

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE

*Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj*

*Faculté des Sciences et de la technologie*

*Département d'Electronique*

# **Rapport**

**Projet de Fin de Cycle (PFC)**

**MCIL 3**

FILIERE : Electronique

Spécialité : Industries Electroniques

Par

- **Bouchtout Zineddine**
- **Nour Ahmed Wassim**

*Intitulé*

*Etude et réalisation d'un robot mobile collaboratif d'assistance à la  
manutention*

*Présenté le : 26/06/2022*

*Devant le Jury composé de :*

<i>Nom &amp; Prénom</i>	<i>Grade</i>	<i>Qualité</i>	<i>Etablissement</i>
<b>M. B. BOUKEZZATA</b>	<b>MCB</b>	<b>Président</b>	<b>Univ-BBA</b>
<b>M. A.H. LATOUI</b>	<b>MCA</b>	<b>Encadreur</b>	<b>Univ-BBA</b>
<b>M. F. KHERRAT</b>	<b>MAA</b>	<b>Examineur</b>	<b>Univ-BBA</b>

*Année Universitaire 2021/2022*

## **Remerciement :**

En premier lieu, je remercie notre seigneur dieu tout-puissant pour m'avoir guidé et donné de la patience, la volonté et la force pour pouvoir achever ce modeste travail.

J'adresse mes sincères remerciements à notre encadreur Mr LATOUI Abdelhakim pour son encadrement de qualité, ces conseils précieux, son encouragement ainsi que pour sa disponibilité et sa vaillance sur l'avancement du travail.

Je remercie le président du jury ainsi que tous les membres d'avoir accepté et pris de leur temps pour lire et juger ce travail.

Je tiens aussi à remercier tout le personnel du département d'électronique commençant par le chef de département allant jusqu'aux enseignants qui ont été à la hauteur pour leurs efforts durant toutes ces dernières années. N'oubliant pas aussi le personnel chargé des démarches administratives au sein du département ainsi que le personnel du laboratoire CAO qui ont mis à notre disposition tout le matériel nécessaire pour la réalisation de travail, que je tiens aussi à les remercier énormément.

Je tiens à remercier aussi chaleureusement du fond du cœur ma famille qui a été toujours à mes côtés en me soutenant que ça soit moralement ou financièrement durant toutes ces années et qui m'a soutenu à chaque moment difficile.

Enfin, je remercie toutes personnes que je n'ai pas cité mais que je n'ai pas oublié pour tout moindres effort ou conseil qui m'ont donné de près comme de loin. A tous, un grand MERCI.

## **Dédicaces :**

Je profite l'occasion pour dédier ce modeste travail pour tous ceux qui me sont chers.

A mes chers parents (que dieux les bénisse) qui étaient et qui sont toujours à mes côtés veillant sur moi et sur mon bien être et qui ne ménagent aucun effort pour ma réussite.

A ma chère sœur ainsi que mon cher frère, qui étaient toujours à cote de moi dans tous mes besoins.

A toute ma famille de près ou de loin et surtout je précise mon cher oncle Cherif pour ses conseils précieux.

A tous mes amis de loin ou de près, mes amis de cartier, mes amis de l'université ainsi qu'à toutes personnes qui m'a aidé à réaliser ce modeste travail.

## **Résumé :**

Un robot mobile est système mécanique, électronique et informatique agissant physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné. Dans ce contexte s'inscrit notre travail qui concerne l'étude et la réalisation d'un robot mobile collaboratif d'assistance à la manutention. Dans notre travail, nous présentons un robot mobile capable de s'adapter avec son environnement en évitant les différents obstacles présents tout au long de sa trajectoire au moyen d'un capteur ultrasonique dont il sera équipé avec. Ce qui concerne la commande de ce robot, elle sera devisée en deux parties, la première consiste à programmer une carte de commande à base d'une carte Arduino UNO, cette dernière permettra de générer les codes nécessaires qui seront transmis à un pont en H qui sera la partie commandante le fonctionnement des moteurs afin de permettre au robot de se déplacer dans toutes les directions selon la situation. La deuxième partie consiste à réaliser une application mobile constituée de plusieurs boutons permettant aussi de commander le fonctionnement des moteurs dans la direction désirée, cette application mettra l'utilisateur en contact avec le robot grâce au module BLUETOOTH dont la carte de commande sera équipée avec.

## **ملخص**

الروبوت المتنقل هو نظام ميكانيكي إلكتروني يعمل في البيئة التي يتواجد فيها و ذلك بغرض القيام بالمهمة المخولة له. يتعلق بدراسة و انجاز روبوت متنقل تعاوني للمساعدة في التعامل مع في هذا السياق يتجلى العمل المنجز و الذي الاجهزة المحمولة. في سياق هذا المشروع سنقدم روبوتا متنقلا قادرا على التأقلم مع البيئة المتواجده فيها و ذلك من خلال تفادي مختلف العقبات المتواجده على طول المسار و ذلك بفضل مستشعر الموجات ما فوق الصوتية الذي يكون الروبوت مجهزا به. فيما يتعلق بالتحكم بالروبوت التحكم سيكون بطريقتين مختلفتين الاولى عن طريق تصميم وحدة تحكم تكون مبنية على اساس بطاقة اردوينو اونو مبرمجة هذه الاخيرة ستسمح باصدار الاوامر اللازمة و التي تحول بدورها الى وحدة التحكم بالمحركات الجزء المتحكم بتشغيل المحركات من أجل السماح للروبوت بالتحرك في جميع الاتجاهات حسب الحالة المتواجده بها. الطريقة الثانية تتضمن استحداث تطبيق للهاتف الذكي مكون من عدة ازرار يمكن الضغط عليها للتحكم بكيفية تشغيل المحركات و ذلك لضمان السير في الاتجاه المرغوب فيه التطبيق يسمح بخلق فرصة تواصل بين الروبوت و المستخدم و هذا عن طريق استخدام وحدة بلوتوث و التي ستكون مدمجة ضمن وحدة التحكم المصممة.

## **Abstract :**

A mobile robot is a mechanical, electronic and computer system acting physically its environment in order to achieve an assigned objective. In this context, our work on the study and implementation of a collaborative mobile robot for handling assistance. In our work, we present a mobile robot able to adapt to its environment by avoiding the various obstacles present throughout its trajectory by means of an ultrasonic sensor which it will be equipped with. Regarding the control of this robot, it will be devised in two parts, the first designer to program a control board based on an arduino UNO, the latter will generate the necessary codes that will be transmitted to an H bridge which will be the part controlling the operation of the engines in order to allow the robot for moving in all directions depending on the situation. The second part is to create a mobile application consisting of several buttons allowing also to order the operation of the motors in the desired direction, this application will put the user in contact with the robot thanks to the BLUETOOTH module, which the control board will be equipped with.

## Table des matières

Introduction générale .....	11
1. Introduction générale :.....	1
Chapitre 1 : Généralités sur les robots mobiles.....	3
1.1 Introduction : .....	4
1.2 Aperçu historique :.....	4
1.3 Définition de la robotique :.....	6
1.3.1 Etymologie du mot robotique :.....	7
1.4 Définition du robot :.....	7
1.4.1 Etymologie du mot « Robot » : .....	7
1.4.2 Robot/automate.... Confusion ?.....	8
1.4.3 Robot mobile :.....	8
1.5 Classification de robots mobiles : .....	8
1.5.1 Classification selon le degré d'autonomie :.....	8
1.5.1.1 Robot télécommandé :.....	8
1.5.1.2 Robot semi-autonome :.....	9
1.5.1.3 Robot autonome : .....	9
1.5.2 Classification des robots mobiles selon le type de locomotion :.....	9
1.5.2.1 Les robots à roues :.....	9
1.5.2.1.1 Robots unicycle :.....	9
1.5.2.1.2 Robot tricycle : .....	9
1.5.2.1.3 Robot voiture :.....	10
1.5.2.1.4 Robot omnidirectionnel :.....	10
1.5.2.2 Les robots à chenilles :.....	11
1.5.2.3 Les robots mobile à pattes ou les robots mobiles marcheurs :.....	12
1.5.2.4 Robots présentant un autre type de locomotion :.....	13
1.6 Architecture des robots mobile :.....	14
1.6.1 La structure mécanique et la motricité :.....	14
1.6.2 Les organes de sécurité :.....	14
1.6.2.1 Les capteurs proprioceptifs :.....	15
1.6.2.2 Les capteurs extéroceptifs : .....	15
1.6.3 Traitement des informations et gestion des taches :.....	15
1.6.4 Le système de localisation : .....	15
1.7 Les domaines d'application des robots mobiles :.....	15
1.7.1 L'industrie : .....	15

1.7.2	Le domaine militaire :.....	16
1.7.3	La santé :.....	16
1.7.4	Usage civile :.....	17
1.7.5	Usage domestique :.....	17
1.8	Les avantages et les inconvénients des robots :.....	19
1.7.5	Les avantages :.....	19
1.7.6	Les inconvénients :.....	19
1.8	Les problématique de la robotique :.....	19
1.9	Conclusion :.....	20
	Chapitre 2 : Conception et réalisation d'une carte de commande d'un robot mobile. ....	21
2.1	Introduction :.....	22
2.1	Commande d'un robot :.....	22
2.2	historique du projet Arduino :.....	22
2.3	Présentation de la carte Arduino :.....	23
2.4	Les différents types de cartes Arduino :.....	23
2.4.1	Arduino UNO :.....	23
2.4.2	Arduino NANO :.....	24
2.4.3	Arduino Due :.....	24
2.4.4	Arduino Mega 2560 :.....	25
2.4.5	Arduino Leonardo :.....	25
2.5	Les différentes applications à base d'Arduino :.....	26
2.6	Carte Arduino UNO :.....	26
2.6.1	Présentation :.....	26
2.6.2	Architecture :.....	27
2.6.2.1	Le microcontrôleur ATmega 328 :.....	31
2.6.2.2	Les mémoire :.....	31
2.6.2.2.1	Les mémoires FLASH :.....	31
2.6.2.2.2	Les mémoires SRAM : (Static-Random-Access-Memory) .....	31
2.6.2.2.3	Les méméoirs de types EEPROM : (Electrical Erasable Programmable Read Only Memory) 31	31
2.6.3	Alimentation de la carte Arduino UNO :.....	32
2.6.4	Les broches (entrées/sorties) numériques de la carte Ardubio UNO :.....	32
2.7	Partie programme :.....	33
2.7.1	Description de l'environnement Arduino IDE :.....	33
2.7.2	Interface du logiciel :.....	34
2.7.3	Structure d'un programme écrit avec le logiciel Arduino IDE :.....	35

2.8	Module BLUETOOTH HC-06 : .....	37
2.8.1	Qu'est ce que BLUETOOTH : .....	37
2.8.2	Présentation du module BLUETOOTH HC-06 : .....	37
2.8.3	Brochage du module BLUETOOTH HC-06 : .....	37
2.9	Moteur à courant continu CC : .....	38
2.9.1	Présentation : .....	38
2.9.2	Principe de fonctionnement d'un moteur à courant continu : .....	39
2.10	Le driver moteur L298N : .....	39
2.10.1	Présentation du driver : .....	39
2.10.2	Brochage du circuit intégré L298N : .....	41
2.10.3	Brochage du driver L298N : .....	42
2.11	Le pont en H : .....	44
2.11.1	Présentation d'un pont en H : .....	44
2.12	Capteur ultrasonique : .....	45
2.12.1	Principe de fonctionnement : .....	45
2.12.2	Brochage du capteur ultrasonique : .....	46
2.12.3	Carctéristique du capteur ultrasonique : .....	47
2.13	Conclusion : .....	47
	Chapitre 3 : Mise à l'eassaie du robot mobile réalisé. ....	48
3.1	Introduction : .....	49
3.1	Fonctionnemnt général du robot : .....	49
3.1.1	Organigramme du fonctionnement du robot : .....	50
3.2	Aperçu historique sur Android : .....	52
3.2.1	Présentation d'Android : .....	53
3.3	MIT app inventor : .....	53
3.3.1	Création d'une application mobile en utilisant le logiciel MIT App Inventor : .....	54
3.3.2	Présentation de la partie design : .....	59
3.3.3	Présentation de la partie blocs : .....	60
3.3.4	Liste des éléments utilisés lors de la réalisation de l'application : .....	60
3.3.5	Organigramme de l'application : .....	62
3.4	Conclusion : .....	64
	Conclusion générale.....	65

## Liste des figures :

Figure 1 : la roue, premier mécanisme de la mobilité connu par l'etre humain. ....	4
Figure 2 : le canard de Jacques De Vaucanson .....	5
Figure 3 : automate turc de Von Kemplen      Figure 4 : automate Euphonia d'Eugène Faber.....	5
Figure 5 : images de quelques domaines d'application de la robotique.....	7
Figure 6 : robot unicycle.....	9
Figure 7 : robot tricycle .....	10
Figure 8 : robot voiture .....	10
Figure 9 : robot omnidirectionnel.....	11
Figure 10 : robot à chenilles .....	12
Figure 11 : robot marcheur.....	13
Figure 12 : robots présentant un autre type de locomotion.....	14
Figure 13 : robot utilisé dans le domaine industriel .....	16
Figure 14 : robot utilisé dans le domaine militaire .....	16
Figure 15 : robot utilisé dans le domaine médicale .....	17
Figure 16 : robot mobile utilisé dans le domaine civil.....	17
Figure 17 : robot utilisé dans le domaine domestique.....	18
Figure 18 : Arduino UNO .....	23
Figure 19 : Arduino NANO .....	24
Figure 20 : Arduino DUE .....	25
Figure 21 : Arduino MEGA 2560.....	25
Figure 22 : Arduino LEONARDO .....	26
Figure 23 : Arduino UNO .....	27
Figure 24 : les différentes parties de la carte Arduino UNO.....	29
Figure 25 : microcontrôleur ATmega 328.....	31
Figure 26 : interface du logiciel Arduino IDE .....	34
Figure 27 : la structure d'un programme écrit sur Arduino IDE.....	36
Figure 28 : brochage du module BLUETOOTH HC-06.....	38
Figure 29 : moteur à courant continu .....	38
Figure 30 : principe de fonctionnement d'un moteur à courant continu .....	39
Figure 31 : signal PWM.....	40
Figure 32 : circuit intégré L298N.....	41
Figure 33 : brochage du circuit intégré L298N.....	41
Figure 34 : brochage du driver moteur L298N.....	42
Figure 35 : architecture interne du circuit intégré L298N.....	44

Figure 36 : circuit électrique d'un pont en H .....	45
Figure 37 : capteur ultrasonique.....	45
Figure 38 : principe de fonctionnement d'un capteur ultrasonique.....	46
Figure 39 : brochage du capteur ultrasonique .....	46
Figure 40 :images réelles du robot réalisé .....	50
Figure 41 : logo d'Android .....	53
Figure 42 : interface du site web de MIT App Inventor.....	54
Figure 43 : création d'une application avec MIT App Inventor.....	55
Figure 44 : ouvrir un compte au site MIT App Inventor .....	56
Figure 45 : commencer un nouveau projet .....	56
Figure 46 :donner un nom au projet créé .....	57
Figure 47 : iteraface de travail.....	57
Figure 48 : partie design .....	58
Figure 49 : partie blocs .....	58
Figure 50 : contenu de la partie design.....	59
Figure 51 : contenu de la partie blocs .....	60
Figure 52 : lien et code QR générés pour télécharger l'aaplication .....	62
Figure 53 : application mobile créée.....	62

## Liste des tableaux :

Tableau 1 : types des robots mobile à roues.....	11
Tableau 2 : domaines d'application de la robotique.....	18
Tableau 3 : caractéristique de la carte Arduino UNO.....	28
Tableau 4 : les broches de l'Arduino UNO.....	30
Tableau 5 : les différentes zones du logiciel Arduino IDE.....	35
Tableau 6 : brochage du module BLUETOOTH HC-06.....	37
Tableau 7 : la configuration des broches du driver L298N.....	40
Tableau 9 : brochage du driver moteur L298N.....	44
Tableau 10 : brochage du capteur ultrasonique.....	46
Tableau 11 : caractéristiques du capteur ultrasonique.....	47
Tableau 12 : les différentes version d'Android.....	52
Tableau 13 : liste des éléments utilisés pour la conception de l'application mobile.....	61

## **Mots clés :**

- Robot mobile.
- Autonome.
- Android.
- Arduino.
- Détection d'obstacles.

# **Introduction** **général**



## 1. Introduction générale :

Depuis l'existence, l'homme a toujours essayé de faire tourner le monde en sa faveur, tout en utilisant ce que lui offre ce dernier pour découvrir, inventer et fabriquer des outils et des machines qui lui faciliteront les différentes tâches de son quotidien.

Avec le temps et la succession des générations ainsi que les découvertes qu'apportent ces dernières, l'humanité a pu affranchir de nouveaux horizons technologiques, et c'est ainsi que la robotique c'est fut présenté à l'humanité.

D'une manière brève, la robotique facilite la vie des gens en leurs rendant les tâches complexes, répétitives ou encore pénible plus paisible et cela en s'introduisant dans la quasi-totalité des secteurs occupés par l'être humain : les différentes chaînes de production dans l'industrie (automobile, alimentaire...), l'agriculture, la médecine, la sécurité et l'assistance à domicile, dans les domaines de précision tel que l'industrie des composants électronique, les milieux inaccessibles ou qui représentent un danger pour l'être humain (les espaces de haute pression 'les espaces sous-marins' ou de basse pression 'les espaces spatiaux' ainsi que les espaces de haute températures, de fortes radiations...

La robotique regroupe deux grandes branches : la robotique de manipulation et la robotique mobile où se présente l'un des problèmes majeurs de la robotique qui est la planification de mouvement. Ainsi de ce problème, notre travail fait le point de départ qui a pour objectif principal de réaliser un robot mobile collaboratif à base d'une carte Arduino UNO dédié aux manipulations d'assistance à la manutention. Il s'agira de la programmation d'un robot autonome qui pourra se mouvoir librement et réagir aux différentes situations présentes dans son environnement tout en évitant les obstacles présents sur sa trajectoire grâce à des capteurs, de plus il sera commandé à distance par un smartphone à l'aide d'une communication Bluetooth. Accomplir ce travail et révéler tous les défis n'est pas une tâche facile, pour ce faire il faut tout d'abord d'élaborer une carte de commande qui sera le cerveau du robot, elle comprendra une carte Arduino UNO et un pont en H afin d'avoir le contrôle et de pouvoir alimenter les deux moteurs à courant continu qui commanderont les quatre roues qui donneront à leurs tour une mobilité au robot ( tourner à gauche et à droite, avancer tout droit et faire des demi-tours) elle sera aussi dotée d'un capteur ultrasonique qui sera de son tour équipé sur un servomoteur afin d'augmenter son champ de détection, ce capteur accomplira la tâche difficile qui est la détection d'obstacles ce qui rendra le mouvement du robot plus optimisé. Et enfin cette carte de commande portera aussi un module Bluetooth pour assurer une communication sans fil considéré comme deuxième méthode de commande du robot et cela par la création d'une application mobile Android (marche et arrêt, marche avant et arrière, tourner à gauche ou à droite).

Sur le plan formel, ce mémoire contiendra trois chapitres qui se présentent de la façon suivante :

En gros, le premier chapitre est un chapitre introductif à la robotique, il présentera les généralités de la robotique ainsi que les robots mobiles passant par leur histoire d'évolution,

---

les classes et types de robots mobile ainsi que leurs domaines d'applications tout en commençant par les notions de base des robots mobile ce qui nous emmènera vers le deuxième chapitre.

Ce deuxième chapitre sera consacré à la présentation et la description détaillé de la carte de commande utilisé pour le contrôle du robot et du matériel et les composants utilisés pour la conception et la réalisation de cette carte de commande ainsi que les logiciels utilisés.

Pour le troisième chapitre qui sera réservé exclusivement pour la description de la réalisation pratique et la mise en essai du robot mobile réalisé.



**Chapitre 1 :**  
**Généralités sur les**  
**robots mobiles.**

---

## 1.1 Introduction :

D'une manière générale, un robot est une créature artificielle, c'est un système mécatronique (alliant la mécanique, l'électronique, l'informatique ainsi que les nouvelles technologies de l'information et de la communication) paramétré et programmé à base des algorithmes ce qui lui donnera de la capacité de réaliser d'une façon autonome les différentes tâches dans le monde réel.

