

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE 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

# Mémoire

Présenté pour obtenir

LE DIPLOME DE MASTER

Filière : Electronique

Spécialité : Electronique des Système embarqués

Présenter par :

ZE HAR Lamri

MEHARGA Sofiane

## Thème

*Conception et réalisation d'un robot mobile suiveur de  
ligne*

Soutenu le : 14/09/2019

Devant le jury :

- Dr.DAACHI                      Président
- Dr.BOUKEZATA              Examineur
- Dr.LATOUI                      Encadreur

Année Universitaire 2018/201

# Remerciement

Nous remercions tout d'abord ALLAH qui nous a donné l'aide, le courage, le pouvoir et la patience pour finaliser ce travail. Nous remercions, notre encadreur Mr LATOUI abdelhakim,, pour son encadrements de grande qualité scientifique, ces précieux conseils, et son encouragement , qui as contribué efficacement à l'avancement de ce travail. Nous adressons de même nos remerciements à Mr DAACHI , pour l'intérêt qu'il a accordé à ce travail en acceptant de présider le jury. Au laboratoire de recherche de génie Electronique (GE) de l'université de Bordj Bou Arreridj. Nous n'oublions pas dans nos remerciements tout le personnel de département d'Electronique. Merci à tous les professeurs qui ont contribué à notre formation .En fin, nous adressons nos plus sincères remerciements à nos familles : nos parents et tous nos proches et amis, qui nous a accompagné, aidé, soutenu et encouragé tout au long de la réalisation de cette mémoire et nous tenons à remercier toutes les personnes qui nous a aidé de prés ou de loin.

# Sommaire

# Sommaire

## Résumé

<b>INTRODUCTION GÉNÉRALE :</b> .....	<b>2</b>
<b>CHAPITRE 1 : GENERALITES SUR LES ROBOTS MOBILE</b> .....	<b>4</b>
<b>1.1 INTRODUCTION : [2]</b> .....	<b>5</b>
<b>1.2. DEFINITION :</b> .....	<b>5</b>
1.2.1 ETYMOLOGIE .....	5
<b>1.3. HISTORIQUE:[35]</b> .....	<b>5</b>
<b>1.4. ROBOTS MOBILES :</b> .....	<b>8</b>
<b>1.5 PROBLEMES EN ROBOTIQUE MOBILE :</b> .....	<b>9</b>
<b>1.6 CLASSIFICATION DES ROBOTS MOBILES : [3]</b> .....	<b>9</b>
<b>1.7ARCHITECTURE DES ROBOTS MOBILES :[6]</b> .....	<b>10</b>
1.7.1LA STRUCTURE MECANIQUE ET LA MOTRICITE : .....	10
1.7.1.1 Robot à roues : .....	10
1.7.1.2 Robots mobiles à chenilles :[11] .....	12
1.7.1.3 Robots mobiles à pattes : [11] : .....	13
1.7.1.4 Autres moyens de locomotion [11] : .....	13
1.7.1.5 La motricité et l'énergie [6] .....	13
1.7.2 Les organes de sécurité : .....	14
1.7.2.1 Capteurs intéroceptifs :[7] .....	14
1.7.2.2 Capteurs extéroceptifs : [7].....	14
1.7.3 TRAITEMENT DES INFORMATIONS ET GESTION DES TACHES [6].....	15
1.7.4 LE SYSTEME DE LOCALISATION : .....	15
<b>1.8 FUSION D'INFORMATIONS :</b> .....	<b>15</b>
<b>1.9LES DIFFERENTS TYPES DE NAVIGATION : [2]</b> .....	<b>16</b>
1.9.1 LES STRATEGIES DE NAVIGATION : .....	16
1.9.1.1Approche d'un objet : .....	16
1.9.1.2 Guidage : .....	16
1.9.1.3 Action associée à un lieu : .....	17
1.9.1.4 Navigation topologique : .....	17
1.9.1.5 Navigation métrique : .....	18
<b>1.10LES APPLICATIONS DU ROBOT MOBILE :</b> .....	<b>19</b>
<b>1.11CONCLUSION :</b> .....	<b>19</b>
<b>CHAPITRE 2 : CONCEPTION ET REALISATION D'UNE CARTE DE COMMANDE D'UN ROBOT MOBILE SUIVEUR DE LIGNE A BASE D'ARDUINO UNO</b> .....	<b>20</b>
<b>2.1. INTRODUCTION :</b> .....	<b>21</b>
<b>2.2 DEFINITION DU MODULE ARDUINO:[14]</b> .....	<b>21</b>
<b>2.3 DEFINITION DU MODULE ARDUINO UNO : [15]</b> .....	<b>21</b>

2.3.1 ATMEGA 328 : [16] .....	22
2.3.1.1 Les mémoires : .....	23
2.3.2 ALIMENTATION DE LA CARTE : [15] .....	23
2.3.3 LES BROCHES NUMERIQUES : [17].....	24
2.3.4 LES ENTREES ANALOGIQUES :[17] .....	25
2.3.5 SPECIFICATIONS DE LA CARTE ARDUINO UNO : [16] :.....	26
2.3.6 L'ENVIRONNEMENT DE DEVELOPPEMENT : .....	26
2.3.6.1 Correspondance : .....	27
<b>2.4 LE MODULE BLUETOOTH HC-05 : .....</b>	<b>27</b>
2.4.1 DEFINITION DE BLUETOOTH : [23].....	27
2.4.2 LES BROCHES DE HC-05 : .....	27
2.4.3 CONFIGURATION DU MODULE HC-05: .....	28
2.4.4CODE DE CONFIGURATION BLUETOOTH HC-05: .....	29
2.4.5 COMMANDES DE CONFIGURATION : .....	29
2.4.5.1 Tester le programme pour allumer une led avec hc-05 :.....	29
<b>2.5 LE CAPTEUR INFRAROUGE KY-033 : [21].....</b>	<b>30</b>
2.5.1 PRINCIPE DE FONCTIONNEMENT : .....	31
2.5.2 SPECIFICATIONS MODULE KY-033 : .....	31
2.5.3BROCHES DE CONNEXION : .....	31
2.5.4PRINCIPE DE FONCTIONNEMENT GENERAL : .....	31
<b>2.6 PARTIE DE PUISSANCE .....</b>	<b>32</b>
2.6.1 L293D : [22].....	32
2.6.1.2 Le pont H – principe : .....	33
2.6.1.3 Principe de fonctionnement d'un pont h avec enable : .....	33
2.6.2 MOTEUR DC :[24] .....	33
2.6.2.1 Les éléments du moteur dc : [25] .....	33
2.6.2.2 Principe de fonctionnement d'un moteur à courant continu : [26] .....	34
2.7.1 ALGORITHME DE SUIVRE LA LIGNE : .....	36
<b>CONCLUSION : .....</b>	<b>36</b>
<b>CHAPITRE 3 : CONCEPTION ET DEVELOPPEMENT D'UNE APPLICATION A DISTANCE D'UN ROBOT MOBILE</b>	
<b>SOUS ANDROID .....</b>	<b>37</b>
<b>3.1 INTRODUCTION : .....</b>	<b>38</b>
<b>3.2 DEFINITION DE SMARTPHONE :[33] .....</b>	<b>38</b>
3.2.1 CARACTERISTIQUES : [34] .....	38
3.2.2 SYSTEMES D'EXPLOITATION DE SMART PHONE :[34] .....	39
3.2.2.1Google Android : [34].....	39
3.2.2.2 Interface Apple OS : .....	39
3.2.2.3 Windows Phone .....	40
<b>3.3 HISTOIRE D'ANDROID : [27].....</b>	<b>40</b>
3.3.1 DEFINITION DE L'ANDROID : [28] .....	40
3.3.2 LES VERSIONS D'ANDROID : [29] .....	41
<b>3.4 DEFINITION DE MIT INVENTOR : [30].....</b>	<b>42</b>
3.4.1 LES AVANTAGES D'APP INVENTOR SONT NOMBREUX : .....	43

3.4.2 COMMENT SE CONNECTER A MIT INVENTOR ET L'UTILISER : [31] .....	43
3.4.2.1 <i>L'interface graphique</i> :.....	45
3.4.2.2 <i>l'aspect des blocs</i> : .....	46
3.4.3 LES ETAPES DE CREATIONS L'APPLICATION : .....	46
3.4.3.1 <i>Labels</i> :.....	47
3.4.3.2 <i>Horizontal arrangement</i> :.....	47
3.4.3.3 <i>Listpicker</i> :.....	48
3.4.3.4 <i>Table Arrangement</i> :.....	48
3.4.3.5 <i>Non visible –component</i> : .....	49
<b>3.5 CONCLUSION :</b> .....	<b>50</b>
<b>CONCLUSION GENERALE :</b> .....	<b>52</b>

