFACE HOMME -MACHINE (IHM) 60
III.6.1 Technologie utilisée : Node-RED 60
III.6.2 Structure du flow 60
III.6.3 Avantages de cette solution 60
III.6.4 Installation de Dashboard 61
III.6.5 Émission et réception des données 61
III.6.6 Configuration de la communication série dans Node-RED 62
III.6.7 Paramétrage du port série 62
III.6.8 Flow développé 64
III.7 MÉTHODOLOGIE DE TEST 66
III.7.1 Tests unitaires 67
III.7.2 Test des capteurs de niveau 67
III.7.3 Test du capteur de présence bouteille 67
III.7.4 Test de commande des pompes 67
III.7.5 Test de la communication série 68
III.7.6 Tests fonctionnels globaux 68
III.7.7 Tests fonctionnels globaux 69
III.8 CONCEPTION 3D DU PROTOTYPE AVEC FUSION 360 80
III.8.1 Objectifs de la modélisation 80
III.8.2 Vue d'ensemble du boîtier final 80
III.8.3 Implantation des composants internes 81
III.8.4 Détail des pièces imprimées en 3D 82
III.8.5 Intégration physique des capteurs et gestion des câbles 83
III.8.6 Avantages de l'approche avec Fusion 360 84
III.9 VALIDATION FINALE DU SYSTÈME 85
III.9.1 Fiabilité de la détection 85
III.9.2 Précision du dosage 85
III.9.3 Robustesse de la communication série 85
III.9.4 Facilité d'utilisation 85

III.10 ÉTUDES COMPLÉMENTAIRES POUR VALORISATION DU SYSTÈME 86
III.10.1Étude de précision 87
III.10.2Étude économique 87
III.10.3Étude comparative avec un modèle du marché 88
III.11 OPTIMISATION ET INTÉGRATION 88
III.11.1 Optimisation du processus de remplissage 88
III.11.2 Amélioration de la robustesse matérielle 89
III.11.3 Évolutions envisagées 89
III.11.4 Intégration dans un environnement de production 89
III.12 CONCLUSION 90

Bibliographie **3**

Table des figures

I.1 *Schéma de remplissage gravitaire* [1] 4
I.2 *Schéma de remplissage sous vide* [1] 5
I.3 *Remplissage isobarométrique* [1] 5
I.4 *Schéma d'un système de débitmétrique*[1] 6
I.5 *Débitmètre à effet électromagnétique* [2] 7
I.6 *Débitmètre massique* [2] 7
I.7 *Débitmètre volumétrique à roues ovales* [2] 8
I.8 *Principe d'une pompe péristaltique* [2] 9
I.9 *Principe de fonctionnement d'une pompe à piston*[2] 10
I.10 *Fonctionnement d'une pompe à piston rotatif*[2] 10
I.11 *Principe de fonctionnement d'une pompe à engrenages* [3] 11
I.12 *Principe du remplissage pondéral par cellule de pesée* [3] 12
I.13 *Schéma d'un système automatisé*[3] 13
I.14 *Constitution de la chaîne d'information*[3] 13
I.15 *Constitution de la chaîne d'énergie*[3] 14

II.1 *Structure de la carte Arduino Uno* [4]. 19
II.2 *Brochage du microcontrôleur ATmega328P*[5] 21
II.3 *Vue physique du microcontrôleur ATmega328P*[5] 22
II.4 *Interface graphique du logiciel Arduino IDE*[6] 24
II.5 *Sélection de la carte et du port dans l'IDE Arduino* [6] 25
II.6 *Sélection manuelle de la carte via le menu Tools dans l'IDE*[6] 25
II.7 *Téléchargement des cartes via le Board Manager*[6] 26
II.8 *Sélection du port COM dans le menu Tools*[6] 27
II.9 *Icône du moniteur série dans l'IDE Arduino*[6] 28
II.10 *Exemple de code pour afficher des données sur le moniteur série*[6] 28
II.11 *Résultat affiché dans le moniteur série*[6] 28

II.12 *Erreur de compilation liée à une librairie manquante*[6] 30
II.13 *Accès au gestionnaire de librairies dans le menu Tools*[6] 30
II.14 *Installation d'une librairie via le gestionnaire*[6] 31
II.15 *Ajout d'une librairie .zip depuis le menu Sketch*[6] 31
II.16 *DC WATER PUMP 365 (6–24V)*[7] 32
II.17 *Module relais 2 canaux 5V – Vue réelle*[8] 33
II.18 *Schéma de connexions du module relais 5V 2 canaux*[8] 35
II.19 *Principe de fonctionnement du capteur YF-S201*[9] 36
II.20 *Débitmètre à effet Hall YF-S201* [9] 37
II.21 *Capteur de niveau sans contact XKC Y25 T12V*[10] 38
II.22 *Module capteur infrarouge – émission et réception IR avec réglage de seuil*[11] 39

III.1 *Schéma général de fonctionnement de la machine de remplissage* 42
III.2 *Schéma de câblage des débitmètres avec Arduino Uno* 45
III.3 *Schéma de câblage du capteur de présence de bouteille avec Arduino Uno* 46
III.4 *Schéma de câblage des pompes via module relais avec Arduino Uno* 46
III.5 *Connexion du module relais à deux canaux avec Arduino Uno* 47
III.6 *Schéma de câblage des débitmètres YF-S201 avec Arduino Uno* 48
III.7 *Schéma global du câblage du système de remplissage automatique* 49
III.8 *Montage physique des débitmètres et du capteur de présence IR.* 50
III.9 *Intégration finale des composants dans le boîtier* 51
III.10 *Organigramme global du fonctionnement du système de remplissage* 53
III.11 *Configuration des broches pour les capteurs et actionneurs sur Arduino Uno* 54
III.12 *Fonction setup() pour l'initialisation de la communication série et des capteurs/actionneurs* 55
III.13 *Gestion du mode manuel et automatique dans la fonction loop() du microcontrôleur* . . 56
III.14 *Fonction fillBottle() pour le dosage séquentiel des liquides* 57
III.15 *Fonction manualControl() pour le pilotage manuel des pompes* 58
III.16 *Affichage du menu utilisateur via la fonction showMenu* 58
III.17 *Installation de la palette Node-RED Dashboard via Manage Palette* 61
III.18 *Nœuds serial in et serial out pour la communication série sous Node-RED* 62
III.19 *Configuration interne du noeud serial* 63
III.20 *Interface Node-RED de commande* 64
III.21 *Interface de contrôle développée sous Node-RED* 66
III.22 *Confirmation visuelle que les deux réservoirs (parfum et alcool) sont pleins .* 70
III.23 *Choix du mode automatique sur l'interface Node-RED* 71

III.24 *Choix du volume 50 ml sur l'interface Node-RED* 71
III.25 *Choix du type de mélange sur l'interface Node-RED* 72
III.26 *Bouton "Remplir" pour lancer le cycle de remplissage* 72
III.27 Résultat du test automatique de remplissage pour 50 ml 73
III.28 *Choix du mode automatique sur l'interface Node-RED* 74
III.29 *Choix du volume 100 ml sur l'interface Node-RED* 74
III.30 *Choix du type de mélange sur l'interface Node-RED* 75
III.31 *Bouton "Remplir" pour lancer le cycle de remplissage* 75
III.32 *Résultat du test automatique de remplissage pour 100 ml* 76
III.33 *Activation du mode manuel via l'interface Node-RED* 77
III.34 *Activation de la pompe parfum avec "ONP"* 77
III.35 *Arrêt de la pompe parfum avec "OFFP"* 77
III.36 *Activation de la pompe alcool avec "ONA"* 78
III.37 *Arrêt de la pompe alcool avec "OFFA"* 78
III.38 *Quitter le mode manuel avec "EXIT"* 79
III.39 *Résultat du test manuel de remplissage pour 160 ml* 79
III.40 *Modélisation complète de la machine réalisée sous Fusion 360* 81
III.41 *Disposition optimisée des composants internes* 82
III.42 *Ensemble des pièces du système (capot, boîtier, supports)* 83
III.43 *Cache imprimé en 3D pour dissimuler les câbles électriques du système* . . . 84
III.44 *Vue d'ensemble de la machine de remplissage entièrement assemblée* 86

Liste des tableaux

II.1 *Les principaux types de cartes Arduino et leurs caractéristiques techniques*[4] 18

II.2 *Principales caractéristiques techniques du microcontrôleur ATmega328P*[5]. 20

III.1 *Récapitulatif des résultats des tests fonctionnels* 80

III.2 *Résultats de l'étude de précision (volume mesuré en ml)* 87

III.3 *Estimation du coût de réalisation du prototype* 87

III.4 *Comparaison fonctionnelle avec un modèle existant* 88

List de Abbreviations

ADC :	Analog to Digital Converter (Convertisseur Analogique-Numérique)
AI :	Artificial Intelligence (Intelligence Artificielle)
ATM :	Atmel (fabricant du microcontrôleur ATmega328P)
ATmega328P :	Microcontrôleur 8 bits utilisé avec Arduino Uno
CPU :	Central Processing Unit (Processeur Central)
DAQ :	Data Acquisition (Acquisition de données)
DC :	Direct Current (Courant Continu)
GPIO :	General Purpose Input/Output (Entrée/Sortie Générale)
HMI :	Human Machine Interface (Interface Homme-Machine)
IDE :	Integrated Development Environment
LCD :	Liquid Crystal Display (Affichage à Cristaux Liquides)
LED :	Light Emitting Diode (Diode Électroluminescente)
MCU :	Microcontroller Unit (Microcontrôleur)
PCB :	Printed Circuit Board (Carte à Circuit Imprimé)
PO :	Partie Opérative
PC :	Partie Commande
PWM :	Pulse Width Modulation (Modulation de Largeur d'Impulsion)
RAM :	Random Access Memory (Mémoire Vive)
ROM :	Read Only Memory (Mémoire Morte)
USB :	Universal Serial Bus

INTRODUCTION GENERALE

L'évolution des techniques constitue un moteur fondamental dans l'amélioration des procédés industriels et commerciaux, en particulier dans les secteurs où la précision du dosage des liquides est essentielle. L'automatisation du remplissage et du dosage se généralise dans les petites officines, les laboratoires et les commerces spécialisés afin d'améliorer l'efficacité, de réduire les pertes et de garantir la qualité des produits.

Le conditionnement de petites quantités de liquides pose des difficultés concrètes dans les petites pharmacies, les boutiques de cosmétiques ou les surfaces alimentaires. En l'absence de solutions adaptées, ces structures utilisent souvent des méthodes manuelles entraînant des erreurs de dosage, une variabilité de qualité et des pertes de produit. Ces défaillances affectent à la fois la rentabilité commerciale et la satisfaction des clients.

L'introduction de technologies modernes permet désormais de concevoir des solutions automatisées fiables, compactes et accessibles. Un système de remplissage automatique, intégrant des capteurs de précision et des actionneurs numériques, garantit une meilleure régularité, un gain de temps et le respect des normes d'hygiène, tout en assurant la traçabilité des opérations.

Dans cette perspective, notre projet a pour objectif la conception d'un système de dosage et de remplissage de précision compact, économique et adapté aux besoins réels des petites entreprises. Il vise à optimiser la fiabilité, le contrôle et l'efficacité du conditionnement tout en restant simple à mettre en œuvre.

Ce mémoire est structuré en trois chapitres :

— Le **chapitre I** présente un état de l'art des techniques de remplissage et de dosage, les systèmes automatisés existants et une analyse des besoins des petites structures.

— Le **chapitre II** décrit les composants matériels et logiciels retenus pour la conception du système, ainsi que les choix technologiques effectués.

— Le **chapitre III** détaille la mise en œuvre pratique du système, de la conception au prototypage, avec les résultats de tests et les ajustements apportés.

Une conclusion générale viendra clore ce travail en récapitulant les apports du projet, en

évaluant leur portée dans les petites structures commerciales et en ouvrant des perspectives d'amélioration et de développement futur [1].

Chapitre I

État de l'Art et Technologies Existantes

I.1 Introduction

Les techniques de remplissage employées par l'industrie sont des méthodes et des procédés importants qui permettent de remplir entièrement des conteneurs, des réservoirs ou des emballages avec des produits liquides, solides, granulaires ou gazeux. Plusieurs techniques se révèlent indispensables au sein de nombreux secteurs industriels. L'agroalimentaire, la pharmacie et la chimie, notamment avec les cosmétiques, sont quelques exemples où la précision, la rapidité et la qualité du remplissage représentent des défis majeurs. Pour satisfaire les besoins actuels de la production, les techniques de remplissage emploient des technologies avancées et des systèmes automatisés pour assurer des conditions stériles, mesurer des volumes ou contrôler les niveaux.

I.2 PRÉSENTATION DES TECHNIQUES DE REMPLISSAGE

I.2.1 Remplissage à Niveau

Cette technique de dosage est déterminée par un niveau : lorsque le liquide obstrue l'orifice d'échappement de l'air, l'écoulement s'arrête. La quantité dosée (en poids) peut varier à cause des tolérances dimensionnelles de la verrerie.

— Précision de dosage variable selon les tolérances du contenant.

— Hauteur constante de remplissage.
— Dosage de volumes variés.

Plusieurs techniques permettent d'obtenir un remplissage à niveau :

— Remplissage gravitaire
— Remplissage sous vide
— Remplissage isobarométrique
— Remplissage isobarométrique sous atmosphère contrôlée

I.2.1.1 Remplissage gravitaire

Le remplissage gravitaire repose sur l'utilisation du poids du liquide, qui s'écoule naturellement depuis une cuve surélevée jusqu'au contenant. Cette méthode simple ne nécessite pas de mécanisme actif, mais impose que la cuve soit positionnée en hauteur.

— Adapté aux contenants rigides ou souples
— Conçu pour produits fluides/liquides
— Nécessite une bonne étanchéité entre bec et bouteille

La Figure I.1 illustre le principe du remplissage gravitaire.

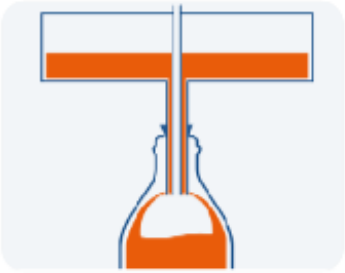


FIGURE I.1 – *Schéma de remplissage gravitaire* [1]

I.2.1.2 Remplissage sous vide

Le remplissage sous vide consiste à créer une dépression au niveau du bec de dosage afin d'aspirer le liquide dans le contenant. Le surplus de liquide est dirigé vers un vase de récupération, assurant ainsi un remplissage précis sans débordement[1].

— Réservé aux contenants rigides
— Adapté aux produits liquides fluides

La Figure I.2 présente un exemple de système de remplissage sous vide.

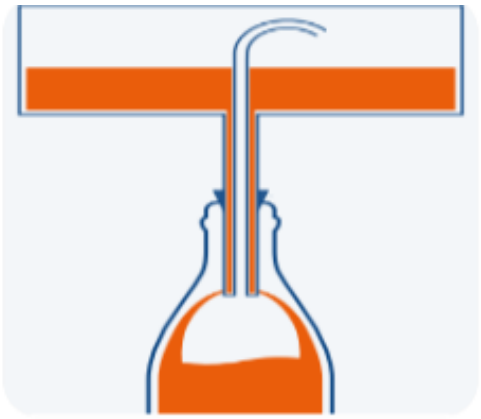


FIGURE I.2 – *Schéma de remplissage sous vide* [1]

I.2.1.3 Remplissage isobarométrique

Ce procédé consiste à pressuriser l'orifice d'échappement avec du CO_2, afin d'équilibrer la pression entre la cuve et le contenant. Cela permet un remplissage sans dégazage du produit, notamment pour les boissons gazeuses.

— Réservé aux contenants rigides

— Spécifique aux produits contenant du CO_2 (ex. : bière)

La Figure I.3 illustre le remplissage isobarométrique.

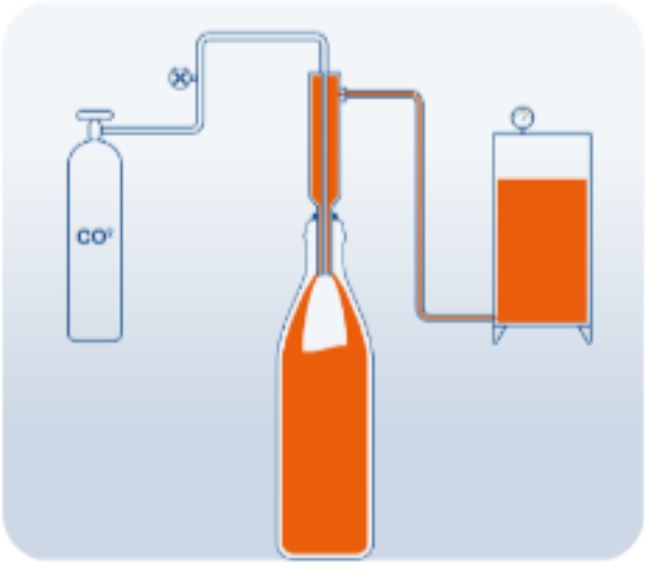


FIGURE I.3 – *Remplissage isobarométrique* [1]

I.2.1.4 Remplissage isobarométrique sous atmosphère contrôlée

Cette technique combine une mise sous vide préalable et une injection de gaz neutre (CO_2 ou N_2) afin de protéger les produits sensibles à l'oxygène. Elle est notamment utilisée dans le conditionnement du vin et d'autres liquides oxydables.

— Contenants rigides.

— Produits sensibles à l'oxydation (ex. : vin) [1].

I.2.2 Remplissage débitmétrique

Le remplissage débitmétrique repose sur un capteur installé sur le bec de remplissage, permettant de garantir un volume ou un poids constant pour chaque dose. La vanne de dosage se ferme automatiquement lorsque la valeur cible est atteinte.

— Précision typique de ±0.5%

— Aucun contact entre bec et contenant (hygiène assurée)

Deux types de débitmètres peuvent être utilisés dans cette configuration : à effet électromagnétique et massique.

Permet un volume ou poids constant par débitmétre installé sur le bec de remplissage. Fermeture automatique par vanne.

— Précision ±0.5%

— Pas de contact bec/contenant (hygiénique)

Deux types : **à effet électromagnétique** et **massique**.

La Figure I.4 montre le principe général d'un système de dosage débitmétrique.

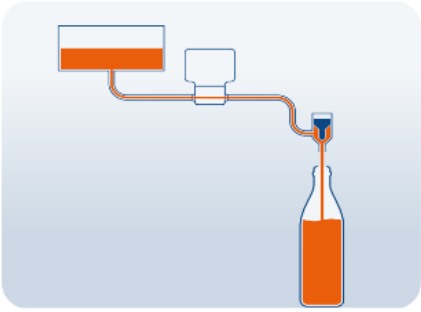


FIGURE I.4 – *Schéma d'un système de débitmétrique*[1]

I.2.2.1 Débitmètre à effet électromagnétique

Ce type de capteur mesure la vitesse d'un liquide en utilisant des électrodes internes qui détectent les variations de champ magnétique générées par le fluide conducteur. Il est exclusivement destiné aux liquides conducteurs tels que ceux à base d'eau, avec une conductivité minimale de 10 μS/cm. Il peut être utilisé avec des contenants rigides ou souples.

— Compatible avec contenants rigides ou souples

— Réservé aux liquides conducteurs (eau, solutions ioniques)

La Figure I.5 illustre le principe de fonctionnement d'un débitmètre à effet électromagnétique.

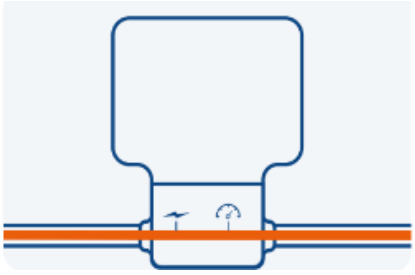


FIGURE I.5 – *Débitmètre à effet électromagnétique* [2]

I.2.2.2 Débitmètre massique

Le débitmètre massique permet de mesurer la masse d'un liquide en circulation à travers un tube, indépendamment de sa densité ou de sa viscosité. Ce type de capteur est particulièrement adapté pour les liquides non conducteurs tels que les huiles ou les solvants. En raison de sa technologie avancée, il représente une solution coûteuse, mais extrêmement précise.

— Mesure indépendante de la densité

— Adapté aux liquides comme l'huile

— Technologie onéreuse

La Figure I.6 présente le fonctionnement d'un débitmètre massique.

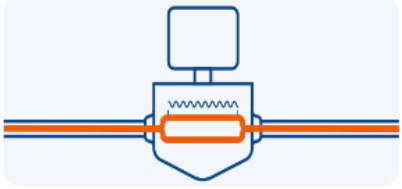


FIGURE I.6 – *Débitmètre massique* [2]

I.2.2.3 Débitmètre volumétrique à éléments rotatifs

Le débitmètre volumétrique repose sur la mise en rotation d'une paire de roues ovales. Chaque rotation permet de mesurer un volume précis de fluide, ce qui en fait un dispositif idéal pour le dosage de grands volumes. Néanmoins, cette technologie implique un entretien rigoureux et un nettoyage complexe en raison de la mécanique interne.

— Adapté au dosage de grands volumes.
— Entretien et nettoyage techniques [2].

La Figure I.7 montre un exemple de débitmètre volumétrique à éléments rotatifs.

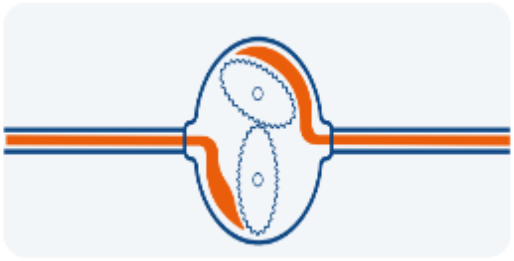


FIGURE I.7 – *Débitmètre volumétrique à roues ovales* [2]

I.2.3 Remplissage Volumétrique

La technique de dosage volumétrique repose sur Une pompe assurant à la fois l'aspiration et le dosage du produit. Dans un premier temps, la pompe aspire un volume de produit (situé dans une cuve) pour l'injecter dans un second temps dans le récipient. Il existe plusieurs types de pompes (volumétrique, à lobes, à membranes, péristaltiques), mais nous ne parlerons ici que des techniques les plus souvent utilisées dans l'univers du remplissage industriel.

— Dosage des produits liquides, visqueux, pâteux et même avec morceaux.
— Large choix de pompes.
— Haute précision.

Plusieurs pompes peuvent être utilisées pour effectuer un remplissage volumétrique :

— Pompe péristaltique.
— Pompe à piston.
— Pompe à piston rotatif.
— Pompe à engrenages [2].

I.2.3.1 Pompe péristaltique

Le principe de la pompe péristaltique repose sur l'écrasement séquentiel d'un tuyau souple par des galets en rotation. Cette action crée un volume fermé qui est déplacé progressivement et injecté dans le contenant. Le liquide ne circule jamais en contact avec les parties mécaniques de la pompe, mais uniquement à l'intérieur du tuyau, ce qui rend le procédé particulièrement hygiénique. Pour changer de produit, il suffit simplement de remplacer le tuyau.

Ce type de pompe est apprécié pour sa précision de dosage, sa flexibilité et sa simplicité de nettoyage. Toutefois, elle génère un flux pulsatoire et peut provoquer la formation de mousse lors du remplissage de certains produits liquides.

Les principaux avantages et limitations de cette technologie sont :

— Technique hygiénique et précise.
— Configuration physique et logicielle facile (taille de tuyaux, valeur de dose, ré-aspiration).
— Flux pulsatoire lié aux galets et au diamètre du tuyau.
— Risque de création de mousse avec certains liquides [2].

La Figure I.8 présente le principe de fonctionnement d'une pompe péristaltique.