Dans ce chapitre nous présentons dans un premier temps un aperçu historique sur les robots et leur évolution à travers le temps, par la suite nous nous intéressons aux différents types de robots, leurs domaines d'application, leur architecture, leurs avantages et inconvénients ainsi que les problèmes et les difficultés que rencontre la robotique et les robots mobiles qui ralentissent leurs développements.

## 1.2 Aperçu historique :

Parmi les mécanismes qu'a connus l'humain et qui ont introduit à la mobilité et à l'automatisation des machines, on trouve la roue, apparue pour la première fois vers 3500 avant J-C, qui est considérée comme le premier ouvrage technique significatif pour l'histoire de l'humanité.

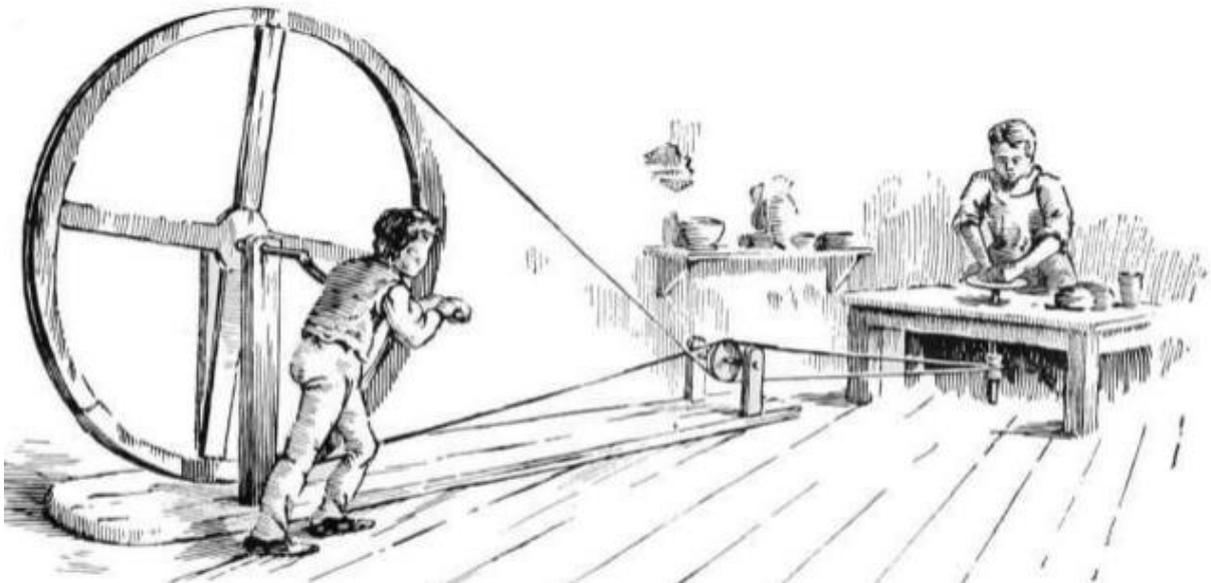


Figure 1 : la roue, premier mécanisme de la mobilité connu par l'homme.

L'autre mécanisme est bien la clepsydre (horloge à eau) inventées en Egypte pour la première fois vers 1600 avant J-C, elles sont bien utilisées pour la détermination de la durée d'un événement. Elles représentent aussi le berceau des premiers automates apparus.

Les premiers représentants de la longue histoire de la robotique sont bien les automates, apparus et connus dès l'Antiquité en se présentant sous différents œuvres tels que le pigeon volant d'Archaeas de Tarente ou les fameuses scènes théâtrales animées de Héron d'Alexandrie.

---

En XVI<sup>ème</sup> siècle, Léonard De Vinci aurait construit le premier androïde capable de se lever, s'asseoir et de coordonner les mouvements de ces bras.

Considéré comme l'âge d'or des automates, le XVIII<sup>ème</sup> siècle a connu le célèbre canard de Jacques De Vaucanson qui pouvait se nourrir et dégivrer sa nourriture en marchant.

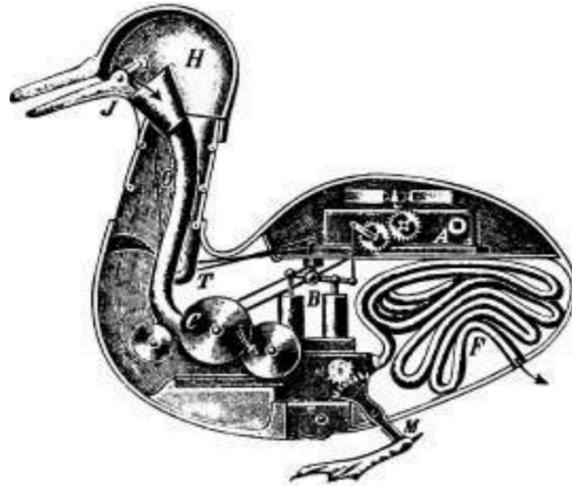


Figure 2 : le canard de Jacques De Vaucanson

Le XIX<sup>ème</sup> siècle a marqué de son tour l'apparition de l'automate Euphonie d'Eugène Faber qui était supposé de communiquer avec les spectateurs, ainsi que l'automate turc de Von Kempfen qui jouait aux échecs.



Figure 3 : automate turc de Von Kempfen

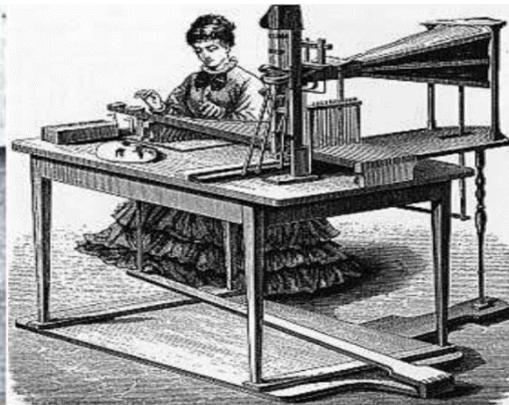


Figure 4 : automate Euphonia d'Eugène Faber

L'apparition des robots au monde réel n'est connue qu'au XX<sup>ème</sup> siècle suite aux travaux et recherches des ingénieurs et des scientifiques poussés par des biologistes ainsi que des psychologues développant à leur tour des hypothèses sur les mécanismes du corps humain et les secrets du cerveau et de la conscience, ces engagements et recherches scientifiques ont connus par la suite la fabrication des machines sous formes des animaux, à noter que le nom « animât » a été attribué aux animaux artificielles en 1985 par S.W Wilson. Ces machines ont la capacité d'apprentissage grâce à leur équipement des capteurs (organes sensoriels) qui recueillent les données de leur environnement et influence leur fonctionnement en les rendant plus adaptif :

---

1915 : le chien électrique de Hammond et Meissner, premier de son genre, pouvant se déplacer selon la luminosité de l'endroit où il se présente grâce à son capteur optique.

1950 : les tortues cybernétiques équipées de capteurs tactiles et lumineux de Grey Walter.

1953 : le renard électronique de Ducrocq.

L'année 1973 a marqué l'apparition du premier robot humanoïde « intelligent » nommé Wabot-1 présenté par l'université de Waseda, doté d'une vision et du pouvoir de manipulation des objets ainsi que la mobilité et emmener une conversation en japonais. L'année qui la succède a connu le robot intelligent d'Hitachi « Hi-T-Hand » spécifié par son pouvoir de manipulation des aiguilles à travers des trous grâce à la détection des forces.

Cela n'est qu'un brève aperçu historique citant quelques dates qui ont marquées la chaîne d'évolution du robot et de la robotique.

### **1.3 Définition de la robotique :**

La robotique est une activité multidisciplinaire, elle accumule l'étude ainsi que les différentes techniques de conception et de réalisation de robots ou plus simplement des machines automatiques. Savoir la pratiquer, nécessite la réunion des techniques et des connaissances scientifiques des domaines de l'électronique, de l'informatique et de la mécanique.

La robotique actuelle se présente dans de multiples domaines d'application, citant :

- La robotique industrielle.
  - La robotique domestique.
  - La robotique scientifique.
  - La robotique éducative.
  - La robotique médicale.
  - La robotique militaire.
  - La robotique de loisir.
  - La robotique de service.
  - La robotique de transport.
-



Figure 5 : images de quelques domaines d'application de la robotique

### 1.3.1 Etymologie du mot robotique :

Contrairement au terme « informatique » qui est issu de la langue française, le mot robotique vient de l'anglais « robotics » imaginé par le romancier Isaac Asimov et popularisé dans un livre apparu en 1942 « Runaround » dans lequel sont décrit « les trois lois de la robotique ».

### 1.4 Définition du robot :

Un robot est une machine physique commandée en position, programmable et reprogrammable, polyvalente et capable de manipuler des matériaux d'une manière automatique. Ou encore on peut bien définir un robot comme suit :

Un robot est un dispositif alliant de la mécanique, de l'électronique et de l'informatique, équipé de capteurs et d'effecteurs ce qui lui donne la possibilité de percevoir son environnement et de trouver une manière pour s'adapter aux différentes situations d'une façon automatique grâce au système électronique ou informatique qui le contrôle, afin de réaliser des tâches dans n'importe quel domaine et n'importe quelle situation étant difficile, répétitive, complexe ou dangereuse.

#### 1.4.1 Etymologie du mot « Robot » :

Le mot robot est inventé par l'écrivain dramaturge tchécoslovaque Karel Capek dans sa pièce théâtrale projetée en 1920 intitulée R.U.R (Rossum's Universal Robots). Ce mot est issu

---

de la langue tchèque mais utilisé dans d'autres pays de l'Europe de l'est tel que la Slovaquie, la Pologne et la Biélorussie sous l'appellation « robota » qui a pour sens de « esclave » ou « assistant ».

### **1.4.2 Robot/automate.... Confusion ?**

Robot, automate, deux termes souvent confondus plus au moins à cause de leurs définitions qui se rapprochent, pour éclaircir un peu les choses, on prend deux points majeurs ce sont la prise de décision et l'adaptation à l'environnement, ces derniers sont les deux aspects fondamentaux d'un robot autonome qui est capable d'analyser son environnement et d'agir en conséquence selon les besoins, contrairement à un automate qui est un dispositif assurant un enchaînement précis automatique et continu d'opérations mais qui lui sera non capable de prendre des décisions ou de s'adapter à son entourage.

### **1.4.3 Robot mobile :**

L'appellation « robot mobile » englobe d'une manière générale tous les types de robots ayant la possibilité ou la capacité de se mouvoir d'une manière autonome sans intervention humaine. Cependant, la différence entre ces derniers s'intitule à la manière ou la méthode dont ces robots se déplacent.

Un robot mobile est un véhicule capable de se déplacer dans un environnement totalement ou partiellement inconnu et cela ce fait possible grâce à son équipement des moyens de locomotion ainsi que des moyens de perception et de raisonnement.

## **1.5 Classification de robots mobiles :**

La classification des robots est faite par la prise en considération de plusieurs critères et points essentiels. Parmi ces critères, on prend ces deux points majeurs : le degré d'autonomie et le type de locomotion.

### **1.5.1 Classification selon le degré d'autonomie :**

Cette classification est considérée comme la plus intéressante car elle nous donne une image précise sur le degré auquel l'objectif principal d'un robot qui est le remplacement complet ou partiel de la touche humaine dans plusieurs secteurs d'activité est atteint.

L'autonomie est un comportement basique programmé initialement sans aide extérieure. Ou bien d'une façon brève, c'est de l'auto-gouvernance.

La prise de décisions et le pouvoir d'acquiescer et de traitement d'informations recueillies du milieu extérieur inconnu ce qui permettra d'accomplir une tâche sous contrôle humain réduit ou inexistant, caractérisent un robot mobile autonome. Selon ces critères, on peut classer les robots mobiles comme suit :

#### **1.5.1.1 Robot télécommandé :**

Tâche imposée par un opérateur (machine ou humain) en lui précisant la tâche à réaliser (avant, reculer, tourner à gauche ou à droite...).

---

### 1.5.1.2 Robot semi-autonome :

Ce type de robot est autonome jusqu'au moment où un opérateur l'interrompt et le contrôle pour recevoir des commandes.

### 1.5.1.3 Robot autonome :

On dit qu'un robot est autonome si et seulement s'il est capable de s'adapter à l'environnement où il se trouve.

## 1.5.2 Classification des robots mobiles selon le type de locomotion :

### 1.5.2.1 Les robots à roues :

Les robots à roues sont des robots qui utilisent des roues motorisées afin d'avoir la capacité de se mouvoir dans leur environnement. Cette conception est la plus simple et la plus utilisée mais de préférence elle nécessite un sol relativement plat. Plusieurs classes de robots à roues sont distinguées et déterminées par rapport au nombre et la position des roues utilisées. Dans ce qui suit on cite les quatre classes principales des robots à roues qui existent et qui sont le plus utilisés :

#### 1.5.2.1.1 Robots unicycle :

Un robot de type unicycle est actionné par deux roues indépendantes, assurant ainsi avec deux autres roues folles sa stabilité lors de son déplacement. Son centre de rotation est présent sur ces deux roues motrices. Ce type de robots n'assure pas un déplacement latéral instantané, d'une autre manière, il est impossible d'avoir un déplacement perpendiculaire aux roues de locomotion car un robot unicycle est un système non-holonyme.

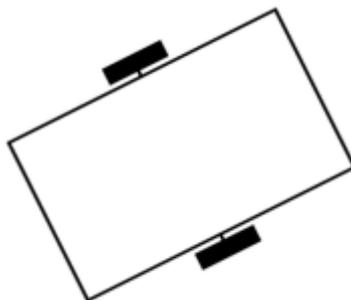


Figure 6 : robot unicycle

#### 1.5.2.1.2 Robot tricycle :

Ce type de robot est spécifié par ces deux roues fixes placées sur un même axe et d'une troisième roue orientable placée sur l'axe longitudinal. Le mouvement du robot tricycle est assuré par la vitesse des deux roues fixes et la roue orientable. Ceci reste toujours un type non-holonyme car il est impossible d'effectuer un mouvement dans une direction perpendiculaire aux roues fixes ou bien de faire des rotations simples qui restera aussi une tâche difficile à compléter à cause d'un rayon de braquage limité par la roue orientable.

---

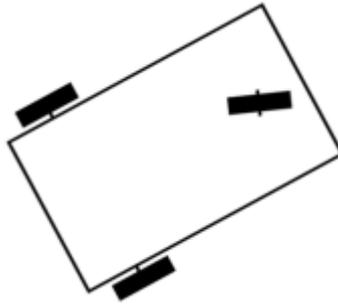


Figure 7 : robot tricycle

### 1.5.2.1.3 Robot voiture :

Le robot voiture est semblable au robot de type tricycle d'un point de vu, la seule différence entre ces deux types est que le robot voiture possède deux roues orientables fixées sur un même axe au lieu d'une seule roue comme il est le cas pour le robot tricycle. Cette différence marque un point avantageux pour ce type en lui offrant un point d'appui de plus et ainsi plus de stabilité. Comme il est mentionné en première phrase, ces deux derniers types de robots dont on en a parlé, ont en similaires toutes les autres propriétés (ce sont des robots non-holonome).

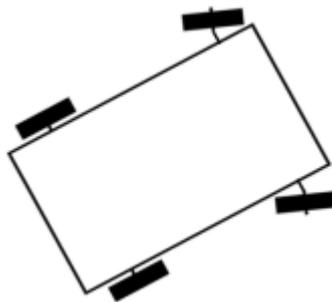


Figure 8 : robot voiture

### 1.5.2.1.4 Robot omnidirectionnel :

Pour ce type de robot, il vient avec une option avantageuse et trouvée nulle part pour les autres types de robots mentionnés précédemment, cette option se présente dans la possibilité de ce dernier de se déplacer dans toutes les directions en toute délicatesse. Ceci n'est possible qu'avec sa structure mécanique spécifique constituée de trois roues décentrées orientables placées en triangle équilatéral.

L'énorme avantage qui se présente en robot omnidirectionnel c'est qu'il est holonome, mais ceci n'est une tâche facile car se fait dépend d'une complexité mécanique bien plus grande.

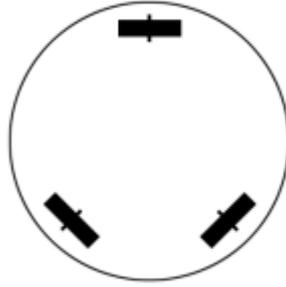


Figure 9 : robot omnidirectionnel

Pour en finir avec les robots à roues, voici le tableau suivant récapitulera quelques points essentiels de chaque type de robots en nous donnant une idée générale sur les points en communs et la différence entre chaque type :

Type de robot mobile à roues	Spécifications
Robot unicycle	<ul style="list-style-type: none"> <li>- non-holonome.</li> <li>+ stable.</li> <li>+ rotation sur sois même.</li> <li>+ complexité mécanique faible.</li> </ul>
Robot tricycle	<ul style="list-style-type: none"> <li>- non-holonome.</li> <li>- peu stable.</li> <li>- pas de rotation sur sois même.</li> <li>+ complexité mécanique modérée.</li> </ul>
Robot voiture	<ul style="list-style-type: none"> <li>- non-holonome.</li> <li>+ stable.</li> <li>- pas de rotation.</li> <li>+ complexité mécanique modérée.</li> </ul>
Robot omnidirectionnel	<ul style="list-style-type: none"> <li>+ holonome.</li> <li>+ stable.</li> <li>+ rotation sur sois même.</li> <li>- complexité mécanique importante.</li> </ul>

Tableau 1 : types des robots mobile à roues

### 1.5.2.2 Les robots à chenilles :

L'utilisation des roues est le type de locomotion le plus utilisé, mais ce dernier n'est avantageux qu'aux sols qui sont relativement plats et perd son efficacité contre les environnements et les terrains accidentés ce qui déminue la mobilité d'un robot équipé de

---

ce type de système de locomotion. C'est en cette situation que les chenilles présentes leur avantage, un avantage qui permet d'augmenter le niveau d'adhérence et de franchir les obstacles présent tout au long de la trajectoire du robot afin d'avoir une meilleure mobilité pour bien terminer la tâche qui lui est dédié.



Figure 10 : robot à chenilles

### 1.5.2.3 Les robots mobiles à pattes ou les robots mobiles marcheurs :

Pour les situations les plus extrêmes où les terrains sont plus accidentés ou présentent une différence de hauteur comme par exemples des escaliers, les robots marcheurs ou les robots à pattes montrent compétence et efficacité. Contrairement aux robots à roues ou encore ceux à chenilles, les robots marcheurs fournissent un niveau supplémentaire de mobilité, ils peuvent marcher facilement sur des petits fossés et même passer des obstacles sans même pas avoir la nécessité de changer la trajectoire pour les éviter.

Les robots marcheurs présentent une variété de types et de modèles, par exemple on trouve les robots qui ressemblent à des êtres humains ou bien d'une manière générale les robots marcheurs à deux pattes ou appelés encore des "bipèdes", il y a aussi ceux qui sont inspirés des animaux. Bien évidemment on trouve les robots marcheurs à quatre pattes appelées des "quadrapèdes" et même d'autres à six pattes appelés des "hexapodes".

---

Tous ces points rendent les robots marcheurs peut être invincibles mais faisant face à la réalité, la conception et la réalisation de ce type de robot doivent surmonter plusieurs difficultés. Parmi les difficultés qui se présentent en premières lignes c'est le problème de stabilité, comment peut-on assurer ce critère qui est indispensable soit lorsque le robot est en mouvement ou bien lorsqu'il est immobile. Le deuxième point qui se relève est comment faire bouger les jambes ou les bien les pattes du robot afin qu'il puisse faire une marche avant ou bien une marche arrière tout en gardant le premier problème en vue. Enfin, et encore plus difficile, le problème qui se pose est qu'elle est la manière pour faire tourner le robot dans un coin afin qu'il puisse se déplacer.

Il y a bien des alternatives pour fuir ces problématiques, donnant à titre d'exemple le robot qui utilise des larges pieds "de canard" ou encore des robots à plusieurs pattes tels que les robots à quatre pattes ou encore à six pattes qui font preuves de compétences en rendant l'équilibre une tâche facile tout en marchant à des vitesses plus au moins rapides sans tomber ainsi que de pouvoir faire des virages et se déplacer dans environnement accidentés. Mais ces alternatives présentent à leur tour de nombreuses complexités.