# Résumé

Dans notre travail nous nous sommes concentré à la réalisation et la conception d'un système très compatible capable de s'adapter à son environnement et mise en œuvre un travail avec précision. Notre travail rentre de ce cadre et nous nous proposons de réaliser un robot mobile capable de suivre un parcours bien déterminé en toute autonomie. Il s'agira plus précisément de développer un système de commande qui permet au robot d'interagir avec son environnement au moyen de capteurs pour suivre une ligne noire tracée sur un fond Blanc. Ce travail est divisé en deux parties, la première partie qui consiste à faire une conception et réalisation d'une carte arduino uno à base d'un microcontrôleur ATMEGA, Celle-ci permet de générer les commandes appropriées à un pont H qui entrainera à son tour deux moteurs pour commander un robot et permet de commander toutes les actions de robot, la deuxième partie est la conception et la réalisation d'une application Android qui consiste en plusieurs boutons permettant de gérer le robot sur plusieurs sens et un bouton stop et autre pour suivre la ligne. Les essais mis en œuvre dans notre laboratoire qui confirment le bon fonctionnement de notre projet.

## ملخص

ركزنا في عملنا على تحقيق وتصميم نظام متوافق للغاية قادر على التكيف مع بيئته وتنفيذ عمل هيدروكربوني في هذا الإطار ونقترح إنجاز روبوت متقل قادر على متابعة مسار محدد في الحيز الكامل. ومنه تطوير نظام تحكم مسبق أكبر للروبوت بالتفاعل مع بيئته عن طريق أجهزة الاستشعار لإتباع خط أسود مرسوم على خلفية بيضاء. ينقسم عملنا على جزئين أساسيين: الجزء الأول الذي يتكون من تصميم وتنفيذ بطاقة arduino uno استناداً إلى المتحكم ATMEGA، وهو يتيح إنشاء الأوامر المناسبة لجسر H والذي بدوره يهتد ويتحكم في أربعة محركات DC للتحكم في الروبوت والسيطرة في جميع إجراءات الروبوت، الجزء الثاني هو تصميم وتنفيذ تطبيق أندرويد يتكون من عدة أزرار لإدارة الروبوت في عدة اتجاهات وتوزر للتوقف أو لمتابعة الخط. الاختبار المنفذ في مختبرنا أكد تحسن سير نظامنا المحقق التطبيق المطور في نهاية هذا المشروع وتحقيق النتائج المقررة.

# Abstract

In our work we focused on the realization and design of a very compatible system able to adapt to its environment and implement a job with precision. Our work comes from this framework and we propose to realize a mobile robot capable of following a determined path in complete autonomy. It will be more precisely to develop a control system that allows the robot to interact with its environment by means of sensors to follow a black line drawn on a white background. This work is divided into two essential parts the first part consists in making a design and realization of an arduinouno card based on a microcontroller ATMEGA, It allows to generate the appropriate commands to an H bridge that will lead to turns two motors to control a robot and allows to control all robot actions, the second part is the design and implementation of an Android application that consists of several buttons to manage the robot in several directions and a stop button and other to follow the line. The tests implemented in our laboratories and confirmed the good functioning of our system and the application developed at the end of this project.

# Liste des figures

Figure 2 : historique de robot .....	7
Figure 3 : Robots mobiles .....	8
Figure 4 : robot mobile a roues .....	10
Figure 5 : robot mobile à roue tricycle .....	11
Figure 6 : robot mobile à roue Omnidirectionnel.....	11
Figure 7 : robot mobile à roue différentiel .....	12
Figure 8 : robot mobile a chenille .....	12
Figure 9 : robot sous-marin et robot drone .....	13
Figure 10 : robot mobile à pattes .....	13
Figure 11 : broches de connexion avec hc-05.....	28
Figure 12 : code de configuration du module hc-05.....	29
Figure 13 : un code pour tester le module hc-05 avec une led .....	30
Figure 14 : le capteur infrarouge ( ky-033 .....	31
Figure 15 : les broches de I293D .....	32
Figure 16 : constitution du moteur DC.....	34
Figure 17 : Un système de frottement du moteur a courant contenu .....	35
Figure 18 : algorithme d'un code pour suivre la ligne .....	36
Figure 19 : le logo officielle de l'android .....	39
Figure 20 : le logo Apple iOS .....	39
Figure 21 : le logo de Windows phone.....	40
Figure 22 : Première smartphone équipée du système Android – T-Mobile G1 .....	40
Figure 23 : les versions android .....	42
Figure 24 : L'interface officielle de MIT APP Inventor .....	43
Figure 25 : Page officielle de mit inventor .....	44
Figure 26 : le menu pour choisir la langue .....	44
Figure 27 : Commencer nouveau projet .....	44
Figure 28 : créations un nouveau projet dans mit App inventor .....	45
Figure 29 : l'interface graphique de mit App inventor .....	45
Figure 30 : les éléments de l'aspect graphique du mit App inventor.....	46
Figure 31 : interface de bloc (coté de programmation).....	46
Figure 32 : le label .....	47
Figure 33 : l'horizontal arrangement .....	47
Figure 34 : List picker.....	48
Figure 35 : table arrangement .....	48
Figure 36 : Bluetooth client (composant non visible ).....	49
Figure 37 : le clock (Composant non visible) [32] .....	49
Figure 38 : l'aspect finale de notre interface graphique.....	50

# Liste des tableaux

<b>Tableau 1 : les applications du robot mobile .....</b>	<b>19</b>
<b>Tableau 2 : de fonctionnement d'un pont h avec enable.....</b>	<b>33</b>
<b>Tableau 3 : les versions android.....</b>	<b>41</b>

# Abréviation

1. PWM : Pulse Worth Modulation
2. UART : Universal Asynchronous Receiver Transmitter
3. DC : Direct Current
4. IOs : anciennement iPhone OS
5. MIT :Massachusetts Institute of Technology
6. C++ : langage de programmation
7. USB : Universal Serial Bus
8. Apk : Androïde Package Kit
9. JAVA : Langage, plate-forme (OS) et concept d'agents

# **INTRODUCTION GÉNÉRALE**

## **INTRODUCTION GÉNÉRALE :**

Contrairement aux robots manipulateurs prévus pour travailler exclusivement dans des espaces connus et de manière répétitive, les robots mobiles sont destinés à évoluer de manière autonome dans des environnements peu ou pas structurés. Par ailleurs, dans l'industrie, la logistique et la grande distribution, ce sont les déplacements des opérateurs, répétitifs et souvent liés au transport de lourds chariots. C'est pourquoi le recours à des robots mobiles à roues motorisés devient plus que nécessaire afin d'éviter le port des charges et les allers-retours entre les zones par les opérateurs. Cependant, les robots mobiles suiveurs de ligne sont très utilisés dans l'industrie afin de transporter de multiples charges d'un poste de travail à un autre.

Notre travail rentre de ce cadre et nous nous proposons de réaliser un robot mobile capable de suivre un parcours bien déterminé en toute autonomie. Il s'agira plus précisément de développer un système de commande qui permet au robot d'interagir avec son environnement au moyen de capteurs pour suivre une ligne noire tracée sur fond blanc. De plus, il sera commandé à distance (mise en marche et arrêt) par smartphone via une communication Bluetooth.

Pour ce faire, il faudra, dans un premier temps, réaliser une carte de commande qui comprendra outre une carte Arduino UNO dotée de module Bluetooth, un pont H pour alimenter deux moteurs à courant continu. Ces derniers entraîneront à leur tour le robot dans les deux sens de marche (avant et arrière) et y offre de plus la possibilité de le faire tourner dans les deux sens (à droite ou à gauche). Par ailleurs, des capteurs IR (type KY-033) seront utilisés (placés à l'avant du robot) pour identifier la ligne noire (tracée sur le sol) afin d'obliger le robot de suivre cette ligne. Dans un deuxième temps, nous proposons le développement d'une application Android pour pouvoir commander le robot via une liaison Bluetooth : un bouton pour le démarrage du robot et le deuxième pour l'arrêt.