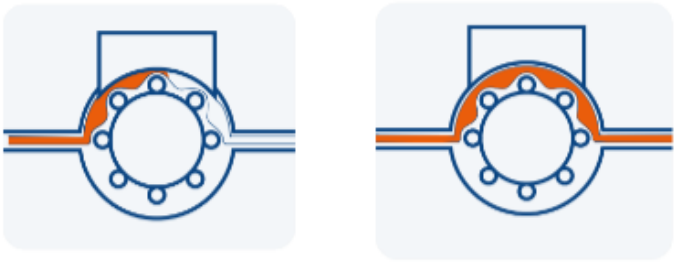


FIGURE I.8 – *Principe d'une pompe péristaltique* [2]

I.2.3.2 Pompe à piston

Grâce à un système d'aspiration, le produit présent dans une cuve est introduit dans la chambre de dosage du piston. Lorsque la quantité prédéfinie est atteinte, le piston pousse le produit vers la vanne puis vers la tête de dosage. Dans certains cas le piston est équipé d'une trémie pour faciliter l'aspiration.

— Polyvalence des produits dosés : liquides, semi-liquides, pâteux, avec morceaux.
— Mauvaise capacité d'aspiration sur de grandes longueurs (surtout produits visqueux).
— Présence de nombreux joints qui constituent autant de pièces d'usure et de zones contaminées [2].

La Figure I.9 présente le principe de fonctionnement d'une pompe à piston.

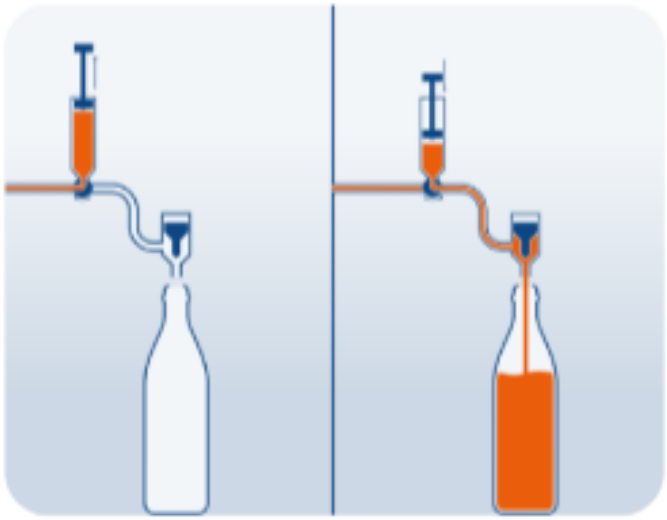


FIGURE I.9 – *Principe de fonctionnement d'une pompe à piston*[2]

I.2.3.3 Pompe à piston rotatif

Dans un premier temps, le liquide est aspiré dans la chambre de dosage (jusqu'à l'obstruction d'un trou marquant l'arrêt et l'obtention de la quantité souhaitée). Le piston est alors mis en rotation, obstruant un premier trou tout en dégageant un second trou. Le piston distribue alors le liquide dosé par le second trou qui n'est plus obstrué.

- Technique hygiénique : pas de joint et donc pas de zone de contamination.
- Entretien simplifié : moins de pièces d'usure présentes.
- Mauvaise capacité d'aspiration [2].

La Figure I.10 présente le principe de fonctionnement d'une pompe à piston rotatif.

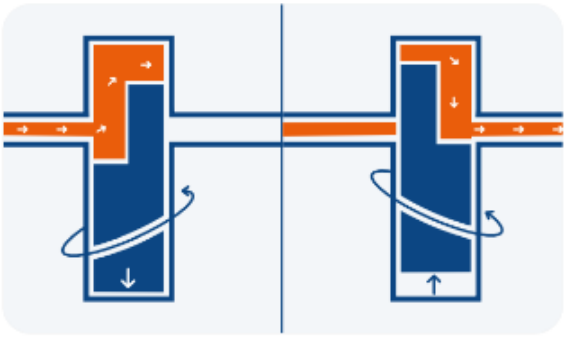


FIGURE I.10 – *Fonctionnement d'une pompe à piston rotatif*[2]

I.2.3.4 Pompe à engrenages

La pompe à engrenage est un dispositif volumétrique dans lequel deux roues dentées tournent en sens opposé à l'intérieur d'un carter fermé, muni de deux orifices (entrée et sortie). Le fluide est capté dans les espaces entre les dents des engrenages et les parois internes du carter. En tournant, les roues entraînent le produit emprisonné vers l'orifice de sortie, assurant ainsi un dosage proportionnel au nombre de rotations.

Ce système de dosage est particulièrement adapté aux produits visqueux ou filants, comme la colle ou le miel, grâce à sa forte capacité d'aspiration et à son contrôle régulier du volume. Il permet également le remplissage de tous types de contenants, qu'ils soient rigides ou souples.

Les avantages principaux de cette technologie sont :

— Compatibilité avec tous types de contenants.

— Excellente capacité d'aspiration.

— Dosage précis de produits visqueux et filants [3].

La Figure I.11 présente le principe de fonctionnement d'une pompe à engrenages.

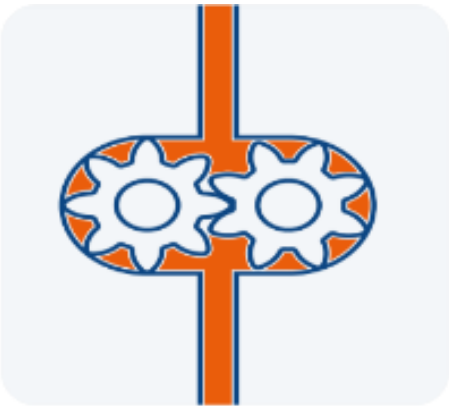


FIGURE I.11 – *Principe de fonctionnement d'une pompe à engrenages* [3]

I.2.4 Remplissage pondéral

La remplisseuse pèse le produit dosé à l'aide d'une balance. Le contenant vide est acheminé au niveau du bec de remplissage où une tare est effectuée automatiquement. Le contenant est ensuite rempli jusqu'à ce que le poids demandé soit atteint.

La balance (ou cellule de pesage) permet de couper ou d'activer le flux de la pompe et même de changer son débit (petite vitesse en fin de dosage). La balance n'est pas en contact avec le produit, ce qui facilite le nettoyage de la machine.

— Technique hygiénique.

— Précision [3].

La Figure I.12 illustre le principe de fonctionnement du remplissage pondéral basé sur une balance.

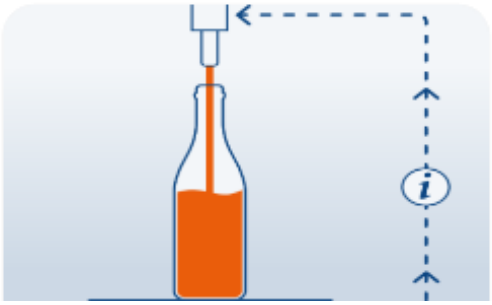


FIGURE I.12 – *Principe du remplissage pondéral par cellule de pesée* [3]

I.3 ANALYSES DES SYSTÈMES AUTOMATISÉS

I.3.1 Système automatisé

Un système automatisé est composé de plusieurs éléments qui exécutent un ensemble de tâches programmées sans que l'intervention de l'homme ne soit nécessaire.

Exemples : le passage à niveau automatique, la porte de garage, etc.

Il est composé de :

— **Chaîne d'information / Partie commande (PC)** : elle donne les ordres et reçoit les informations de l'extérieur ou de la partie opérative. Elle peut se présenter sous trois formes :
  — Un boîtier de commande.
  — Un microprocesseur (cerveau électronique).
  — Un ordinateur.

— **Chaîne d'énergie / Partie opérative (PO)** : c'est la machine qui exécute les ordres. Elle contient :
  — Les **capteurs** : ils recueillent les informations et les transmettent à la partie commande.
  — Les **actionneurs** : ils produisent des actions physiques (déplacement, chaleur, lumière, son)[3].

I.3.2 Schéma d'un système automatisé

Le schéma d'un système automatisé se compose de deux chaînes principales : la chaîne d'information et la chaîne d'énergie. Ces chaînes travaillent ensemble pour produire une action de manière autonome à partir d'une consigne donnée.

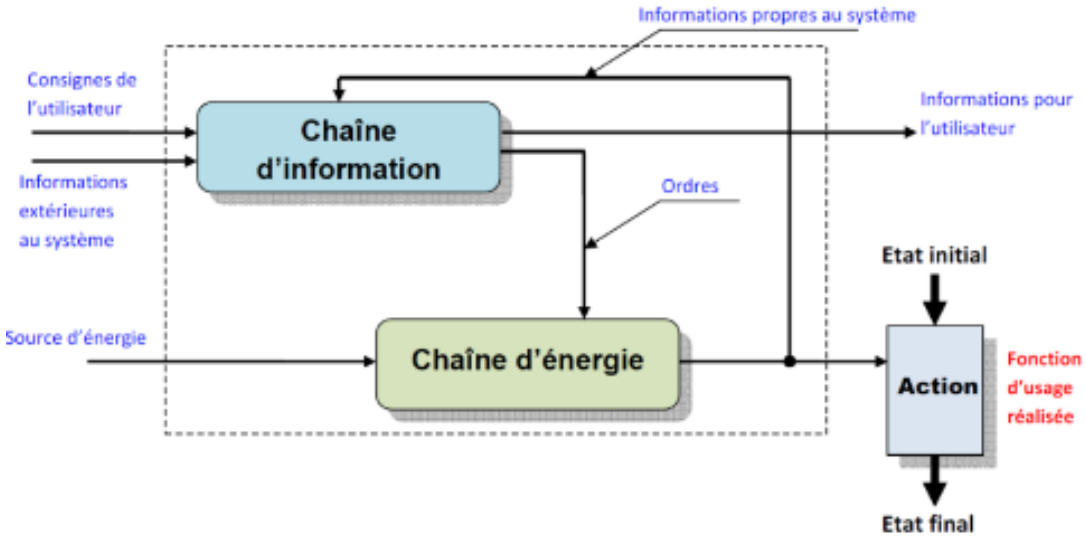


FIGURE I.13 – *Schéma d'un système automatisé*[3]

I.3.3 Chaîne d'information

Définition : c'est la partie du système automatisé qui capte l'information et qui la traite. On peut découper cette chaîne en plusieurs blocs fonctionnels :

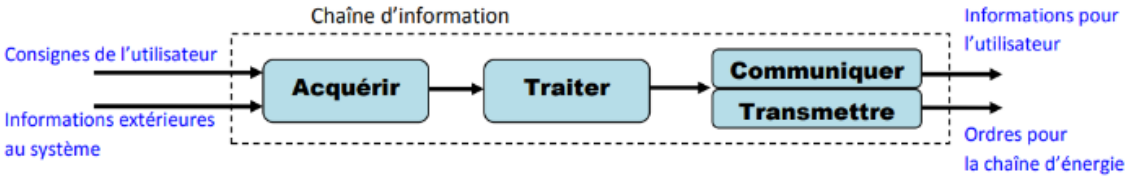


FIGURE I.14 – *Constitution de la chaîne d'information*[3]

Traiter : c'est la partie commande composée d'un automate ou d'un microcontrôleur.

Communiquer : cette fonction assure l'interface avec l'utilisateur et/ou d'autres systèmes.

Transmettre : cette fonction assure l'interface avec l'environnement de la partie commande [3].

I.3.4 Chaîne d'énergie

Définition : dans un système automatisé, on appelle une chaîne d'énergie l'ensemble des procédés qui vont réaliser une action. On peut découper cette chaîne en plusieurs blocs fonctionnels :

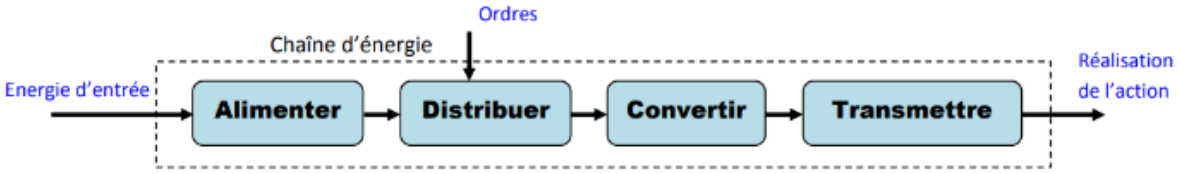


FIGURE I.15 – *Constitution de la chaîne d'énergie*[3]

Alimenter : mise en forme de l'énergie externe en énergie compatible pour créer une action.

Distribuer : distribution de l'énergie à l'actionneur réalisée par un distributeur ou un contacteur.

Convertir : l'organe de conversion d'énergie appelé actionneur peut être un vérin, un moteur...

Transmettre : cette fonction est remplie par l'ensemble des organes mécaniques de transmission de mouvement et d'effort : engrenages, courroies, accouplement, embrayage.... .

I.4 BESOINS SPÉCIFIQUES DES PETITES ENTREPRISES

Les petites entreprises du secteur du remplissage ont des besoins spécifiques pour optimiser leur production tout en respectant des contraintes budgétaires et de flexibilité. Voici les principaux besoins identifiés :

I.4.1 Flexibilité et modularité des équipements

Les PME doivent souvent diversifier leur production pour répondre aux demandes changeantes du marché. L'adoption de systèmes de conditionnement flexibles et modulaires permet d'ajuster rapidement les lignes de production à différentes tailles et formes de produits sans nécessiter d'investissements majeurs dans de nouveaux équipements. Cette approche est

particulièrement avantageuse pour les entreprises souhaitant élargir leur gamme de produits tout en maîtrisant les coûts [4].

I.4.2 Automatisation pour petites entreprises

L'installation de machines de conditionnement et de remplissage automatiques peut contribuer considérablement à accroître la productivité des petites entreprises. Les machines permettent de réduire les coûts manuels, de faciliter la production à une vitesse accrue et de garantir la qualité constante. Il est essentiel de choisir des machines qui conviennent à des volumes de production modestes, offrant une précision et une efficacité augmentées sans l'exigence de dépenses prohibitives [4].

I.4.3 Solutions compactes et polyvalentes

Les contraintes d'espace et de budget sont courantes chez les petites entreprises. Opter pour des équipements de conditionnement compacts et polyvalents permet d'optimiser l'espace de production tout en répondant à une variété de besoins. Ces machines doivent être capables de traiter différents types de produits et formats d'emballage, offrant ainsi une grande adaptabilité aux PME (Petite et Moyenne Entreprise).

En bref, afin de relever avec succès les défis de l'industrie du remplissage, les petites entreprises doivent privilégier des équipements flexibles, automatisés et polyvalents, adaptables à leurs limites budgétaires et spatiales spécifiques [4].

I.5 CONCLUSION

Le chapitre a présenté les principales techniques de remplissage, leur automatisation et a opposé les avantages et les inconvénients. Cela a également mis en évidence les besoins d'une petite entreprise et la possibilité de fournir des solutions flexibles, automatisées et compactes pour augmenter la productivité. En outre, cela servira de base pour les recherches ultérieures.

Chapitre II

Matériels et softwares utilisés

II.1 Introduction

Nous allons présenter au cours de ce chapitre les différents outils matériels et logiciels utilisés pour la réalisation de notre système de remplissage et dosage automatisé.

Concernant les outils matériels, nous avons choisi des composants précis et adaptés à nos besoins techniques : la carte Arduino Uno originale, des débitmètres à effet Hall YF-S201 (1–30 L/min), un relais à 4 canaux (5V), un capteur infrarouge Arduino, des capteurs détecteurs de niveau d'eau sans contact (XKC Y25 T12V), des pompes à eau DC WATER PUMP 365 (6–24V) et un écran tactile SPI 2.8 pouces pour assurer l'affichage et l'interaction utilisateur.

Pour les outils logiciels, nous présenterons particulièrement le logiciel Arduino IDE, utilisé pour programmer et contrôler ces composants, permettant une gestion efficace du système avec une interface utilisateur simple et intuitive.

II.2 CARTE DE COMMANDE ARDUINO

Arduino est un projet inventé en 2005 par une équipe italienne de l'Institut du Design d'Interaction d'Ivrea, composée de six individus :

— Massimo Banzi
— David Cuartielles
— Tom Igoe
— Gianluca Martino
— David Mellis

— Nicholas Zambetti

Cette équipe a créé le "système Arduino". C'est un outil qui permet aux débutants, amateurs ou professionnels de créer des systèmes électroniques plus ou moins complexes.[4]

II.2.1 Une Carte Microcontrôleur Polyvalente

Arduino est une carte microcontrôleur open source, accessible, compacte et largement utilisée pour la commande de systèmes embarqués. Elle dispose d'un port USB pour la programmation, et de plusieurs broches d'entrées/sorties pour connecter des périphériques tels que moteurs, relais, capteurs, LED, haut-parleurs ou encore microphones. Alimentée soit via le port USB, soit par une batterie 9V ou une source externe, elle peut fonctionner en autonomie une fois programmée.

La plateforme Arduino est également soutenue par un environnement de développement gratuit, compatible avec Windows, Mac OS et Linux, ce qui facilite son adoption dans les milieux académiques et industriels[4].
Le caractère open source de son matériel a permis l'émergence de nombreuses variantes adaptées à des besoins spécifiques.

Parmi les cartes les plus courantes, l'**Arduino Uno**, basée sur le microcontrôleur ATmega328P, constitue une solution standard pour les débutants grâce à ses 14 broches numériques (dont 6 PWM), sa mémoire flash de 32 Ko, et sa compatibilité avec une large gamme de shields. L'**Arduino Nano**, plus compacte, offre des caractéristiques similaires mais avec un format réduit facilitant les intégrations limitées en espace. Pour les projets nécessitant un nombre important de broches, l'**Arduino Méga**, dotée de 54 E/S numériques et d'une mémoire flash de 256 Ko, est la plus adaptée.

Certaines cartes se distinguent par des fonctionnalités avancées : l'**Arduino Leonardo** intègre un port USB natif permettant d'émuler un clavier ou une souris ; l'**Arduino Due**, équipée d'un microcontrôleur ARM 32 bits fonctionnant en 3.3 V, convient aux applications exigeantes en performance. Pour les projets IoT, l'**Arduino MKR1000** embarque un module WiFi intégré, tandis que l'**Arduino Zero** est destinée aux utilisateurs avancés grâce à ses performances améliorées.

Enfin, pour des applications ultra-compactes ou à faible consommation énergétique, l'**Arduino Pro Mini** propose un format minimaliste, tout en conservant l'essentiel des capacités de l'Uno. Un résumé comparatif de ces cartes est présenté dans le tableau II.1[4].

TABLE II.1 – *Les principaux types de cartes Arduino et leurs caractéristiques techniques*[4]

Carte Arduino	Microcontrôleur	Tension	Entrées/Sorties	Mémoire Flash	RAM	EEPROM	Spécificités
Arduino Uno	ATmega328P	5V	14 (6 PWM)	32KB	2 KB	1 KB	Débutant, shield-compatible
Arduino Nano	ATmega328p/168	5V	22 (6 PWM)	32KB	2 KB	1 KB	Compact, mini-USB
Arduino Méga	ATmega2560	5V	54 (15 PWM)	256KB	8 KB	4 KB	Projets complexes
Arduino Leonardo	ATmega32u4	5V	20 (7 PWM)	32KB	2.5 KB	1 KB	USB natif, HID support
Arduino Due	ATSAM3X8E	3.3V	54 (12 PWM)	512 KB	96 KB	—	ARM Cortex-M3, 32 bits
Arduino MKR1000	ATSAMW25	3.3V	8 (12 PWM)	256 KB	32 KB	—	WiFi intégré, IoT
Arduino Zero	ATSAMD21G18	3.3V	20 (3 PWM)	256 KB	32 KB	1 KB	Performances avancées
Arduino Pro Mini	ATmega328	3.3V / 5V	14 (6 PWM)	32KB	2 KB	—	Ultra compact, basse conso

II.2.2 Pourquoi Arduino Uno ?

On utilise la carte **Arduino Uno** à cause des raisons suivantes dans le domaine de remplissage et de dosage de précision :

— Facilité d'utilisation.

— Compatibilité avec une grande variété de capteurs et d'accessoires.

— Programmation simple en langage Arduino.

— Fiabilité et robustesse reconnues.

— Coût abordable.

L'Arduino Uno est une carte microcontrôleur fondée sur l'**ATmega328P** d'Atmel. Elle propose 14 broches d'entrée/sortie numériques (dont 6 peuvent être employées comme sorties PWM), ainsi que 6 entrées analogiques. Elle permet de délivrer jusqu'à 40 mA par broche, avec 50 mA pour le port 3,3V. Les 32 Ko de mémoire Flash, 2 Ko de SRAM et 1 Ko d'EEPROM sont des ressources suffisantes pour la majorité des projets de prototypage. Ses dimensions sont de 68,6 mm × 53,4 mm, ce qui en fait un format assez compact pour une palette d'applications considérables.

L'Arduino Uno est équipée d'un connecteur USB et d'un connecteur d'alimentation d'un diamètre de 2,1 mm, cette organisation étant identique à celle de l'Arduino Mega. Elle est compatible avec quasiment tous les shields pour Arduino, ce qui facilite son extension pour couvrir des besoins spécifiques. Comme l'Arduino Uno intègre un convertisseur USB-série, la programmation et les échanges avec un ordinateur deviennent d'une simplicité exemplaire.

La Figure II.1 ci-dessous illustre la carte Arduino Uno avec ses principaux composants visibles[4].

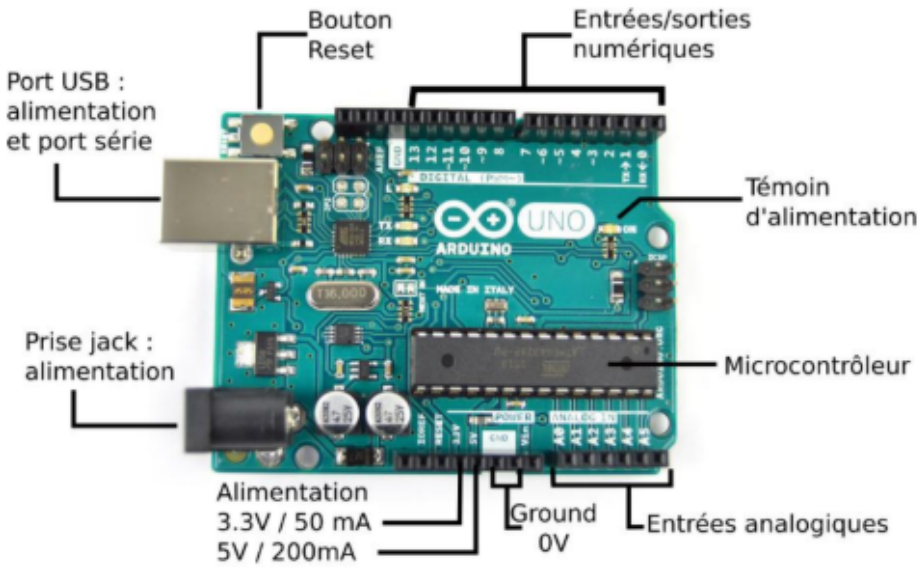


FIGURE II.1 – *Structure de la carte Arduino Uno* [4].

II.2.3 Microcontrôleur : ATmega328P

Le **ATmega328P** est un microcontrôleur 8 bits CMOS basse consommation, basé sur l'architecture AVR à jeu d'instructions RISC optimisé. Grâce à son exécution d'instructions en un seul cycle d'horloge, il peut atteindre un débit de traitement de l'ordre de 1 MIPS par MHz, ce qui le rend particulièrement adapté aux systèmes embarqués nécessitant un bon compromis entre performance et faible consommation énergétique.

Ce microcontrôleur, largement utilisé sur les cartes Arduino Uno, intègre une mémoire flash de 32 Ko (dont 0,5 Ko utilisés pour le bootloader), 2 Ko de mémoire SRAM, ainsi qu'une mémoire EEPROM de 1 Ko pour les données persistantes. Il fonctionne avec une tension de 1.8V à 5.5V et supporte une fréquence d'horloge allant jusqu'à 20 MHz. Il possède 23 lignes d'entrées/sorties numériques, 6 canaux PWM, 6 entrées analogiques, et des interfaces de communication telles que **SPI**, **I2C (TWI)** et **USART**.

Les principales caractéristiques techniques du microcontrôleur ATmega328P sont synthétisées dans le tableau II.2 ci-dessous[5].

TABLE II.2 – *Principales caractéristiques techniques du microcontrôleur ATmega328P*[5].