Figure 11 : robot marcheur

#### **1.5.2.4 Robots présentant un autre type de locomotion :**

Il est indispensable de signaler que parmi les types de locomotion mentionnés en dessus il existe d'autres moyen de locomotion. Par exemple, les robots sous-marins, les robots volants ou appelés également des "drones", les robots pour l'exploration spatiale...etc. les robots utilisant ces types de locomotion ont la particularité d'être employés pour des applications bien visées et c'est ainsi pour quoi on les trouve aussi bien spécialisés par des architectures spécifiques dédiés à la mission dont ils sont fabriqués pour accomplir.

---

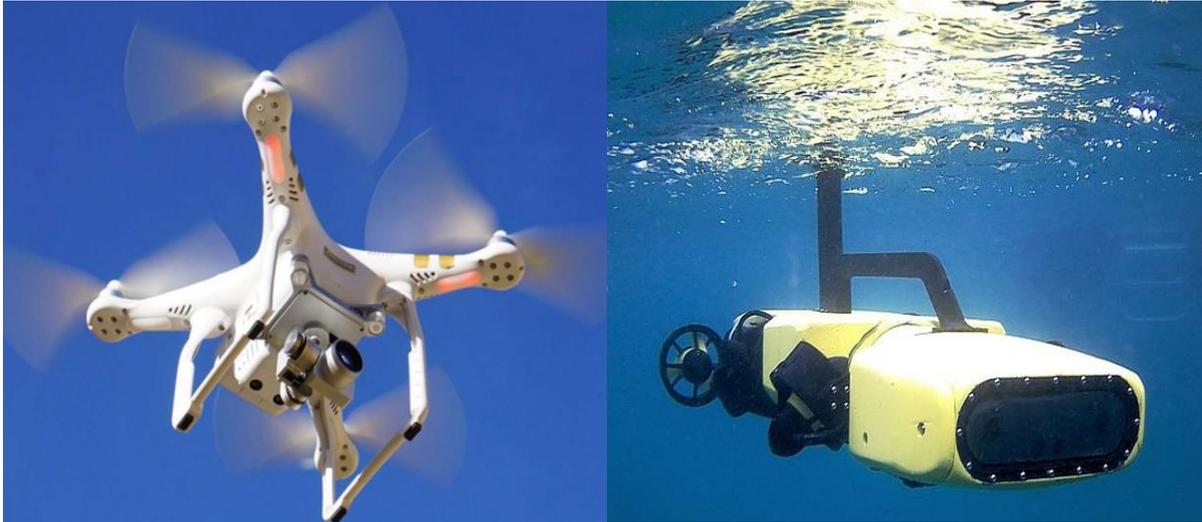


Figure 12 : robots présentant un autre type de locomotion

## 1.6 Architecture des robots mobile :

L'architecture générale des robots mobiles est composée de quatre parties essentielles se présentent comme suit :

- La structure mécanique et la motricité.
- Les organes de sécurité.
- Le système de traitement d'informations et gestion de tâches.
- Le système de localisation.

### 1.6.1 La structure mécanique et la motricité :

Pour ce qui est de la structure mécanique assurant la motricité, on a déjà abordé les différents types de locomotion des robots mobiles dans la parties classification des robots mobiles, donc ça sera du répétitif si on l'aborde de nouveau.

### 1.6.2 Les organes de sécurité :

Afin d'augmenter la durée de vie d'un robot et avoir un meilleur rendement, l'idée de mettre le robot en sécurité et de le garder loin des sources perturbantes est indispensable, et pour cela le robot doit être équipé des organes de sécurité. Comme leur nom l'indique, les organes de sécurité sont des dispositifs qu'on équipe au robot afin de le protéger de tout ce qui peut le nuire et le déraille de sa mission. Des capteurs sont employés dans ces conditions, il existe plusieurs types de capteurs chacun est spécifié pour une tâche précise, dans notre cas deux types de capteurs sont utilisés, les capteurs de proxémiques ou bien appelés aussi les capteurs périmétriques (ultrasons, hyperfréquences, infrarouges...) assurant la détection des obstacles avant collision, et les capteurs de contact qui détectent un choc ou une collision avec l'environnement où il se présente le robot (contact électrique sur pare-chocs, résistance variable...).

Ces derniers se regroupent en deux catégories différentes :

---

### **1.6.2.1 Les capteurs proprioceptifs :**

Les capteurs proprioceptifs fournissent des informations permettant de mesurer l'état du robot lui-même (vitesse de roues, position, charge de la batterie...). Ces informations qui aident le robot en cas de mouvement, sur son déplacement dans son environnement.

### **1.6.2.2 Les capteurs extéroceptifs :**

Ces capteurs ont pour objectif de mesurer l'état de l'environnement (cartographie, température...) afin d'acquérir des informations sur l'environnement proche du robot.

### **1.6.3 Traitement des informations et gestion des tâches :**

Le traitement des informations et la gestion des tâches permettent d'établir les commandes qui donneront par la suite au robot mobile la possibilité de se mouvoir et d'activer les organes nécessaires tout en gardant une cohérence avec l'objectif.

La génération de plan, un problème qui se pose car elle consiste à trouver la manière avec laquelle le robot va se déplacer en tenant compte des connaissances aprioriques ou même celles obtenues lors d'évolution.

D'une manière générale, la génération de plan se fait suivant trois points majeurs :

- La stratégie de navigation.
- La modélisation de l'espace.
- La planification.

### **1.6.4 Le système de localisation :**

Le système de localisation d'un robot mobile comme son l'indique, permet de localiser et de définir le positionnement du robot dans l'environnement où il se trouve ou même d'éléments de référence ou de particulier qui se trouvent au même environnement que le robot.

## **1.7 Les domaines d'application des robots mobiles :**

La robotique actuelle est utilisée dans pas mal de domaine différent, cette intervention n'est dû qu'au développement consécutive de cette branche et l'adoption des nouvelles technologies afin de répondre aux besoins de toutes les branches. Utilisée principalement en premier temps dans l'industrie et ensuite généralisé dans d'autres domaines plus exigeant que nous citerons dans ce qui suit :

### **1.7.1 L'industrie :**

Comme on l'a déjà cité, les premières applications de la robotique étaient dans l'industrie comme était le premier but des robots remplacer l'être humain dans les tâches difficiles et répétitives. Les robots sont implantés dans les chaînes de production, d'emballage, de montage.... On les trouve aussi comme des soudeurs, peintres, manipulateurs....

---



Figure 13 : robot utilisé dans le domaine industriel

### 1.7.2 Le domaine militaire :

La robotique s'est fait une place dans le coté militaire en proposant des modèles miniaturisés et discrets dotés de capteurs idéal pour les missions d'espionnage, ainsi que les robots volants "drones" et robots sous-marins qui offrent la possibilité d'exploration et de surveillance des frontières tout en gardant la vies des soldats à l'abrie.



Figure 14 : robot utilisé dans le domaine militaire

### 1.7.3 La santé :

La robotique est également présente en domaine médicale, elle est considérée comme un appui important pour les assistances médicales et chirurgicale. Les robots utilisés dans ce domaine ne sont pas tout à fait autonomes, ils sont observés et commandés par des médecins compétents à l'aide des joysticks et des pédales, mais cette intervention humaine ne sera plus nécessaire au futur vu les croissances technologiques que connait la robotique.



Figure 15 : robot utilisé dans le domaine médicale

#### **1.7.4 Usage civile :**

De plus en plus les robots font preuve de compétence, ils se présentent même dans le domaine civile en remplaçant les gens chargés des taches civiles et des travaux à intérêt général tel que le nettoyage de la ville, l'aide de la population, s'occuper des lieux publics....



Figure 16 : robot mobile utilisé dans le domaine civil

#### **1.7.5 Usage domestique :**

De plus en plus les robots font part de notre vie quotidienne, ils ont pour but de décharger leur propriétaire des activités domestiques peu attractives. Ceux-ci sont capable de faire du ménage, surveillance du domicile, tondre la pelouse, nettoyer des piscines...



Figure 17 : robot utilisé dans le domaine domestique

Pour en finir avec les différentes applications de la robotique, on met un tableau récapitulatif donnant un aperçu sur d'autres domaines où la robotique et les robots font part.

Domaines	Applications
Agricole	<ul style="list-style-type: none"> <li>- Cueillette de fruit.</li> <li>- Traite, moisson, traitement de vignes.</li> </ul>
Mine	<ul style="list-style-type: none"> <li>- Assistance d'urgence.</li> <li>- Exploration de mines.</li> </ul>
Chimique	<ul style="list-style-type: none"> <li>- Manipulations des produits chimiques.</li> <li>- Débarrassements des déchets chimiques.</li> <li>- Surveillance de site.</li> </ul>
Nucléaire	<ul style="list-style-type: none"> <li>- Surveillance des centrales nucléaire.</li> <li>- Manipulations des matériaux radioactifs.</li> <li>- Démantèlement des centrales.</li> </ul>
Sécurité civile	<ul style="list-style-type: none"> <li>- Surveillance et exploration des sites isolés.</li> <li>- Neutralisation des activités terroristes.</li> <li>- Déminage.</li> <li>- Pose d'explosif.</li> <li>- Surveillance des munitions.</li> </ul>
Espace	<ul style="list-style-type: none"> <li>- Exploration de l'espace.</li> <li>- Prise d'échantillons pour des analyses scientifiques.</li> </ul>
Sous-marin	<ul style="list-style-type: none"> <li>- Exploration et inspection des fonds marins.</li> <li>- Pose de câbles.</li> <li>- Recherche de navires immergés.</li> <li>- Recherche de module.</li> </ul>

Tableau 2 : domaines d'application de la robotique

## **1.8 Les avantages et les inconvénients des robots :**

### **1.7.5 Les avantages :**

- Un robot ayant une morphologie similaire à celle de l'être humain peut travailler dans un environnement dédié à un usage humain sans avoir la nécessité de le modifier, il peut même travailler et collaborer avec des travailleurs humains.
- Remplacer l'homme en son absence, par exemple un robot humanoïde pourrait être utilisé pour des applications de télé présence.
- Simplifier la vie de l'homme en le remplaçant dans les tâches fatigantes et complexes et accroître le taux de productivité.
- Accomplir des tâches qui demandent de la précision en une durée bien limitée ce qui donnera une production de masse des produits sans erreur humaine.
- Réduction des coûts de production en éliminant les coûts internes afin de compenser les salaires humains.

### **1.7.6 Les inconvénients :**

- La perte de l'emploi est l'opposition majoritaire de l'utilisation des robots aux milieux industriels, ce qui provoquera une augmentation du chômage et une baisse de consommation.
- Le coût d'un robot performant est monstrueux, sans négliger le coût de sa maintenance qui est également élevé.
- La sécurité et le contrôle social, comment peut-on s'assurer si la masse des informations recueillies par ces robots de compagnie ne seront pas utilisées dans un but de contrôle politique.

## **1.8 Les problématiques de la robotique :**

Malgré les avancées technologiques et les succès que connaît la robotique et son intervention dans quasiment tous les secteurs, cette dernière rencontre un certain nombre de problèmes. Bien évidemment, on trouve un problème lié au choix d'aspects matériels qui consiste à trouver et choisir une structure mécanique du système, sa motorisation et alimentation ainsi que le choix d'une architecture informatique spécifique du système de commande et de contrôle.

Le choix de la structure est effectué d'une manière pour laquelle les problèmes de modélisation, planification et de commande seront résolus. De même les moteurs électriques à courant continu avec ou sans collecteur, alimentés par des convertisseurs de puissance fonctionnant sur des batteries, font un choix classique des actionneurs et de l'alimentation.

Enfin pour ce qui est de problèmes d'aspect logistique, les architectures de contrôle-commande des robots mobiles sont similaires à ceux des systèmes automatiques et robotiques. Cependant deux niveaux de spécialisation sont distingués :

- Couche décisionnelle : chargé de la planification et la gestion des événements.
  - Couche fonctionnelle : chargé de la génération en temps réels des commandes.
-

Ça reste évident que l'architecture du robot reste dépendante des avancements technologiques et ses offres du moment.

## **1.9 Conclusion :**

Dans ce premier chapitre qui est un chapitre introductif, on a pu accomplir notre tâche qui consiste à s'introduire à la robotique et aux robots mobiles d'une façon générale.

Nous avons donné un aperçu historique sur les différentes étapes d'évolution de la robotique, la définition de la robotique et des robots mobiles, leur classification selon leur type de locomotion et leur type d'autonomie, architecture générale des robots mobiles, les différentes application dont ils sont utilisés ainsi que les différents avantages et inconvénients des robots mobiles et les problématiques de la robotique.

---

**Chapitre 2 :**  
**Conception et**  
**réalisation d'une**  
**cartre de commande**  
**d'un robot mobile.**

---

## 2.1 Introduction :

Un système embarqué est un système électronique et informatique à la fois, c'est un élément constitutif d'un système plus complexe pour lequel il rend des services bien précis. Il est constitué de parties matérielles et logicielles conçues spécifiquement afin de réaliser une tâche bien précise. Arduino qui est une plateforme rassemblant un matériel et un logiciel "OPEN SOURCE" et qui sera aussi notre appui dans notre projet dans lequel on commandera un robot mobile à roue.

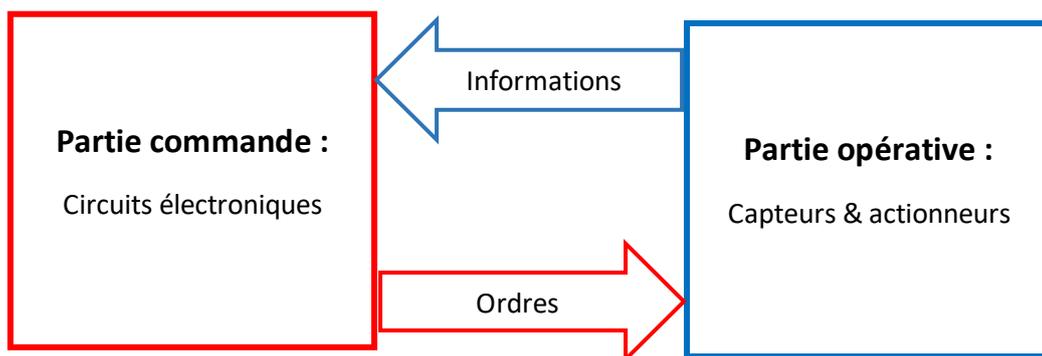
Dans ce chapitre on va aborder en détaille la façon avec laquelle on va commander notre robot ainsi que les composants et le matériel utilisé afin de le réaliser.

## 2.1 Commande d'un robot :

Un robot est système de commande automatisé contenant deux parties différentes mais reliées entre elles.

Partie opérative : partie contenant des actionneurs ainsi que des capteurs.

Partie commande : cette partie contient des circuits électroniques ( carte Arduino, carte de commande...).



## 2.2 Historique du projet Arduino :

Issu et conçu par une équipe d'enseignants et d'étudiants (David Mellis, Tom Igoe, Gianluca Martino, David Cuartielles, Massimo Banzi ainsi que Nicholas Zambetti) de l'école de design d'interaction d'Ivera (en Italie). Le projet Arduino (2005) est inspiré de la carte Wiring (2003). Wiring, une carte électronique développée par Hernando Barragon en 2003 accompagné par un environnement de programmation libre et ouvert. Pour ce travail Hernando réutilisait les sources du projet procession (un langage de programmation développé par Casey Reas et Ben Fry en 2001) afin d'arriver à un dispositif simple dans son utilisation et surtout moins coûteux par rapport aux autres outils de création de projets.

Le projet Arduino consiste un environnement particulièrement adapté à la production artistique ainsi qu'au développement de conceptions qui peuvent trouver leurs réalisations dans la production industrielle.

---

L'origine du nom Arduino est issue du nom du bar dans lequel l'équipe fondatrice avait l'habitude de se trouver. Aussi le nom d'un roi italien, personnage historique de la ville "Arduino d'Ivrée", et même encore un prénom italien qui signifie "l'ami fort".

## 2.3 Présentation de la carte Arduino :

Arduino est composé principalement de deux parties majeures : partie matérielle et partie logicielle.

Partie matérielle : il s'agit d'une carte électronique considérée comme un hébergement du micro-contrôleur lequel autour sont implantés d'autres composants électroniques ainsi qu'un accès à toutes les entrées et sorties assurant la connectivité de la carte avec d'autres composants.

Parties logicielles : il s'agit d'un environnement de programmation Arduino (IDE : Intégrates Développement Environnement) c'est une application écrite en JAVA inspiré du langage Processing. Cette interface permet d'écrire, de modifier un programme et de le convertir en une série d'instructions compréhensible pour la carte.

Le système Arduino donne la possibilité de fusionner les performances de la programmation à celles de l'électronique.

## 2.4 Les différents types de cartes Arduino :

### 2.4.1 Arduino UNO :

Le type de carte Arduino le plus populaire, alimenté par un processeur Atmega328 fonctionnant à 16MHz, il comprend une mémoire de programme de 32 Ko, 1 Ko d'EEPROM, 2 Ko de RAM, 14 entrées et sorties numériques, 6 entrées analogiques et un rail d'alimentation de 5V et 3.3V.



Figure 18 : Arduino UNO

### 2.4.2 Arduino NANO :

Un arduino NANO est considéré comme un arduino UNO réduit, ce qui lui donnera ainsi plus d'avantages par exemple pour ses petites dimensions il sera très pratique pour les espaces restreints il est aussi pratique pour les projets qui nécessitent une réduction de poids. Les cartes arduino Nano sont souvent l'option moins chère de l'arduino UNO, donc ils sont retables pour les grands projets.

Comme un arduino UNO, un arduino Nano est doté d'un processeur Atmega328 fonctionnant à 16MHz (les cartes Arduino Nano antérieurs de la version v3.0 utilisaient l'Atmega168 ce qui correspond à la moitié des spécifications), 32Ko de mémoire programme, 1Ko d'EEPROM, 2Ko de RAM, 14 entrées et sorties numériques, 6 entrées analogiques et des rails d'alimentation de 5V et 3.3V.

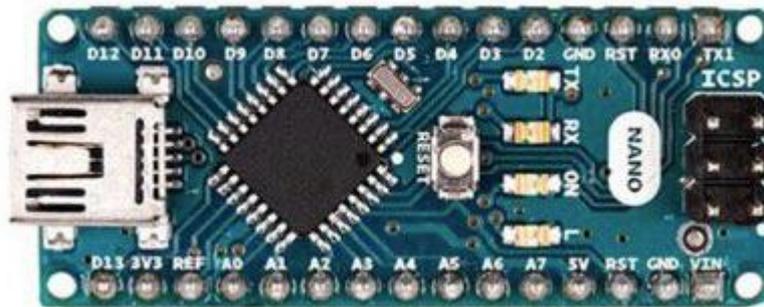


Figure 19 : Arduino NANO

### 2.4.3 Arduino Due :

Considéré comme l'une des plus grandes cartes Arduino, de plus elle est la première carte à être alimentée par un processeur ARM.

Equipé d'un Cortex-M3 ATSAM3X8E cadencé à 84 MHz, il dispose effectivement de 512Ko de ROM et de 96Ko de RAM, 54 broches d'entrées/sorties numériques, 12 canaux PWM, 12 entrées analogiques, 2 sorties analogiques et de ce qui est important à noter que l'Arduino Due est doté d'une alimentation de 3.3V contrairement à l'Arduino UNO ou encore Nano car un e surtension endommagerait directement la carte.