Outre cette introduction, le présent manuscrit est organisé en trois chapitres. Le premier chapitre introduit les concepts de base de la robotique mobile où nous présentons les notions de base qui y sont inhérentes en passant par son historique, son évolution et nous terminons par les différentes classes et domaines d'applications des robots mobiles.

Le deuxième chapitre est consacré à une présentation détaillée de la conception de la carte de commande du robotsuiveur de ligne. Le troisième chapitre propose le développement d'une application Android qui permettrade positionner le robot sur la ligne noire, tracée sur le sol, et la mise éventuelle de son arrêt. De plus, les essais effectués au terme de projet sont également présentés dans ce même chapitre.

# **Chapitre 1 :**

# **Généralités sur les**

# **Robots mobiles**

## **1.1 Introduction : [2]**

il existe diverses définitions du terme robot, mais elles tournent en général autour de celle-ci : un robot est un dispositif mécatronique (alliant mécanique, électronique et informatique) conçu pour accomplir automatiquement des tâches imitant ou reproduisant, dans un domaine précis.

Dans notre premier chapitre, nous donnerons un aperçu des robots mobiles, de leurs types et modes de fonctionnement, ainsi que de la façon dont ils s'adaptent au centre dans lequel ils se trouvent.

## **1.2. Définition :**

Un robot est un dispositif mécatronique (alliant mécanique, électronique et informatique) conçu pour accomplir automatiquement des tâches imitant ou reproduisant, dans un domaine précis, La conception de ces systèmes est l'objet d'une discipline scientifique. [1]

### **1.2.1 Etymologie**

Le terme robot apparaît pour la première fois de l'auteur Karel Čapek

Le mot « robota » qui signifie « travail forcé, corvée ». [1]

## **1.3. Historique:[35]**

Premier robot est probablement dû à Léonard De Vinci, qui présenta en <<1495>> un chevalier en armure capable de s'asseoir, relever sa visière et bouger ses bras.

1739 : Le canard de Jacques Vaucanson

Un des automates les plus surprenants est certainement le Canard de Vaucanson (1739). Il est capable de boire, manger, cancaner et digérer comme un véritable animal

1774 : Les 3 automates de Pierre Jacquet-Droz

1950 : Les tortues de Grey Walter, premiers robots mobiles autonomes

L'énergie utilisée est des panneaux solaires pouvant fournir jusqu'à 140 watts d'énergie électrique.

1961 :Unimate, premier robot industrielle

Les premières applications consistaient en des tâches de manutention dans des ateliers de mécanique automobile, puis des travaux de soudage par points dans les ateliers de montage et d'assemblage des grands constructeurs automobiles. Les matériaux utilisés sont des bouts de métal et de fer.

1967 : Shakey, premier robot mobile contrôlé par ordinateur, (Stanford research Institute)

Il a pour tête une grande caméra qui peut tourner et s'incliner et son corps est constitué par un immense ordinateur reposant sur une plate-forme à trois roulettes. Si Shakey parvient à se mouvoir par lui-même. Quant à son nom, il est dû aux secousses qu'il éprouve lors de tels déplacements : Shakey signifie "qui tremble"., par ses divers capteurs : sa caméra, mais aussi un télémètre pour mesurer les distances et des senseurs tactiles pour percevoir les obstacles

1973 : Wabot- 1 , premier robot à marche bipède

Également une source d'énergie embarquée : généralement d'accumulateurs électriques. Ils y remplacent les ouvriers dans les tâches pénibles et dangereuses (peinture, soudage, emboutissage, etc.).

1991 Robot mouche, laboratoire mouvement et perception

1997 Deep Blue, ordinateur d'IBM, bat Kasparov aux échecs

Deep Blue est un superordinateur spécialisé dans le jeu d'échec

2001 Minirobot de Sandia

Il se compose d'un microprocesseur de 8Ko de ROM, de deux moteurs actionnant les roues et d'un capteur de températures.

2000 Asimo, HondaIl mesure 120 cm pour 52 kg.Le petit androïde autonome mesure 1.20 m et pèse 52 kg. Il se déplace à 1.6 km/h mais il est capable de courir à allure modérée (12 km/h), de sautiller, de danser et de monter ou descendre des escaliers. A terme, les robots ASIMO devront pouvoir venir en aide aux personnes handicapées, âgées ou malades. Ils pourront aussi effectuer des tâches dangereuses pour les humains.

2005 Nao : NAO est un robot de 58 cm de haut et de 5 kg développé par AldebaranRobotics. C'est un robot entièrement programmable, autonome et interactif. NAO est utilisé dans de

prestigieuses universités et instituts de recherche du monde entier pour la recherche et l'éducation

- 1947 : Premier manipulateur électrique télé opéré.
- 1954 : Premier robot programmable.
- 1961 : Apparition d'un robot sur une chaîne de montage de General Motors.
- 1961 : Premier robot avec contrôle en effort.
- 1963 : Utilisation de la vision pour commander un robot.



Figure 2 : historique de robot

Nous avons mentionné quelques historiques de robots, mais il existe encore un développement de robots dans de nombreux domaines.

## 1.4. Robots mobiles :

Un robot mobile est un véhicule doté de moyens de locomotion qui lui permettent de se déplacer. Suivant son degré d'autonomie il peut être doté de moyens de perception et de raisonnement.

C'est un système mécanique, électronique et informatique qui agit physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné. Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditions de fonctionnement. Elle est dotée de fonctions de perception, de décision et d'action. Ainsi, le robot devrait être capable d'effectuer.

De différentes manières, des tâches diverses et d'accomplir correctement sa propre tâche, même s'il rencontre de nouvelles situations inopinées.

L'appellation « robot mobile » rassemble tous les types de robots dont la caractéristique commune est la faculté de se mouvoir. Cependant, la manière qui dépend du domaine d'utilisation des robots, fait la différence de ces derniers.

Ainsi, la mobilité par les roues est, par exemple, la structure mécanique la plus communément appliquée. Cette technique assure, selon l'agencement et les dimensions des roues, un déplacement

Dans toutes les directions avec une accélération et une vitesse importantes. [3]



Figure 3 : Robots mobiles

## **1.5 Problèmes en robotique mobile :**

On distingue sans trop d'ambiguïté un certain nombre de problèmes en robotique mobile. Bien évidemment, l'aspect matériel, qui consiste à choisir et dimensionner aussi bien la structure mécanique du système que sa motorisation, son alimentation et l'architecture informatique de son système de contrôle-commande apparaît comme le premier point à traiter. Le choix de la structure est souvent effectué parmi un panel de solutions connues et pour lesquelles on a déjà résolu les problèmes de modélisation, planification et commande. Le choix des actionneurs et de leur alimentation est généralement assez traditionnel. La plupart des robots mobiles sont ainsi actionnés par des moteurs électriques à courant continu avec ou sans collecteur, alimentés par des convertisseurs de puissance fonctionnant sur batterie. De la même façon, les architectures de contrôle-commande des robots mobiles ne sont pas différentes de celles des systèmes automatiques ou robotiques plus classiques. On y distingue cependant, dans le cas général, deux niveaux de spécialisation, propres aux systèmes autonomes : une couche décisionnelle, qui a en charge la planification et la gestion (séquentielle, temporelle) des événements et une couche fonctionnelle, chargée de la génération en temps réel des commandes des actionneurs. Bien évidemment, l'architecture du robot dépend fortement de l'offre et des choix technologiques du moment [12].

## **1.6 CLASSIFICATION DES ROBOTS MOBILES : [3]**

La classification des robots mobiles se fait suivant plusieurs critères :

Degré d'autonomie, système de locomotion, énergie utilisée ...). La classification la plus intéressante, et la plus utilisée est selon leur degré d'autonomie. Un robot mobile autonome est un système doté de capacités décisionnelles et de moyens d'acquisition et de traitement de l'information qui lui permettent d'accomplir sous contrôle humain réduit, un certain nombre de tâches, dans un environnement non complètement connu. On peut citer quelques types:

- Véhicule télécommandé par un opérateur qui lui impose chaque tâche élémentaire à réaliser.
- Véhicule télécommandé au sens de la tâche à réaliser.
- Véhicule semi- autonome réalisant sans l'aide de l'opérateur des tâches prédéfinies.
- Véhicule autonome qui réalise des tâches semi- définies.

Les principaux problèmes particuliers liés à la conception de tels robots sont:

- La conception mécanique liée à la mobilité.
- La détermination de la position et de l'orientation.
- La détermination du chemin optimal.