Architecture	RISC 8 bits
Fréquence d'horloge	16 MHz
Mémoire Flash	32 KB (0.5 KB utilisés par le bootloader)
SRAM	2 KB
EEPROM	1 KB
Broches	28 broches (E/S numériques, analogiques, alimentation)

II.2.3.1 Brochage du microcontrôleur ATmega328P

Le brochage du microcontrôleur ATmega328P permet une compréhension claire des fonctions affectées à chaque broche. Ces broches assurent l'interfaçage entre le microcontrôleur et les différents composants électroniques.

Les caractéristiques essentielles liées aux broches sont les suivantes :

— Tension de fonctionnement : 5V
— Courant DC par broche d'E/S : 20 mA
— Courant DC pour la broche 3.3V : 50 mA
— Canaux PWM : 6
— Canaux ADC : 6 canaux 10 bits
— Protocoles de communication : UART, SPI, I2C [5]

La Figure II.2 ci-dessous illustre la disposition des broches du microcontrôleur ATmega328P avec leurs fonctions respectives.

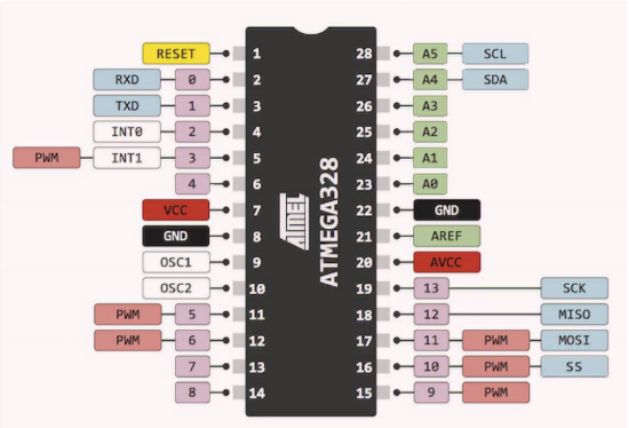


FIGURE II.2 – *Brochage du microcontrôleur ATmega328P*[5]

II.2.4 Présentation physique de l'ATmega328P

Le microcontrôleur ATmega328P, conçu par Microchip Technology, est largement utilisé dans une grande variété de projets électroniques, notamment dans les systèmes embarqués, les cartes Arduino (comme l'Arduino Uno), et les dispositifs de prototypage rapide. Ce composant se distingue par sa structure compacte (souvent disponible en boîtier DIP à 28 broches ou SMD) qui facilite son intégration dans des circuits imprimés, même de petite taille.

Son succès repose sur sa faible consommation d'énergie, qui le rend idéal pour des applications alimentées par batterie, ainsi que sur ses performances solides grâce à un cœur AVR 8 bits cadencé jusqu'à 20 MHz. Il est également reconnu pour sa fiabilité, sa robustesse en environnements industriels, et sa compatibilité avec divers périphériques (EEPROM, ADC, PWM, UART, SPI, etc.), ce qui le rend adapté aussi bien à des usages pédagogiques qu'à des applications professionnelles exigeantes.

La Figure II.3 ci-dessous présente une vue physique typique de l'ATmega328P.

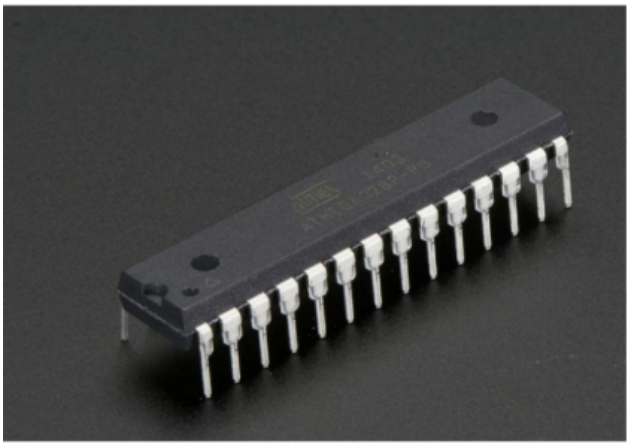


FIGURE II.3 – *Vue physique du microcontrôleur ATmega328P*[5]

II.2.4.1 Les entrées / sorties

La carte Arduino Uno dispose de 14 E/S numériques et de 6 entrées analogiques.

- **Entrées/sorties numériques**

Chacune des 14 broches numériques (numérotée de 0 à 13) sur l'Arduino Uno peut être utilisée comme une entrée ou une sortie. Elles fonctionnent à 5 volts. Chaque broche peut fournir ou recevoir 40 mA en état de fonctionnement. Cependant, il est recommandé de ne pas dépasser cette valeur pour éviter des dommages permanents au microcontrôleur. Un courant total maximum de 200 mA est supporté par l'ensemble des broches.

- **Les entrées / sorties analogiques**

L'Arduino Uno dispose de 6 entrées analogiques, chacune pouvant fournir une mesure d'une résolution de 10 bits (c'est-à-dire sur 1024 niveaux, de 0 à 1023). Par défaut, ces broches mesurent entre 0V (valeur 0) et 5V (valeur 1023). Les broches analogiques peuvent également être utilisées en tant que broches numériques.

II.2.4.2 L'interface de communication

L'interface de communication de l'Arduino Uno permet des échanges de données avec d'autres appareils, qu'ils soient ordinateurs, autres microcontrôleurs ou périphériques divers. Voici un aperçu des principales interfaces de communication disponibles :

— **Communication série (UART)** : utilise les broches 0 (RX) et 1 (TX) pour la communication série entre l'Arduino et d'autres appareils comme un ordinateur ou un module Bluetooth.

— **Communication I2C** : utilise les broches A4 (SDA) et A5 (SCL), permet de connecter l'Arduino à divers capteurs via une communication à deux fils.
— **Communication SPI** : utilise les broches 10 (SS), 11 (MOSI), 12 (MISO) et 13 (SCK), pour des communications rapides avec des modules de mémoire, réseau, etc.

II.2.5 Partie Logicielle

Une carte basée sur un microcontrôleur doit être dotée d'une interface de programmation, comme c'est le cas ici. L'environnement de développement open source pour Arduino peut être téléchargé gratuitement pour Mac, Windows et Linux via :

`https://www.arduino.cc/en/main/software`

Les fonctions principales du logiciel Arduino :

— Écrire et compiler des programmes pour la carte Arduino.
— Se connecter à la carte pour transférer les programmes.
— Communiquer avec la carte une fois le programme chargé.

L'environnement (IDE) comprend :

— Une **barre de menus** (interface graphique classique).
— Une **barre de boutons** pour un accès rapide aux fonctions principales.
— Un **éditeur** de texte pour écrire le code avec navigation par onglets.
— Une **zone de messages** qui indique l'état des actions.
— Une **console texte** qui affiche les résultats de compilation.

La Figure II.14 présente l'interface utilisateur typique de l'environnement de développement Arduino IDE.[6]

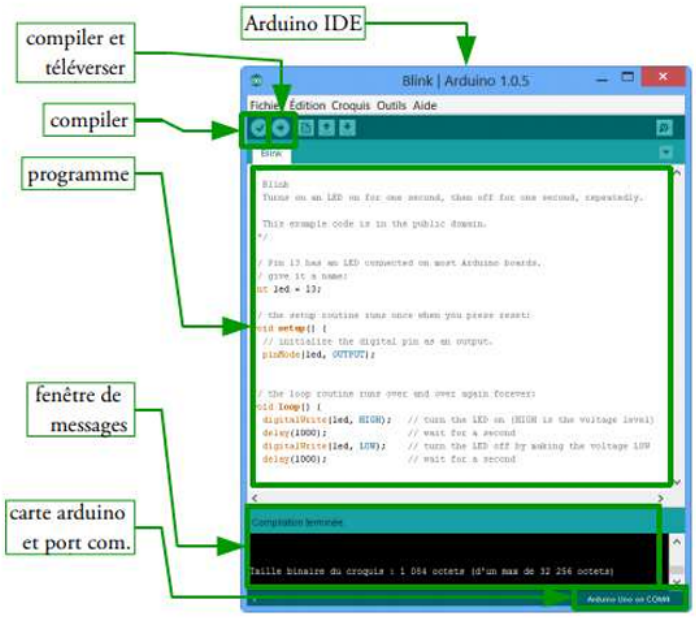


FIGURE II.4 – *Interface graphique du logiciel Arduino IDE*[6]

II.2.5.1 Fonctionnalités avancées de l'IDE Arduino

Le logiciel Arduino intègre également un **terminal série** (fenêtre séparée) qui permet d'afficher des messages textes reçus de la carte Arduino et d'envoyer des caractères vers la carte. Cette fonctionnalité facilite la mise au point des programmes, en permettant d'afficher sur l'ordinateur l'état de variables, de résultats de calculs ou de conversions analogique-numérique. C'est un élément essentiel pour tester, améliorer et corriger ses programmes.

Le code écrit avec le logiciel Arduino est appelé un programme (ou *sketch* en anglais). Ces programmes sont écrits dans l'éditeur de texte, qui dispose des fonctions classiques de copier/coller et de rechercher/remplacer.

La **zone de messages** affiche l'état des opérations comme les sauvegardes ou les exportations, et signale les erreurs. La **console texte** montre les messages produits par le logiciel, incluant les erreurs détaillées et d'autres informations utiles.

La **barre de boutons** permet de vérifier la syntaxe, transférer les programmes, créer, ouvrir et enregistrer le code, ainsi qu'ouvrir le moniteur série. Enfin, la **barre des menus** donne accès à toutes les fonctionnalités du logiciel Arduino.

II.2.5.2 Transfert des programmes vers la carte Arduino

Une fois le programme vérifié et terminé, il est temps de le téléverser sur la carte. Pour cela, vous devez brancher votre carte Arduino sur votre ordinateur.[6]

▷ **Choisir le bon type de carte**

Le logiciel Arduino IDE ne va pas téléverser le programme de la même manière si c'est une carte Arduino Uno ou Nano, car le microprocesseur à l'intérieur n'est pas le même.[6]



FIGURE II.5 – *Sélection de la carte et du port dans l'IDE Arduino* [6]

Si votre carte n'est pas directement reconnue, vous pouvez la sélectionner manuellement dans le menu Tools puis Board :

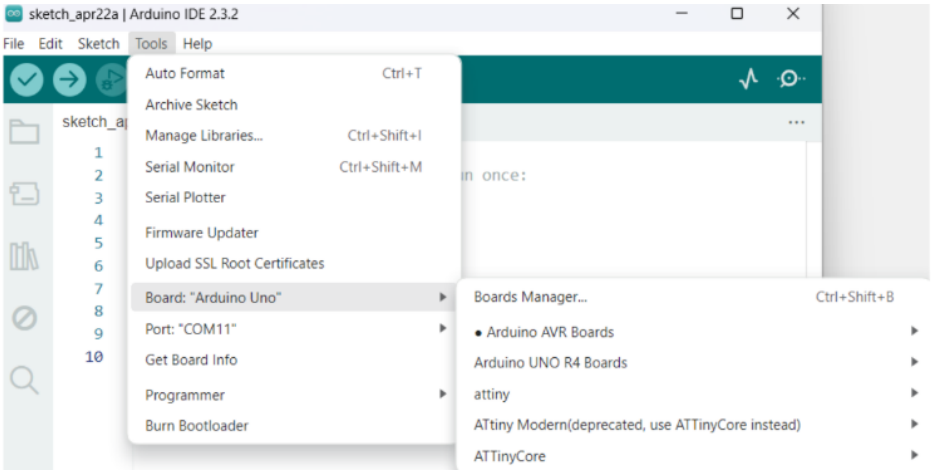


FIGURE II.6 – *Sélection manuelle de la carte via le menu Tools dans l'IDE*[6]

Si vous ne trouvez pas votre carte dans la liste, vous pouvez cliquer sur **Board Manager** et télécharger le module correspondant à votre carte.

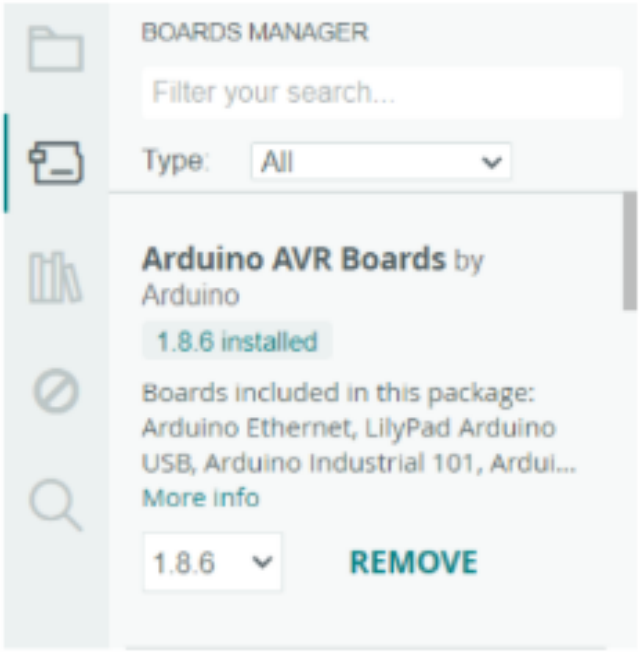


FIGURE II.7 – *Téléchargement des cartes via le Board Manager*[6]

Choisir le Port COM

Si votre carte Arduino n'est pas reconnue directement, vous devez choisir le port COM sur lequel elle est branchée. Le logiciel Arduino IDE propose souvent automatiquement ce port, comme dans l'exemple suivant :

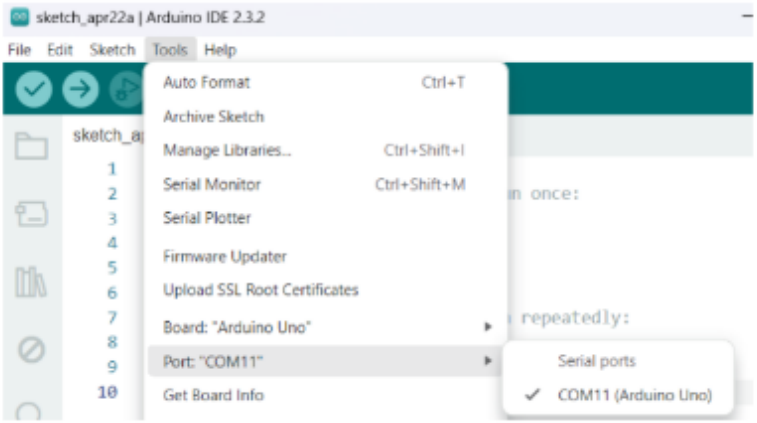


FIGURE II.8 – *Sélection du port COM dans le menu Tools*[6]

Il vous reste alors à téléverser le programme avec le bouton prévu à cet effet.

Moniteur série

Le moniteur série est une interface entre l'utilisateur et la carte Arduino. Il permet d'afficher les informations reçues de la carte, telles que la température lue par un capteur ou l'angle d'un servomoteur.

Ce moniteur ne peut être utilisé que si la carte est connectée et que le programme a bien été téléversé. Il peut être ouvert via le bouton situé en haut à droite de l'IDE Arduino.

L'utilisateur peut aussi entrer des valeurs depuis le moniteur série pour contrôler un servomoteur par exemple.

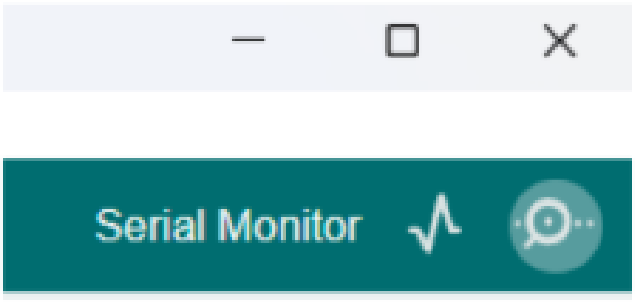


FIGURE II.9 – *Icône du moniteur série dans l'IDE Arduino*[6]

Pour avoir des valeurs sur le moniteur série, vous devez le paramétrer dans votre code :

```
void setup() {
    Serial.begin(9600) ; // Initialisation de la communication avec le moniteur
série (9600 bits/s)
}
void loop() {
    Serial.print("Distance en cm :"); // Permet d'afficher sur le moniteur série
sur la même ligne
    Serial.println("la mesure"); // Permet de sauter une ligne après le texte
}
```

FIGURE II.10 – *Exemple de code pour afficher des données sur le moniteur série*[6]

Le moniteur série doit être configuré avec un nombre précis de bit/s, ici 9600. Ceci va être utile pour recevoir les informations depuis le moniteur série. Assurez-vous que le débit du moniteur série correspond à la valeur définie dans le code. Vous pouvez laisser par défaut 9600 bit/s si votre projet ne nécessite pas de débit spécifique.



FIGURE II.11 – *Résultat affiché dans le moniteur série*[6]

II.2.5.3 Langage Arduino

Le langage Arduino permet de programmer la carte Arduino UNO sur l'IDE. Il est dérivé du langage C, largement utilisé en électronique. La principale différence réside dans la

structure du programme : Arduino utilise obligatoirement deux fonctions : `void setup()` et `void loop()`, alors que le C classique repose sur `main()`.

A. Structure du programme

— Toutes les actions doivent se terminer par un point-virgule (`;`)

— Toutes les fonctions doivent être entourées d'accolades `{ }`

B. `void setup()`

Cette fonction s'exécute une seule fois au démarrage. Elle contient l'initialisation des composants, des broches, du moniteur série... Elle est obligatoire même vide.

C. `void loop()`

Elle s'exécute en boucle infinie. Elle sert à lire les capteurs, afficher sur le moniteur série, contrôler LEDs, servomoteurs... Elle est aussi obligatoire.

II.2.5.4 Bibliothèque sur Arduino

Les librairies (ou bibliothèques) sont un ensemble de fonctions prêtes à l'emploi. Si le programme contient une ligne `#include`, cela signifie qu'une librairie est appelée. Elles ne sont pas obligatoires, mais simplifient énormément la programmation.

II.2.5.5 Quand installer une librairie ?

Lors de la compilation, Arduino IDE vérifie la présence des librairies nécessaires. Si une librairie n'est pas trouvée, une erreur s'affiche comme ci-dessous :

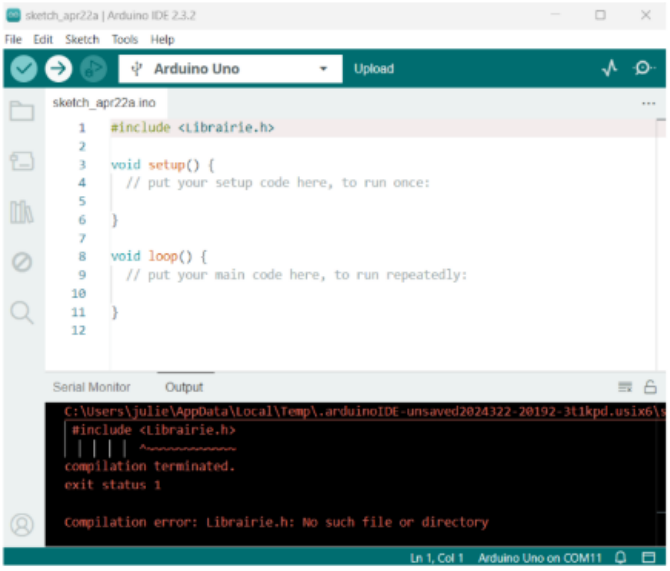


FIGURE II.12 – *Erreur de compilation liée à une librairie manquante*[6]

Ajouter une librairie sur Arduino IDE

- **Gestionnaire de librairie**

Dans Arduino IDE, allez dans `Tools > Manage Libraries...`

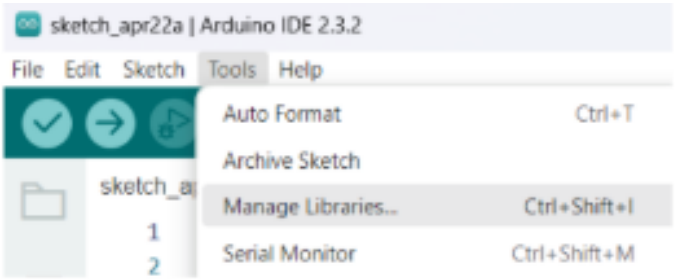


FIGURE II.13 – *Accès au gestionnaire de librairies dans le menu Tools*[6]

Dans la fenêtre, tapez le nom de la librairie, puis cliquez sur `Install` :

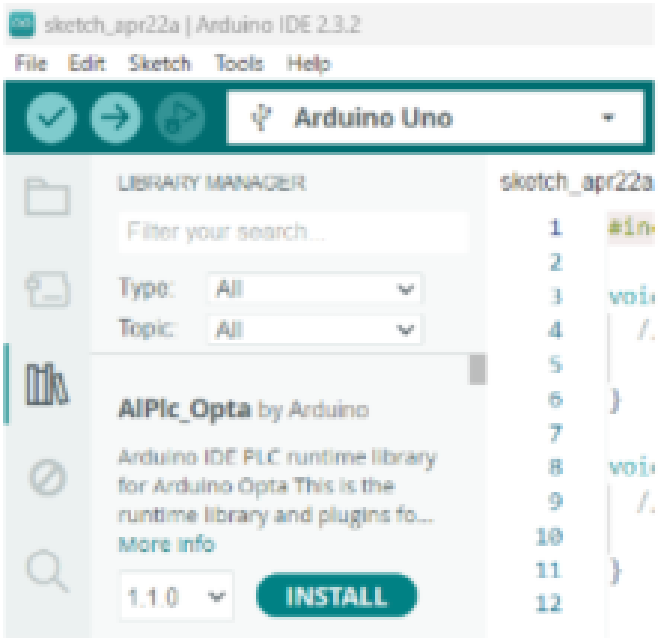


FIGURE II.14 – *Installation d'une librairie via le gestionnaire*[6]

- **Importer une librairie .ZIP**

Vous pouvez aussi ajouter une librairie à partir d'un fichier .zip local via `Sketch > Include Library > Add .ZIP Library...`

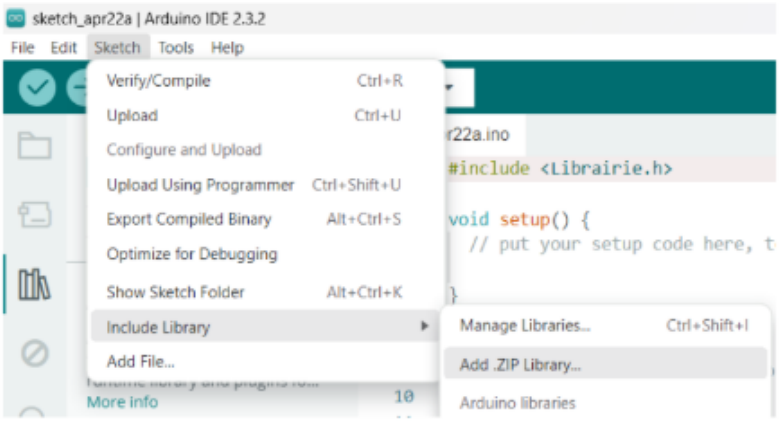


FIGURE II.15 – *Ajout d'une librairie .zip depuis le menu Sketch*[6]

II.3 LES CAPTEURS, LES ACTIONNEURS ET LES MODULES DE TRANSMISSION

II.3.1 DC WATER PUMP 365 (6–24V)

II.3.1.1 Présentation

La DC Water Pump 365 (6–24V) est une pompe à eau électrique de petite taille, idéale pour les projets nécessitant un contrôle précis du débit de liquide. Elle fonctionne dans une plage de tension de 6 à 24V, ce qui la rend adaptée à divers environnements. Ce dispositif est robuste, fabriqué à partir de matériaux de haute qualité garantissant une longue durée de vie.

La pompe est compacte, facile à installer et à entretenir. Elle offre une excellente performance en matière de pompage, même à haute température (jusqu'à 80°C pour une utilisation à court terme). Cette pompe est souvent utilisée dans des applications telles que les systèmes de remplissage automatisés et les dispositifs de transfert de liquide[7].
La Figure II.16 montre l'aspect physique de cette pompe.