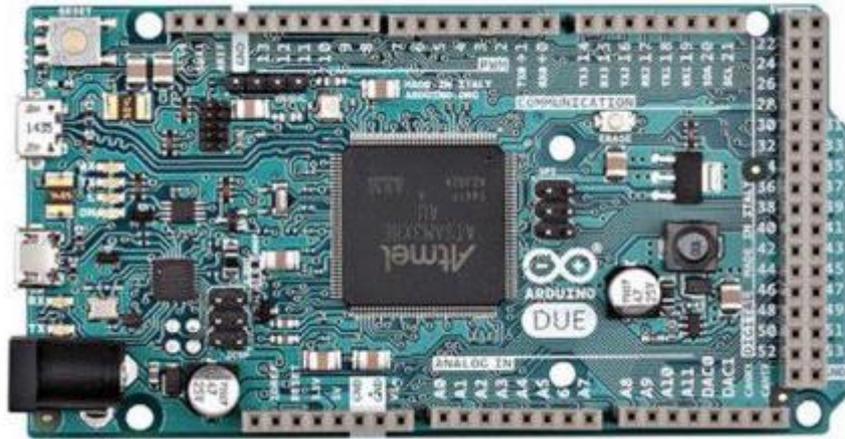


Figure 20 : Arduino DUE

#### 2.4.4 Arduino Mega 2560 :

L'Arduino Mega est un peu similaire à l'Arduino Due, en ce sens ils partagent quelques caractéristiques, ils ont le même brochage ils disposent également de 54 entrées/sorties.

L'Arduino Mega est alimenté par un processeur ATmega 2560 cadencé à 16MHz, il comprend 256Ko de ROM, 8Ko de RAM, 4Ko d'EEPROM, alimentation à 5V ce qui facilite son utilisation avec la plupart des appareils électroniques, il dispose aussi de 15 entrées analogiques et 15 canaux PWM.



Figure 21 : Arduino MEGA 2560

#### 2.4.5 Arduino Leonardo :

La carte Arduino Leonardo est basée sur le processeur Atmega32u4 cadencé à 16MHz ce qui permet la gestion du port USB et d'augmenter la flexibilité dans la communication avec l'ordinateur. Il dispose de 32Ko de mémoire flash, 2.5Ko mémoire SRAM, 1Ko d'EEPROM, 20 broches d'entrées/sorties dont 6 PWM et une intensité par E/S de 40mA, 6 entrées

analogiques, bus série I2C et SPI, une gestion d'interruptions, fiche micro USB et une alimentation de 7V à 12V sur connecteur ou bien une alimentation via port USB.



Figure 22 : Arduino LEONARDO

## 2.5 Les différentes applications à base d'Arduino :

Plusieurs applications sont possibles et réalisable à base d'une carte Arduino, on cite :

- Contrôle des appareils domestiques.
- Télécommander un appareil mobile.
- Réalisation des robots.
- Réalisation d'animations lumineuses.
- Réalisation de jeux ludiques et interactifs.

## 2.6 Carte Arduino UNO :

### 2.6.1 Présentation :

La carte Arduino UNO est un circuit électronique imprimé programmable permettant de faire fonctionner un ensemble de composants. Elle possède des ports de connexion assurant une liaison avec d'autres composants ou avec un ordinateur.



Figure 23 : Arduino UNO

## 2.6.2 Architecture :

Une carte Arduino UNO comporte essentiellement :

- Un microcontrôleur Atmega 328.
  - Une horloge à 16MHz.
  - 32Ko de mémoire programme.
  - 1Ko de mémoire EEPROM.
  - 2Ko de mémoire RAM.
  - Des broches de connexion dont 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme sortie PWM) et 6 entrées analogiques.
  - Une connexion USB.
  - Une alimentation de 5V ou de 3.3V.
  - Un bouton RESET et d'autres composants.
-

Microcontrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6 (utilisables en entrées/sorties numériques)
Intensité max disponible par broche E/S (5V)	40 mA (200 mA cumulé pour l'ensemble des broches E/S)
Intensité max disponible pour la sortie 3.3V	50 mA
Intensité max disponible pour la sortie 5V	Fonction de l'alimentation utilisée 500mA max si port USB est utilisé seul
Mémoire programme flash	32 Ko (ATmega328)
Mémoire SRAM	2 Ko (ATmega328)
Mémoire EEPROM	1 Ko (ATmega328)
Vitesse d'horloge	16MHz

Tableau 3 : caractéristique de la carte Arduino UNO

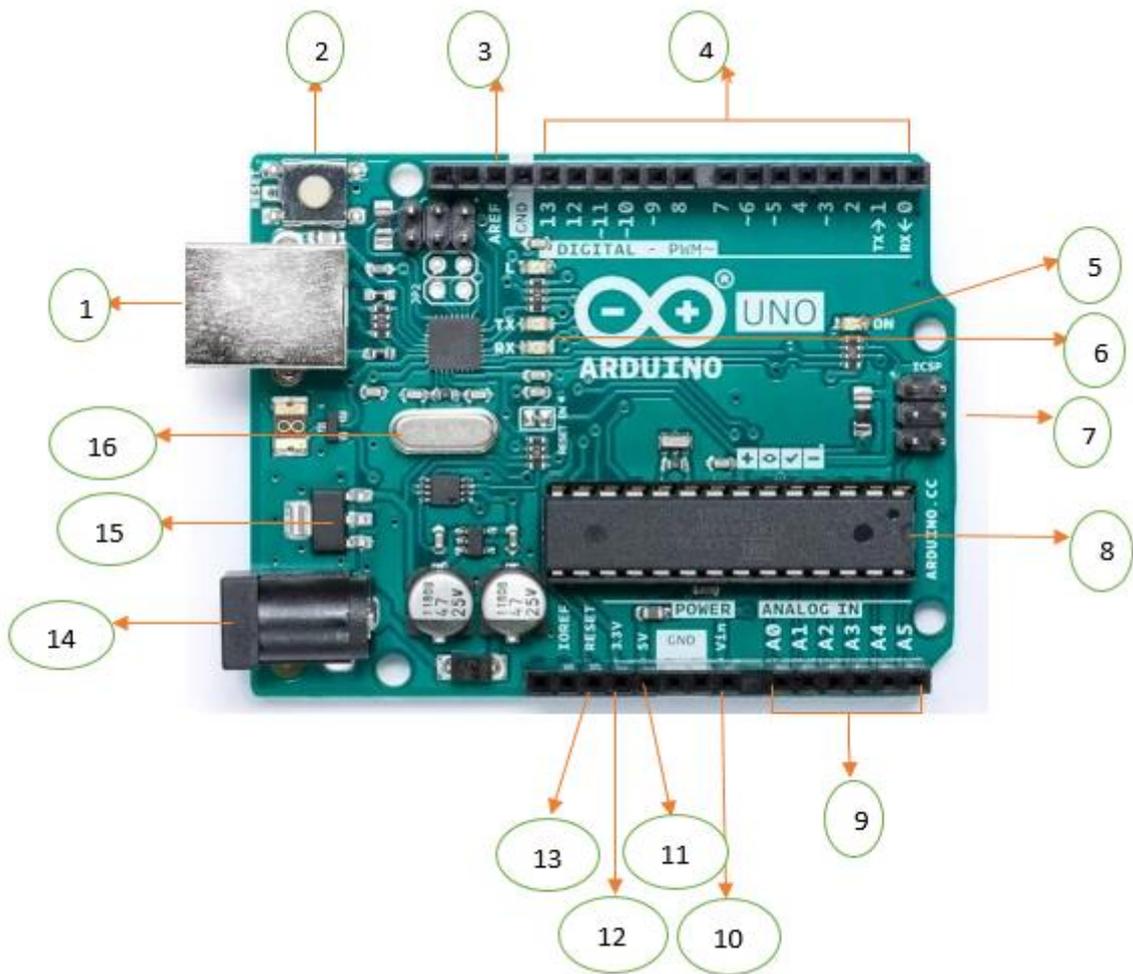


Figure 24 : les différentes parties de la carte Arduino UNO

1	<b>Connexion USB</b> : utilisée pour relier la carte Arduino avec l'ordinateur afin de téléverser les codes écrits sur Arduino IDE, elle est même utilisée comme source d'alimentation pour la carte.
2	<b>Bouton RESET</b> : un bouton qui sert à réinitialiser la carte Arduino.
3	<b>Broche AREF</b> : "Analog REference" utilisée pour désigner un voltage extérieur de référence.
4	<b>Entrées/Sorties numériques</b> : la carte Arduino UNO dispose de 14 broches d'entrées/sorties numériques dont 6 broches peuvent être utilisées pour générer des signaux PWM.
5	<b>LED indicatrice</b> : la LED indicatrice s'allume lorsque la carte Arduino est branchée à une source d'alimentation, dans le cas contraire, un problème de connexion de la carte est possible.
6	<b>TX/RX LED</b> : TX pour "Transmit" et RX pour "Receive" ces deux LED clignotent lors de la transmission ou réception de données, la vitesse de clignotement peut varier selon le débit en bauds utilisé par la carte.
7	<b>ICSP PIN</b> : La plupart du temps, ICSP est un AVR, un tout petit en-tête de programmation pour l'Arduino composé de MOSI, MISO, SCK, RESET, VCC et GND. Il est souvent appelé SPI (Serial Peripheral Interface) qui pourrait être considéré comme une expansion de la sortie. En fait, vous asservissez le périphérique de sortie au maître du bus SPI.
8	<b>MICROCONTROLLEUR</b> : chaque carte arduino est dotée d'un microcontrôleur qui est considéré comme le cerveau de la carte, ce circuit intégré peut se différencier d'une carte à l'autre, les microcontrôleurs utilisés dans les différentes cartes arduino sont généralement de ATmel Company.
9	<b>Les entrées analogiques</b> : la carte arduino est équipée de six broche d'entrée analogique permettant de convertir un signal analogique obtenu des différents capteurs analogiques tel que le capteur de température ou d'humidité en un signal électrique compris par le microprocesseur.
10	<b>Vin pin</b> : cette broche peut être aussi utilisée comme source d'alimentation externe pour la carte Arduino.
11	<b>5V pin</b> : cette broche peut fournir 5V en sortie pour alimenter les différents périphériques associés à la carte Arduino.
12	<b>3.3V pin</b> : cette broche peut fournir 3.3V en sortie pour alimenter les différents périphériques associés à la carte Arduino.
13	<b>RESET pin</b> : cette broche permet de réinitialiser la carte Arduino, en fait il en existe deux méthodes avec lesquelles réinitialiser une carte Arduino, la première en utilisant le bouton de réinitialisation et la deuxième en associant un bouton à la broche RESET de la carte.
14	<b>BARREL JACK POWER</b> : (connecteur JACK) une carte Arduino peut être alimenter par deux méthodes différentes, soit par la connexion USB ou bien par l'utilisation de la prise 2.1mm d'alimentation du connecteur JACK de la carte.
15	<b>Régulateur de tension</b> : la fonction principale du régulateur de tension est de contrôler la tension d'alimentation fournit à la carte Arduino et la stabiliser.
16	<b>Oscillateur à cristal</b> : l'oscillateur à cristal aide arduino à traiter les questions de temps. le nombre imprimé sur le dessus du cristal arduino est 16000H9H. il nous dit que la fréquence est de 16 MHz.

Tableau 4 : les broches de l'Arduino UNO

### 2.6.2.1 Le microcontrôleur ATmega 328 :

L'ATmega 328 est un microcontrôleur à puce possédant un cœur de processeur de 8 bits RISC (Reduced Instructions Set Computer) à architecture Harvard modifiée, créé par Atmel dans la famille megaAVR (Microchip Technology a acquis Atmel en 2016).

L'ATmega 328 est connu pour ses spécifications performantes, il est doté d'un processeur AVR 8 bits combinant une mémoire de type flash de 32 Ko, une SRAM de 2 Ko et une EEPROM de 1 Ko. 23 lignes de brochages d'entrées/sorties à usage général, deux interruptions internes et externes, trois compteurs/horloges avec modes de comparaison, 32 registres de travail à usage général, USART (récepteur émetteur synchrone et asynchrone universel) programmable série, un port série SPI (interface périphérique série), un convertisseur analogique/numérique à 10 bits et une horloge de surveillance programmable avec un oscillateur exécutant des instructions puissantes en un cycle d'horloge.

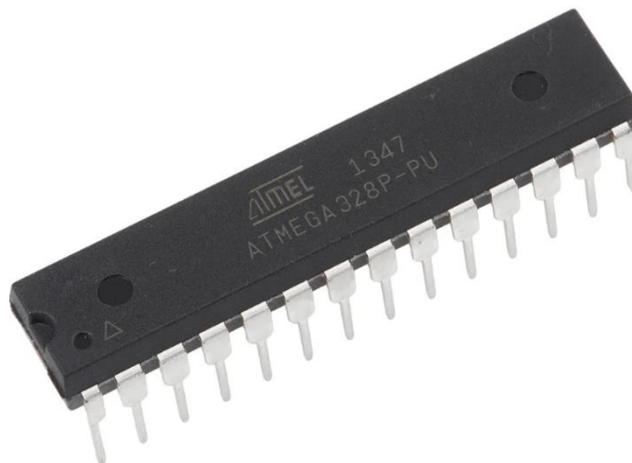


Figure 25 : microcontrôleur ATmega 328

### 2.6.2.2 Les mémoires :

L'ATmega 328 utilise trois types de mémoire : les mémoires FLASH, les SRAM et l'EEPROM.

#### 2.6.2.2.1 Les mémoires FLASH :

Utilisée pour stocker des programmes à exécuter, elle non volatile et rapide, elle est considérée comme la mémoire idéale, le seul inconvénient c'est qu'elle est couteuse. L'ATmega est doté d'une mémoire de ce type de capacité de 32Ko.

#### 2.6.2.2.2 Les memories SRAM : (Static-Random-Access-Memory)

C'est une mémoire qui rapide mais couteuse, utilisée pour le stockage temporaire de données (les variables de programme par exemple), de plus c'est une mémoire volatile. L'ATmega est doté d'une SRAM de 2Ko.

#### 2.6.2.2.3 Les mémoires de types EEPROM : (Electrical Erasable Programmable Read Only Memory)

C'est une mémoire non volatile (conserve les données même si l'alimentation est coupée) elle est peu couteuse mais lente, c'est un type de mémoire qui est à lecture uniquement,

---

effaçable électriquement et programmable. L'ATmega en possède une de ce type d'une capacité de 1Ko.

### **2.6.3 Alimentation de la carte Arduino UNO :**

La carte Arduino UNO peut être alimentée suivant deux méthodes différentes, la première via la connexion USB (qui fournit 5V), la deuxième méthode est en choisissant une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

Pour ce qui est de l'alimentation extérieure, on en a le choix entre soit un adaptateur secteur (qui peut fournir typiquement de 7V jusqu'à 12V) ou des piles ou bien des accumulateurs (accus). L'adaptateur secteur est connecté à la carte en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte.

Les fils en provenance des piles ou des accus peuvent être insérés dans les connecteurs des broches de la carte appelées GND (la masse) et Vin (tension positive en entrée) du connecteur d'alimentation.

La plage de voltage pour le fonctionnement d'une carte Arduino UNO via une alimentation externe est de 6V jusqu'à 12V. cependant, si la carte est alimentée avec une alimentation fournissant moins de 7V, la broche 5V pourrait fournir moins de 5V ce qui provoquera un état instable pour la carte. Par contre si on utilise une alimentation de plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Ainsi, la plage idéale pour l'alimentation d'une carte Arduino UNO est entre 7V et 12V.

On trouve les différentes broches d'alimentation de la carte Arduino UNO :

Vin : la tension d'entrée positive est utilisée lorsque la carte est alimentée par une alimentation externe ou bien si l'alimentation est fournie par le jack d'alimentation.

5V : la tension régulée, cette broche est utilisée pour alimenter le microcontrôleur ainsi que les autres composants de la carte Arduino UNO qui nécessitent une tension d'alimentation parfaitement stable dite la tension régulée.

3.3V : une alimentation de 3.3V fournie par le circuit intégré FTDI (c'est un circuit faisant l'adaptation du signal entre le port USB de l'ordinateur et le port série de l'Atmega) ceci reste quand même intéressant pour certains circuits externes nécessitant une tension de 3.3V au lieu celle de 5V.

GND : broche de la masse ou de 0V.

### **2.6.4 Les broches (entrées/sorties) numériques de la carte Arduino UNO :**

La carte Arduino UNO dispose de 14 broches peuvent être utilisées soit comme entrées numériques, soit comme sorties numériques cela reste au choix de l'utilisateur de configurer les broches convenablement à ses besoins.

Certaines broches sont spécifiées par des fonctions spéciales :

---

Broches 0 et 1 : assurant une communication série "RX(0) et TX(1)" ces deux broches sont utilisées pour la réception et la transmission de données séries, elles connectées aux broches correspondantes du circuit intégré ATmega8u2 programmé en convertisseur USB vers série de la carte.

Broches 2 et 3 : peuvent être utilisées pour générer des interruptions externes sur une valeur basse, sur un front montant ou descendant ou sur un changement de valeur.

Broches 3,5,6,9,10 et 11 : peuvent être utilisées pour générer des signaux PWM (Pulse Width Modulation) via le logiciel en utilisant la fonction `analogwrite()`.

Broches 10,11,12 et 13 : peuvent être configurées et utilisées pour une communication SPI (Serial Peripheral Interface) tout en utilisant une librairie pour communication SPI.

Broches 4 et 5 : peuvent être configurées et utilisées pour supporter les communications de protocole I2C (Interface 2 fils) tout en utilisant la librairie `Wire/I2C`.

Broches 13 : la carte Arduino UNO dispose d'une LED intégrée reliée à la broche 13, lorsque cette broche est à l'état haut, la LED est allumée, si non elle éteinte.

Les broches analogiques de la carte Arduino UNO :

La carte UNO dispose de 6 broches analogiques numérotées de A0 à A5 pouvant fournir chacune une mesure d'une résolution de 10bits (10bits à cause du convertisseur analogique/numérique CAN qui donne des résultats sur 10bits avoir 1024 niveaux) à l'aide de la fonction `analogRead()`.

## **2.7 Partie programme :**

### **2.7.1 Description de l'environnement Arduino IDE :**

De plus en plus l'électronique est remplacée par l'électronique programmée ou ce qu'on appelle de l'électronique embarquée ou de l'informatique embarquée.

Comme toutes les machines, une carte Arduino nécessite un compilateur afin de traduire le langage ou bien le programme informatique en un langage machine (langage constitué d'un ensemble de ligne de 0 et de 1) compréhensible pour la machine mais non pas pour l'être humain. Le logiciel d'Arduino rend cette tâche possible et encore plus facile.

Un logiciel Arduino est Environnement de Développement Intégré (en anglais IDE : Integrated Development Environments) open source et gratuit téléchargeable sur le site officiel Arduino compatible avec tous les systèmes d'exploitation (Windows, MacOS et Linux) écrit en JAVA inspiré du langage Processing.

L'IDE Arduino permet d'éditer des programmes (appelés aussi des sketches) écrits en langage C/C++. Une fois le programme écrit ou modifié au clavier, l'IDE compile ce programme du langage C/C++ au langage machine de l'Arduino, lors de la compilation la console utilisée donne des informations sur l'enchaînement de la compilation et indique la présence d'erreur au programme afin de les rectifier. Revenant à notre tâche qui est la programmation de la carte Arduino, donc le programme après avoir été compilé, doit être

---

transmit vers la carte, on appelle cela le téléversement du programme de la console vers la mémoire de la carte, cette option est assurée par l'IDE. Le processus du téléversement (upload) est réalisé grâce à la connexion USB de la carte Arduino qui la relie à l'ordinateur et permet au logiciel de s'installer dans la mémoire de cette carte, une fois dans la mémoire de la carte le logiciel téléversé est appelé micrologiciel. Au cours de cette étape, la console donne des informations sur le déroulement tout en indiquant les messages d'erreurs. L'IDE permet également de communiquer avec la carte Arduino grâce au moniteur série de la carte, de telle façon que si la connexion entre la carte et la console est active (câble USB), il peut communiquer avec l'ordinateur pendant le fonctionnement du programme en mémoire d'Arduino.

## 2.7.2 Interface du logiciel :

Pour la première vue, l'interface du logiciel Arduino apparait trop moins compliqué comparant à d'autres logiciel (tel que Proteus, Matlab).