## **1.7 ARCHITECTURE DES ROBOTS MOBILES :[6]**

L'architecture des robots mobiles se structure en quatre éléments :

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

### **1.7.1 La structure mécanique et la motricité :**

On peut dénombrer quatre types de structures mécaniques assurant la motricité.

#### ***1.7.1.1 Robot à roues :***

La mobilité par roues est la structure mécanique la plus communément

appliquée. Cette technique assure selon l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importantes

.Le franchissement d'obstacles ou l'escalade de marches d'escaliers est possible.



**Figure 4 : robot mobile a roues**

Il existe plusieurs classes de robots à roues déterminées, principalement, par la position et le nombre de roues utilisées. Nous citerons ici les classes principales de robots à roues. [5]

#### 1.7.1.1.1 Robot Tricycle :

Deux roues fixes indépendantes de même axe de rotation et une roue centrée orientable sur l'axe longitudinal du robot équivalent à une voiture usuelle.

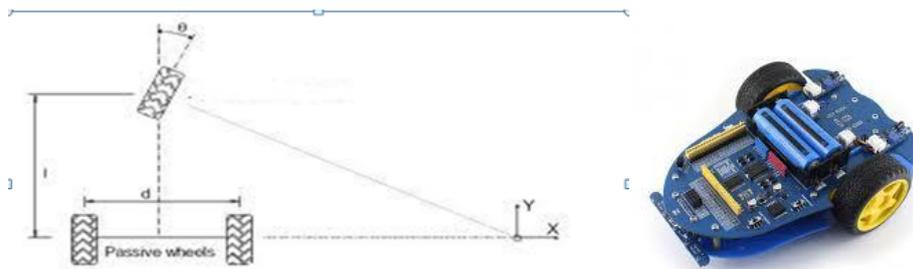


Figure 5 : robot mobile à roue tricycle

#### 1.7.1.1.2 Robot Omnidirectionnel:

Équipé de trois à quatre roues, peut se déplacer dans toutes les directions du plan et s'orienter indépendamment. [5]

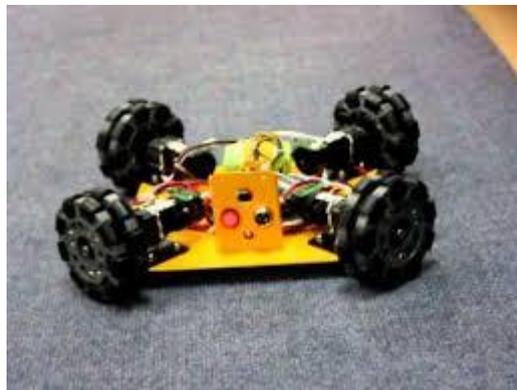
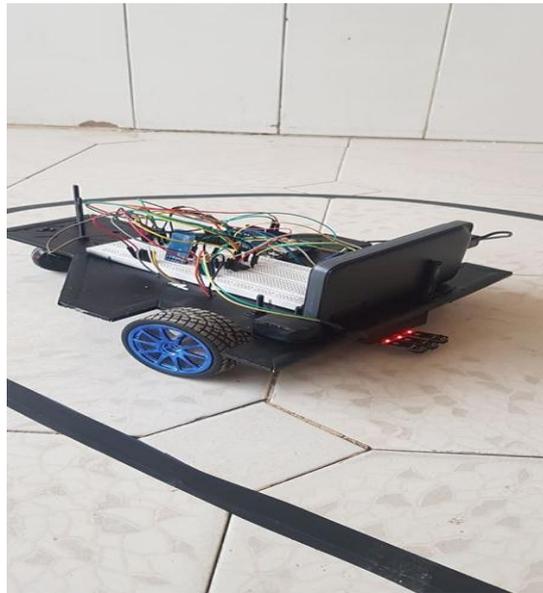


Figure 6 : robot mobile à roue Omnidirectionnel

#### 1.7.1.1.3 Robot différentiel :

Deux roues fixes indépendantes de même axe de rotation et une roue libre auto directionnelle [5]

→ l'orientation se fait par la différence de marche des deux roues de propulsion



**Figure 7 : robot mobile à roue différentiel**

#### ***1.7.1.2 Robots mobiles à chenilles :[11]***

Lorsque le terrain est accidenté, les roues perdent leur efficacité de locomotion. Ceci limite la capacité d'évolution du robot mobile équipé de ce type de système de locomotion. Dans Ces conditions, les chenilles sont plus intéressantes, car elles permettent d'augmenter l'adhérence au sol et de franchir des obstacles plus importants l'adhérence au sol et de franchir des obstacles plus importants



**Figure 8 : robot mobile a chenille**

### 1.7.1.3 Robots mobiles à pattes : [11] :



Figure 9 : robot mobile à pattes

.Dans la situation où le terrain est encore plus accidenté, avec des grandes différences de hauteur comme par exemple un escalier ou un terrain très accidenté, les robots à roues ou à chenilles ne sont plus efficaces, Les robots mobiles à pattes ont des points d'appui discrets sur le terrain et sont donc la solution à ce problème de mouvement. Par contre, la conception et le contrôle d'un engin à pattes est très complexes. En plus, sa vitesse d'évolution est généralement très réduite

### 1.7.1.4 Autres moyens de locomotion [11] :

Il est d'usage de mettre dans ce groupe tous les robots mobiles qui utilisent un moyen de locomotion différent des trois précédents. Par exemple, les robots mobiles qui se déplacent par reptation, les robots sous-marins, les robots pour l'exploration spatiale, les robots volants également appelés « drones », etc. Les applications de ce type de robots sont très spécialisées et les architectures des robots sont en général spécifiques à l'application visée.

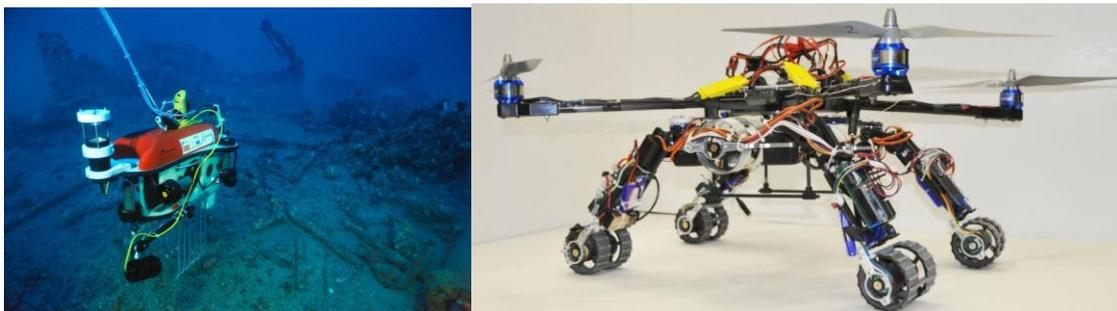


Figure 10 : robot sous-marin et robot drone

### 1.7.1.5 La motricité et l'énergie [6]

Les déplacements des robots sont réalisés par des moteurs de types électrique, thermique ou hydraulique. L'énergie électrique la plus fréquemment employée offre l'avantage d'une commande aisée. Par contre le transport et la génération présentent des difficultés. Plusieurs méthodes sont employées :

- Par batteries qui sont soit rechargées périodiquement de manière automatique ou manuelle, soit par un échange avec d'autres lorsqu'elles sont déchargées.
- Par groupe électrogène embarqué dont l'inconvénient constitue la masse élevée. L'énergie de base est alors thermique.
- Par cordon ombilical qui réduit l'autonomie du robot. L'énergie thermique est essentiellement employée par des véhicules de forte puissance comme énergie de base pour la traction ou pour activer un compresseur hydraulique.
  - Par l'énergie solaire à partir des panneaux solaire qu'ils sont adaptés sur le corps du robot

### ***1.7.2 Les organes de sécurité :***

Un robot, selon la tâche qui lui est confiée, peut être amené à travailler au voisinage du personnel. A ce titre, il est obligatoire qu'il soit doté d'organes garantissant la sécurité. Des capteurs sont disponibles tout autour du mobile afin de détecter un obstacle sur un domaine le plus étendu possible. Deux types de capteurs sont employés : les capteurs périmétriques assurant la détection avant collision (ultra-son, infrarouge...) et les capteurs de contact détectant une collision ou un choc avec l'environnement (contact électrique sur pare-choc...). Ce sont des dispositifs redondants par rapport aux capteurs précédents

Ils sont classés dans deux classes:

#### ***1.7.2.1 Capteurs intéroceptifs : [7]***

Fournissent des données sur l'état interne du robot (vitesse, position, orientation,...). Ces informations renseignent le robot en cas de mouvement, sur son déplacement dans l'espace (la localisation). Ce sont des capteurs que l'on peut utiliser directement. Nous citons par exemple: l'odomètre, radar doppler...