FIGURE II.16 – *DC WATER PUMP 365 (6–24V)*[7]

II.3.1.2 Caractéristiques techniques

— Matériel : Plastiques techniques
— Taille : env. 85 × 40 × 35 mm (3.3 × 1.6 × 1.4 in)

- Diamètre de sortie d'eau :
  - Intérieur : env. 5 mm (0.2 in)
  - Extérieur : env. 7 mm (0.3 in)
- Tension de fonctionnement : DC 6–12V (9V1A ou 12V1A recommandé)
- Courant de fonctionnement : 0.5–0.7 A
- Débit : 1.5–2 L/min (environ)
- Aspiration maximale : env. 2 m (6.6 ft)
- Levage maximum : env. 3 m (9.8 ft)
- Durée de vie : jusqu'à 2500 heures
- Température de l'eau : jusqu'à 80 °C
- Tension 6V : puissance 6 W/h

II.3.2 Relais 5V à 2 canaux

II.3.2.1 Présentation

Le module relais 5V 2 canaux est une carte d'interface pouvant être contrôlée directement par divers microcontrôleurs tels qu'Arduino, AVR, PIC, ARM, etc.

Il permet de contrôler deux dispositifs indépendants comme des moteurs, électroaimants ou lampes. Ce module est essentiel dans les projets nécessitant une interface entre microcontrôleur et appareil à haute puissance AC/DC.

Il fonctionne en 5V, consomme peu, et s'intègre facilement à un circuit logique TTL. Il peut commuter des charges jusqu'à 250V AC ou 30V DC[8].
La Figure II.17 présente la vue réelle du module relais 5V.

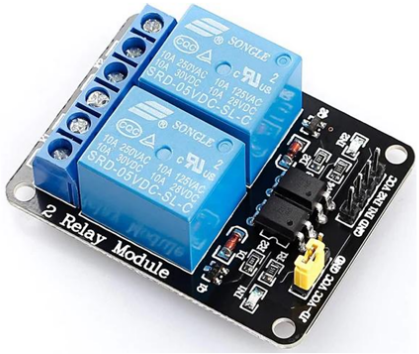


FIGURE II.17 – *Module relais 2 canaux 5V – Vue réelle*[8]

II.3.2.2 Caractéristiques techniques

- Tension d'alimentation : 5V DC
- Courant requis : supérieur à 100 mA
- Charge supportée : 250V 10A AC ou 30V 10A DC
- Dimensions : 50.5 × 38.5 × 18.5 mm
- Poids : 31 g
- PCB : couleur noire, trous de fixation (Ø 3.1 mm)
- Type : SPDT (contact inverseur)
- Isolation : optocoupleur, bonne résistance aux interférences
- Niveau bas (LOW) active le relais, allume la LED d'état

II.3.2.3 Connexions (Pinout)

Le module dispose d'une entrée (signal de commande) et de sorties (connexion de charge) :

Entrée – côté commande :

- **VCC** : alimentation du module (5V)
- **GND** : masse
- **IN1** : commande relais 1 (LOW = actif)
- **IN2** : commande relais 2 (LOW = actif)
- **JD-VCC** : alimentation séparée pour les relais (si jumper retiré)

Sortie – côté charge :

- **NO (Normally Open)** : connecté à COM si relais actif
- **COM (Commun)** : point de bascule commun
- **NC (Normally Closed)** : connecté à COM si relais inactif

La Figure II.18 ci-dessous illustre le schéma des connexions typiques du module.

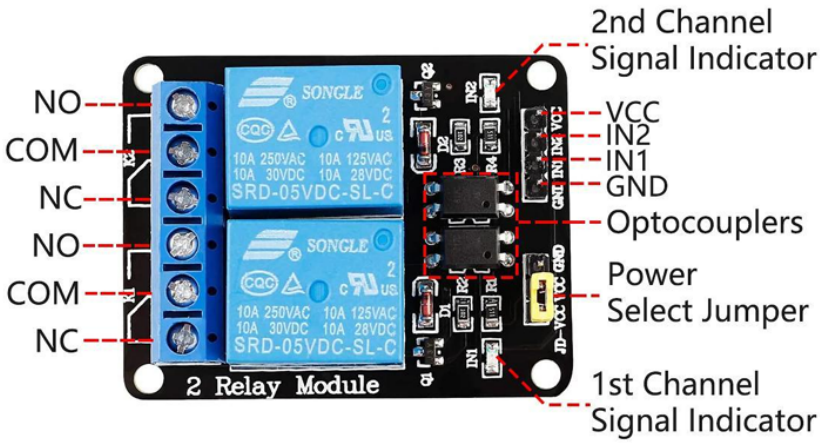


FIGURE II.18 – *Schéma de connexions du module relais 5V 2 canaux*[8]

II.3.3 débitmètre YF-S201

II.3.3.1 Présentation

Le capteur YF-201 permet de mesurer avec précision le débit d'eau dans un tuyau. Il est couramment utilisé dans les systèmes de surveillance, d'irrigation, de refroidissement et même de facturation d'eau pour des bâtiments résidentiels ou industriels.

II.3.3.2 Principe de fonctionnement

Le capteur YF-S201 fonctionne selon l'effet Hall. Une turbine interne tourne avec l'écoulement de l'eau et génère un champ magnétique détecté par un capteur Hall, produisant des impulsions proportionnelles au débit[9].
La Figure II.19 illustre ce principe de fonctionnement.

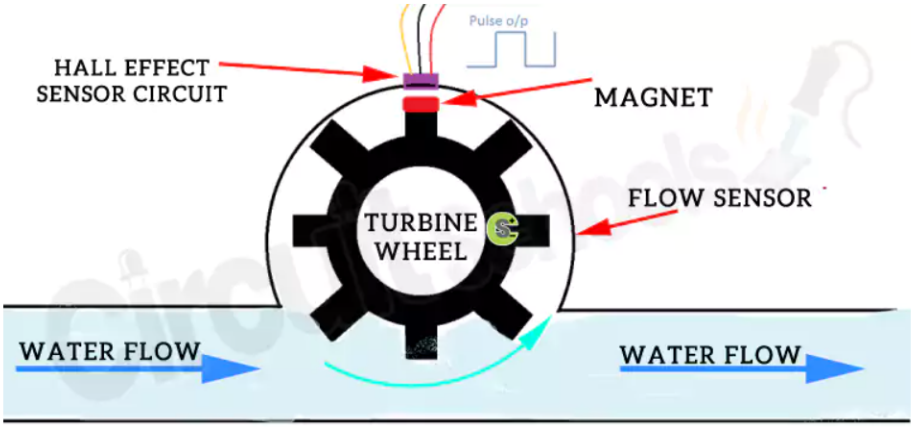


FIGURE II.19 – *Principe de fonctionnement du capteur YF-S201*[9]

La fréquence de sortie est liée au débit par :

$$f(\text{Hz}) = K \times Q(\text{L/min})$$

— $K = 7.5$ pour capteur 1/2”
— $K = 5.5$ pour capteur 3/4”
— $K = 3.5$ pour capteur 1”

II.3.3.3 Spécifications techniques

— Tension : DC 4.5 – 24V (normale : 5–18V)
— Sortie : 5V TTL
— Précision : ±10%
— Débit : 1 – 30 L/min
— Impulsions par litre : 450
— Température : capteur 80°C, liquide 120°C
— Pression maximale : 1.75 MPa
— Filetage : 1/2"
— Durée de vie : > 300000 cycles[9]

La Figure II.20 présente l’aspect réel du capteur YF-S201.



FIGURE II.20 – *Débitmètre à effet Hall YF-S201* [9]

II.3.4 Capteur de niveau sans contact XKC Y25 T12V

Ce capteur détecte le niveau de liquide sans entrer en contact direct avec le fluide. Il est adapté aux liquides corrosifs ou en environnement sous pression, utilisant une puce haute vitesse pour détecter le niveau à travers les parois de récipients.

Il ne dépend pas du type de liquide ou du matériau du récipient et peut être intégré facilement dans des applications industrielles sécurisées. La sensibilité est réglable via un bouton SET[10].

II.3.4.1 Spécifications techniques

- Tension de fonctionnement : 5–24V DC
- Courant : 5mA
- Sortie : 0V (LOW) ou InVCC (HIGH)
- Courant de sortie : 1 – 50mA
- Temps de réponse : 500ms
- Température de fonctionnement : 0 – 105°C
- Épaisseur de paroi détectée : 0 – 13mm
- Étanchéité : IP67
- Dimensions : 28×28mm
- Matériau : ABS

La Figure II.21 illustre l'aspect physique du capteur de niveau sans contact.



FIGURE II.21 – *Capteur de niveau sans contact XKC Y25 T12V*[10]

II.3.5 Capteurs Infrarouges (IR)

Les capteurs infrarouges (IR) sont des dispositifs capables de détecter et mesurer le rayonnement infrarouge émis par les objets. Ils sont utilisés dans les domaines de la sécurité, l'automobile, la médecine et même l'astronomie. Grâce à leur sensibilité à la chaleur, ils peuvent fonctionner même dans l'obscurité[11].

II.3.5.1 Principe de fonctionnement

Un capteur IR utilise un émetteur (LED IR) et un récepteur (phototransistor ou photodiode). L'émetteur envoie un faisceau infrarouge, qui est réfléchi si un obstacle est présent. Le récepteur capte cette réflexion, ce qui déclenche un signal logique.

Certains modules incluent un potentiomètre pour ajuster la distance de détection, et fournissent une sortie numérique directement compatible avec Arduino. La Figure II.22 montre un module IR standard avec ajustement de seuil intégré.

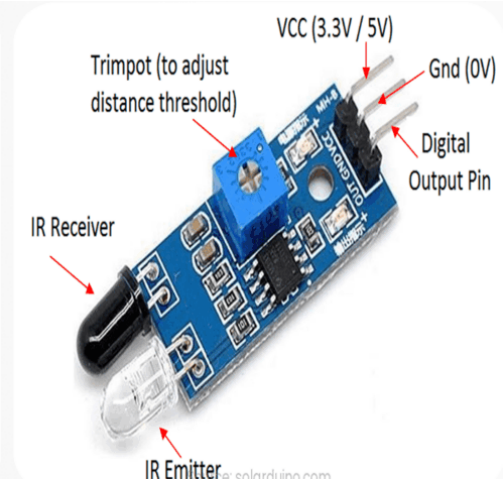


FIGURE II.22 – *Module capteur infrarouge – émission et réception IR avec réglage de seuil*[11]

II.4 CONCLUSION

Dans ce chapitre, nous avons présenté les composants matériels et logiciels nécessaires à la conception de notre système de comptage d'eau intelligent.

La sélection minutieuse de ces outils nous permet d'assurer précision, fiabilité et compatibilité avec les objectifs du projet. Le chapitre suivant portera sur la mise en œuvre concrète du système, incluant l'assemblage, la programmation, et les tests fonctionnels sur le terrain.

Chapitre III

Réalisation Pratique du Système

III.1 INTRODUCTION

Après avoir présenté les composants matériels et logiciels utilisés dans le projet, ce chapitre est consacré à l'implémentation technique et pratique du système développé, à savoir une machine de remplissage et de dosage de précision pour petits volumes liquides, conçue dans un cadre industriel compact et automatisé.

Ce chapitre expose les aspects essentiels de la phase de réalisation du projet. Il met en avant les différentes étapes de conception, de programmation, d'intégration des capteurs et d'élaboration de l'interface de commande, organisées selon les rubriques suivantes :

— Cahier des charges
— Solution proposée
— Connexions des capteurs et des actionneurs avec l'Arduino
— Algorithme de dosage et logique de fonctionnement
— Interface Homme-Machine (IHM) via Node-RED
— Tests fonctionnels et validation du système

III.2 CAHIER DES CHARGES

Dans le but de réaliser une machine de remplissage et dosage de précision pour petits volumes liquides, nous avons établi les principales exigences qui définissent clairement les objectifs techniques et fonctionnels du projet.

— Le système doit permettre le remplissage automatisé de volumes liquides variant de 1 à 250 ml avec une précision de l'ordre de $\pm 0.5\%$.

— L'architecture repose sur une solution matérielle embarquée, utilisant un Arduino Uno comme unité de commande centrale.
— L'intégration de capteurs de niveau sans contact et de capteurs de présence de bouteille est requise pour garantir la sécurité du processus.
— Le contrôle des pompes de dosage est assuré via un module relais 4 canaux, permettant une commande fiable et isolée.
— Une interface utilisateur (IHM) est développée à l'aide de la plateforme Node-RED, afin d'assurer la supervision, le choix des volumes et la visualisation en temps réel des statistiques de production.
— Le système doit permettre une traçabilité des opérations via la communication série entre l'Arduino et l'IHM.
— Le dispositif doit être modulaire, facilement adaptable à différents types de contenants, et prêt à être intégré dans un environnement de production semi-automatique.

III.3 FONCTIONNEMENT GÉNÉRAL DE LA MACHINE DE REMPLISSAGE PROPOSÉE

La machine de remplissage automatique que nous avons développée assure deux fonctions principales de manière autonome et contrôlée :

— Effectuer le remplissage de contenants avec précision, selon un volume sélectionné (50 ml ou 100 ml)
— Garantir la sécurité du processus à travers la vérification en temps réel des niveaux de cuves et de la présence de la bouteille

Le fonctionnement général du système est illustré dans la Figure III.1.

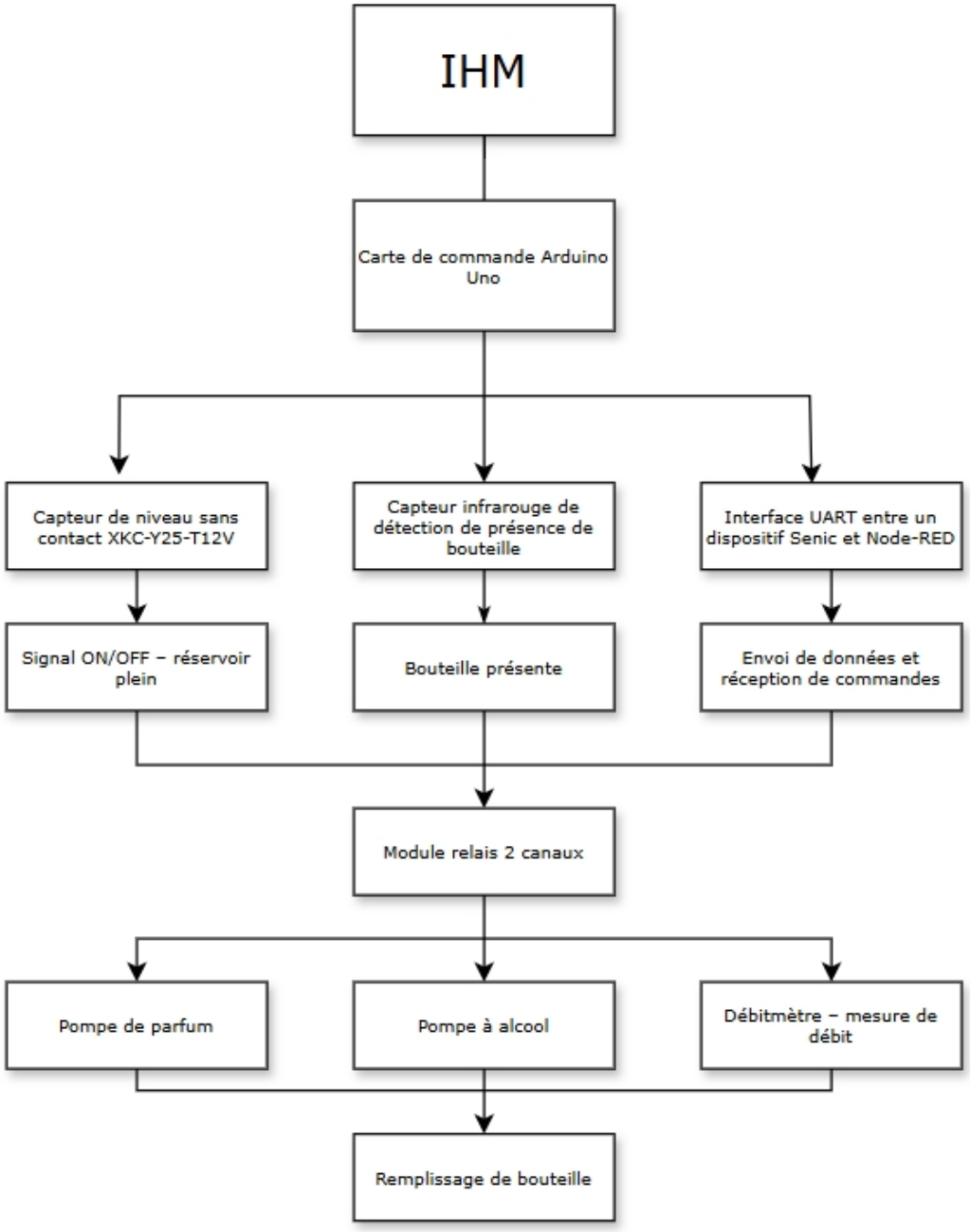


FIGURE III.1 – *Schéma général de fonctionnement de la machine de remplissage*

Le fonctionnement repose sur un microcontrôleur Arduino Uno qui communique avec plusieurs capteurs (capteurs de niveau, capteur de présence) et actionneurs (pompes). L'utilisateur interagit avec le système via une interface homme-machine Node-RED, accessible depuis un navigateur web connecté au réseau local.

Le système adopte un fonctionnement semi-automatisé, basé sur les étapes suivantes :

1. L'utilisateur sélectionne le volume souhaité via l'IHM (50 ml ou 100 ml).
2. Le microcontrôleur vérifie les conditions de sécurité : présence de la bouteille et niveaux suffisants dans les réservoirs d'alcool et de parfum.
3. En cas de validation, les pompes sont activées successivement pour délivrer la quantité requise.
4. Une fois le remplissage terminé, le système envoie un message de confirmation vers l'interface, qui met à jour les statistiques de production.

Le système est conçu pour être modulaire et évolutif, ce qui permet d'ajouter d'autres fonctionnalités à l'avenir comme :

— La gestion de plusieurs têtes de remplissage.
— Le contrôle à distance via Wi-Fi.
— L'automatisation de la sélection de recettes selon le type de produit.

Modes de fonctionnement prévus :

— **Mode automatique** : le système pourrait, à terme, lancer des cycles de remplissage en fonction d'un planning ou d'un déclencheur externe.
— **Mode manuel** : l'utilisateur lance manuellement les cycles de remplissage via l'interface Node-RED, après avoir sélectionné le volume désiré et validé la présence de la bouteille et les niveaux des cuves.

Dans notre prototype actuel, le remplissage est effectué sur une seule ligne avec deux pompes distinctes (alcool et parfum), le tout supervisé par une interface Node-RED connectée en série à la carte Arduino.

III.3.1 Supervision et interface utilisateur

Le système est relié à un ordinateur via une communication série. Une interface utilisateur simple et fonctionnelle a été conçue avec Node-RED, permettant :

— La saisie des paramètres de remplissage (volume, type de mélange, opérateur, lot).
— Le déclenchement automatique du remplissage.
— L'affichage en temps réel de l'état du système (prêt, en cours, erreur, terminé).

III.3.2 Exemple de scénario réel

Dans un laboratoire de parfumerie artisanale, un opérateur est chargé de remplir manuellement des flacons de 50 ml et 100 ml d'un mélange composé d'alcool et d'un

concentré de parfum. Ce travail répétitif est sujet à des erreurs de dosage et engendre une perte de temps.

Grâce au système automatisé développé :

1. L'opérateur place un flacon vide sous le bec de remplissage et accède à l'interface Node-RED depuis une tablette connectée.
2. Il sélectionne le volume souhaité (50 ml ou 100 ml) sur l'IHM.
3. Le système vérifie automatiquement :
   — La présence du flacon via un capteur infrarouge
   — Le niveau minimal dans les réservoirs d'alcool et de parfum
4. En cas de conditions valides :
   — La pompe d'alcool est activée pour une durée calibrée
   — Puis la pompe de parfum démarre à son tour
5. Une fois le cycle terminé, l'IHM affiche :
   — Un message : "Remplissage terminé : 50 ml"
   — Le compteur de bouteilles passe à +1
   — Le volume total distribué est mis à jour
6. L'opérateur retire la bouteille et place la suivante

Ce scénario reflète un usage réel en environnement de production semi-automatisé, tout en maintenant la précision, la sécurité et la traçabilité requises dans les secteurs cosmétique, pharmaceutique ou alimentaire.

III.4 PRINCIPE DE MONTAGE DES DIFFÉRENTS COMPOSANTS

Dans notre système de remplissage automatique, nous avons intégré plusieurs capteurs et actionneurs de manière sécurisée et modulaire, avec une séparation claire entre la commande et la puissance. Chaque composant a été soigneusement monté pour garantir la fiabilité du fonctionnement.

III.4.1 Capteurs de niveau (XKC Y25 T12V)

Nous avons utilisé deux capteurs de niveau sans contact pour surveiller le niveau des cuves d'alcool et de parfum.

Principe de montage :

Chaque capteur possède trois fils : rouge (VCC), noir (GND) et bleu (Signal).

— Le fil rouge est connecté au 5V de l'Arduino

— Le fil noir au GND

— Le fil bleu à l'entrée numérique pin 6 pour la cuve alcool et pin 7 pour la cuve parfum

La Figure III.2 montre le schéma de câblage des débitmètres avec l'Arduino Uno. Ces capteurs détectent la présence de liquide à travers la paroi sans contact direct, ce qui les rend adaptés aux milieux sensibles.

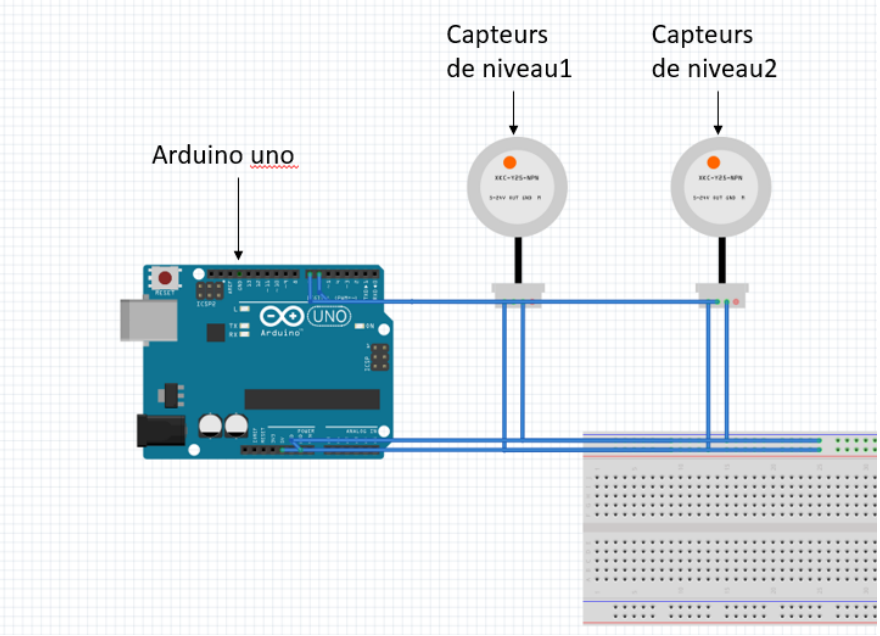


FIGURE III.2 – *Schéma de câblage des débitmètres avec Arduino Uno*

III.4.2 Capteur de présence bouteille (IR)

Un capteur infrarouge permet de détecter la présence d'une bouteille sous les buses avant de lancer le remplissage.

Principe de montage :

Le capteur IR contient trois fils : VCC (rouge), GND (noir) et Signal (jaune).

— Connecté respectivement à 5V, GND, et pin 8 de l'Arduino

Ce montage empêche toute activation des pompes en l'absence d'un contenant. La Figure III.3 montre le schéma de câblage du capteur de présence de bouteille avec l'Arduino Uno.