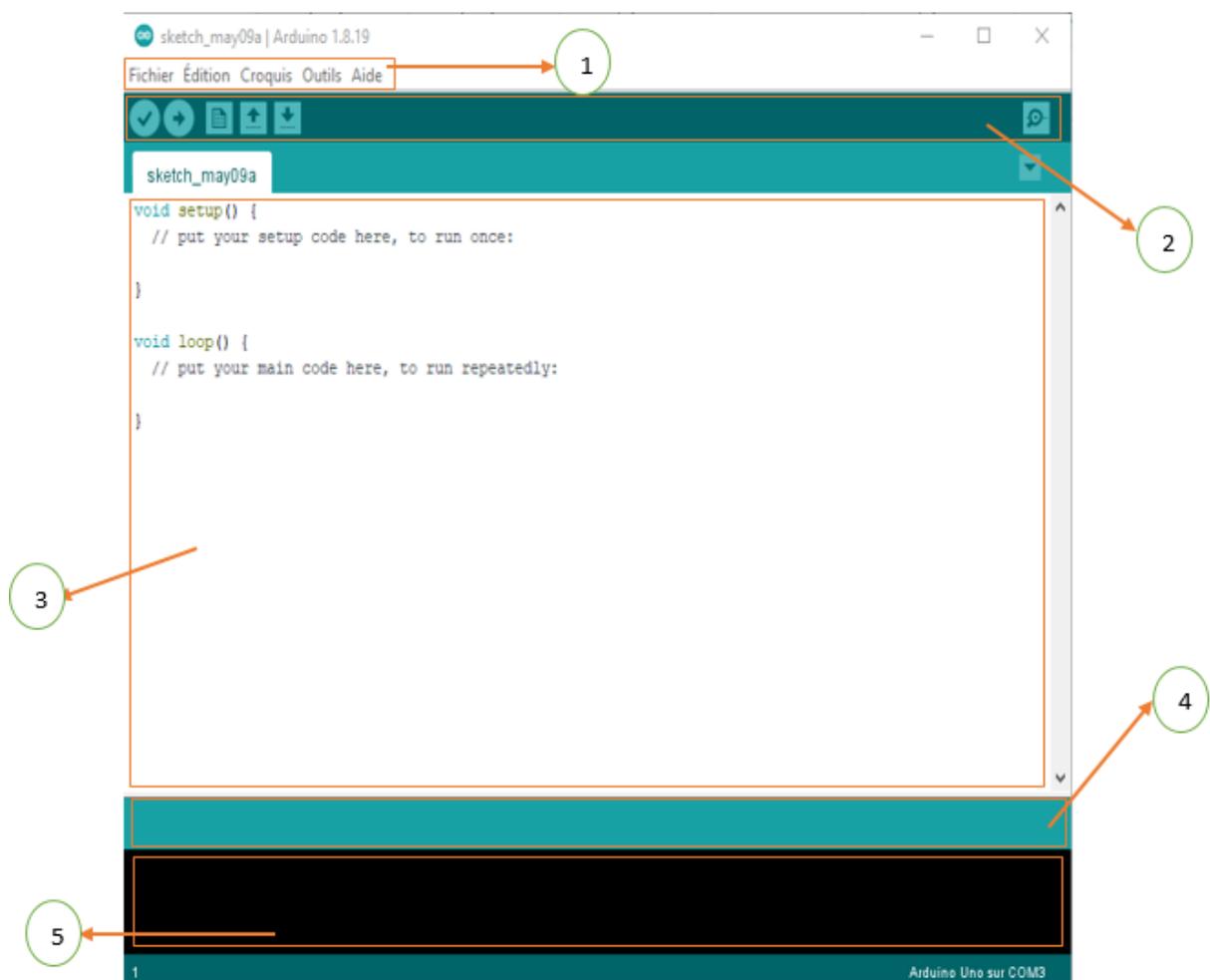


Figure 26 : interface du logiciel Arduino IDE

1	Barre de menu.
2	Barre des boutons.
3	Fenêtre de création et d'édition de programme.
4	Zone de messages d'actions en cours de réalisation.
5	Débogueur (affichage des erreurs de programmation).

Tableau 5 : les différentes zones du logiciel Arduino IDE

Donnant plus de détails sur les icônes de chaque boutons figurant sur la barre de boutons :

-  Permet de vérifier le programme tout en cherchant et indiquant les erreurs apparues.
-  Permet de téléverser le programme écrit de l'ordinateur vers la carte Arduino.
-  Permet de créer un nouveau fichier.
-  Permet d'ouvrir un fichier existant.
-  Permet de sauvegarder le programme en cours.
-  Moniteur série.

#### Arduino Uno sur COM3

Avant de téléverser le programme vers la carte Arduino, il faut bien configurer le type de la carte Arduino utilisée ainsi que le numéro du port USB.

### 2.7.3 Structure d'un programme écrit avec le logiciel Arduino IDE :

Après avoir compiler et téléverser le programme vers la carte, le microcontrôleur de la carte Arduino lit et exécute le programme de haut vers le bas. Il est nécessaire de déclarer toute variable qui sera utilisée par la suite dans une fonction au programme.

Un programme écrit avec le logiciel Arduino IDE a une structure minimale constituée de trois parties comme suit :

- Une entête : sert à déclarer tout type de variables, constantes, indication d'utilisation de toute sorte de bibliothèque.
-

- Void Setup : (initialisation) cette partie du programme n'est exécutée qu'une seule fois, elle comprend les fonctions qui devront être exécutées avant au démarrage (par exemple la configuration des broches soit comme entrées ou comme sorties).
- Void Loop : (boucle) cette partie est lue et exécutée en boucle, elle comprend les fonctions réalisables en boucle.

On peut bien ajouter en plus de cette structure minimale :

- Des sous-programmes qui peuvent être appelés n'importe quel moment dans la boucle, comme ça présente aussi un avantage pour réaliser des parties de code répétitives.
- Des rappels "callback" qui sont des fonctions rappelées automatiquement depuis une bibliothèque.

The image shows a screenshot of the Arduino IDE interface. The main window displays a C++ sketch for an Arduino Uno. The sketch is divided into three sections, each highlighted with an orange box and labeled with a blue box:

- En-tete**: This section contains the variable declarations:
 

```
int trig = 7;
int echo = 8;
int LED = 2;
long duration;
long distance;
```
- Initialisation**: This section contains the `void setup()` function:
 

```
void setup() {
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(LED, OUTPUT);
  Serial.begin(9600);
}
```
- Boucle**: This section contains the `void loop()` function:
 

```
void loop() {
  digitalWrite(trig, LOW);
  delay(2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  duration = pulseIn(echo, HIGH);
  distance = duration/58;
  Serial.print("distance = ");
  Serial.print(distance);
  delay(1000);
  if(distance < 20) {
    digitalWrite(LED, HIGH);
  } else { digitalWrite(LED, LOW);
  }
}
```

The IDE window title is "sketch\_mar13c | Arduino 1.8.19". The menu bar includes "Fichier", "Édition", "Croquis", "Outils", and "Aide". The status bar at the bottom indicates "1" and "Arduino Uno sur COM3".

Figure 27 : la structure d'un programme écrit sur Arduino IDE

## 2.8 Module BLUETOOTH HC-06 :

### 2.8.1 Qu'est-ce que BLUETOOTH :

BLUETOOTH est un standard de communication assurant une communication bidirectionnelle entre plusieurs périphériques en transférant et envoyant des fichiers et des données sur une très courte distance, fonctionnant sur une plage de fréquence comprise entre 2.4 GHz et 2.483 GHz. Son principal avantage se présente dans le fait de pouvoir de réaliser une connexion entre deux appareils sans avoir à faire aucune liaison filaire.

### 2.8.2 Présentation du module BLUETOOTH HC-06 :

Le module BUETOOTH HC-06 est un dispositif permettant d'établir une connexion entre la carte Arduino et un autre périphérique (n'importe quel appareil doté d'une option de connexion BLUETOOTH). Ce module peut être configuré au mode esclave contrairement au module BLUETOOTH HC-05 qui est configuré au mode maitre, la différence entre ces deux modules se présentent dans le fait qu'un module maitre peut demander à un autre appareil BLUETOOTH de s'appairer avec lui, tandis qu'un module esclave ne peut pas effectuer une demande d'appairage avec un autre module ou appareil disposant d'une connexion BLUETOOTH, il ne peut que recevoir des demandes d'appairage.

### 2.8.3 Brochage du module BLUETOOTH HC-06 :

RX	Broche de réception : connectée au pin de transmission "TX" de la carte Arduino.
TX	Broche de transmission : connectée au pin de réception "RX" de la carte Arduino.
GND	Masse : connectée à la broche GND de la carte Arduino.
Vcc (+5V)	Broche d'alimentation : connectée à la broche +5V de la carte Arduino.

Tableau 6 : brochage du module BLUETOOTH HC-06

---

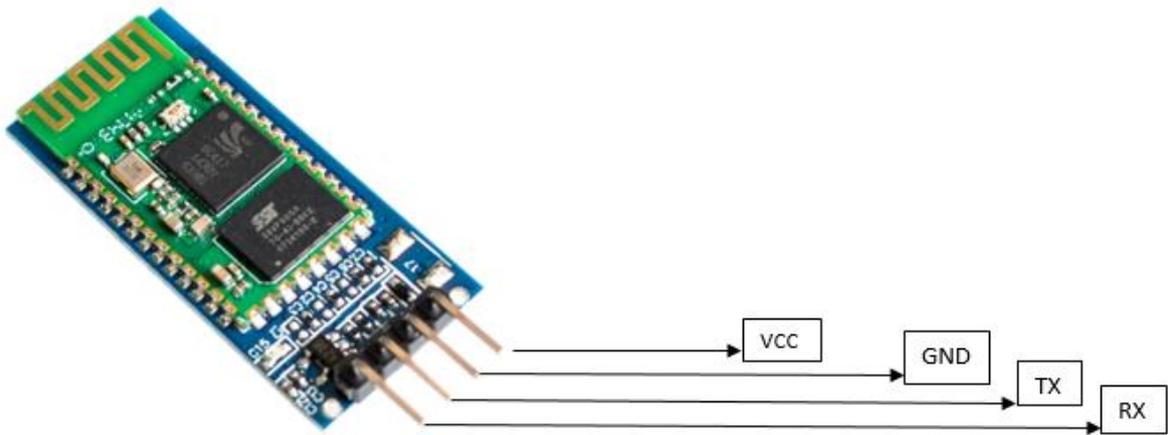


Figure 28 : brochage du module BLUETOOTH HC-06

## 2.9 Moteur à courant continu CC :

### 2.9.1 Présentation :

Un moteur à courant continu est un dispositif assurant la conversion de l'énergie électromagnétique (il permet de convertir de l'énergie électrique en mouvement). Le rôle majeur d'un moteur à courant continu peut apparaître dans la conversion d'énergie électrique en énergie mécanique, mais ce dernier peut assurer la conversion d'une énergie mécanique à une énergie électrique en jouant le rôle d'un générateur d'électricité comme c'est le cas pour la dynamo.

La taille et la puissance des moteurs à courant continu peut varier des simples moteurs alimentant des jouets à des mécanismes qui alimentent les véhicules et les ascenseurs.



Figure 29 : moteur à courant continu

Un moteur à courant continu est constitué essentiellement de deux parties majeures qui sont le stator qui constitue la partie statique ou fixe et le rotor qui constitue à son rôle la partie tournante.

Le stator : c'est la partie statique du moteur, elle est constituée des aimants permanents assurant un champ d'induction (magnétique) nécessaire pour la rotation de la partie rotative et c'est pourquoi on l'appelle aussi inducteur.

Le rotor : c'est la partie tournante du moteur qui subit l'induction du stator et c'est pourquoi on l'appelle aussi l'induit.

## 2.9.2 Principe de fonctionnement d'un moteur à courant continu :

Un moteur à courant continu suit un principe fondamental afin de produire un mouvement. Il fonctionne grâce aux propriétés des aimants permanents et au champs magnétiques générés lorsqu'un courant électrique traverse un conducteur.

Le stator qui est la partie statique, est équipé d'aimants permanents qui génèrent à leur tour un champ magnétique traversant le rotor, dans ce rotor qui constitue la partie tournante du moteur, un courant continu circule d'avant en arrière. La circulation d'un courant continu dans un conducteur placé au milieu d'un champ magnétique produit une sorte de force qui est la force électromotrice générée dans le conducteur ce qui le fait tourner.

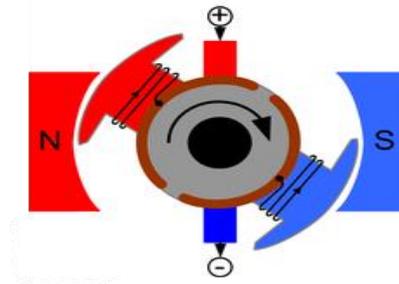


Figure 30 : principe de fonctionnement d'un moteur à courant continu

## 2.10 Le driver moteur L298N :

### 2.10.1 Présentation du driver :

Le shield L298N est une carte électronique compatible avec la carte Arduino UNO permettant de commander des moteurs pas à pas ou des moteurs à courant continu tout en donnant la possibilité de contrôler leur vitesse ainsi que leur sens de rotation.

Ce module est basé sur la puce L298N de STMicroelectronics et qui est un circuit intégré contenant essentiellement :

- Deux ponts en H permettant chacun de contrôler un moteur à courant continu dans un sens ou dans l'autre.
- Une combinaison logique de commande qui se constitue des broches d'activation des ponts (Ena ET Enb) qui permettent le lancement ou l'arrêt des moteurs en les alimentant en tout ou rien ou bien les utiliser pour commander la vitesse de rotation des moteurs en les alimentant par des entrées PWM de la carte Arduino. Des broches de sélection de ponts (IN1, IN2, IN3 et IN4) permettant à leur tour de commander le sens de rotation des moteurs.

Pour bien avoir un contrôle sur la vitesse des moteurs les broches d'activation des ponts devraient être pilotés via un signal à rapport cyclique variable (PWM), tel que la vitesse d'un moteur à courant continu est proportionnelle à sa tension moyenne. Donc il suffit juste de faire varier la tension d'alimentation moyenne du moteur et cela est fait via une entrée de modulation en largeur d'impulsion (PWM).

---

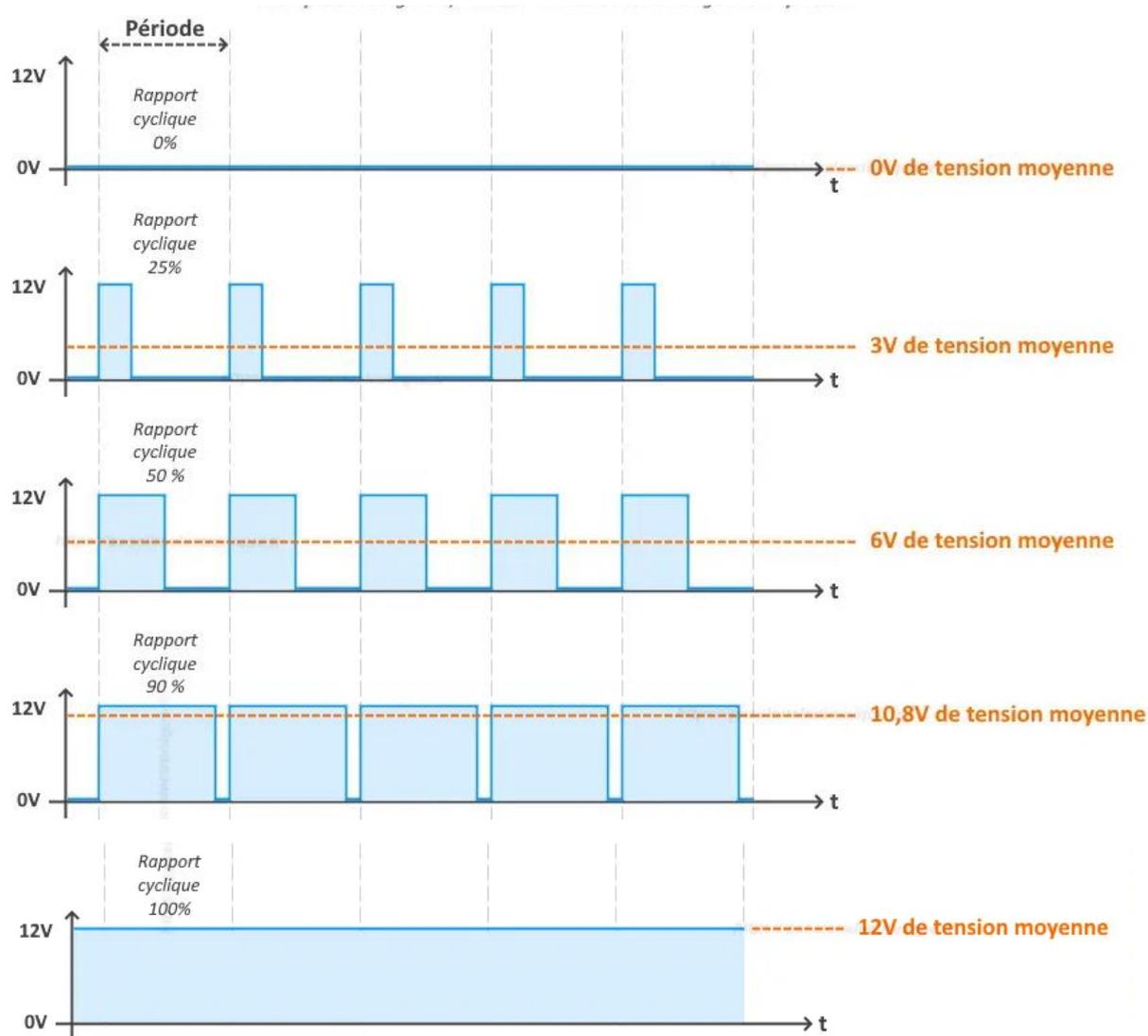


Figure 31 : signal PWM

Le tableau suivant résume la manière dont les broches de sélection de ponts devraient être configurés pour commander le sens des moteurs :

ENA	IN1	IN2	Résultat en sortie.
ENB	IN3	IN4	Résultat en sortie.
L	X	X	Moteur en roue libre (à l'arrêt sans frein).
H	L	L	Blocage de moteur (arrêt rapide, freinage fort).
H	L	H	Marche arrière.
H	H	L	Marche avant.
H	H	H	Blocage du moteur (arrêt rapide, freinage fort).

Tableau 7 : la configuration des broches du driver L298N

### 2.10.2 Brochage du circuit intégré L298N :

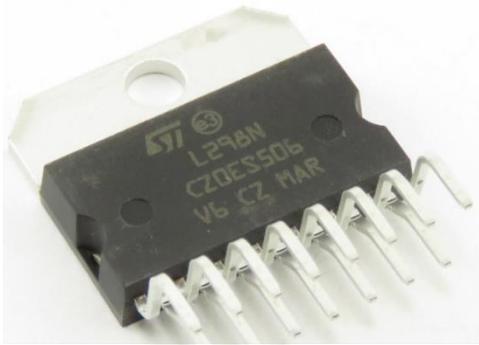


Figure 32 : circuit intégré L298N

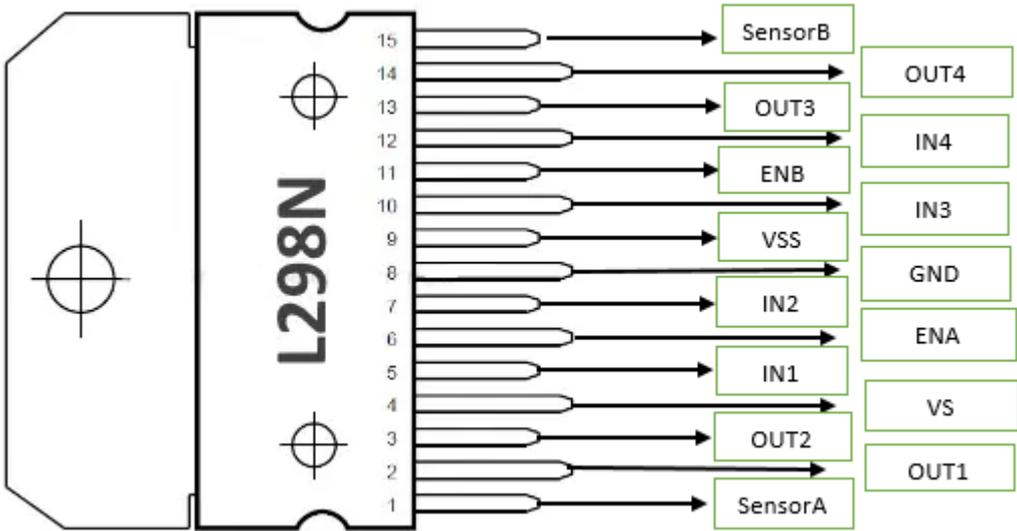


Figure 33 : brochage du circuit intégré L298N

Broches	Signification
SensorA	Capteur de courant pont A.
OUT1	Sortie 1 (pont A).
OUT2	Sortie 2 (pont A).
VS	Alimentation partie puissance.
IN1	Entrée 1 (pont A).
ENA	Activation/désactivation (pont A).
IN2	Entrée 2 (pontA).
GND	Masse.
VSS	Alimentation partie commande.
IN3	Entrée 3 (pont B).
ENB	Activation/désactivation (pont B).
IN4	Entrée 4 (pont B).
OUT3	Sortie 3 (pont B).
OUT4	Sortie 4 (pont B).
SensorB	Capteur de courant pont B.