#### ***1.7.2.2 Capteurs extéroceptifs : [7]***

Ont pour objectif d'acquérir des informations sur l'environnement proche du véhicule. Ils fournissent des mesures caractéristiques de la position que le robot peut acquérir dans son environnement par la détection des objets qui contourne. Ces informations peuvent être de natures très variées. Nous citons comme exemple : ultrason, infrarouge, laser, les caméras,...etc.

Dans un robot, la perception est assurée par l'utilisation de capteurs. Ces derniers donnent une information à propos de l'environnement ou des composants internes (position d'un moteur, état d'une LED). Cette information est utilisée pour calculer l'ordre approprié à envoyer aux actionneurs.

Il existe une grande variété de capteurs. Par exemple :

Ultra-son, infrarouge ou LASER. Ces derniers sont à la base des scanners laser permettant à l'unité centrale du robot de prendre "conscience" de son environnement en 3D. et des caméras

### **1.7.3 Traitement des informations et gestion des tâches [6]**

L'ensemble de traitement des informations et gestion des tâches constitue le module Information central qui établit les commandes permettant au mobile de réaliser un déplacement. Et d'activer les divers organes en accord avec l'objectif. Nous nous limiterons au problème de Génération de plan qui consiste à établir la manière dont le robot se déplace par rapport à des Connaissances à priori (statiques) ou obtenues en cours d'évolution (dynamiques). comme (La stratégie de navigation, La modélisation de l'espace, La planification...).

### **1.7.4 Le système de localisation :**

Les méthodes de La localisation instantanée est un des points les plus importants et les plus délicats des robots mobiles. Elle permet de définir le positionnement : Du mobile dans l'environnement ou d'éléments particuliers de l'environnement par rapport au mobile.

## **1.8 Fusion d'informations :**

Est un processus qui permet de collecter tous les résultats détectés par les capteurs afin de traiter les informations. Les capteurs fonctionnent individuellement mais doivent être intégrés pour tout savoir sur le robot, La fusion de capteurs regroupe les techniques permettant d'utiliser conjointement

Les différents capteurs dont dispose un robot. Comme nous l'avons vu, un capteur peut superviser la commande proprioceptive, ou bien être intègre dans la loi de commande.

Cette séparation se retrouve dans le cas de plusieurs capteurs.

Dans le premier cas, l'objectif est de regrouper les données fournies afin d'évaluer au mieux la pose du robot. Il s'agit de la fusion de données dans un large éventail des techniques et des applications est exposé.

Ces méthodes sont communes aux systèmes d'information à grande échelle [13]

En résumé, les informations proprioceptives sont simples à utiliser, mais dérivent au cours du temps, tandis que les perceptions ne dérivent pas, mais souffrent des problèmes de variabilité perceptuelle et d'Ambiguïté. La solution pour résoudre ces problèmes est de fusionner ces deux types d'information. Il est par exemple possible d'utiliser les informations proprioceptives afin de distinguer deux positions physiquement différentes mais similaires pour le système perceptif. [2]

## **1.9 Les différents types de navigation : [2]**

### **1.9.1 Les stratégies de navigation :**

Les stratégies de navigation permettant à un robot mobile de se déplacer pour rejoindre un but sont extrêmement diverses, de même que les classifications qui peuvent en être faites. Afin de situer les différentes méthodes de navigation que nous allons étudier dans un contexte général, nous reprenons ici une classification établie. Cette classification comporte cinq catégories, de la plus simple à la plus complexe :

#### ***1.9.1.1 Approche d'un objet :***

Cette capacité de base permet de se diriger vers un objet visible à partir des capteurs spéciaux (capteur infrarouge, capteur ultrasonique...).

#### ***1.9.1.2 Guidage :***

Cette capacité permet d'atteindre un but qui n'est pas un objet matériel directement visible, mais un point de l'espace caractérisé par la configuration spatiale d'un ensemble d'objets remarquables, qui l'entourent ou qui en sont voisins. La stratégie de navigation, consiste alors à se diriger dans la direction qui permet de reproduire cette configuration. Cette capacité semble utilisée par certains insectes, comme les abeilles, et a été utilisée sur divers robots. Cette stratégie utilise également des actions réflexes et réalise une navigation locale qui requiert caractérisant le but soient visibles

### 1.9.1.3 Action associée à un lieu :

Action associée à un lieu. En chaque lieu, représenté par un cercle, l'action accomplir pour rejoindre le but A est représentée par une flèche indiquant la direction à suivre À partir de ce lieu. Cette stratégie permet de rejoindre un but distant dans l'environnement Mais repose sur des chemins figés.

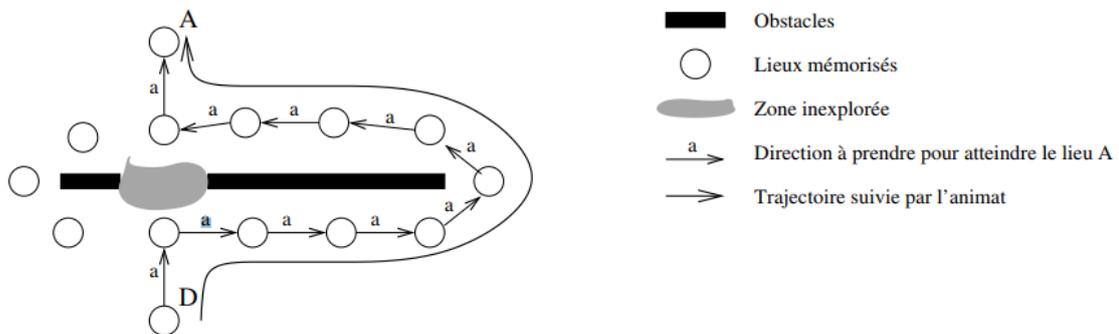


Figure 10 : Action associée à un lieu

### 1.9.1.4 Navigation topologique :

Cette stratégie permet de mémoriser un ensemble de lieux et les possibilités de passer de l'un à l'autre, indépendamment de tout but. Pour rejoindre un but, il faut alors une étape de planification qui permet de rechercher, parmi tous les chemins possibles, le chemin rejoignant le but. Dans notre exemple, le chemin le plus court entre D et A peut alors être calculé, mais uniquement parmi les lieux et les chemins déjà connus. Cette stratégie permet, par exemple, de contourner l'obstacle par la gauche mais ne permet pas de le traverser en ligne droite de D à A.

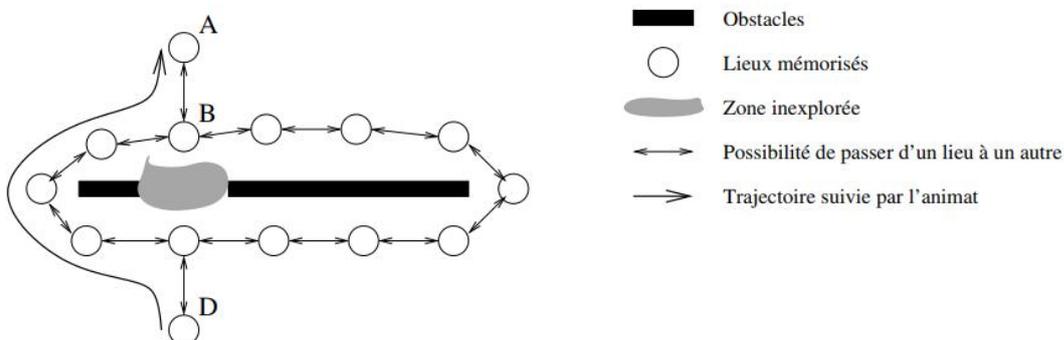


Figure 11 : navigation topologique

### 1.9.1.5 Navigation métrique :

Cette stratégie permet de calculer le chemin le plus court entre deux lieux mémorisés, permettant même de planifier des raccourcis au sein des zones inexplorées de l'environnement. Pour cela, la carte mémorise la position métrique relative de chacun des lieux visités par le robot. Ainsi il est possible de prévoir un déplacement entre deux lieux, même si la possibilité de ce déplacement n'est pas enregistrée dans la carte.