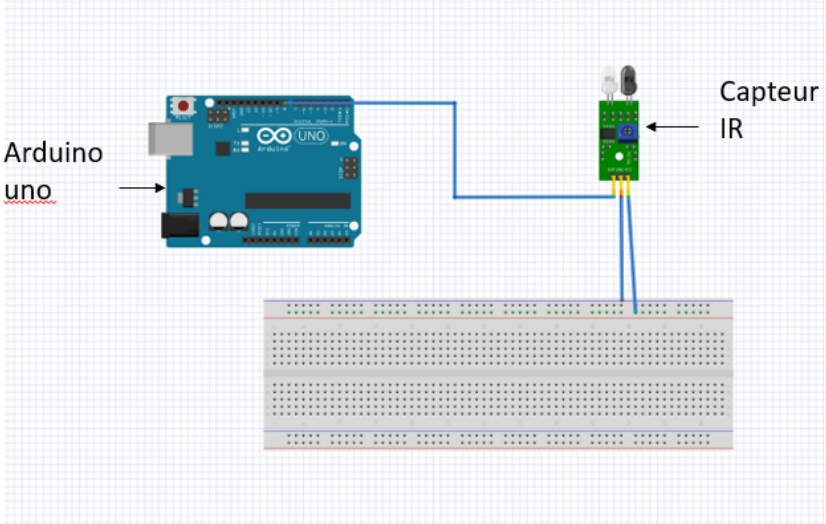


FIGURE III.3 – *Schéma de câblage du capteur de présence de bouteille avec Arduino Uno*

III.4.3 Pompes à liquide (DC WATER PUMP 365)

Deux pompes à courant continu sont utilisées pour doser l'alcool et le parfum.

Principe de montage :

Chaque pompe est connectée à un module relais 5V afin d'isoler la partie puissance (12V) de la carte Arduino. Le relais est activé/désactivé via les pins 4 et 5 pour les pompes alcool et parfum respectivement. La Figure III.4 illustre le schéma de câblage des pompes via module relais avec Arduino Uno.

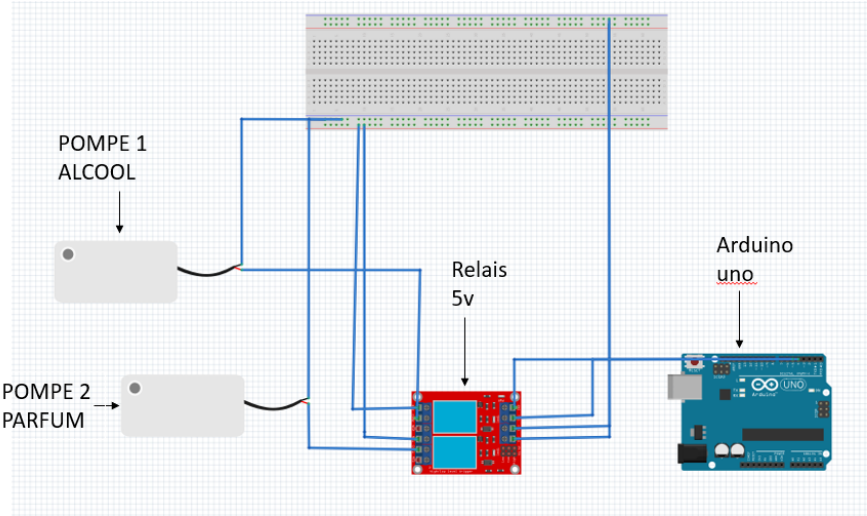


FIGURE III.4 – *Schéma de câblage des pompes via module relais avec Arduino Uno*

III.4.4 Module relais 2 canaux

Le module relais sert à contrôler les pompes sans risquer d'endommager la carte Arduino.

Raccordement à l'Arduino :

— VCC → 5V Arduino

— GND → GND Arduino

— IN → pin de commande (4 ou 5)

Sortie relais :

— COM : borne commune

— NO : normalement ouvert (contact fermé lors de l'activation)

— NC : normalement fermé (contact fermé au repos)

Le système utilise les bornes COM et NO pour assurer un contact uniquement durant le cycle de remplissage. La Figure III.5 montre la connexion du module relais à deux canaux avec l'Arduino Uno.

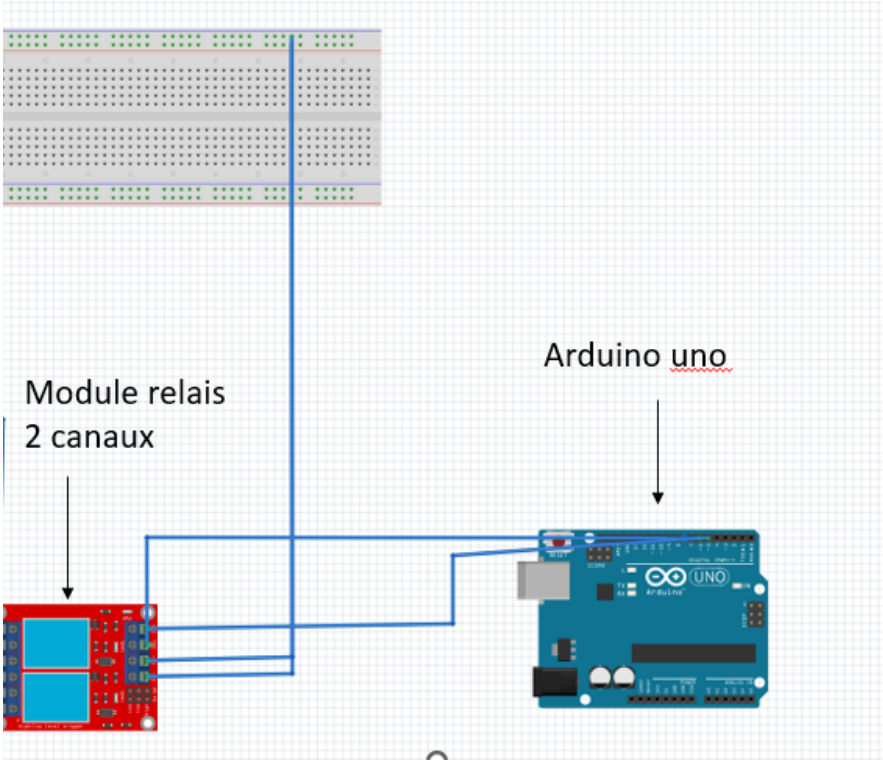


FIGURE III.5 – *Connexion du module relais à deux canaux avec Arduino Uno*

III.4.5 Montage et câblage des débitmètres

Deux débitmètres ont été utilisés pour mesurer les volumes d'alcool et de parfum transitant dans les conduites :

— Débitmètre pour l'alcool : connecté sur la conduite d'alimentation de la pompe à alcool.

— Débitmètre pour le parfum : connecté sur la conduite d'alimentation de la pompe à parfum.

La Figure III.6 présente le schéma de câblage des débitmètres YF-S201 avec Arduino Uno.

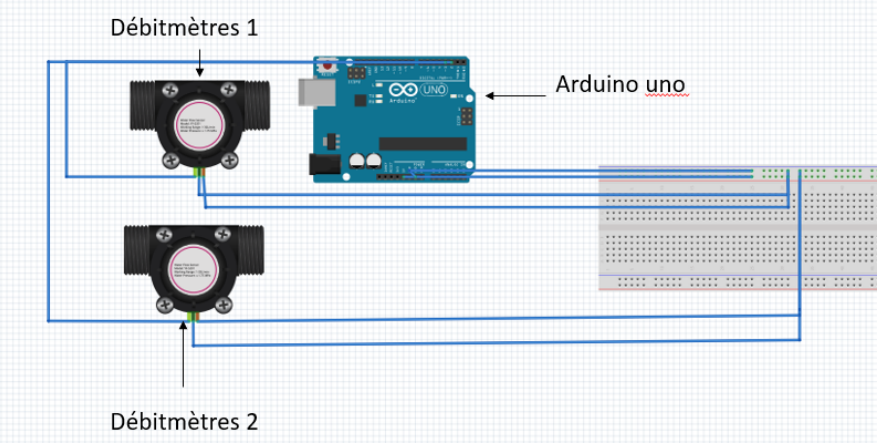


FIGURE III.6 – *Schéma de câblage des débitmètres YF-S201 avec Arduino Uno*

Connexion aux entrées numériques de l'Arduino :

— Débitmètre alcool : fil de signal connecté sur la broche D2 de l'Arduino.

— Débitmètre parfum : fil de signal connecté sur la broche D3 de l'Arduino.

Les autres fils :

— VCC au 5V de l'Arduino

— GND au GND de l'Arduino

III.4.6 Montage complet du système

Tous les composants principaux ont été fixés sur une base plane rigide afin d'assurer la stabilité de l'ensemble.

La disposition suit un schéma fonctionnel :

— Les réservoirs d'alcool et de parfum sont placés en amont des pompes.

— Les pompes sont connectées aux becs de remplissage dirigés vers la zone de placement de la bouteille.
— Les capteurs de niveau (alcool et parfum) sont installés sur les réservoirs respectifs pour surveiller le niveau de liquide disponible.
— Le capteur de présence de bouteille est positionné à l'entrée de la zone de remplissage.
— Le boîtier Arduino est monté sur un support sécurisé à proximité, permettant un accès facile aux câbles USB et aux entrées/sorties. La Figure III.7 présente le schéma global du câblage du système de remplissage automatique.

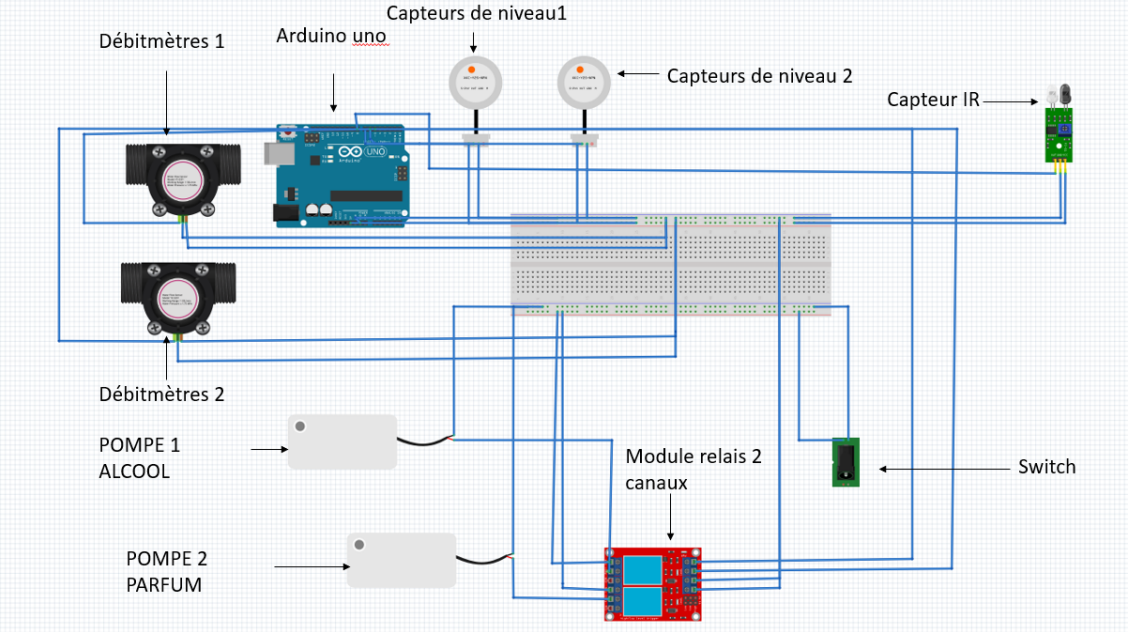


FIGURE III.7 – *Schéma global du câblage du système de remplissage automatique*

III.4.7 Illustrations des circuits réels et montage final de la machine

Afin de compléter les schémas de câblage présentés précédemment, nous avons intégré ci-dessous des images du montage réel du système. Ces photographies permettent d'illustrer l'implémentation physique des différents composants utilisés.

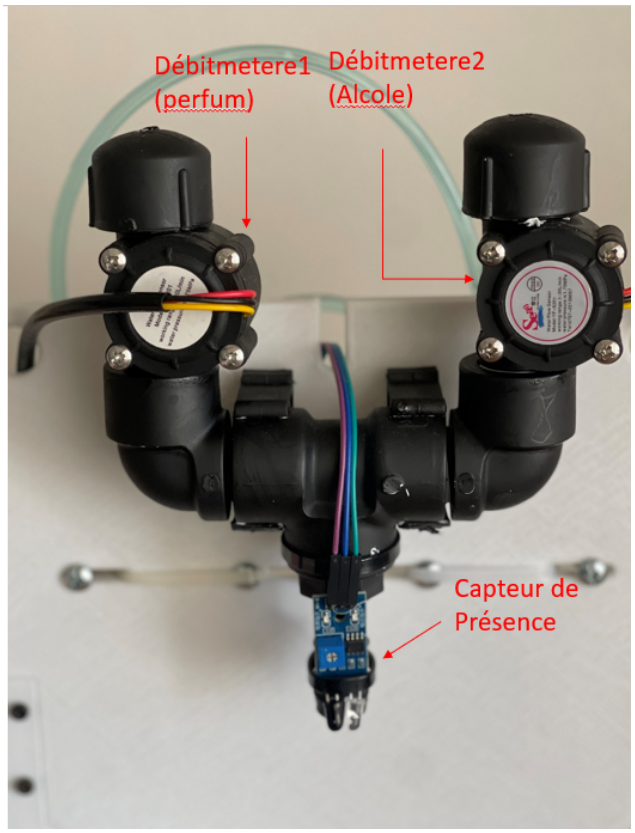


FIGURE III.8 – *Montage physique des débitmètres et du capteur de présence IR.*

La Figure III.8 présente la jonction des deux entrées liquides, à savoir le parfum et l'alcool, chacune mesurée respectivement par un débitmètre YF-S201. Ces capteurs sont fixés à l'aide de supports imprimés en 3D. Un capteur infrarouge (IR) est positionné à la base du montage pour détecter la présence d'une bouteille à remplir.

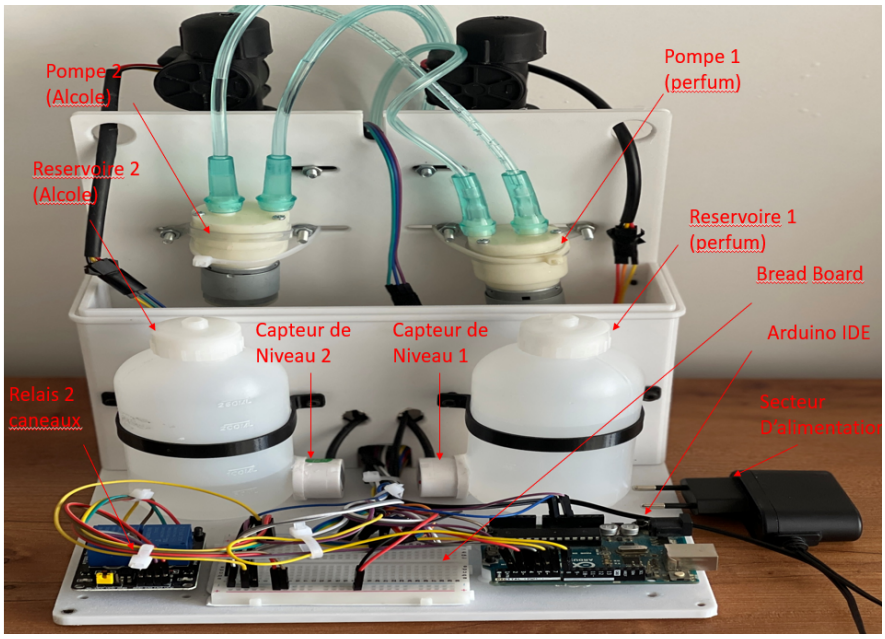


FIGURE III.9 – *Intégration finale des composants dans le boîtier*

Comme illustré en Figure III.9, l'ensemble des composants est monté de manière compacte dans un boîtier imprimé en 3D. Ce boîtier accueille :

— Deux pompes péristaltiques (une pour chaque liquide).
— Deux capteurs de niveau pour surveiller le niveau de liquide dans chaque réservoir.
— Deux réservoirs distincts (alcool et parfum).
— Un module relais à deux canaux assurant la commutation des pompes.
— Une carte Arduino Uno assurant le contrôle du système.
— Une breadboard pour le prototypage rapide et l'organisation des connexions.
— Une alimentation secteur dédiée.

III.5 PROGRAMME DE CONTRÔLE DU REMPLISSAGE (ARDUINO)

Le programme développé sur Arduino constitue le cœur du système de commande de la machine de remplissage. Il gère l'ensemble des interactions entre les capteurs, les actionneurs, et l'interface utilisateur via la liaison série. Le code a été conçu pour être simple, structuré, et évolutif. Dans cette section, nous présentons une analyse détaillée de chaque partie du programme.

III.5.1 Algorithme du mode automatique

Le mode automatique repose sur un ensemble d'étapes conditionnelles. Voici l'algorithme structuré du fonctionnement :

1. Initialisation des ports, du port série, et mise à l'état bas des pompes.
2. Affichage du menu via l'IHM : sélection du volume et du type de liquide.
3. Lecture des entrées série : attente de la sélection volume/type.
4. Vérification des capteurs :
   — Présence bouteille (capteur IR)
   — Niveau réservoir alcool et parfum
5. Si toutes les conditions sont valides :
   — Démarrage de la pompe sélectionnée
   — Lecture du débitmètre en temps réel
   — Arrêt automatique une fois le volume atteint
6. Affichage "Remplissage terminé" et retour à l'état initial

III.5.2 Organigramme du système de contrôle

Le fonctionnement général du système a été modélisé à l'aide d'un organigramme illustrant les différentes étapes selon le mode de fonctionnement choisi par l'utilisateur (manuel ou automatique). Cette représentation permet de visualiser l'enchaînement logique des décisions, des vérifications de capteurs, et de l'exécution du remplissage.

L'organigramme commence par l'initialisation de l'Arduino et de ses ports. L'utilisateur sélectionne ensuite le mode (manuel ou automatique). En mode manuel, les boutons de l'interface IHM permettent un contrôle direct des pompes. En mode automatique, la logique de vérification des capteurs, de sélection du volume et du type de liquide, puis de dosage est exécutée automatiquement. En fin de cycle, le système affiche un message de confirmation et se réinitialise. La Figure III.10 présente l'organigramme global du fonctionnement du système de remplissage, mettant en évidence l'enchaînement des étapes et les décisions prises à chaque étape du processus.

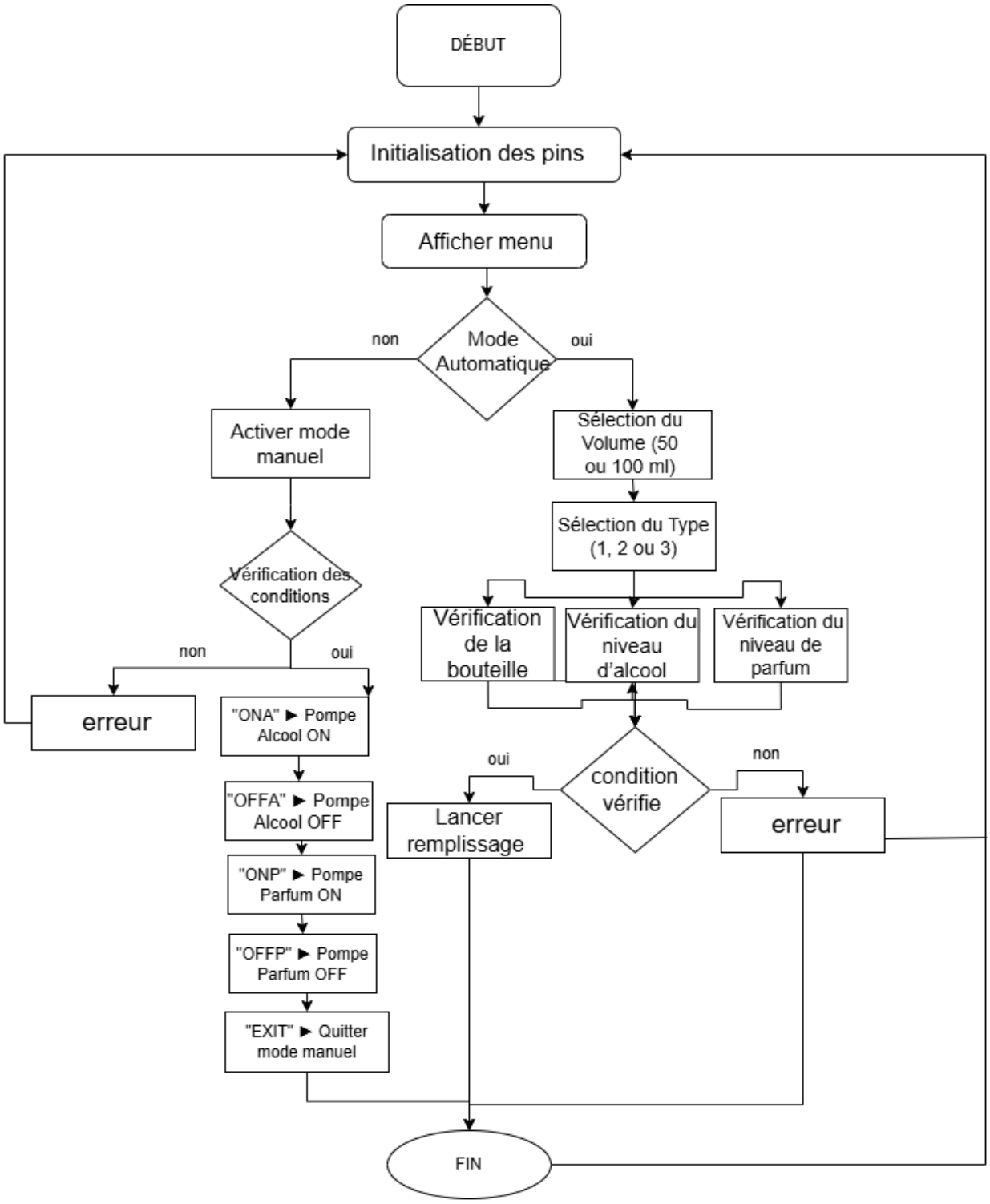


FIGURE III.10 – *Organigramme global du fonctionnement du système de remplissage*

III.5.3 Déclaration des constantes et des variables

— **Définition des broches** : Chaque broche numérique de l'Arduino est assignée à un composant spécifique (capteur ou pompe).

— **Variables globales** : `bottleSize` stocke la taille demandée (50 ou 100 ml), `type` stocke le type de mélange choisi, et `manualMode` détermine si l'utilisateur est en mode manuel ou automatique. Afin de configurer les broches et initialiser les composants de manière optimale, la Figure III.11montre la configuration des broches pour les capteurs et actionneurs sur l'Arduino Uno.

```
1   const int flowAlcohol = 2;
2   const int flowPerfume = 3;
3   const int pumpAlcohol = 4;
4   const int pumpPerfume = 5;
5   const int levelAlcohol = 6;
6   const int levelPerfume = 7;
7   const int bottleSensor = 8;
8
9   int bottleSize = 0;
10  int type = 0;
11  bool manualMode = false;
```

FIGURE III.11 – *Configuration des broches pour les capteurs et actionneurs sur Arduino Uno*

III.5.4 Initialisation

— `Serial.begin(9600)` : Initialise la communication série à 9600 bauds, utilisée pour dialoguer avec Node-RED.

— `pinMode()` : Définit le rôle de chaque broche (entrée ou sortie).

— `digitalWrite(HIGH)` : Au repos, les pompes sont désactivées (relais ouvert par logique inverse).

— `showMenu()` : Affiche les instructions à l'utilisateur dans le moniteur série. La Figure III.12 présente la fonction setup() pour l'initialisation de la communication série et des capteurs/actionneurs.

```
void setup() {
  Serial.begin(9600);
  pinMode(flowAlcohol, INPUT);
  pinMode(flowPerfume, INPUT);
  pinMode(pumpAlcohol, OUTPUT);
  pinMode(pumpPerfume, OUTPUT);
  pinMode(levelAlcohol, INPUT);
  pinMode(levelPerfume, INPUT);
  pinMode(bottleSensor, INPUT);

  digitalWrite(pumpAlcohol, HIGH);
  digitalWrite(pumpPerfume, HIGH);

  showMenu();
}
```

FIGURE III.12 – *Fonction setup() pour l'initialisation de la communication série et des capteurs/actionneurs*

III.5.5 Boucle principale

— **Lecture ligne par ligne** : Chaque commande envoyée depuis Node-RED est reçue comme une ligne indépendante.