Tableau 8 : les différentes broches du circuit intégré L298N

### 2.10.3 Brochage du driver L298N :

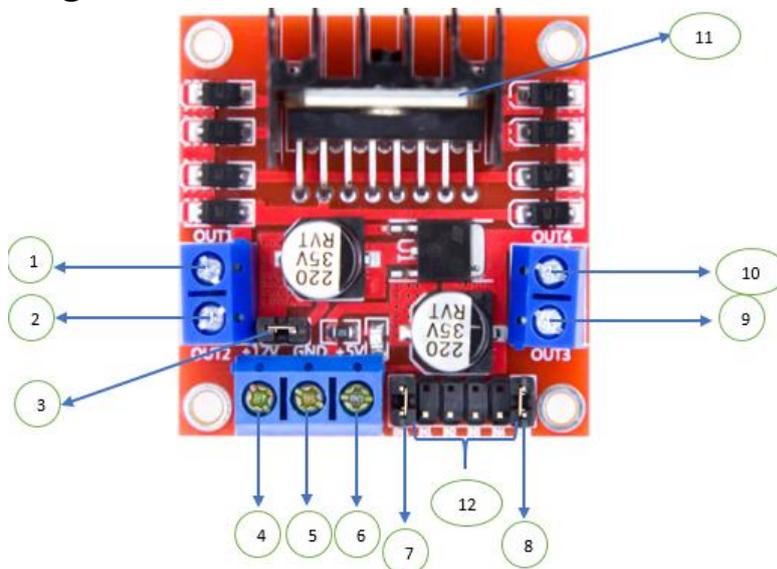


Figure 34 : brochage du driver moteur L298N

Broches	Spécification
1	Borne positive du moteur 1.
2	Borne négative du moteur 2.
3	Cavalier régulateur.
4	Entrée d'alimentation de 5V à 35V.
5	Masse.
6	Sortie pour alimenter l'Arduino à 5V.
7	Cavalier d'activation/désactivation du moteur 1.
8	Cavalier d'activation/désactivation du moteur 2.
9	Borne négative du moteur 2.
10	Borne positive du moteur 2.
11	Circuit intégré L298N (double pont en H).
12	Entrées de commande du sens des moteurs 1 et 2.

---

Tableau 9 : brochage du driver moteur L298N

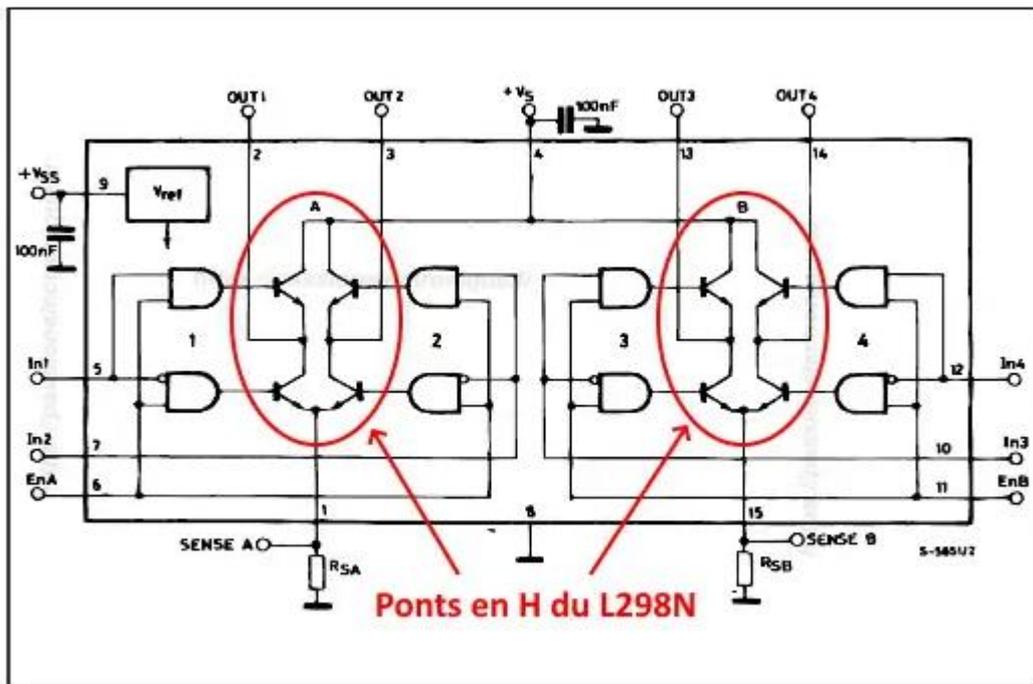


Figure 35 : architecture interne du circuit intégré L298N

## 2.11 Le pont en H :

### 2.11.1 Présentation d'un pont en H :

Un pont en H est un circuit électronique de puissance en forme de H ce qui explique d'où vient l'inspiration de ce nom, destiné essentiellement au contrôle et au pilotage d'un moteur à courant continu (sens de rotation ainsi que la vitesse de rotation) et cela en permettant à la tension de circuler dans les deux sens sur une carte. Le sens de rotation d'un moteur est contrôlé par le sens du courant qui parcourt la carte du pont en H, et pour le contrôle du sens de courant, le pont en H met en œuvre quatre transistors montés de telle façon que le courant puisse passer dans un sens bien précis.

D'une manière brève un pont en H permet :

- Fournir le courant nécessaire pour le fonctionnement du moteur.
- Donner la possibilité d'inverser le sens du courant et donc le sens de rotation du moteur ainsi que la possibilité de pouvoir varier la vitesse de rotation du moteur ou bien ce qu'on appelle une modulation en PWM.

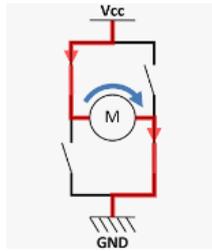


Figure 36 : circuit électrique d'un pont en H

## 2.12 Capteur ultrasonique :

Un capteur à ultrason est un dispositif électronique utilisé essentiellement pour détecter la présence d'un objet et mesurer la distance qui le sépare du capteur, tout en utilisant des ondes sonores supérieures à 20 000 MHz, c'est-à-dire au-delà des capacités de l'oreille humaine, et ensuite les convertir en un signal électrique compréhensible par la machine.



Figure 37 : capteur ultrasonique

### 2.12.1 Principe de fonctionnement :

Le principe suivi par un capteur à ultrason est simple, ce processus se résume en l'envoi d'une onde sonore à une fréquence supérieure à la plage d'audition humaine. Pour être claire, un capteur à ultrason est constitué de parties principales :

Un émetteur : qui émet une onde sonore à l'aide de cristaux piézoélectriques.

Un récepteur : qui reçoit l'onde après qu'elle a voyagé vers et depuis la cible ou l'obstacle.

Le capteur ultrason envoie des ultrasons à partir de son émetteur en attendant qu'elle se rejette sur un objet et revient vers le récepteur du capteur qui agit comme un microphone pour recevoir le son ultrasonique.

Afin de calculer la distance qui sépare entre l'objet présent et le capteur, le capteur mesure le temps nécessaire entre l'émission du son et son contact avec le récepteur. Le calcul de cette distance se résume par l'équation suivante :

$$D = \frac{1}{2} \times T \times C$$

D'où

---

D : est la distance.

T : est le temps.

C : est la vitesse du son dans l'air qui est approximative à 343 m/s.

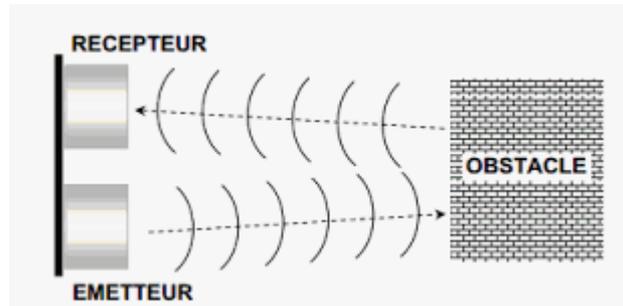


Figure 38 : principe de fonctionnement d'un capteur ultrasonique

### 2.12.2 Brochage du capteur ultrasonique :

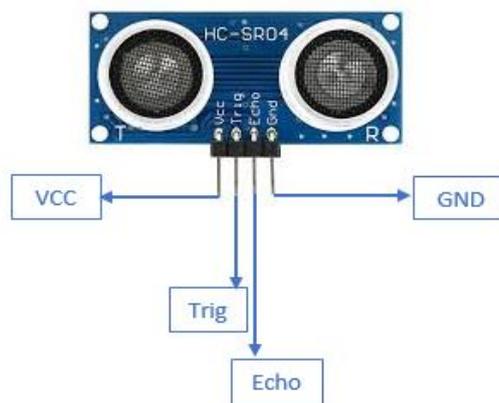


Figure 39 : brochage du capteur ultrasonique

VCC	Cette broche présente la broche d'alimentation +5V du capteur.
Trig (Trigger)	Présente une broche d'entrée utilisée pour initialiser la mesure en transmettant des ondes ultrasonores tout en gardant cette broche à 1 pendant 10 $\mu$ s.
Echo	Présente une broche de sortie qui passe au niveau haut pendant le temps qui équivaut la durée pendant laquelle l'onde revient au capteur.
GND	Masse 0V.

Tableau 10 : brochage du capteur ultrasonique

### 2.12.3 Caractéristique du capteur ultrasonique :

Dimensions	45 × 20 × 15 mm
Tension d'alimentation	5V (CC.
Courant d'alimentation	15Ma
Fréquence de modulation	40Hz
Portée maximale	4m
Portée minimale	2cm
Précision	3cm
Angle de mesure efficace	15°
Sortie	0 – 5V
Signal de déclenchement d'impulsion TTL	10µs
Fréquence de fonctionnement	40KHz

Tableau 11 : caractéristiques du capteur ultrasonique

### 2.13 Conclusion :

Dans ce chapitre nous avons abordé les trois parties principales constituant notre projet qui se présente à la réalisation d'un robot mobile, ces parties se résument en : partie mécanique, partie électronique et partie informatique, tout en étudiant et en mettant la lumière sur les différents composants électroniques les plus connus au domaine de la robotique mobile ainsi que leurs mode et principe de fonctionnement et les outils utilisés pour la réalisation du projet.

---

**Chapitre 3 : Mise à  
l'éassaie du robot  
mobile réalisé.**

---

### **3.1 Introduction :**

Aujourd'hui, et avec toutes les évolutions et les innovations de la robotique, cette dernière s'est imposée dans quasiment tous les domaines et tous les secteurs. Parmi les domaines où la robotique s'est fortement imposée, on trouve le domaine industriel, ce domaine qui de plus en plus exige de nouvelles technologies et les applications les plus innovantes et pratiques au-delà des applications classiques de la robotique dans le domaine industriel. Partant de cette ligne de réflexion, l'idée de voir un robot mobile assistant à côté des ouvriers n'est plus impossible et même elle est devenue une exigence aux grands usines et chaînes de production, l'intégration de ce type de robots dans les différentes tâches industrielles est tolérable à cause des services qu'ils donnent et leur utilisation qui est simple et pratique ainsi que leur autonomie ce qui souligne leur plus grand avantage.

### **3.1 Fonctionnement général du robot :**

Le robot réalisé dans ce projet fait part des robots mobile à roues, plus précisément les robots mobile voiture (quatre roues motorisées), il est dédié aux applications de la manutention dans les milieux industriels en faisant part de la procédure du déplacement des marchandises ou bien dans le cas général des charges et cela en les déplaçant à l'endroit envisageable en suivant une trajectoire pratique et bien précise.

Pour se faire, le robot est équipé d'une carte de commande à base d'Arduino UNO dotée d'un module BLUETOOTH ce qui rend le robot autonome grâce au programme inscrit dans la carte de commande et pour toute sorte de défaillance, le module BLUETOOTH donne la possibilité de pouvoir commander le robot à distance grâce à une application mobile. En résumant le principe de fonctionnement du robot mobile, la fameuse question qui se pose est : lorsqu'on est dans un milieu industriel, il est fort possible de voir des matériaux utilisés lors de la fabrication misent un peu partout sur le sol ce qui présentera au robot un obstacle, alors comment faire pour régler ce problème ? ce problème n'est qu'une petite tâche constituante le quotidien du robot, grâce au capteur ultrasonique qui sera le robot équipé avec, ce capteur aidera le robot à détecter et éviter les obstacles présent sur son chemin et cela sans avoir besoin du moindre geste humain pour le faire sortir de cette situation.

---

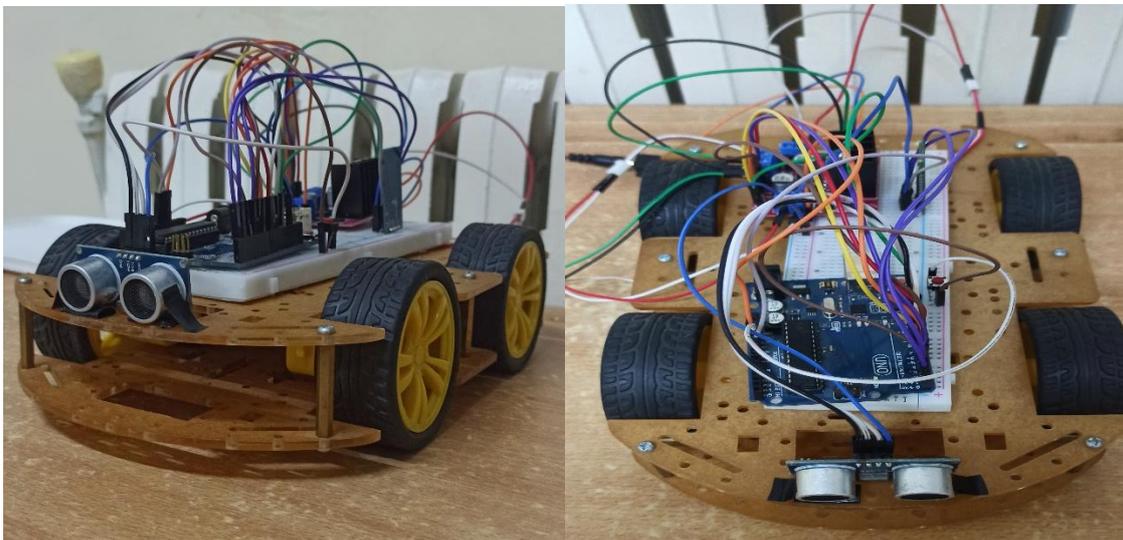
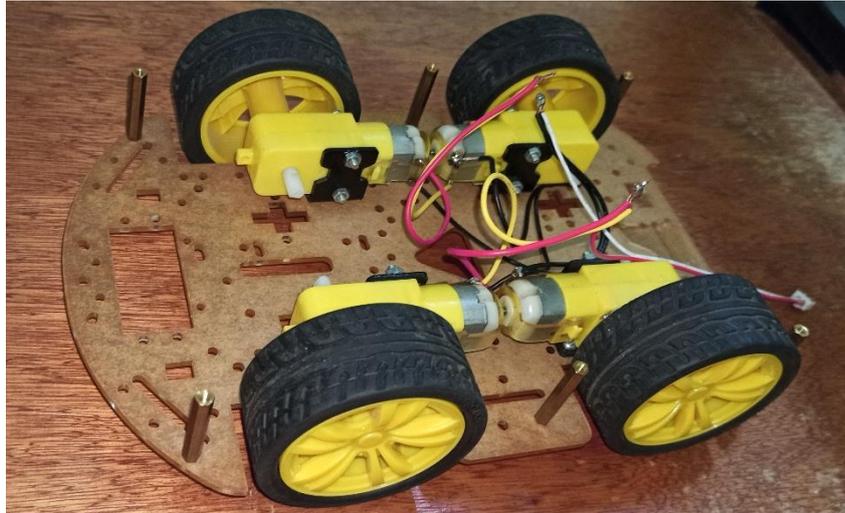
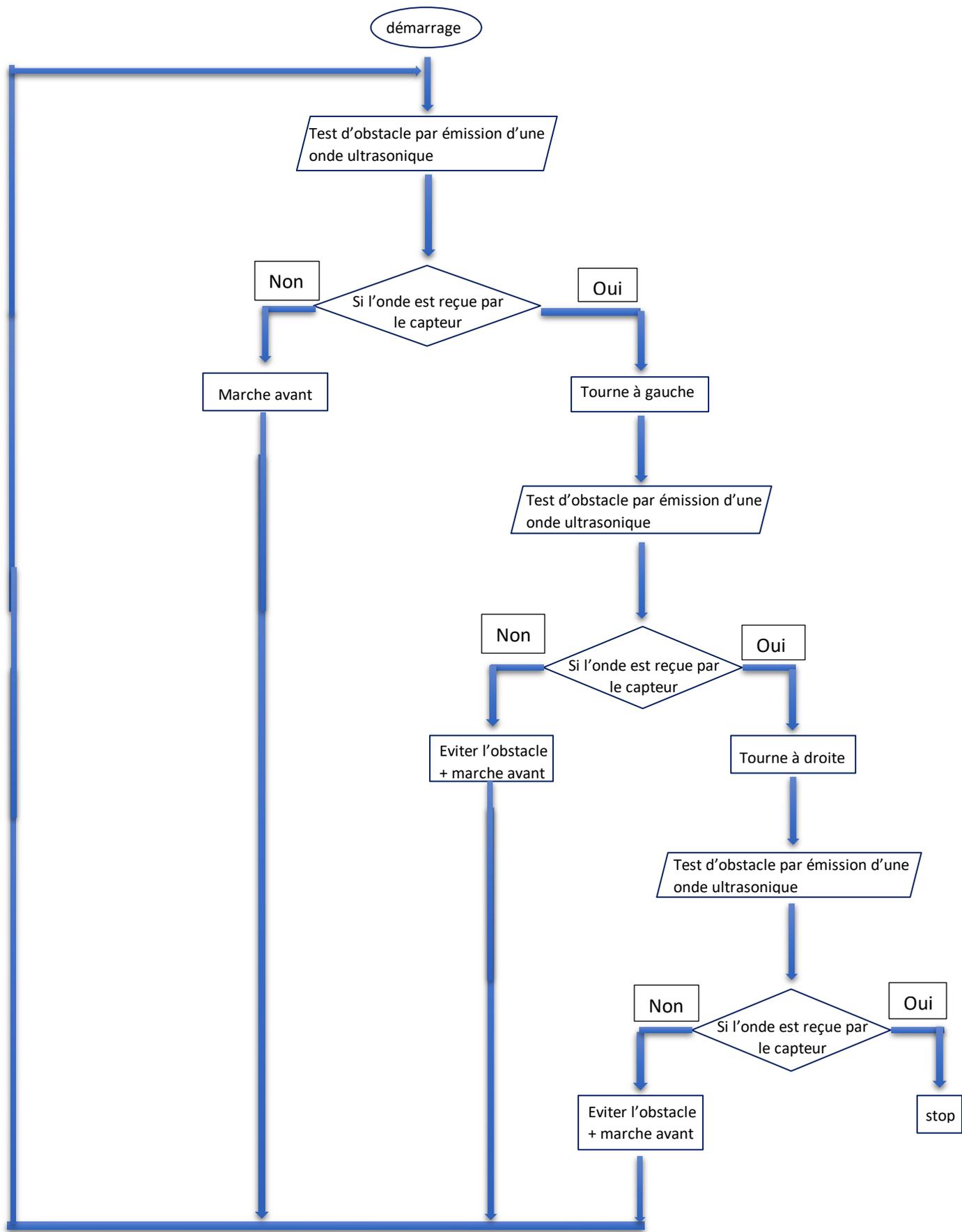


Figure 40 :images réelles du robot réalisé

### 3.1.1 Organigramme du fonctionnement du robot :

L'organigramme suivant donne un aperçu sur le principe de fonctionnement du robot.