Dans cet exemple, cette stratégie permet de d'aller du lieu A au lieu D'en traversant la zone inexplorée.

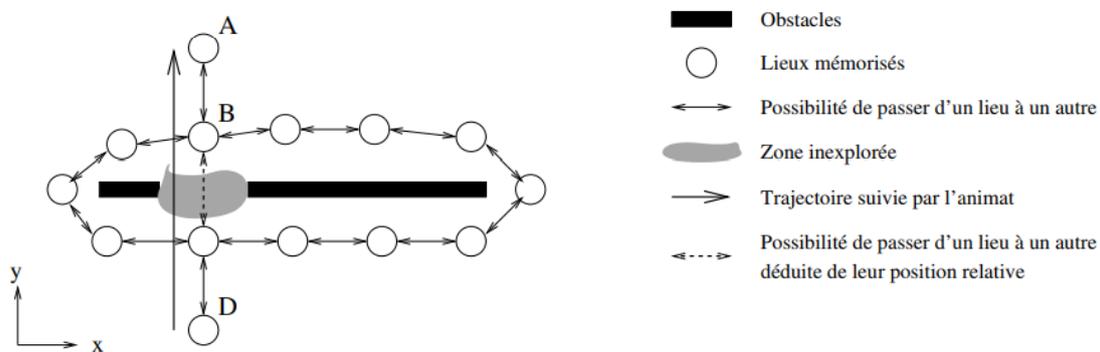


Figure 12 : navigation métrique

Construire des méthodes et des outils permettant, aujourd'hui, de maîtriser tous les éléments permettant à un véhicule à remorque de calculer automatiquement des trajectoires sans collision entre deux configurations puis de suivre ces trajectoires en évitant de manière réactive les obstacles imprévus. Ces travaux ont été intégrés à bord [8]. De notre robot qui navigue de manière autonome et effectue des avec précision. Pour traiter en particulier la planification automatique et de la commande de mouvement en présence d'obstacles pour les systèmes articulés à roues, ainsi que la sécurité active des automobiles pour l'aide à la conduite. Pouvoir naviguer de façon autonome dans un environnement complexe est une capacité indispensable à tout robot mobile évoluant en intérieur ou en extérieur. Pour cela, le robot par le biais de capteurs, doit se construire une carte du monde qui intègre les différentes couches de représentation nécessaires à une tâche de navigation autonome. Pour définir une couche de représentation topologique. La représentation topologique a pour objectif de coder l'accessibilité aux différentes parties de l'environnement. [9]

## 1.10 Les applications du robot mobile :

Le domaine d'application du robot :

Domaines	Application
Industrie nucléaire	<ul style="list-style-type: none"> <li>✓ Surveillance de cites.</li> <li>✓ Manipulation de matériaux radioactifs.</li> <li>✓ démantèlement de central.</li> </ul>
Sécurité civil	<ul style="list-style-type: none"> <li>✓ neutralisation d'activité terroriste</li> <li>✓ déminage.</li> <li>✓ pose explosif.</li> <li>✓ Surveillance de munition.</li> </ul>
Chimique	<ul style="list-style-type: none"> <li>✓ Surveillance de cite.</li> <li>✓ Manipulation de matériaux toxique</li> </ul>
Mine	<ul style="list-style-type: none"> <li>✓ Assistance d'urgence</li> </ul>
Agricole	<ul style="list-style-type: none"> <li>✓ Cueillette de fruit</li> <li>✓ Traite, moisson, traitement des vignes</li> <li>✓</li> </ul>
Nettoyage	<ul style="list-style-type: none"> <li>✓ Coque de navire</li> <li>✓ Nettoyage industriel</li> </ul>
Espace	<ul style="list-style-type: none"> <li>✓ Exploration</li> </ul>
industrie	<ul style="list-style-type: none"> <li>✓ convoyage</li> <li>✓ surveillance</li> </ul>
Sous-marine	<ul style="list-style-type: none"> <li>✓ pose de câbles</li> <li>✓ recherche de module</li> <li>✓ recherche de navire immergé</li> <li>✓ inspection des fonds mains</li> <li>✓ inspection des fonds marins</li> </ul>
Militaire	<ul style="list-style-type: none"> <li>✓ surveillance</li> <li>✓ pose d'explosifs</li> <li>✓ manipulation de munitions</li> </ul>

Tableau 1 : les applications du robot mobile

## 1.11 Conclusion :

Dans ce premier chapitre, nous avons présenté un aperçu général des robots mobiles alors on a étudié, leurs classification et leurs types et l'architecture général (La structure mécanique, les organes de sécurité, Le système de traitement des informations ...) et de leurs domaines d'application, ainsi que la façon de perception et localisation et de traitement d'information, ainsi que les type de navigation de robot mobile dans un environnement extérieur. Nous nous sommes concentrés sur les aspects fondamentaux du processus système sur lesquels reposent les robots mobiles et sur la structure de base.

## **Chapitre 2 :**

# **Conception et réalisation d'une carte de commande d'un robot mobile suiveur de ligne à base d'Arduino Uno.**

## 2.1. Introduction :

Aujourd'hui, l'électronique est développée de plus en plus remplacée par l'électronique programmable c'est à dire un système embarqué ou un système informatique pour le bon fonctionnement en a créé un système complexe performant adapter sur tous les domaines et pour accessible à toutes les personnes ,dans notre chapitre nous sommes appuyés sur carte arduino pour commander un robot mobile à roue qui se compose de moteurs à courant continu (moteur DC) qui sont alimentées par un pont H (L293D) , Nous contrôlons notre robot à distance par un smart phone à partir d'une application android nous l'avons créé et développé et pour cela nous adoptons une liaison sans fil (module Bluetooth ) a base d'un HC-05.

## 2.2 Définition du module Arduino:[14]

La carte Arduino est un circuit imprimé conçu pour héberger un microcontrôleur et donner accès à toutes ses entrées et sorties. Elle comprend aussi quelques autres composants électroniques qui permettent de faire fonctionner le microcontrôleur ou d'en étendre les fonctionnalités.

## 2.3 Définition du module Arduino uno : [15]

La carte Arduino UNO est un circuit électronique à base de microcontrôleur ATmega328 programmable permettant de faire fonctionner des composants (moteur, LED...). Elle possède des «ports» permettant par exemple de se connecter à un ordinateur ou de s'alimenter.



Figure 13 : la carte arduino uno

Il se compose de :

Un microcontrôleur ATMEGA 328

- ✓ Broches d'alimentation ;
- ✓ De 14 broches numériques d'entrées/sorties (dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée));
- ✓ de 6 entrées analogiques ;

Un quartz 16Mhz;

- ✓ D'une connexion USB;
- ✓ D'un bouton de réinitialisation (reset). Et d'autres composants

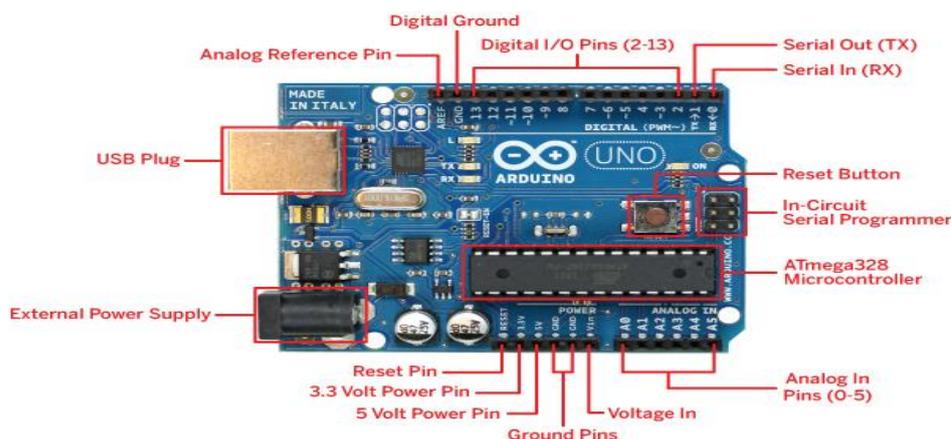


Figure 14 : les composants (constitution) de la carte arduino uno

### 2.3.1 ATMEGA 328 : [16]

L'ATMega328 est un microcontrôleur 8bits dont la programmation peut être réalisée en langage C/C++. L'intérêt principal des cartes ARDUINO est leur facilité de mise en oeuvre. et pour traiter les données et gère les opérations dans la carte arduino, L'ATMEGA 328 est un microcontrôleur basé sur RISC AVR 8 bits hautes performances qui combat la mémoire flash de 32 ko avec des capacités de lecture en cours d'écriture, EEPROM 1024B, SRAM de 2 ko, 23 lignes E/S à usage général, trois horloges/compteurs flexibles avec modes de comparaison, un port série SPI, un convertisseur A/N, une horloge programmable de surveillance avec oscillateur exécutant des instructions puissantes en un seul cycle d'horloge.