— **Mode manuel** : Si `MANUEL` est reçu, le système entre dans un sous-mode où l'utilisateur peut contrôler directement chaque pompe.

— **Remplissage automatique** : Si deux valeurs numériques sont reçues (volume et type), et que les capteurs valident la présence de la bouteille et le niveau des cuves, le cycle est lancé. La Figure III.13 montre la gestion du mode manuel et automatique dans la fonction loop() du microcontrôleur.

```
void loop() {
  if (Serial.available()) {
    String input = Serial.readStringUntil('\\n');
    input.trim();

    if (input == "MANUEL") {
      manualMode = true;
      Serial.println("MODE MANUEL ACTIVÉ");
      manualControl();
    } else {
      if (!manualMode) {
        // Taille (size) et Type
        if (input == "50") bottleSize = 50;
        else if (input == "100") bottleSize = 100;
        else if (input == "1" || input == "2" || input == "3") type = input.toInt();

        if (bottleSize > 0 && type > 0) {
          if (digitalRead(bottleSensor) == LOW &&
              digitalRead(levelAlcohol) == LOW &&
              digitalRead(levelPerfume) == LOW) {
            fillBottle();
            bottleSize = 0;
            type = 0;
            showMenu();
          } else {
            Serial.println("Erreur : bouteille absente ou niveau bas.");
          }
        } else {
          Serial.println("Entrée invalide ou incomplète.");
          showMenu();
        }
      }
    }
  }
```

FIGURE III.13 – *Gestion du mode manuel et automatique dans la fonction loop() du microcontrôleur*

III.5.6 Fonction `fillBottle`

- **Ratios calculés selon le type** :
  - Type 1 = 50/50
  - Type 2 = 30/70
  - Type 3 = 20/80
- **Temps de fonctionnement** : Les durées de pompage sont adaptées en fonction du volume et du ratio sélectionné.
- **Contrôle des pompes** : Les relais sont activés (LOW) puis désactivés (HIGH) selon le délai calculé. La Figure III.14 montre la fonction fillBottle() pour le dosage séquentiel des liquides.

```
// Function to fill the bottle
void fillBottle() {
  float ratioAlcool = 0;
  float ratioParfum = 0;

  if (type == 1) { ratioAlcool = 0.5; ratioParfum = 0.5; }
  else if (type == 2) { ratioAlcool = 0.3; ratioParfum = 0.7; }
  else if (type == 3) { ratioAlcool = 0.2; ratioParfum = 0.8; }

  int totalTime = (bottleSize == 50) ? (durationAlcohol_50ml + durationPerfume_50ml) : (durationAlcohol_100ml + durationPerfume_100ml);
  int timeAlc = totalTime * ratioAlcool;
  int timePar = totalTime * ratioParfum;

  digitalWrite(pumpAlcohol, LOW);
  delay(timeAlc);
  digitalWrite(pumpAlcohol, HIGH);
  delay(300);
  digitalWrite(pumpPerfume, LOW);
  delay(timePar);
  digitalWrite(pumpPerfume, HIGH);

  Serial.print("REMPLI: ");
  Serial.print(bottleSize);
  Serial.print(" ml | Type ");
  Serial.println(type);
  Serial.println("Remplissage terminé.");
}
```

FIGURE III.14 – *Fonction fillBottle() pour le dosage séquentiel des liquides*

La fonction `manualControl` permet à l'utilisateur de contrôler manuellement les pompes du système. Lors de l'activation de cette fonction, le système entre dans une boucle où il intercepte et traite uniquement les commandes suivantes :

`ONA` : Active la pompe à alcool.

`OFFA` : Désactive la pompe à alcool.

`ONP` : Active la pompe à parfum.

`OFFP` : Désactive la pompe à parfum.

`EXIT` : Permet de revenir au mode automatique après avoir effectué les réglages manuels.

Cette fonctionnalité permet à l'utilisateur de tester les pompes individuellement ou de réaliser un remplissage partiel. Le système permet une gestion fine et précise du processus de remplissage, en assurant que chaque pompe fonctionne indépendamment et conformément aux commandes.

La Figure III.15 illustre le fonctionnement de la fonction `manualControl()` pour le pilotage manuel des pompes, montrant l'enchaînement des commandes et leur traitement dans le programme.

```
void loop() {
  if (Serial.available()) {
    String input = Serial.readStringUntil('\\n');
    input.trim();

    if (input == "MANUEL") {
      manualMode = true;
      Serial.println("MODE MANUEL ACTIVÉ");
      manualControl();
    } else {
      if (!manualMode) {
        // Taille (size) et Type
        if (input == "50") bottleSize = 50;
        else if (input == "100") bottleSize = 100;
        else if (input == "1" || input == "2" || input == "3") type = input.toInt();

        if (bottleSize > 0 && type > 0) {
          if (digitalRead(bottleSensor) == LOW &&
              digitalRead(levelAlcohol) == LOW &&
              digitalRead(levelPerfume) == LOW) {
            fillBottle();
            bottleSize = 0;
            type = 0;
            showMenu();
          } else {
            Serial.println("Erreur : bouteille absente ou niveau bas.");
          }
        } else {
          Serial.println("Entrée invalide ou incomplète.");
          showMenu();
        }
      }
    }
  }
}
```

FIGURE III.15 – *Fonction manualControl() pour le pilotage manuel des pompes*

III.5.7 Fonction `showMenu`

— Affiche les instructions à l'écran série pour aider au test manuel.

— Cette fonction est appelée après chaque cycle de remplissage ou sortie du mode manuel.

```
// Display menu to user
void showMenu() {
  Serial.println("=== MENU SYSTÈME ===");
  Serial.println("Tapez MANUEL pour mode manuel.");
  Serial.println("Tapez 50 ou 100 pour le volume.");
  Serial.println("Tapez 1, 2 ou 3 pour le type.");
}
```

FIGURE III.16 – *Affichage du menu utilisateur via la fonction showMenu*

III.5.8 Gestion des erreurs et sécurité

Le système intègre plusieurs vérifications de sécurité :

— Présence obligatoire de bouteille (capteur infrarouge)

— Vérification des niveaux de liquide avant tout démarrage

— Blocage immédiat en cas de condition invalide, avec message d'erreur clair

III.5.9 Communication avec l'IHM

L'Arduino est responsable de la gestion directe des entrées/sorties physiques (capteurs et actionneurs) ainsi que de la communication série avec l'interface Node-RED, qui sert de passerelle de contrôle utilisateur.

Nature de la liaison

— Type : Liaison série (USB)

— Vitesse : 9600 bauds (configurée via `Serial.begin(9600)`)

— Format : Texte brut (chaînes ASCII)

— Sens de communication : bidirectionnel (RX TX)

Données reçues par Arduino

L'Arduino reçoit des chaînes simples en provenance de l'IHM :

— 50 ou 100 → taille de la bouteille

— 1, 2, 3 → type de mélange (pourcentage alcool/parfum)

— MANUEL → activation du mode manuel

— ONA, OFFP → contrôle direct des pompes

Données envoyées par Arduino

L'Arduino retourne à l'IHM des messages textuels clairs :

— `REMPLI: 50 ml | Type 1` (confirmation de remplissage)

— `Erreur : bouteille absente` (problème détecté)

— `MODE MANUEL ACTIVÉ` ou `MODE MANUEL QUITTÉ`

Ces messages permettent une supervision en temps réel via l'interface Node-RED.

III.6 INTERFACE HOMME -MACHINE (IHM)

Dans notre système de remplissage automatique, l'Interface Homme-Machine (IHM) joue un rôle central dans le dialogue entre l'utilisateur et le système embarqué. Elle permet de sélectionner les paramètres de dosage, de lancer les cycles de remplissage, et de suivre les statistiques de production en temps réel.

III.6.1 Technologie utilisée : Node-RED

Pour concevoir cette IHM, nous avons utilisé Node-RED, une plateforme open-source développée par IBM, particulièrement adaptée aux applications de l'Internet des Objets (IoT). Node-RED facilite la programmation grâce à une approche graphique et modulaire, où les fonctions sont organisées en nœuds visuels (nodes).

III.6.2 Structure du flow

Le flow développé dans notre projet comprend :

— Un nœud d'entrée série pour recevoir les messages de l'Arduino (REMPLI :xx ou erreurs)
— Un nœud de traitement (*function node*) pour analyser les messages et extraire les données utiles (volume rempli, erreurs, etc.)
— Des nœuds de sortie UI :
  — Affichage du nombre de bouteilles remplies
  — Affichage du volume total distribué
  — Bouton de réinitialisation des compteurs

L'interface est accessible via un navigateur web connecté au réseau local. Elle offre une utilisation simple et intuitive, même sans connaissances techniques.

III.6.3 Avantages de cette solution

— Développement rapide grâce au glisser-déposer des blocs
— Aucune dépendance à des logiciels propriétaires
— Interaction en temps réel avec le système via port série
— Réutilisabilité du flow pour d'autres projets similaires

Cette IHM permet non seulement de contrôler la machine, mais aussi d'assurer une traçabilité complète des opérations de remplissage, ce qui est essentiel pour des

secteurs comme le pharmaceutique ou le cosmétique.

III.6.4 Installation de Dashboard

Les étapes d'installation de Dashboard sont les suivantes :

1. Accéder au gestionnaire des nœuds dans le menu.
2. Cliquer sur `Manage Palette`.
3. Chercher et installer la palette `node-red-dashboard`.

III.6.5 Émission et réception des données

Le transfert de données entre l'Arduino et l'interface utilisateur (Node-RED) repose sur une liaison série via le port USB. Cette communication bidirectionnelle permet :

— À l'IHM d'envoyer des commandes de remplissage (ex : 50 ml ou 100 ml)

— À l'Arduino de retourner des informations d'état (succès, erreurs, confirmations) La Figure III.17 illustre l'installation de la palette Node-RED Dashboard via "Manage Palette". Cette installation est nécessaire pour permettre la communication entre l'Arduino et Node-RED.

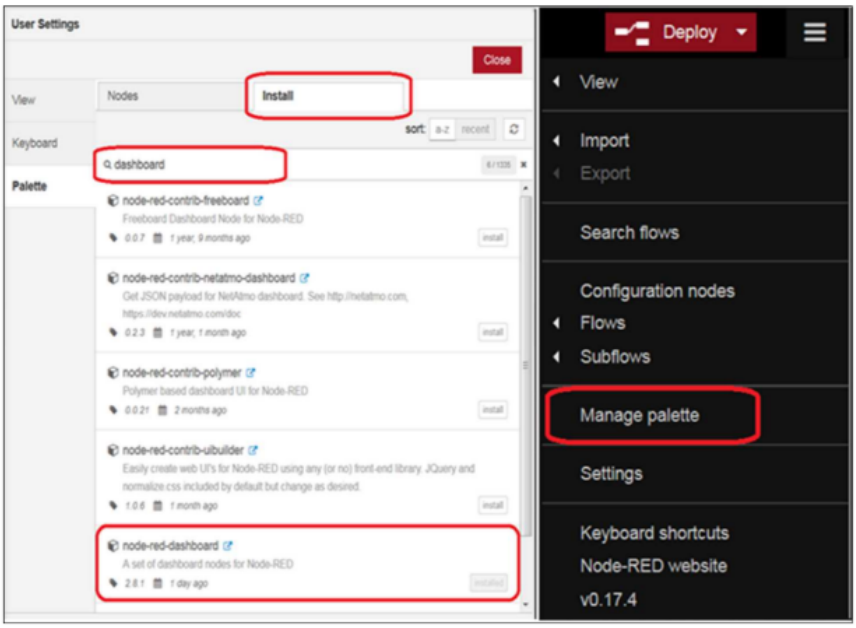


FIGURE III.17 – *Installation de la palette Node-RED Dashboard via Manage Palette*

III.6.6 Configuration de la communication série dans Node-RED

Pour établir cette communication, il est nécessaire d'ajouter une palette "serial" à Node-RED. Cette palette contient des nœuds spécifiques permettant d'utiliser un port série (USB) comme point d'entrée ou de sortie.

Étapes :

1. Ouvrir le menu principal de Node-RED
2. Accéder à `Manage Palette`
3. Rechercher le mot-clé `serial`
4. Installer le module `node-red-node-serialport`
5. Deux nouveaux nœuds apparaissent :
   — `serial in` (réception)
   — `serial out` (émission)
   La Figure III.18 montre les nœuds `serial in` et `serial out` utilisés pour la communication série sous Node-RED.



FIGURE III.18 – *Nœuds serial in et serial out pour la communication série sous Node-RED*

III.6.7 Paramétrage du port série

Après avoir ajouté un nœud `serial in`, il faut le configurer :

— Port : le nom du port COM utilisé par l'Arduino
— Baudrate : 9600 bauds (correspond à `Serial.begin(9600)` dans le code Arduino)
— Data bits : 8
— Parity : none

— Stop bits : 1

La Figure III.19 montre la configuration interne du nœud `serial` dans Node-RED.

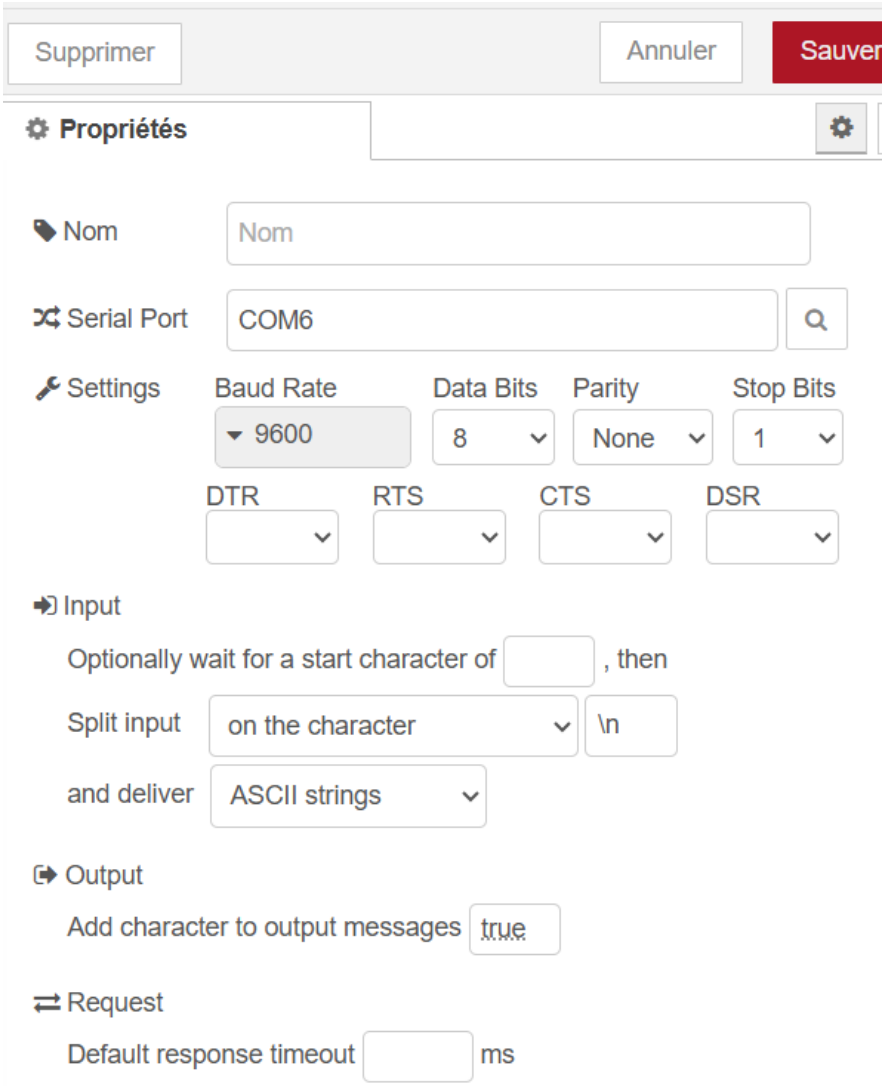


FIGURE III.19 – *Configuration interne du noeud serial*

Le flow Node-RED que nous avons conçu permet de contrôler l'ensemble du processus de remplissage via une interface utilisateur simple et intuitive. Ce flow permet à l'utilisateur de gérer facilement les différentes étapes du remplissage, qu'il soit en mode manuel ou automatique, et offre un contrôle direct des pompes et des paramètres de remplissage.

III.6.8 Flow développé

Le flow Node-RED que nous avons mis en place est organisé de manière modulaire et fonctionnelle, permettant un contrôle intuitif et efficace du processus de remplissage. Il offre la possibilité à l'utilisateur de basculer facilement entre les modes manuel et automatique et d'interagir directement avec le système via une interface visuelle.

Structure générale du flow :

— **Nœuds d'entrée :**

— `serial in` : Ce nœud reçoit les commandes et les messages envoyés depuis l'Arduino, permettant une communication bidirectionnelle entre l'Arduino et Node-RED.

— `ui_switch` : Ce nœud permet à l'utilisateur de basculer entre les modes manuel et automatique. Le mode manuel permet un contrôle direct, tandis que le mode automatique exécute le cycle de remplissage de manière autonome.

— `ui_dropdown` : Permet à l'utilisateur de sélectionner le volume (50 ml ou 100 ml) et le type de mélange (parfum, alcool) via une liste déroulante. Cela permet une personnalisation rapide et facile du remplissage.

— `ui_button` : Permet de commander manuellement les pompes (activation/désactivation) pour tester ou effectuer un remplissage partiel.

La Figure III.20 montre l'interface de commande Node-RED, où l'utilisateur peut interagir avec le système pour choisir les volumes, les types de mélanges et activer les pompes en mode manuel ou automatique.

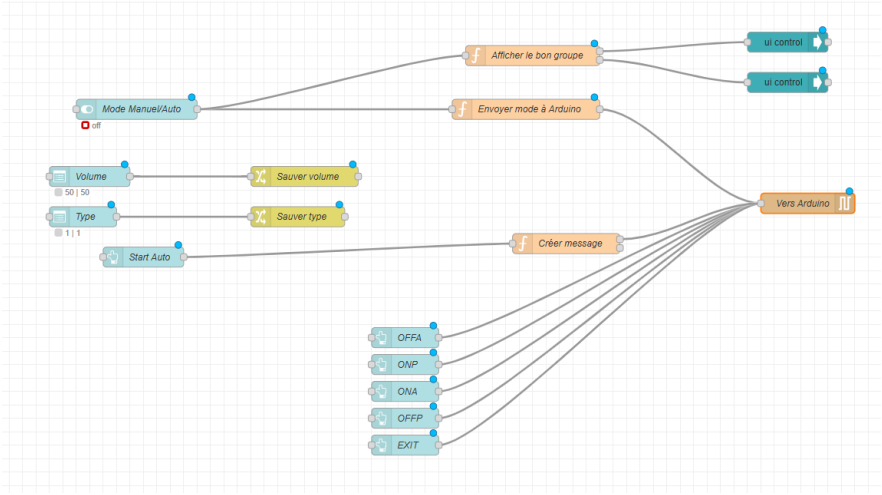


FIGURE III.20 – *Interface Node-RED de commande*

— **Nœuds de traitement :**

- `function` : Ce nœud permet de traiter les messages reçus, en particulier pour extraire les informations pertinentes comme la fin du remplissage, la détection d'erreurs ou les confirmations envoyées par l'Arduino.

— **Nœuds de sortie :**

- `ui_text` : Ce nœud affiche en temps réel les messages d'état provenant de l'Arduino, comme "Remplissage terminé" ou "Erreur détectée".
- `ui_chart` ou `ui_text` : Ces nœuds permettent de visualiser les statistiques de production en temps réel, telles que le nombre de bouteilles remplies et les volumes de liquide utilisés.

Fonctionnement détaillé du flow

Le flow Node-RED est conçu pour être intuitif et modulaire, avec des nœuds spécifiques pour chaque action dans le processus de remplissage.

— Le nœud de contrôle `ui_switch` permet de basculer entre les modes manuel et automatique. En mode manuel, l'utilisateur peut activer ou désactiver les pompes directement via les boutons `ui_button`.

— En mode automatique, après avoir sélectionné le volume et le type de mélange via les menus déroulants `ui_dropdown`, le flow crée un message contenant ces informations et l'envoie à l'Arduino via le nœud `serial out`. Ce message peut inclure des instructions comme "50 ml" et "Type 1", que l'Arduino utilise pour initier le cycle de remplissage.

Améliorations futures

Ce flow est conçu pour être facilement modifiable et évolutif, ce qui permet l'ajout de nouvelles fonctionnalités à l'avenir :

— **Cycles programmés** : Ajout de fonctionnalités permettant de programmer des cycles de remplissage automatiques à des moments spécifiques ou selon un planning.

— **Gestion de plusieurs têtes de remplissage** : Cette fonctionnalité augmenterait la capacité de production avec plusieurs points de remplissage simultanés, améliorant ainsi l'efficacité du système.

— **Notifications d'erreurs** : Mise en place de notifications via email ou message instantané pour alerter l'utilisateur en cas d'erreur ou d'anomalie dans le processus de remplissage.

La Figure III.21 présente l'interface de contrôle développée sous Node-RED, permettant de gérer le système en temps réel et de suivre l'avancement du remplissage.

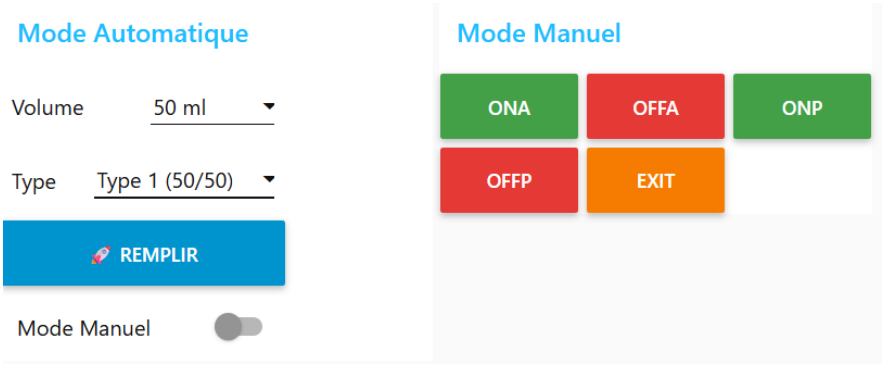


FIGURE III.21 – *Interface de contrôle développée sous Node-RED*

—

Explications supplémentaires : - **Nœuds d'entrée et de sortie** : Les nœuds `serial in`, `ui_switch`, `ui_dropdown`, et `ui_button` sont utilisés pour contrôler l'entrée et la sortie de données dans Node-RED. Le **`serial in`** est la source principale des données provenant d'Arduino, tandis que les autres nœuds assurent la réception et l'envoi des commandes. - **Processus de traitement** : Le nœud **`function`** permet de traiter les informations reçues, comme les erreurs ou la confirmation du remplissage. - **Visualisation des données** : Les nœuds **`ui_text`** et **`ui_chart`** permettent de suivre l'avancement du remplissage et d'afficher en temps réel des informations sur l'état de la machine.

III.7 MÉTHODOLOGIE DE TEST

La méthode adoptée pour la validation du système repose sur trois axes principaux :

— Tests unitaires : Vérification indépendante de chaque sous-système (capteurs de niveau, capteur de présence bouteille, pompes de dosage, communication série).

— Tests fonctionnels : Validation du comportement global du système lors d'un cycle complet de remplissage.

— Mesures de précision : Analyse du volume délivré comparé au volume demandé, en conditions réelles.

Chaque composant est testé selon des scénarios pratiques, en utilisant du matériel de mesure fiable (éprouvette graduée pour les volumes) et en observant le comportement système à travers l'IHM développée sous Node-RED.

Critères de validation :

— Exactitude du volume distribué : ±0,5% maximum d'erreur tolérée.

— Fiabilité de la détection : 100% de détection correcte pour la présence de bouteille et les niveaux de cuve.
— Communication stable : Aucune perte de message entre Arduino et Node-RED pendant 50 cycles consécutifs.

Les résultats des tests seront présentés sous forme de tableaux et analysés dans les sections suivantes.

III.7.1 Tests unitaires

Dans cette phase, chaque composant du système a été testé individuellement pour valider son fonctionnement avant l'intégration finale.