### 3.2 Aperçu historique sur Android :

Android, le nom d'une entreprise américaine dont son nom est choisi par elle-même, il est basé sur le concept d'androïde qui est un robot à l'apparence humaine. L'entreprise Android fut créée en 2003 par la société technologique américaine Android Inc, afin de développer des systèmes opérateurs pour des cameras digitale. Ensuite devenu un opérateur système pour les smartphones durant l'année qui succède sa date de création. En 2005, Android Inc est rachetée par la société américaine de moteur de recherche GOOGLE Inc qui a basé le projet Android sur Linux qui est un opérateur système open source pour les ordinateurs personnels. En novembre 2007, Google a annoncé la création du consortium Open Alliance Handest qui a regroupé une dizaine de sociétés technologiques et de téléphonie mobile (Intel, LG, Samsung, T-Mobile, Nvidia, Motorola...). Ce consortium avait été créé afin de pouvoir développer et promouvoir Android comme un opérateur système Open Source et gratuit avec un support pour les applications tierces. Google assure un développement constant par des mises à jour du système Android résumées dans le tableau suivant :

Version d'Android	Date de lancement
Android 1.0	2008
Android Cupcake 1.5	2009
Android Eclair 2.0	2009
Android Froyo 2.2	2010
Android Gingerbread 2.3	2011
Android Honeycomb 3.0	2011
Android Ice Cream Sandwich 4.0	2011
Android Jelly Bean 4.1	2012
Android KitKat 4.4	2013
Android Lollipop 5.0	2014
Android Marshmallow 6.0	2015
Android nougat 7.0	2016
Android Oreo 8.0	2017
Android Pie 9.0	2018
Android 10 Q	2019
Android 11	2020
Android 12	2021

Tableau 12 : les différentes version d'Android



### 3.2.1 Présentation d'Android :

Android est un projet open source du nom d'AOSP (Android Open Source Project) qui est ensuite utilisé par Google pour développer Android, et réutilisé par les autres constructeurs de téléphones mobile. Android est le système d'exploitation équipant la majorité des téléphones portables du moment, il offre également la possibilité de pouvoir personnaliser le téléphone en téléchargeant des applications diverses (navigateur, maps, GPS, réseaux sociaux...).

Ce système d'exploitation donne vie aux appareils, il permet de parler aux composants et permet également aux utilisateurs de lui parler à travers une interface graphique. Android est donc présent sur des smartphones, smart TV, smart watch, autoradio, la domotique... en fait c'est un système très ouvert et flexible, techniquement, il est basé sur des éléments open source "Linux" ainsi que des composants propres à Google.

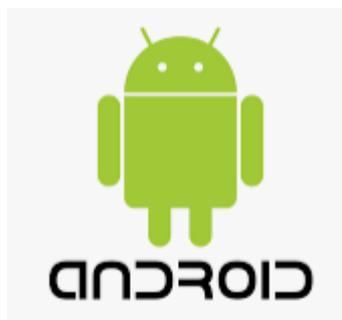


Figure 41 : logo d'Android

### 3.3 MIT app inventor :

App Inventor est un logiciel de développement en ligne, gratuit et open source permettant la conception et le développement des applications mobile pour smartphones et les tablettes Android.

Créé et développé par Google et entretenu et géré par le Massachusetts Institute of Technology (MIT). App Inventor est une plateforme de développement pour Android (disponible aussi pour iOS mais en version BETA "elle est encore sous-développement"), écrite avec JAVA, SCHEME et KAWA, disponible en 19 langues différentes.

App Inventor offre une interface graphique de développement basée sur le pouvoir de faire glisser et déposer des objets visuels pour créer des applications pouvant être téléchargeable et exécutable, il est également basé sur l'assemblage et la manipulation des blocs instructifs pour la programmation de l'application voulue ce qui est similaire à d'autres langages de programmation tel que Scratch et Starlogo.

### 3.3.1 Création d'une application mobile en utilisant le logiciel MIT App Inventor :

Comme il est déjà mentionné, le logiciel App Inventor donne la possibilité de pouvoir créer un application mobile Android téléchargeable et partageable avec d'autres appareils Android. Dans ce qui suit nous citons les étapes indispensables pour la création d'une application mobile avec ce logiciel tout en illustrant ses parties essentielles :

Tout d'abord en cherchant MIT App Inventor sur internet nous nous trouvons sur ce site présenté dans la figure ci-dessous :

Utilisateurs actifs aujourd'hui :	Utilisateurs actifs cette semaine :	Utilisateurs actifs ce mois-ci :	Utilisateurs enregistrés :	Pays :	Applications créées :
87,7 000	401 100	1,1 million	14,9 millions	195	67,8 millions

Figure 42 : interface du site web de MIT App Inventor

Pour pouvoir entamer le travail et commencer à réaliser une application, la création d'un compte est indispensable afin de pouvoir créer et développer des projets en toute sécurité et aussi pouvoir sauvegarder le travail réalisé et le consulter à n'importe quel moment et via n'importe quel appareil.

Créer des applications !

Sur    Educateurs    Nouvelles    Ressources    Blogues    Donner    ENHANCE    🔍

## Tout le monde peut créer des applications Android et iPhone avec un impact mondial

Apprendre encore plus

<b>Utilisateurs actifs aujourd'hui :</b>	<b>Utilisateurs actifs cette semaine :</b>	<b>Utilisateurs actifs ce mois-ci :</b>	<b>Utilisateurs enregistrés :</b>	<b>Pays :</b>	<b>Applications créées :</b>
87,7 000	401 100	1,1 million	14,9 millions	195	67,8 millions

MIT Master Trainer

Devenir maître formateur du MIT en informatique mobile éducative

Le

Figure 43 : création d'une application avec MIT App Inventor

Afin de pouvoir créer votre projet il est nécessaire de vous connecter en utilisant votre compte Gmail.

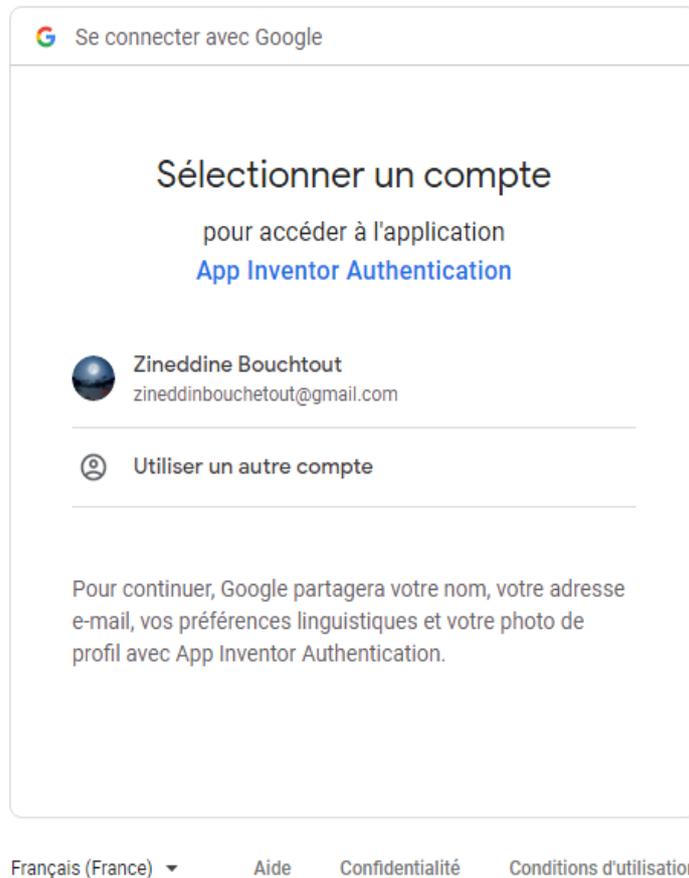


Figure 44 : ouvrir un compte au site MIT App Inventor

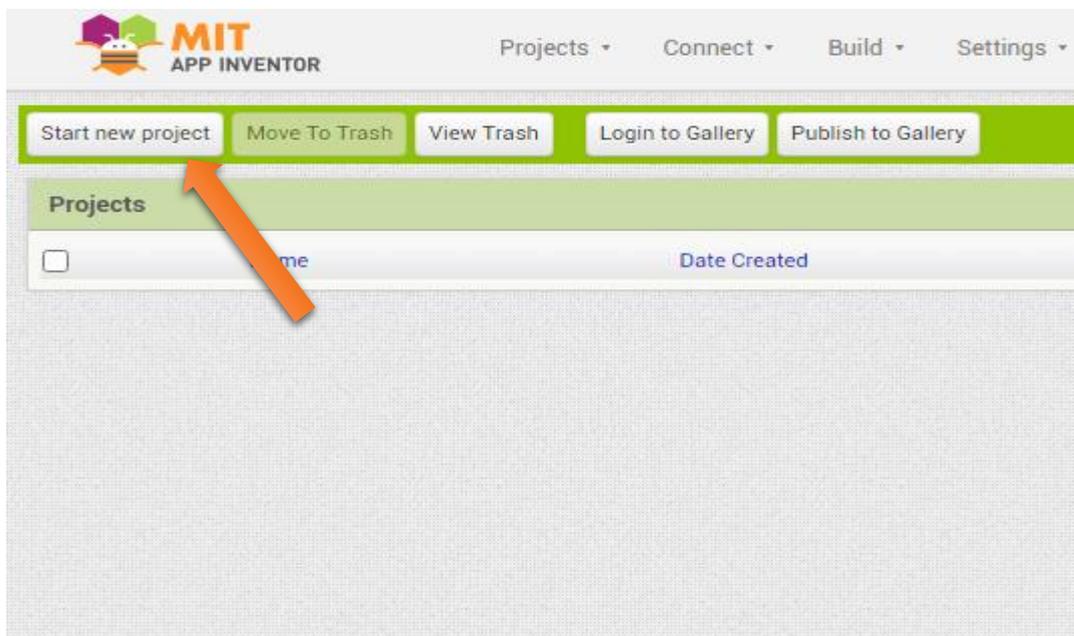


Figure 45 : commencer un nouveau projet

Donner un nom au projet.

---

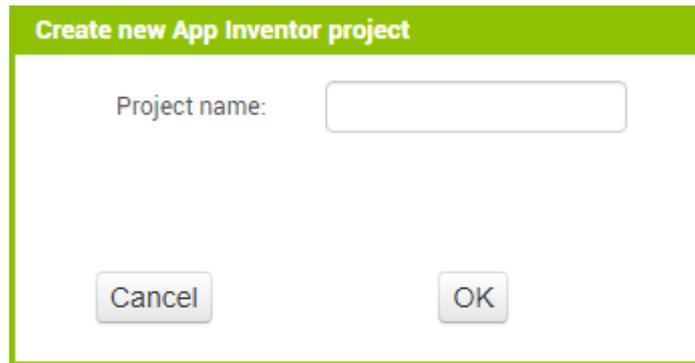


Figure 46 :donner un nom au projet créé

Après avoir nommé le projet on trouve l'interface du logiciel présentée comme il est illustré dans la figure ci-dessous.

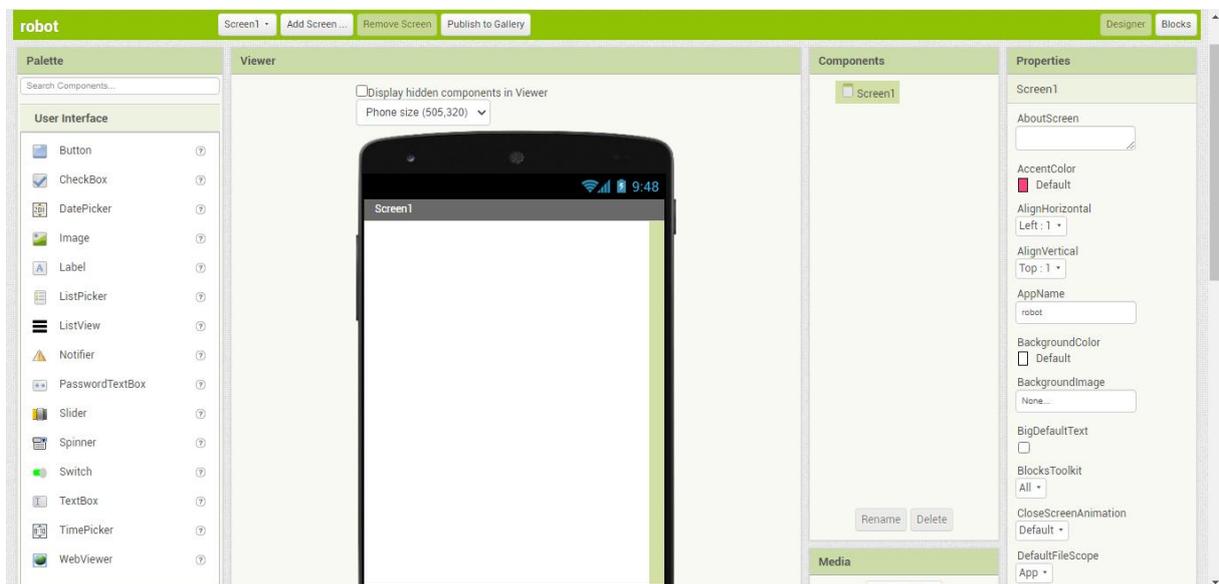


Figure 47 : iteface de travail

Le logiciel App Inventor se constitue essentiellement d'une partie "design" et d'une partie "blocs".

La partie design représentée dans la figure ci-dessous :

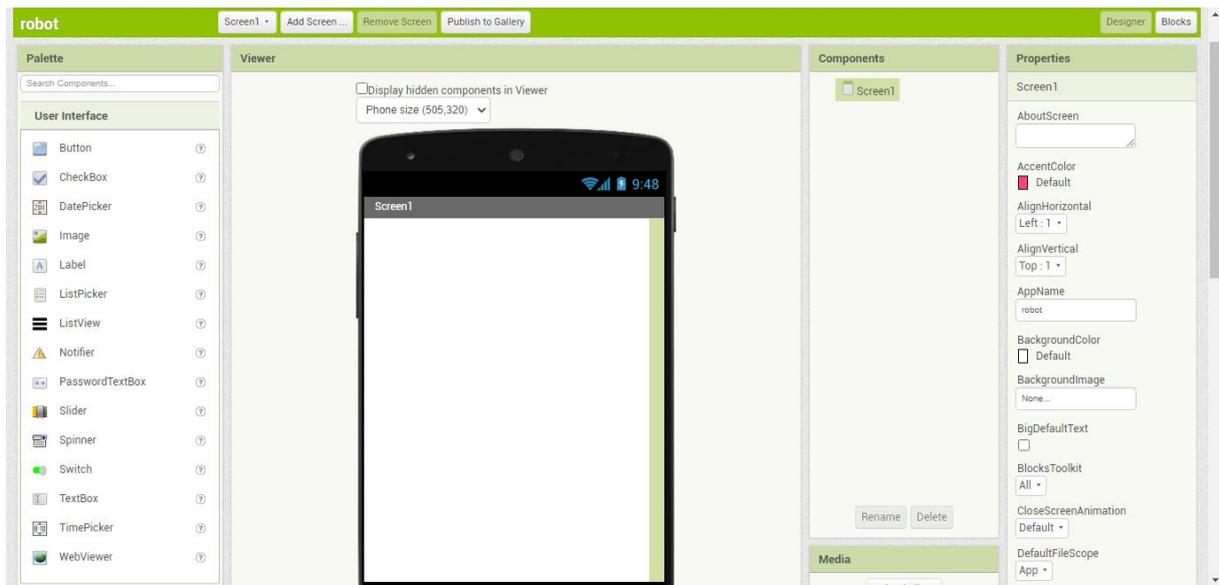


Figure 48 : partie design

La partie blocs est présentée dans la figure ci-dessous.

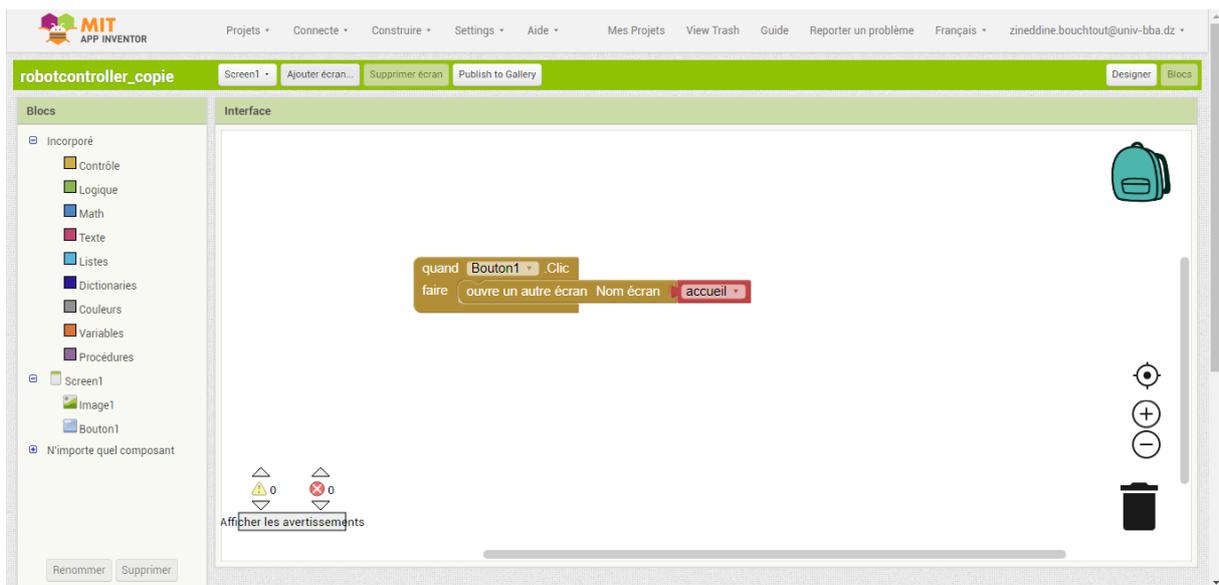


Figure 49 : partie blocs

### 3.3.2 Présentation de la partie design :

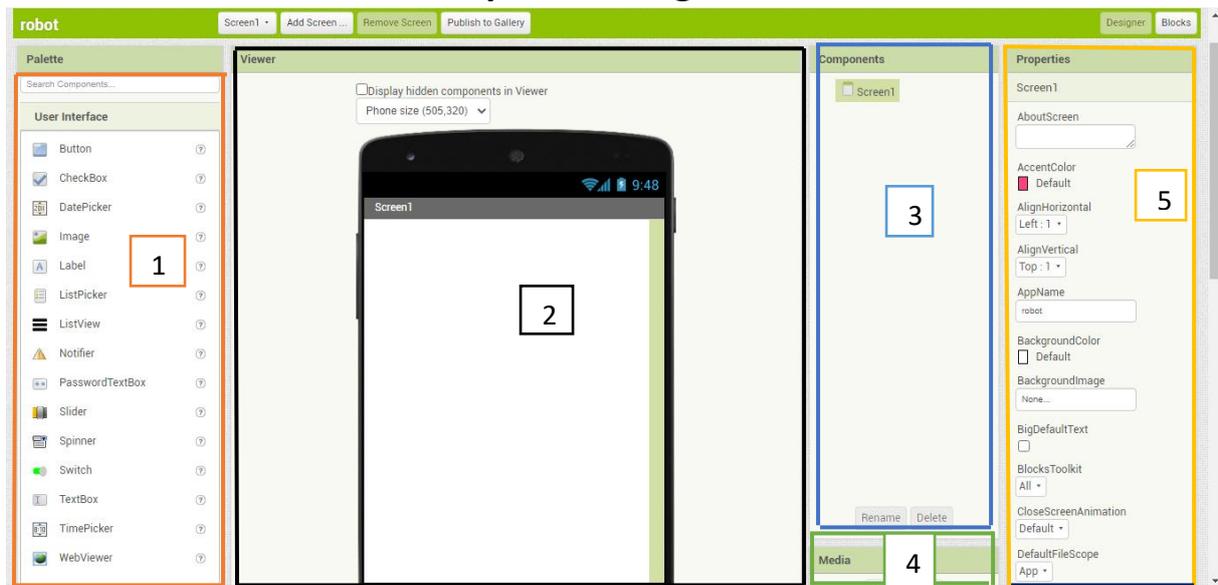


Figure 50 : contenu de la partie design

Cadre 1 : contient l'ensemble des objets glissable utilisés pour la conception de l'application.