Figure 15 : Atmega 328

### 2.3.1.1 Les mémoires :

Les 3 types de mémoires qui coexistent :

La mémoire **FLASH** : Peu coûteuse cette mémoire sert à stocker les programmes à exécuter, c'est une mémoire qui perdure après arrêt de l'alimentation, elle est donnée pour une capacité de 1 000 000 de cycles. L'atmega 328 en est doté de 32 KO.

La mémoire **SRAM** (Static Read Access Memory) : Coûteuse mais rapide, cette mémoire sert à stocker des données temporaires (les variables de votre programme par exemple). C'est une mémoire volatile. Elle possède seulement 2KO

La mémoire **EEPROM** : (Electrically-Erasable Programmable Read-Only Memory ou mémoire morte effaçable électriquement et programmable)

Elle permet le stockage par votre programme de données persistantes, peut coûteuse, mais lente, l'atmega 328 en possède 1 KO. Cette mémoire s'utilise plus rapidement, sa capacité est de 100 000 cycles d'écriture. Cette mémoire permettra à vos programmes de stocker des données de log ou des états devant être conservés de manière pérenne (après arrêt de l'alimentation)

## 2.3.2 Alimentation de la carte : [15]

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte. L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise (2.1mm) positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs

des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte.

Les broches d'alimentation sont les suivantes :

- VIN. La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). Vous pouvez alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- 5V. La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.
- 3V3. Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'ATmega) de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V).
- GND. Broche de masse (ou 0V).

### **2.3.3 Les broches numériques : [17]**

**Pins: 0 (RX) et 1 (TX) :** sont utilisés pour la réception (RX) et d'émission (TX) des données en série. Ceux - ci sont connectés aux broches correspondantes du port USB



### 2.3.5 Spécifications de la carte Arduino Uno : [16] :

Catégorie	Valeur
Microcontrôleur	Atmega 328
Tension de service	5V
Tension d'entrée	7-12V
Tension d'entrée (limites)	6-20V
Port numérique	14 entrées et sorties
Port analogique	6 entrées analogiques
Mémoire	32 KO
SRAM	2 KO (Atmega 328)
EEPROM	1 KO (Atmega 328)
Fréquence d'horloge	16 MHz

Tableau 2 :spécification de la carte arduino uno

### 2.3.6 L'environnement de développement :

Un IDE (environnement de développement) libre et gratuit est distribué sur le site d'Arduino (compatible Windows, Linux et Mac) [18].Langage Arduino est basé sur le langage C/C++, avec des fonctions et des bibliothèques spécifiques à l'Arduino.Les opérations de compilation et de chargement dans la mémoire du microcontrôleur étant ramenées à des clics sur des boutons dans l'IDE (très simple). La communication entre le PC et la carte se fait via le port USB, moyennant installation d'un driver adapté (fourni par ARDUINO). [19]

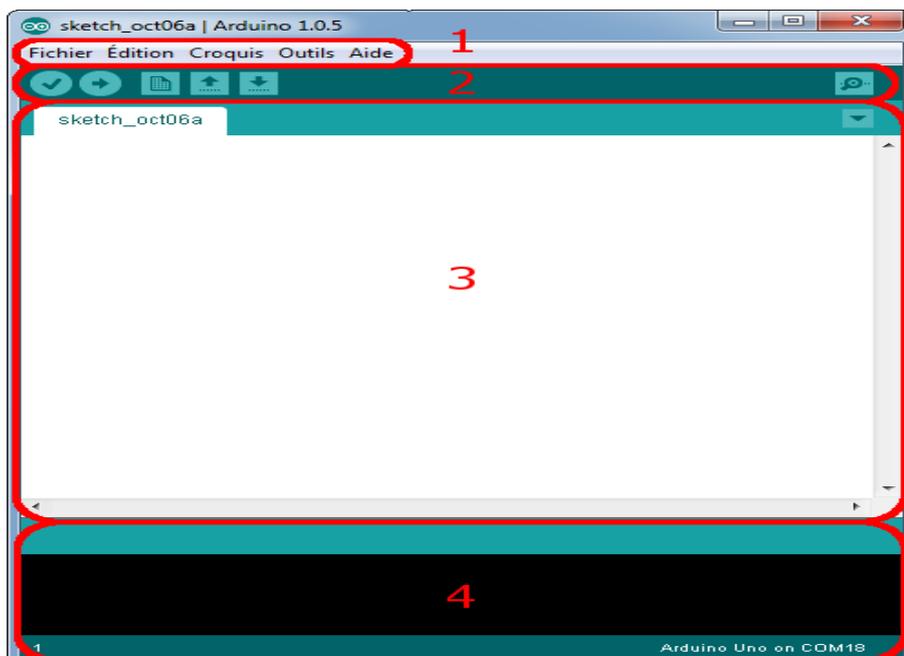


Figure 19 : environnement de développement arduino

### **2.3.6.1 Correspondance :**

**Le cadre 1 :** ce sont les options de configuration du logiciel (Barre de menu).

**Le cadre 2 :** contient les boutons vont nous servir lorsque l'on va programmer nos cartes (Barre des boutons).

**Le cadre 3 :** ce bloc va contenir le programme que nous allons créer (Zone de message des actions en cours).

**Le cadre 4 :** le débogueur corrigé le faute et affiche le message (le Console d'affichage des messages de compilation).

## **2.4 Le module Bluetooth hc-05 :**

### **2.4.1 Définition de Bluetooth : [23]**

**Bluetooth** est une norme de communications permettant l'échange bidirectionnel de données à très courte distance en utilisant des ondes radio sur une bande de fréquence de 2,4 GHz. Son objectif est de simplifier les connexions entre les appareils électroniques en supprimant des liaisons filaires. Elle peut remplacer par exemple les câbles