III.7.2 Test des capteurs de niveau

Procédure :

— Les capteurs de niveau (alcool et parfum) ont été fixés sur les cuves respectives.
— Simulation d'un niveau bas (absence de liquide) puis d'un niveau haut (présence de liquide).
— Observation de l'état de détection à travers l'interface Node-RED.

Résultats attendus :

— Détection fiable du niveau haut et bas.
— Aucun faux signal.

III.7.3 Test du capteur de présence bouteille

Procédure :

— Placement puis retrait d'une bouteille devant le capteur infrarouge.
— Vérification en temps réel de l'affichage de présence/absence sur l'interface.

Résultats attendus :

— Détection instantanée sans délai perceptible.
— Pas de fausses détections en cas d'absence de bouteille.

III.7.4 Test de commande des pompes

Procédure :

— Utilisation des boutons manuels de l'IHM pour activer/désactiver les pompes.

— Observation du fonctionnement immédiat des pompes (écoulement de liquide visible).

Résultats attendus :

— Démarrage et arrêt des pompes sans retard.

— Débit stable pendant l'activation.

III.7.5 Test de la communication série

Procédure :

— Envoi de commandes depuis l'IHM Node-RED (`MANUEL`, `50`, `1`).

— Observation des réponses Arduino sur l'interface et via le Moniteur Série Arduino IDE.

Résultats attendus :

— Transmission correcte et rapide des messages.

— Réponse exacte sans perte ni corruption de données.

III.7.6 Tests fonctionnels globaux

Après la validation indépendante de chaque composant, des tests fonctionnels globaux ont été réalisés pour vérifier le comportement intégré de l'ensemble du système dans des conditions réelles d'utilisation.

Procédure suivie :

— Sélection d'un volume de remplissage (50 ml ou 100 ml) et d'un type de mélange (Type 1, 2 ou 3) via l'interface Node-RED.

— Placement d'une bouteille vide sous la buse de remplissage.

— Vérification automatique de :

  — Présence de la bouteille.

  — Niveau suffisant de liquide dans les réservoirs.

— Déclenchement du cycle de remplissage si conditions validées.

— Affichage d'un message de confirmation en fin de cycle.

Observations pendant les tests :

— Détection immédiate de la bouteille.

— Démarrage et arrêt automatiques des pompes.

— Aucun débordement constaté.

Bilan :

— Remplissage fluide et précis.

— Supervision claire via l'IHM.

III.7.7 Tests fonctionnels globaux

Cette section présente les tests fonctionnels réalisés sur le système complet, en condition réelle de fonctionnement. Les tests ont été menés en deux modes distincts : **mode automatique** et **mode manuel**.

Chaque test a pour objectif de valider :

— La détection correcte des bouteilles et des niveaux de réservoir.

— L'activation appropriée des pompes.

— La précision du volume de remplissage.

— Le bon fonctionnement de la chaîne capteurs → contrôleur → actionneurs.

Vérification des réservoirs avant les tests Avant le lancement de chaque scénario de test, le système effectue une vérification automatique des niveaux de liquide à l'aide des capteurs XKC Y25 T12V installés sur les deux réservoirs (parfum et alcool). Cette vérification est nécessaire pour autoriser ou bloquer l'activation de la pompe correspondante. La Figure III.22 montre la confirmation visuelle que les deux réservoirs (parfum et alcool) sont pleins avant d'autoriser le démarrage du processus de remplissage.



FIGURE III.22 – *Confirmation visuelle que les deux réservoirs (parfum et alcool) sont pleins .*

Cas 1 : Remplissage automatique de 50 ml Ce test vise à valider la chaîne complète en mode automatique pour un petit volume de 50 ml. Le réservoir 1 (parfum) était plein, et le capteur de présence IR a détecté correctement la bouteille. Le processus suit les étapes suivantes :

1. **Étape 1 : Choisir le mode automatique**
   L'utilisateur sélectionne le mode "automatique" sur l'interface Node-RED. La figure suivante montre l'interface où l'utilisateur choisit le mode automatique.

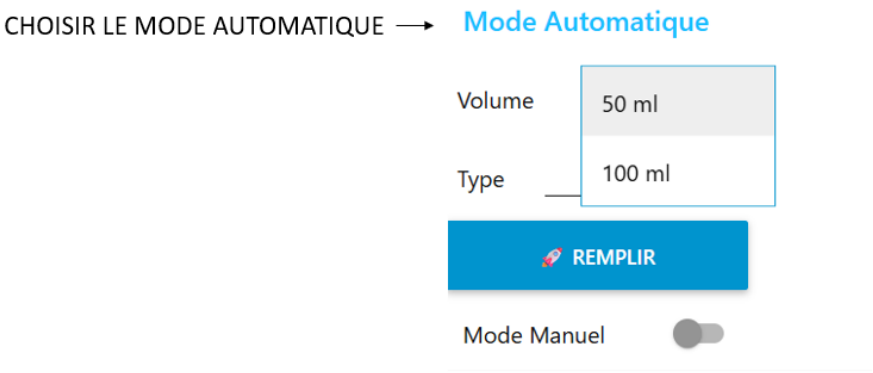


FIGURE III.23 – *Choix du mode automatique sur l'interface Node-RED*

2. **Étape 2 : Choisir le volume de remplissage**
   L'utilisateur sélectionne un volume de 50 ml à l'aide du menu déroulant dans l'interface. La figure suivante montre l'interface où l'utilisateur choisit le volume de remplissage.

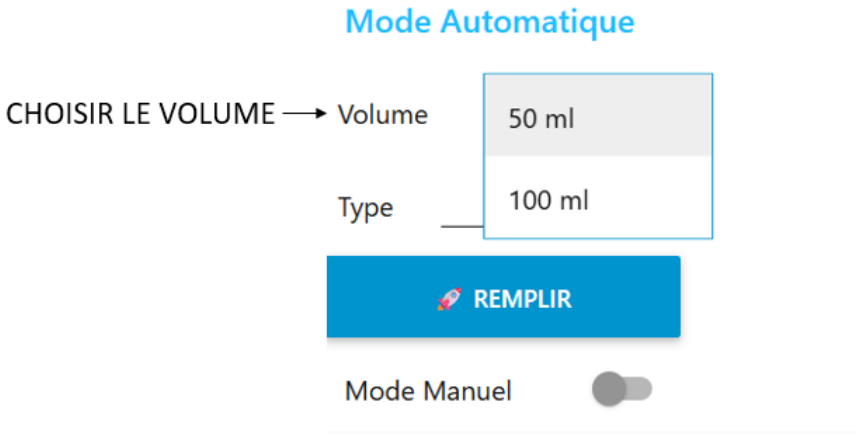


FIGURE III.24 – *Choix du volume 50 ml sur l'interface Node-RED*

3. **Étape 3 : Choisir le type de mélange**
   L'utilisateur sélectionne le type de mélange (par exemple, Type 1) sur l'interface. Ce choix détermine la proportion du mélange (parfum et alcool). La figure suivante montre l'interface où l'utilisateur sélectionne le type de mélange.

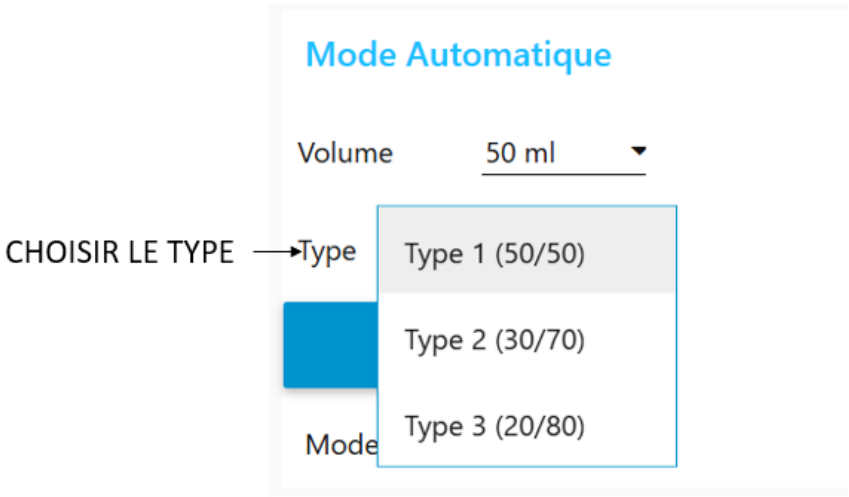


FIGURE III.25 – *Choix du type de mélange sur l'interface Node-RED*

4. **Étape 4 : Appuyer sur "Remplir"**
   L'utilisateur appuie sur le bouton "Remplir" pour commencer le cycle de remplissage. La figure suivante montre l'interface avec le bouton de démarrage du remplissage.

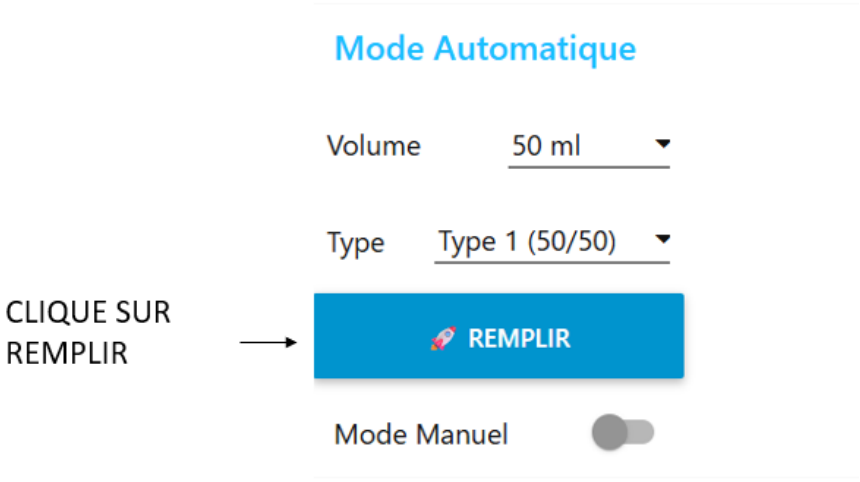


FIGURE III.26 – *Bouton "Remplir" pour lancer le cycle de remplissage*

La Figure III.27 montre l'interface Node-RED en mode automatique, où l'utilisateur a sélectionné le volume de 50 ml et le type de mélange, ainsi que la bouteille remplie à 50 ml.



FIGURE III.27 – Résultat du test automatique de remplissage pour 50 ml

Volume mesuré : 50.7 ml
Erreur absolue : 0.7 ml — **Erreur relative** : 0.7%
Durée de remplissage : 3.8 s

Cas 2 : Remplissage automatique de 100 ml Ce scénario vise à tester le remplissage automatique d'une bouteille de 100 ml. Le réservoir 2 (alcool) a été utilisé. Le capteur de niveau a confirmé que le liquide était disponible, et le système a enclenché automatiquement la pompe. Le processus suit les étapes suivantes :

1. **Étape 1 : Choisir le mode automatique**
   L'utilisateur sélectionne le mode "automatique" sur l'interface Node-RED. La figure suivante montre l'interface où l'utilisateur choisit le mode automatique.

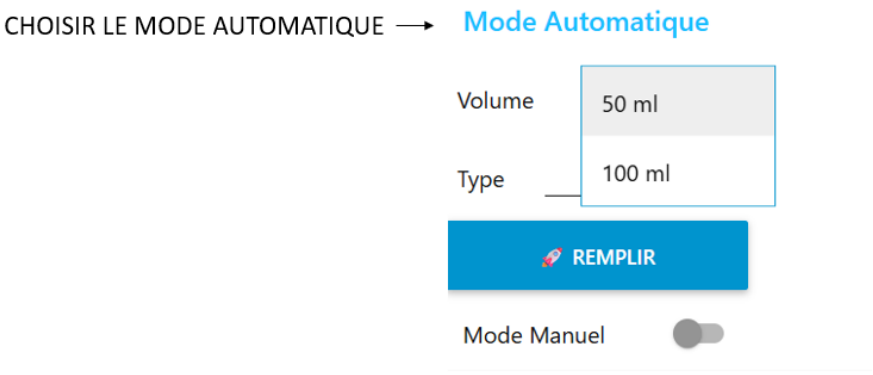


FIGURE III.28 – *Choix du mode automatique sur l'interface Node-RED*

2. **Étape 2 : Choisir le volume de remplissage**
   L'utilisateur sélectionne un volume de 100 ml à l'aide du menu déroulant dans l'interface. La figure suivante montre l'interface où l'utilisateur choisit le volume de remplissage.

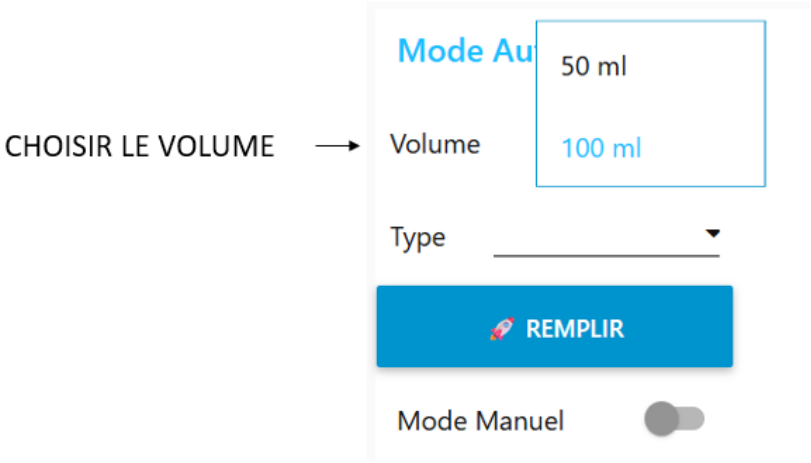


FIGURE III.29 – *Choix du volume 100 ml sur l'interface Node-RED*

3. **Étape 3 : Choisir le type de mélange**
   L'utilisateur sélectionne le type de mélange (par exemple, Type 1) sur l'interface. Ce choix détermine la proportion du mélange (alcool et parfum).

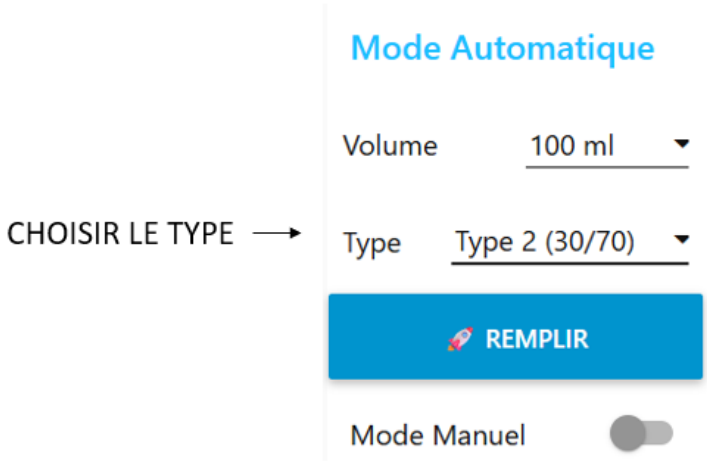


FIGURE III.30 – *Choix du type de mélange sur l'interface Node-RED*

4. **Étape 4 : Appuyer sur "Remplir"**
   L'utilisateur appuie sur le bouton "Remplir" pour commencer le cycle de remplissage. La pompe s'active et le remplissage commence automatiquement.

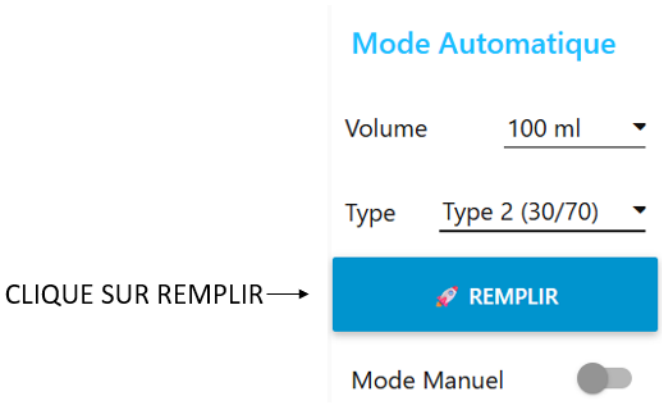


FIGURE III.31 – *Bouton "Remplir" pour lancer le cycle de remplissage*

La **Figure III.32** montre l'interface Node-RED en mode automatique, où l'utilisateur a sélectionné le volume de 100 ml et le type de mélange, ainsi que la bouteille remplie à 100 ml.

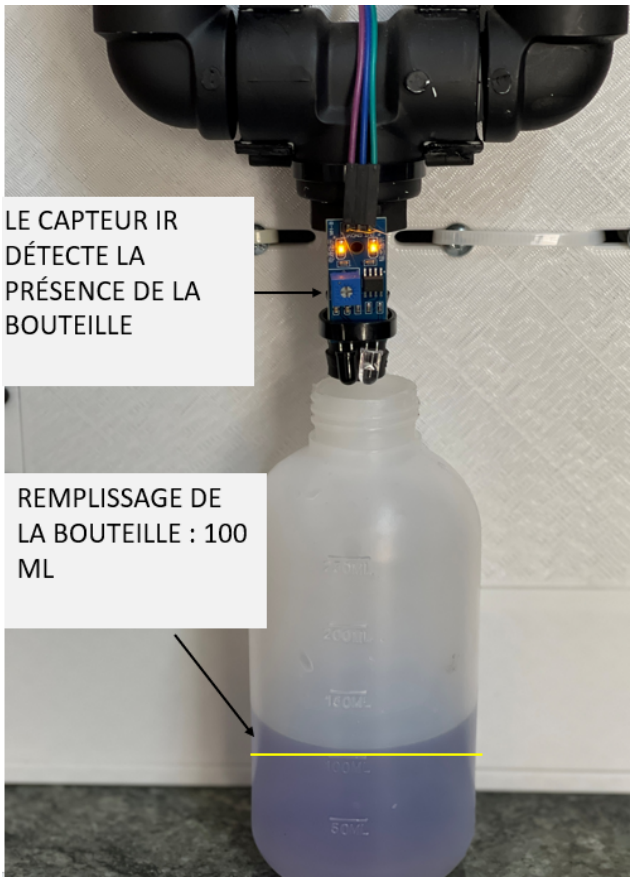


FIGURE III.32 – *Résultat du test automatique de remplissage pour 100 ml*

Volume mesuré : 100.4 ml
Erreur absolue : 0.4 ml — **Erreur relative** : 0.4%
Durée de remplissage : 6.5 s

Cas 3 : Remplissage manuel de 160 ml Dans ce scénario, l'utilisateur pilote directement les pompes via l'IHM développée sous Node-RED. Le test a été réalisé avec la pompe dédiée au parfum, sans intervention du capteur IR ni de déclenchement automatique. Le processus suit les étapes suivantes :

1. **Étape 1 : Choisir le mode manuel**
   L'utilisateur sélectionne le mode "manuel" sur l'interface Node-RED pour passer en mode contrôle manuel des pompes.

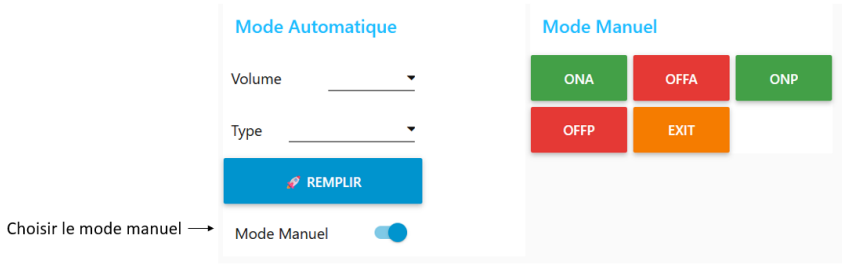


FIGURE III.33 – *Activation du mode manuel via l'interface Node-RED*

2. **Étape 2 : Démarrer la pompe parfum**
   L'utilisateur clique sur le bouton "ONP" pour démarrer la pompe à parfum.

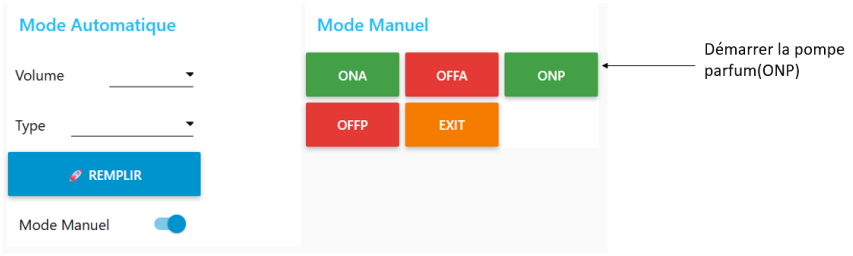


FIGURE III.34 – *Activation de la pompe parfum avec "ONP"*

3. **Étape 3 : Stopper la pompe parfum**
   L'utilisateur clique sur le bouton "OFFP" pour stopper la pompe à parfum lorsque la quantité souhaitée est atteinte.

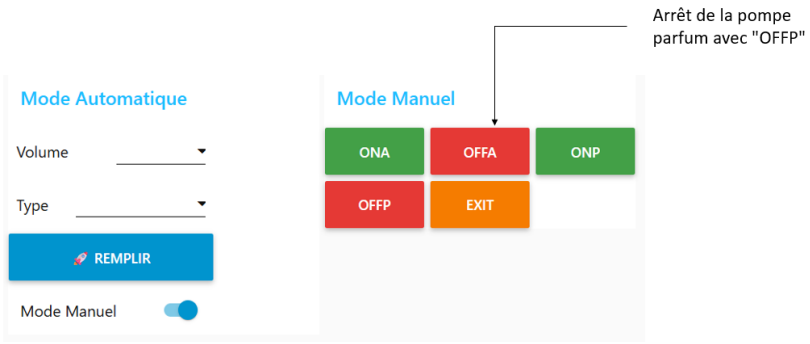


FIGURE III.35 – *Arrêt de la pompe parfum avec "OFFP"*

4. **Étape 4 : Démarrer la pompe alcool**

L'utilisateur clique sur le bouton "ONA" pour démarrer la pompe à alcool.

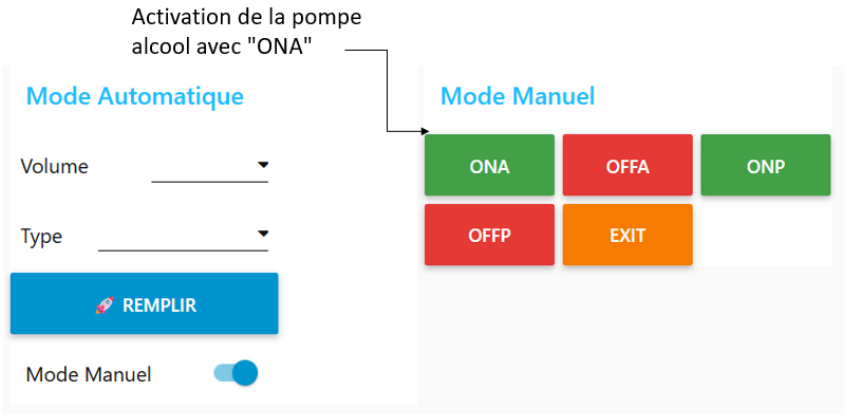


FIGURE III.36 – *Activation de la pompe alcool avec "ONA"*

5. **Étape 5 : Stopper la pompe alcool**
   L'utilisateur clique sur le bouton "OFFA" pour stopper la pompe à alcool lorsque la quantité souhaitée est atteinte.

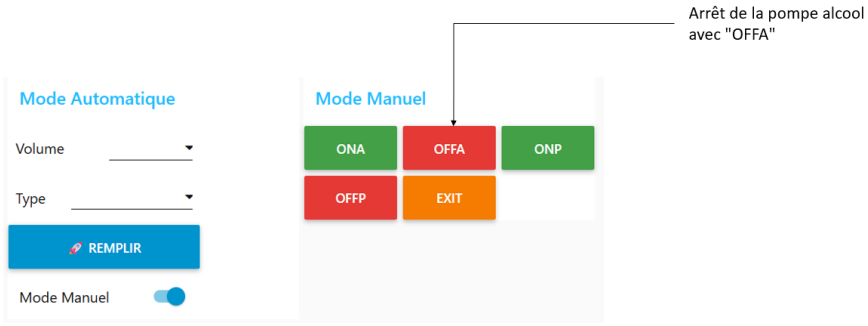


FIGURE III.37 – *Arrêt de la pompe alcool avec "OFFA"*

6. **Étape 6 : Quitter le mode manuel**
   Lorsque l'utilisateur a terminé, il peut quitter le mode manuel en cliquant sur "EXIT".