Cadre 2 : représente la zone de travail, elle contient ce qui sera affiché sur l'écran du smartphone.

Cadre3 : cette zone contient une liste des objets glissés vers la zone de travail, elle permet également de sélectionner un objet parmi les objets présent au lieu de travail.

Cadre 4 : cette zone contient l'ensemble des images téléchargées et utilisées pour la personnalisation des objets glissés.

Cadre 5 : cette zone est réservée à la personnalisation des objets glissés (texte, couleur, couleur de fond, image, police...).

### 3.3.3 Présentation de la partie blocs :

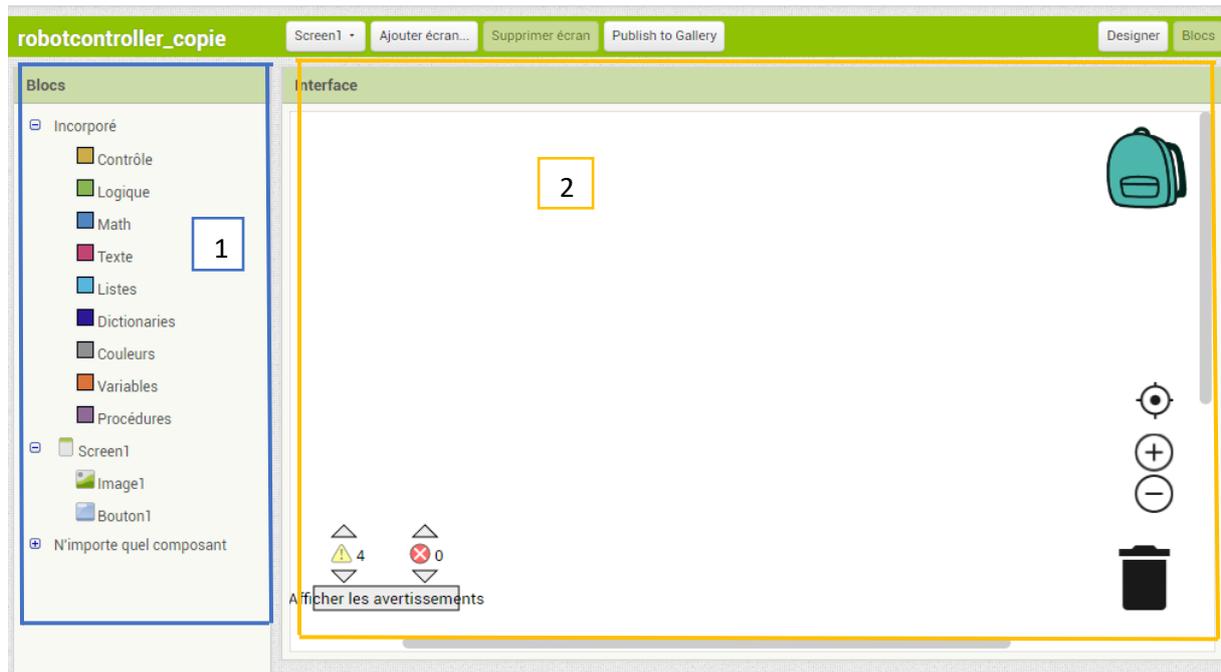


Figure 51 : contenu de la partie blocs

Le cadre 1 : contient l'ensemble des blocs à faire glisser dans la zone de travail afin de constituer le programme final de l'application qui sera sous forme de blocs.

Le cadre 2 : représente la zone de travail, la zone dans laquelle sont glissés les blocs.

### 3.3.4 Liste des éléments utilisés lors de la réalisation de l'application :

 Bouton	Bouton qui a la possibilité de détecter les clics. Beaucoup d'aspect de son apparence peuvent être changé, ainsi que la possibilité qu'il soit cliquable. (Activé), peut être changé dans la partie conception ou dans la partie bloc.
 Label	Un label affiche un morceau de texte, qui est spécifié avec la propriété Texte. Les autres propriétés, qui peuvent être fixées dans la partie Designer ou bloc, contrôlent l'apparence et l'emplacement du texte.
 Arrangement horizontal	Un élément de formatage dans lequel on place les composants qui doivent être affichés de gauche à droite. Si vous voulez avoir des composants affichés l'un sur l'autre, utilisez Arrangement vertical
 Arrangement tableau	Un élément de format où on place des composants qui doivent être affichés sous forme de tableau.
 Image	Composant pour afficher les images. La photo à afficher, et les autres aspects de l'apparence de l'image, peuvent être spécifiés dans la partie Conception ou dans la partie bloc.
 Bluetooth client	Composant client Bluetooth et qui est un composant non visible.
 List picker	Un bouton qui, quand cliqué, affiche à l'utilisateur une liste de textes pour choisir. Les textes peuvent être spécifiés dans la partie Designer ou la partie bloc en fixant la propriété Éléments de chaîne (par exemple, <i>choix 1</i> , <i>choix 2</i> , <i>choix 3</i> ) ou en fixant la propriété Éléments à une liste dans la partie bloc.
 Zone de texte	Une zone qui permet à l'utilisateur de faire entrer un texte.

Tableau 13 : liste des éléments utilisés pour la conception de l'application mobile

Enfin, après avoir conçu l'application des deux côtés (coté design et coté blocs) on clique sur l'icône "construire" et on choisit "ANDROID App (APK)" ce qui générera un code QR permettant de télécharger l'application mobile sur le smartphone.

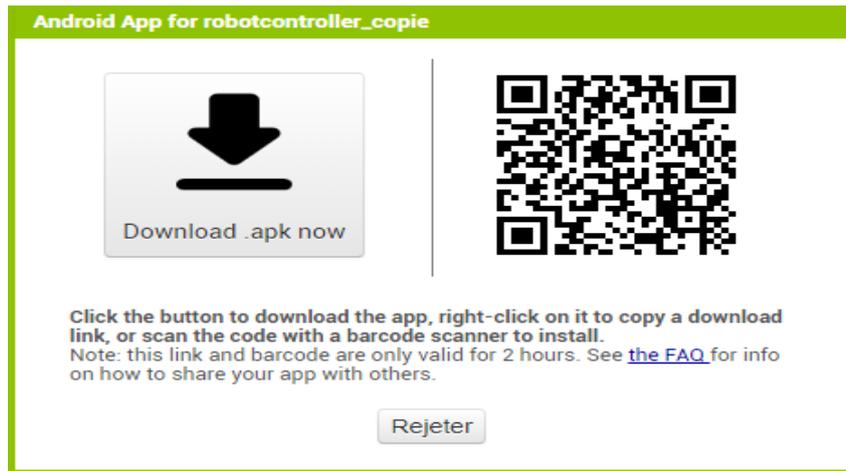


Figure 52 : lien et code QR générés pour télécharger l'application

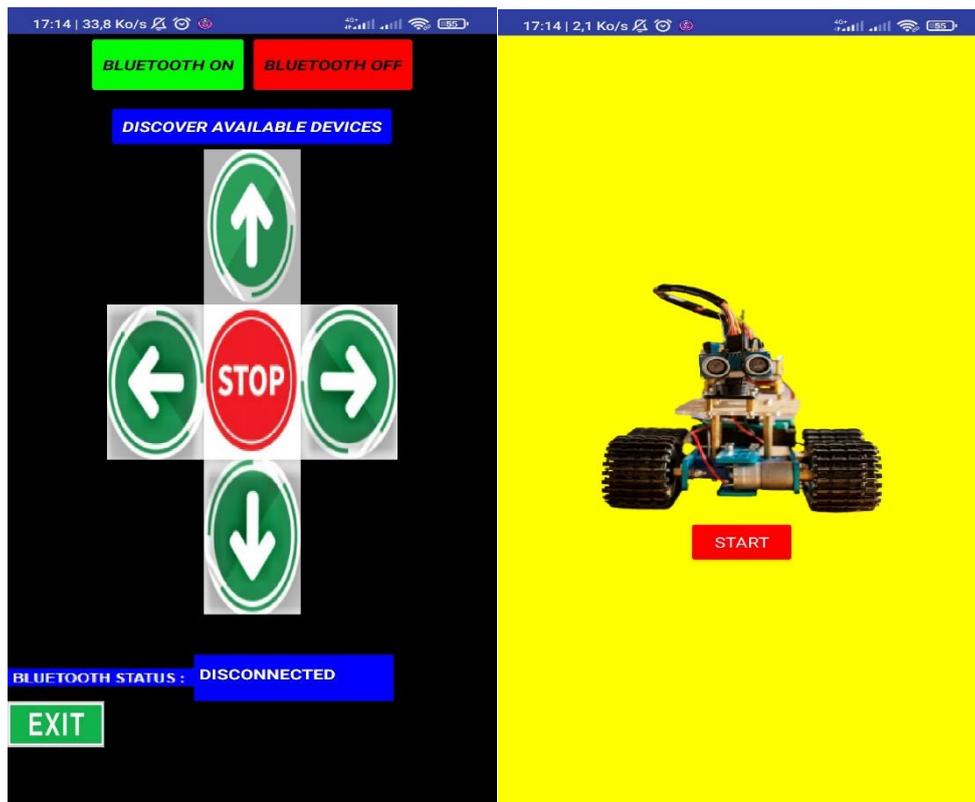
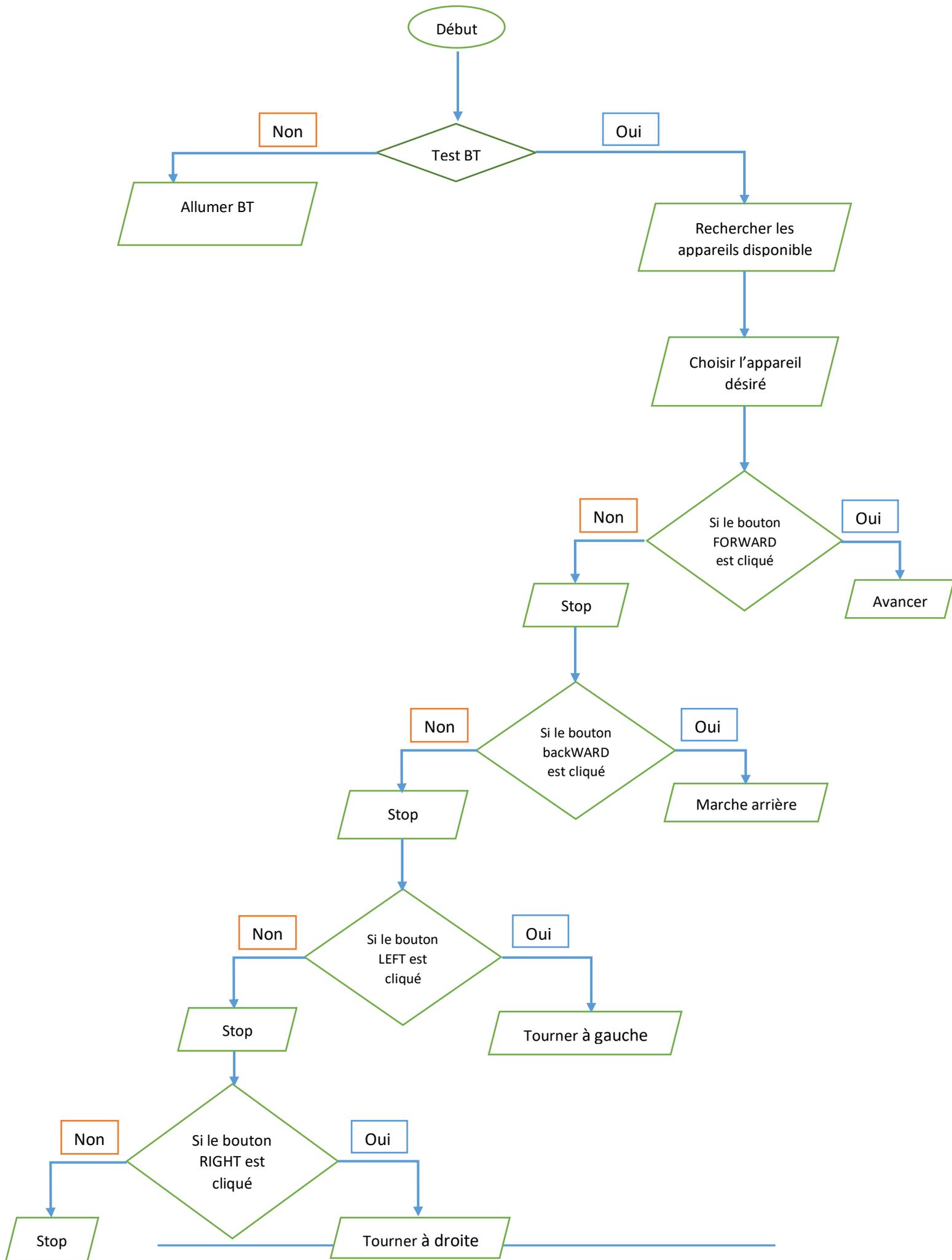


Figure 53 : application mobile créée

### 3.3.5 Organigramme de l'application :





### **3.4 Conclusion :**

Ce dernier chapitre est consacré pour donner une explication générale sur le robot réalisé ainsi que son mode de fonctionnement tout en mettant en évidence l'alternatif du commandement de ce robot qui est un commandement à distance à l'aide d'une application mobile Android qui assurera la liaison avec le robot mobile. Par la suite nous avons donné un aperçu sur le logiciel utilisé pour la conception et la réalisation de cette application en citant les différentes étapes et les outils utilisés pour cette réalisation.

---

**Conclusion**  
**générale.**

---

## **2. Conclusion générale :**

Dans ce travail présenté tout au long de mémoire, nous avons pu étudier et réaliser un robot mobile collaboratif d'assistance à la manutention à base d'Arduino. Ce projet était notre premier pas au domaine de la robotique mobile, un domaine qui nous a été avant étrange. Ce projet qui consiste à la mise en œuvre d'un robot mobile accumule de différentes spécialités : l'automatique, la mécanique et l'électronique informatique, ce qui nous a été bénéfique en nous donnant la chance d'intégrer de nouveaux domaines et d'acquérir de nouvelles connaissances dans le domaine de la robotique qui est loin d'être limité.

Dans ce travail nous nous sommes intéressés à l'étude et la réalisation d'un robot mobile capable de déterminer son propre trajet en commençant d'un point de départ tout en évitant les obstacles et les barrières qui se présentent tout au long de son chemin et enfin pouvoir revenir au même point de départ. Afin de pouvoir surmonter ce défi, nous avons conçu une carte de commande du robot mobile où ce robot est à base de roues motorisées (quatre roues motorisées). Cette carte de commande qui est liée essentiellement au microcontrôleur ATmega 328 qui sera le cerveau du système de commande conçu. Ce dernier se trouve au sein de la carte Arduino UNO sur laquelle est basée notre projet, cette carte assure la liaison et la connexion entre les différents composants utilisés pour commander ce robot. En effet, tout commence par le test effectué par le capteur ultrason dont le robot sera équipé avec, ce dispositif permet de déterminer la présence d'obstacles sur le chemin du robot ce qui permettra à la carte Arduino de générer les commandes nécessaires au driver moteur qui est à base d'un pont en H qui commencera à son tour les moteurs du robot dans les différents sens (avant, arrière, à droite et à gauche) afin de le pouvoir bien guider pour s'en sortir des labyrinthes où il se trouve.

Dans un deuxième temps, nous avons développé une application Android mobile qui assurera la commande du robot mobile à distance et cela en utilisant le module BLUETOOTH qui sera l'outil assurant la connexion entre le robot et l'utilisateur, cette application permettra également de commander le robot dans les différents sens (avant, arrière, à droite et à gauche) ainsi que de pouvoir l'arrêter en cas de défaillance ou d'urgence.

Il est à noter que ce projet est notre premier pas au domaine de la robotique mobile et il restera bien sûr loin d'être parfait et indémaillable. Néanmoins, il restera envisageable de l'améliorer pour répondre à toutes les exigences. Donnât exemple à rendre le robot capable de faire de l'auto-apprentissage de trajectoire afin d'optimiser son trajet tout en tenant compte aussi de l'équilibrage des roues imparfait qui affectera le déplacement du robot.

---

## **Bibliographie :**

- [1] ZEHAR Lamri & MEHARGA Sofiane, Conception et réalisation d'un robot mobile suiveur de ligne, présenté en vue de l'obtention d'un diplôme de MASTER, université de Bordj Bou Arreridj, soutenu le 14/09/2019.
- [2] LAMARA Nabil, Réalisation d'une commande d'un robot mobile, présenté en vue de l'obtention d'un MASTER PROFESSIONNEL, université de Tizi Ouzou année 2016.
- [3] ICHEROUFENE Omar, Conception et réalisation d'un robot mobile télécommandé à base de la PCDUINO V3, présenté en vue de l'obtention d'un MASTER ACADEMIQUE, université de Tizi Ouzou, soutenu le 19/07/2016.
- [4] BELLILI Sana'a, Conception et réalisation d'un robot d'inspection des pipelines à base de carte arduino UNO, en vue de l'obtention d'un diplôme de MASTER, université de Biskra, soutenu le 07/07/2019.
- [5] DJOUGGANE Fadhila, Etude et conception d'un robot manipulateur pour l'ENMTP, en vue de l'obtention d'un diplôme de MAGISTER, université de Batna, année universitaire 2011/2012.
- [6] Eskimon Olyte, zeste de savoir, Arduino : premiers pas en informatique embarquée, 1<sup>er</sup> février 2020.
- [7] [cours-gratuit.com—id-10467.pdf](#)
- [8] <http://fr.flossmanuals.net/arduino/index>
- [9] <https://tinyurl.com/3k5dzb47>
- [10] <https://www.techniques-ingenieur.fr/ressources-documentaires/demande-d-informations/extract/?ctype=Treaty&pageld=42398210&offer=ti661>
- [11] <https://perso.liris.cnrs.fr/laetitia.matignon/index/coursL1robotique.pdf>
- [12] <https://www.ulaval.ca/etudes/cours/glo-4001-introduction-a-la-robotique-mobile>
- [13] <http://www2.ift.ulaval.ca/~pgiguere/cours/IntroRobotique/index.html>
- [14] <https://licence-master-informatique.formation.univ-lorraine.fr/licence-informatique-universite-de-lorraine/l3-en-details/introduction-a-la-robotique-mobile/>
- [15] <https://fr.wikipedia.org/wiki/Arduino>
- [16] <https://arduiblog.com/2019/02/05/presentation-de-larduino/>
- [17] <http://domotics.fr/index.php/2019/02/25/les-differentes-cartes-arduino/>
- [18] <https://en.wikipedia.org/wiki/ATmega328>
- [19] <https://tinyurl.com/a75m4m3a>
-

- [20] <https://arduino.blaisepascal.fr/presentation/logiciel/>
  - [21] <https://www.journaldunet.fr/web-tech/dictionnaire-de-l-iot/1203379-bluetooth-definition-et-fonctionnement/>
  - [22] <https://pecquery.wixsite.com/arduino-passion/copie-de-le-detecteur-a-ultrasons-h-1>
  - [23] <https://tinyurl.com/yc3xbe5b>
  - [24] <https://arduino.blaisepascal.fr/pont-en-h-l298n/>
  - [25] <https://passionelectronique.fr/tutoriel-l298n/>
  - [26] <https://composant-electronique.fr/circuit-integre-l298n#>
  - [27] [http://technomoussi.free.fr/IMG/pdf/Tuto-Driver\\_Moteur\\_L298N-2018.pdf](http://technomoussi.free.fr/IMG/pdf/Tuto-Driver_Moteur_L298N-2018.pdf)
  - [28] <https://tinyurl.com/4trf8mau>
-