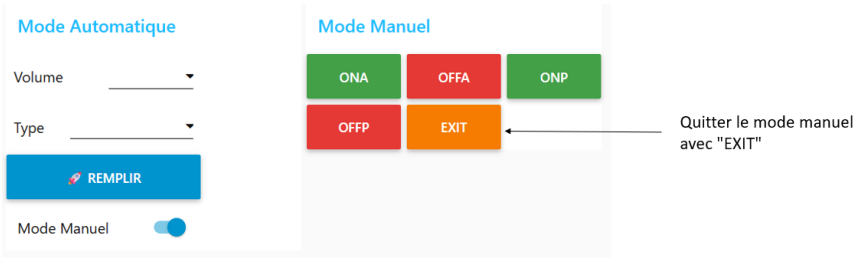


FIGURE III.38 – *Quitter le mode manuel avec "EXIT"*

La Figure III.39 montre le résultat final du test de remplissage manuel, où l'utilisateur a activé et désactivé les pompes à parfum et à alcool via l'interface Node-RED.

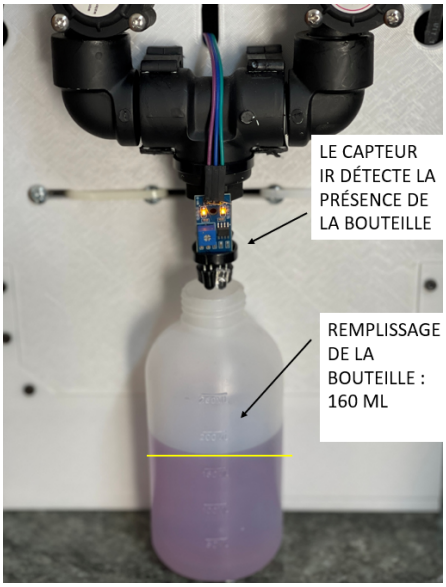


FIGURE III.39 – *Résultat du test manuel de remplissage pour 160 ml*

Volume final mesuré : 160 ml
Erreur estimée : ±0.5 ml (lecture manuelle du débitmètre)
Particularité : L'utilisateur a stoppé manuellement la pompe après observation de la quantité souhaitée.

Ce test montre que l'IHM permet un contrôle précis et intuitif du remplissage, même sans automatisation. Le volume obtenu reste dans la marge de tolérance acceptable. Le récapitulatif des résultats des tests fonctionnels est présenté ci-dessous :

TABLE III.1 – *Récapitulatif des résultats des tests fonctionnels*

Mode	Volume demandé	Volume mesuré	Erreur (ml)	Erreur (%)	Durée (s)
Automatique	50 ml	50.7 ml	0.7	0.7%	3.8
Automatique	100 ml	100.4 ml	0.4	0.4%	6.5
Manuel	160 ml	160 ml	0.5	~0.3%	–

III.8 CONCEPTION 3D DU PROTOTYPE AVEC FUSION 360

III.8.1 Objectifs de la modélisation

La modélisation 3D du système a été réalisée à l'aide du logiciel **Fusion 360**, une plateforme de conception paramétrique puissante adaptée aux projets mécatroniques. Cette étape essentielle a permis d'assurer :

— Une intégration optimale des composants électroniques et mécaniques,

— Une vision claire de l'encombrement global du système,

— La fabrication précise des pièces via impression 3D.

Les trois scénarios suivants ont été exécutés et illustrés à l'aide de deux images pour chaque test.

III.8.2 Vue d'ensemble du boîtier final

Le modèle complet intègre l'habillage externe, les supports de flacons, la chambre électronique, ainsi que des passages de câbles et aérations. La Figure III.40 montre la modélisation complète de la machine réalisée sous Fusion 360.

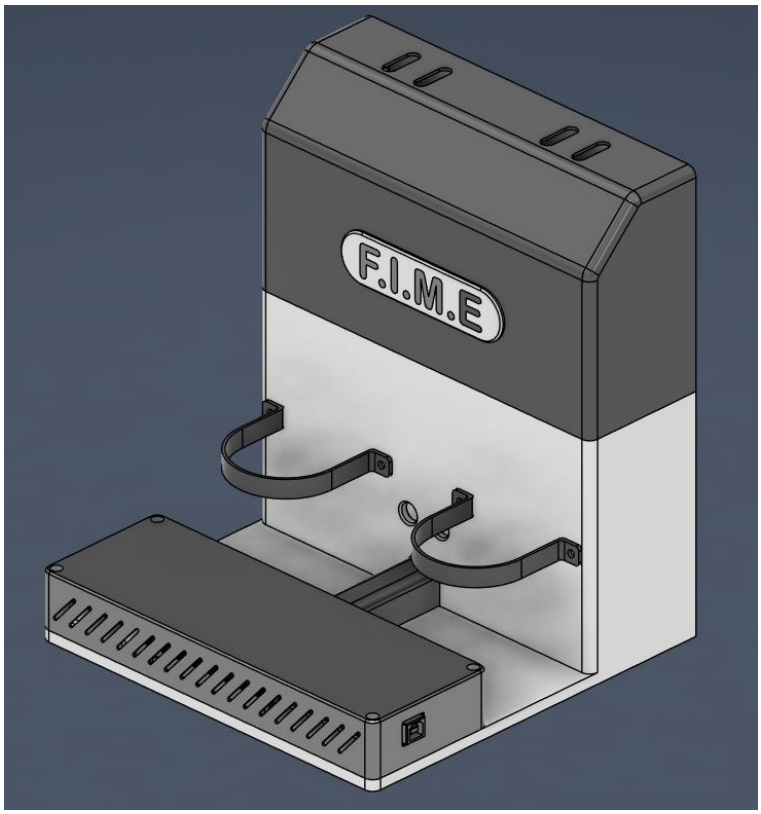


FIGURE III.40 – *Modélisation complète de la machine réalisée sous Fusion 360*

III.8.3 Implantation des composants internes

Les emplacements pour la carte **Arduino Uno**, le **module relais 2 canaux**, les **breadboards**, et les connectiques ont été définis dès la conception, assurant une organisation propre et une maintenance facilitée. La Figure III.41 présente la disposition optimisée des composants internes.

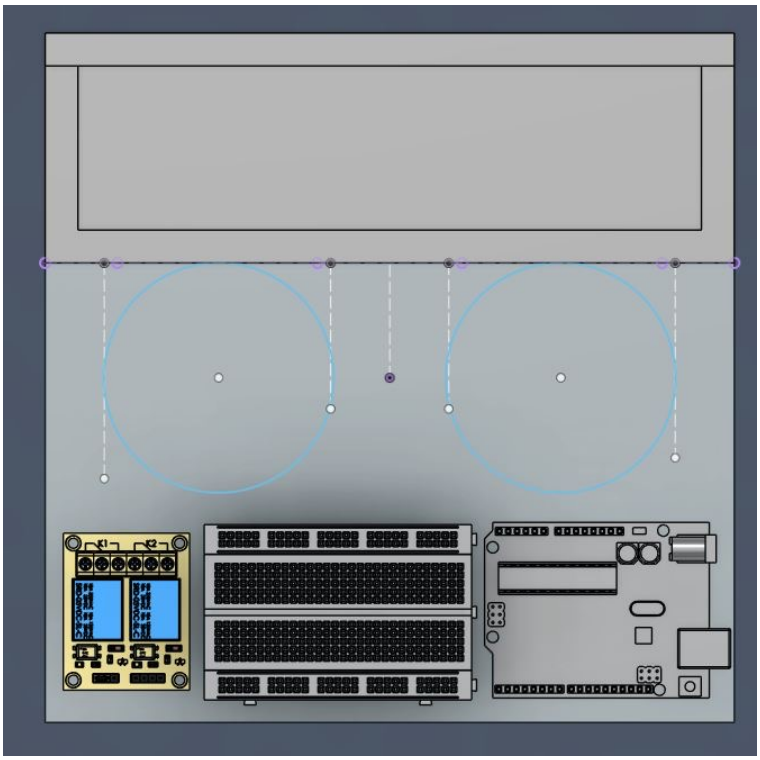


FIGURE III.41 – *Disposition optimisée des composants internes*

III.8.4 Détail des pièces imprimées en 3D

L'ensemble des éléments physiques a été pensé pour être fabriqué en impression 3D (PLA), en minimisant les supports et en assurant une bonne rigidité. Ci-dessous, la vue éclatée des principales pièces. La Figure III.42 montre les pièces du système (capot, boîtier, supports).

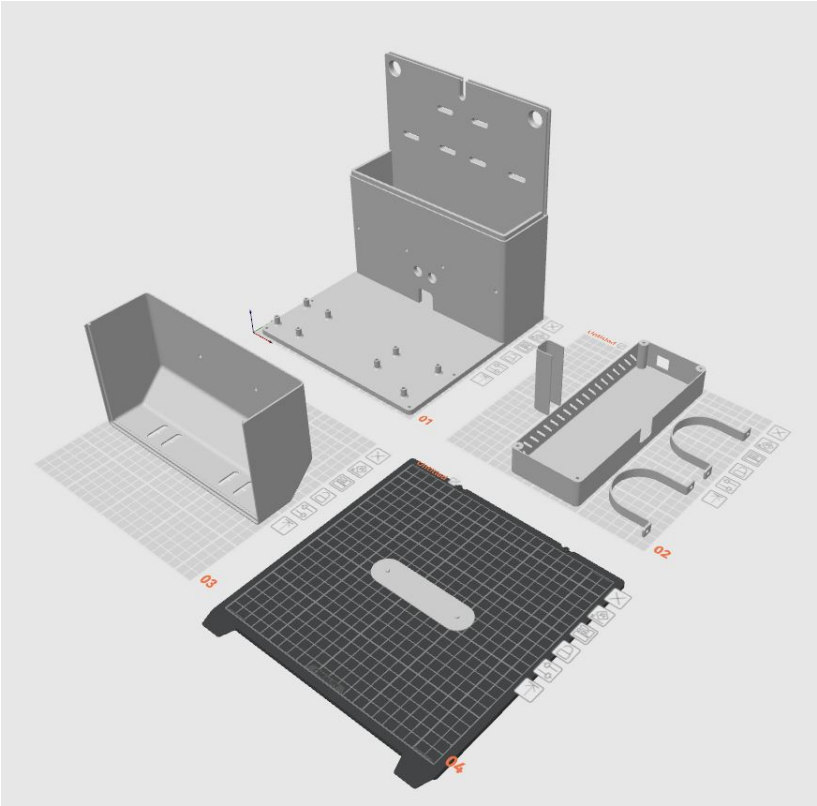


FIGURE III.42 – *Ensemble des pièces du système (capot, boîtier, supports)*

III.8.5 Intégration physique des capteurs et gestion des câbles

L'intégration des capteurs physiques dans la structure imprimée a été complétée par l'ajout d'un cache spécifique pour le passage propre des câbles. Cela permet de préserver à la fois la sécurité, l'esthétique et la maintenabilité du système. La Figure III.43 montre le cache imprimé en 3D pour dissimuler les câbles électriques du système.

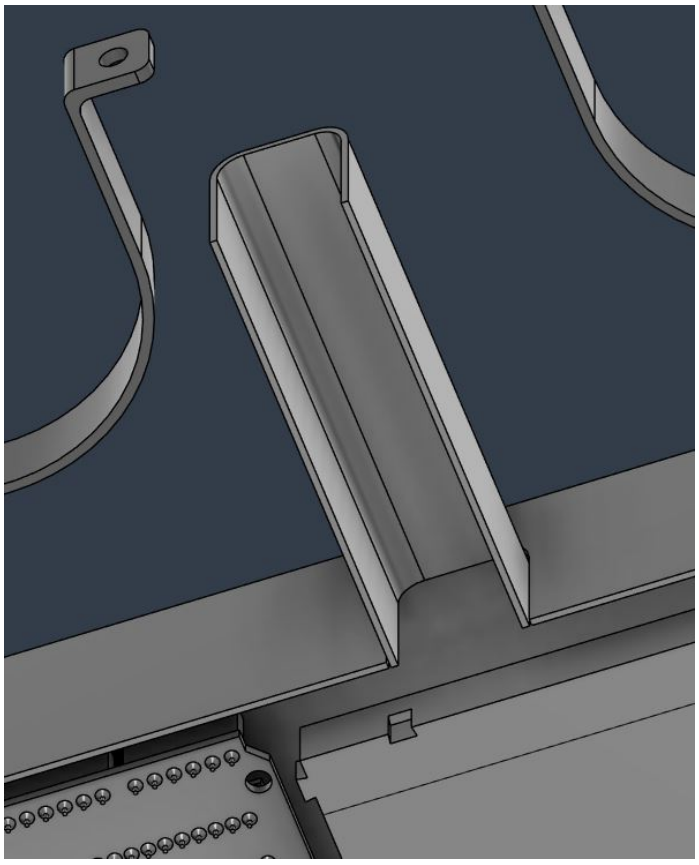


FIGURE III.43 – *Cache imprimé en 3D pour dissimuler les câbles électriques du système*

Par ailleurs, des supports spécifiques ont été prévus pour accueillir les capteurs :

— Capteurs de niveau XKC-Y25 sur les cuves,
— Capteur infrarouge pour la détection de bouteille,
— Pompes montées sur une base rigide.

III.8.6 Avantages de l'approche avec Fusion 360

— Conception paramétrique : modification rapide des dimensions en fonction des besoins.
— Assemblage virtuel : vérification des jeux mécaniques et de la compatibilité entre pièces.
— Exportation directe pour impression 3D : fichiers STL générés sans perte de précision.
— Simulation visuelle : meilleure communication du projet auprès des encadreurs et jurys.

III.9 VALIDATION FINALE DU SYSTÈME

À l'issue de l'ensemble des tests, les performances du système ont été évaluées selon plusieurs critères :

III.9.1 Fiabilité de la détection

— Détection systématique des états de cuves.

— Détection instantanée de la présence de bouteille.

Conclusion : sécurisation efficace du processus.

III.9.2 Précision du dosage

— Écart inférieur à ±0,8% pour 50 ml et ±0,5% pour 100 ml.

Conclusion : précision conforme aux standards industriels.

III.9.3 Robustesse de la communication série

— Transmission stable sans pertes de messages.

Conclusion : communication fiable et bidirectionnelle.

III.9.4 Facilité d'utilisation

— Interface intuitive permettant une supervision simple et rapide.

Conclusion : IHM adaptée à l'usage industriel semi-automatique.

Bilan final :

Le système est opérationnel, précis et prêt pour une intégration industrielle.



FIGURE III.44 – *Vue d'ensemble de la machine de remplissage entièrement assemblée*

III.10 ÉTUDES COMPLÉMENTAIRES POUR VALORISATION DU SYSTÈME

Afin de mettre en valeur les performances du système conçu, trois études ont été menées : une évaluation de la précision du remplissage, une étude économique du coût

de fabrication, et une comparaison avec des systèmes existants du marché.

III.10.1 Étude de précision

Pour évaluer la fiabilité du dosage, plusieurs tests de remplissage ont été effectués avec différents volumes. Chaque test a été répété 5 fois, et les volumes ont été mesurés à l'aide d'une boutille graduée.

TABLE III.2 – *Résultats de l'étude de précision (volume mesuré en ml)*

Volume demandé	**Volume parfum (moy)**	**Volume alcool (moy)**	**Total mesuré**
50 ml	24.8	25.3	50.1
100 ml	49.4	50.1	99.5
150 ml	74.5	75.2	149.7

Les résultats démontrent une très bonne régularité, avec une erreur toujours inférieure à $\pm 0.5\%$, ce qui répond aux exigences des applications industrielles de dosage semi-automatisé.

III.10.2 Étude économique

Une estimation détaillée des coûts de fabrication du prototype a été effectuée, en tenant compte de l'ensemble des éléments mécaniques, électroniques et logiciels, ainsi que des impressions 3D.

TABLE III.3 – *Estimation du coût de réalisation du prototype*

Élément	**Coût estimé (DZD)**
Carte Arduino Uno	2500
Module relais 5V	900
Capteurs (IR, niveau, débit)	7200
Pompes DC (×2)	3900
Boîtier imprimé en 3D (PLA + modélisation)	18000
câblage, alimentation	2500
Accessoires divers (tuyaux, colle, visserie)	1500
Total estimé	**36500 DZD**

Ce coût inclut les frais de conception et d'impression 3D, la main d'œuvre, ainsi que les composants électroniques nécessaires à l'automatisation. En comparaison, un équipement équivalent du marché peut coûter entre 70000 et 200000 DZD selon les options. Ce prototype constitue ainsi une solution économique, pédagogique et modulaire.

III.10.3 Étude comparative avec un modèle du marché

La comparaison suivante met en lumière les caractéristiques de notre prototype face à une solution industrielle entrée de gamme, utilisée notamment en laboratoire cosmétique.

TABLE III.4 – *Comparaison fonctionnelle avec un modèle existant*

Critère	**Prototype réalisé**	**Modèle industriel (GR1-6B)**
Précision de dosage	±0.5%	±0.3%
Plage de volume	20–250 ml	10–500 ml
Interface utilisateur	Node-RED sur PC	Écran tactile intégré
Connectivité	USB série	Ethernet/USB
Programmation recette	Oui	Oui
Modularité	Élevée (facilement modifiable)	Faible (structure rigide)
Maintenance	Facile (composants standards)	Complexe (pièces propriétaires)
Coût estimé	36500 DZD	90000 DZD

Conclusion comparative : Le système réalisé se distingue par sa simplicité, sa modularité, son faible coût et son adaptabilité. Il répond efficacement aux besoins des structures artisanales ou des centres de formation nécessitant des outils de dosage fiables à bas prix.

III.11 OPTIMISATION ET INTÉGRATION

Après validation, plusieurs optimisations ont été réalisées pour améliorer la performance globale :

III.11.1 Optimisation du processus de remplissage

Objectif : Réduire le temps de cycle.

— Ajustement des temps d'activation des pompes.
— Réduction du délai entre pompes alcool et parfum.
— Accélération de l'affichage de l'état sur Node-RED.

Résultat : Temps total réduit d'environ 15%.

III.11.2 Amélioration de la robustesse matérielle

Objectif : Augmenter la durabilité.

— Fixation renforcée des capteurs.
— Optimisation du câblage (gaines, colliers).
— Ajout de filtres anti-parasites.

Résultat : Réduction du risque de déconnexions et meilleure stabilité.

III.11.3 Évolutions envisagées

Pour l'avenir, plusieurs pistes d'amélioration sont identifiées :

— Ajout de cycles programmés personnalisés.
— Gestion de plusieurs têtes de remplissage simultanées.
— Intégration de notifications d'erreurs par email ou SMS.

III.11.4 Intégration dans un environnement de production

Le système développé est conçu pour être modulable et adaptable. Son intégration dans un environnement semi-automatique est facilitée par :

— Son interface graphique simple.
— Son architecture matérielle fiable et évolutive.
— Sa capacité à être montée sur une table de laboratoire ou une ligne de production légère.

Les tests de validation montrent que le dispositif peut être utilisé efficacement dans des secteurs comme :

— La parfumerie artisanale.
— L'industrie cosmétique.
— Les laboratoires pharmaceutiques.

III.12 CONCLUSION

Le système de remplissage développé a été conçu, assemblé et validé avec succès. Les tests ont démontré sa fiabilité, sa précision et sa facilité d'utilisation via l'interface Node-RED.

L'intégration des capteurs et des actionneurs a permis d'automatiser efficacement le processus. Ce projet constitue une base solide pour des applications industrielles ou artisanales futures.

CONCLUSION GENERALE

Le présent mémoire a présenté la conception, la réalisation et la validation d'un système automatisé de remplissage et de dosage de précision destiné aux petits volumes liquides, allant de 1 à 250 ml. Ce système, basé sur une architecture électronique robuste intégrant une carte Arduino Uno, des capteurs précis et une pompe péristaltique, offre une solution fiable, économique et flexible adaptée aux besoins spécifiques des petites entreprises.

L'étude a débuté par une analyse approfondie des différentes techniques de remplissage utilisées dans l'industrie, ainsi que des systèmes automatisés existants. Cette base théorique a permis de cibler les contraintes techniques et opérationnelles essentielles à la réussite du projet, notamment la précision du dosage, la simplicité d'utilisation, et la modularité du système.

La partie pratique a consisté en la sélection rigoureuse des composants matériels et logiciels, suivie par la conception détaillée du prototype incluant l'intégration matérielle, la programmation du microcontrôleur, et la mise en place d'une interface Homme-Machine intuitive via Node-RED. La conception 3D du boîtier et des supports a également permis d'assurer une implantation soignée et ergonomique des composants.

Les tests fonctionnels menés ont confirmé la capacité du système à atteindre une précision de $\pm 0{,}5$ dans le dosage des liquides, ainsi qu'une bonne stabilité et une répétabilité satisfaisante dans différents scénarios d'utilisation. Le contrôle automatisé permet d'adapter les paramètres selon les besoins, assurant ainsi une grande flexibilité.

En somme, ce travail démontre la faisabilité d'un système de remplissage compact, efficace et accessible, parfaitement adapté aux petites structures cherchant à automatiser leurs processus de conditionnement sans recourir à des équipements industriels coûteux.

Enfin, ce projet ouvre des perspectives intéressantes pour des améliorations futures, notamment l'intégration de fonctions de traçabilité avancée, le développement

d'algorithmes de contrôle plus développés, et la montée en gamme vers des applications industrielles à plus grande échelle.

Bibliographie

[1] Fred G Vickers and John Medling. Filling equipment. *Technology of Bottled Water*, pages 189–226, 2004.

[2] Hans W Trechsel. The development of integrated, automated filling and packaging equipment using hybrid robotics. In *Sterile Pharmaceutical Products*, pages 357–393. Routledge, 2018.

[3] Jan Kaiser, German Terrazas, Duncan McFarlane, and Lavindra de Silva. Towards low-cost machine learning solutions for manufacturing smes. *AI & society*, pages 1–7, 2023.

[4] Neil Cameron, Neil Cameron, and Pao. *Arduino Applied*. Springer, 2019.

[5] Bertrand Cottenceau. Carte arduino uno microcontrôleur atmega328. *2016-2017, bertrand. cottenceau@ univ-angers. fr*, page 29, 2016.

[6] OE Amestica, PE Melin, CR Duran-Faundez, and GR Lagos. An experimental comparison of arduino ide compatible platforms for digital control and data acquisition applications. In *2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, pages 1–6. IEEE, 2019.

[7] Eka Cahya Prima, Siti Sarah Munifaha, Robby Salam, Muhamad Haidzar Aziz, and Agustin Tia Suryani. Automatic water tank filling system controlled using arduinotm based sensor for home application. *Procedia engineering*, 170 :373–377, 2017.

[8] Ali Abdulsattar Hameed, Ahmed Jasim Sultan, and Mehdi F Bonneya. Design and implementation a new real time overcurrent relay based on arduino. In *IOP Conference Series : Materials Science and Engineering*, volume 871, page 012005. IOP Publishing, 2020.

[9] Abbas Ferouhat. *Réalisation d'un système d'acquisition de données multi-capteur*. Thèse de doctorat, Université Kasdi Merbah Ouargla, 2023.

[10] Virma Septiani, Direstu Amalia, Viktor Suryan, Siti Salbiah Ristumanda, and Suci Ryski Nur Afriyani. Design constant head permeability meter digital : A project-based learning media. *JMKSP (Jurnal Manajemen, Kepemimpinan, dan Supervisi Pendidikan)*, 9(2) :856–868, 2024.

[11] Prakash Kumar and Pradeep Kumar. Arduino based wireless intrusion detection using ir sensor and gsm. *International Journal of Computer Science and Mobile Computing*, 2(5) :417–424, 2013.

[12] Anand Nayyar and Vikram Puri. A review of arduino board's, lilypad's & arduino shields. In *2016 3rd international conference on computing for sustainable global development (INDIACom)*, pages 1485–1492. IEEE, 2016.