### **2.4.2 Les broches de hc-05 :**

**VCC :** broche d'alimentation. Typiquement connectée à la broche 5V de l'Arduino.

**GND :** masse. Typiquement connectée à la broche GND de l'Arduino

**RX :** broche de réception. Connecté à la broche de transmission (TX) de l'Arduino

**TX :** broche de transmission. Connecté à la broche de réception (RX) de l'Arduino

**State :** retourne 1 lorsque le module est connecté

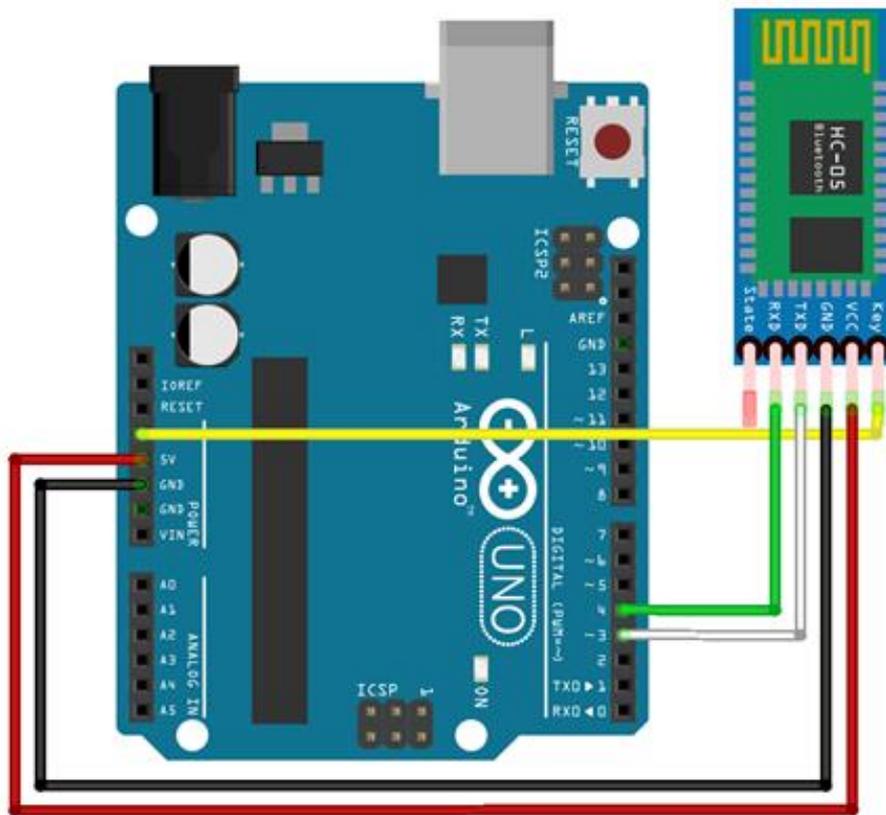
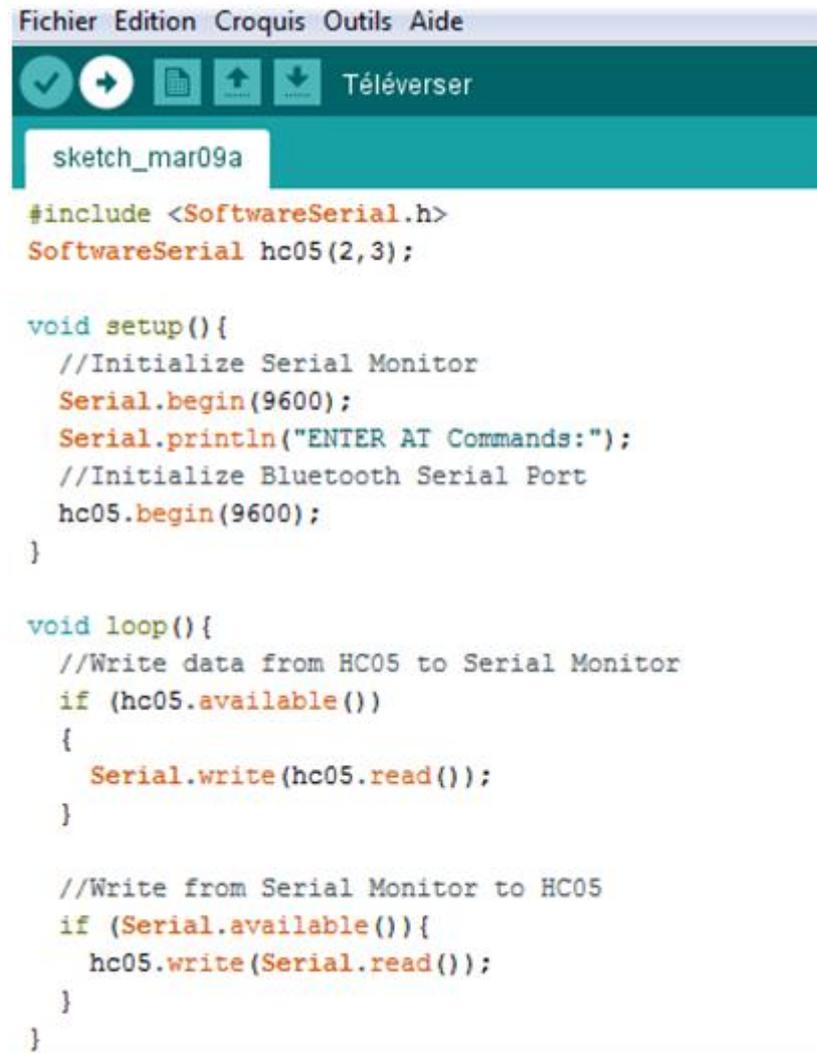


Figure 11 : broches de connexion avec hc-05

### 2.4.3 Configuration du module HC-05:

La configuration du module Bluetooth peut être intéressante pour vérifier le bon fonctionnement du module et pour modifier ses paramètres notamment lorsque vous voulez utiliser le module en mode maître.

## 2.4.4 Code de configuration Bluetooth hc-05:



```
Fichier Edition Croquis Outils Aide
Téléverser
sketch_mar09a
#include <SoftwareSerial.h>
SoftwareSerial hc05(2,3);

void setup(){
  //Initialize Serial Monitor
  Serial.begin(9600);
  Serial.println("ENTER AT Commands:");
  //Initialize Bluetooth Serial Port
  hc05.begin(9600);
}

void loop(){
  //Write data from HC05 to Serial Monitor
  if (hc05.available())
  {
    Serial.write(hc05.read());
  }

  //Write from Serial Monitor to HC05
  if (Serial.available()){
    hc05.write(Serial.read());
  }
}
```

Figure 12 : code de configuration du module hc-05

## 2.4.5 Commandes de configuration :

Pour tester la communication, tapez **AT** dans le monitor série de l'IDE Arduino. Si tout va bien, le module doit répondre OK.

### 2.4.5.1 Tester le programme pour allumer une led avec hc-05 :

```

#include <SoftwareSerial.h>
int led=13;
int RX=2;
int TX=3;
SoftwareSerial bluetoothSerial (TX,RX);
char c;

void setup() {
  bluetoothSerial.begin(9600);
  Serial.begin(9600);
  pinMode(led,OUTPUT);

}

void loop() {
  if (bluetoothSerial.available()) {
    c=Serial.read();
    Serial.println(c);
    if(c=='1'){
      digitalWrite(led.HIGH);

    }
    else if (c=='2'){
      digitalWrite(led.LOW);

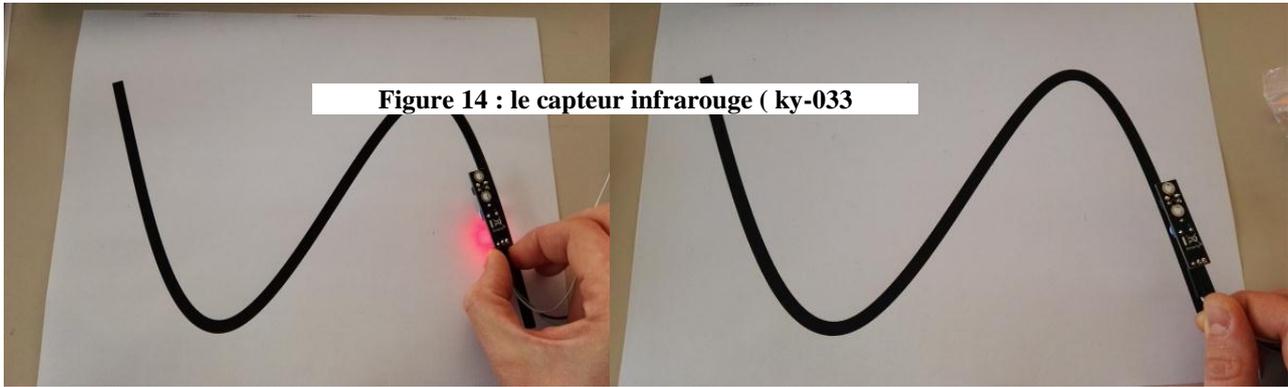
    }
  }
}
}

```

Figure 13 : un code pour tester le module hc-05 avec une led

## 2.5 Le capteur infrarouge ky-033 : [21]

Ce module permet de suivre des lignes réfléchissant ou absorbant la lumière. Lors d'une détection, la sortie du module passe à l'état bas. La portée de détection est réglable.



### 2.5.1 Principe de fonctionnement :

Le principe d'un capteur par réflexion est d'éclairer l'obstacle avec une LED infrarouge, et de mesurer la lumière réfléchie avec une photodiode ou un phototransistor.

### 2.5.2 Spécifications Module KY-033 :

- Tension de fonctionnement: DC 3.3 V-5 V
- Courant de travail:  $\geq 20\text{mA}$
- Distance de détection: 2-40 cm
- Interface IO: interfaces 3 fils
- Angle efficace:  $35^\circ$

### 2.5.3 Broches de connexion :

- ✓ Signal : pour transmettre le signal mesuré.
- ✓ Vcc : pour l'alimentation
- ✓ Gnd : ground

### 2.5.4 Principe de fonctionnement général :

Le principe d'un capteur (infrarouge) par réflexion est d'éclairer l'obstacle avec une LED infrarouge, et de mesurer la lumière réfléchie avec une photodiode ou un phototransistor. Il détecte une longueur d'onde ou lumière consiste de rayonnement électronique avec une longueur d'onde spéciale cette gamme représente la bande infrarouge .L'énergie mise par un objet atteint le capteur IR par intermédiaire du système optique de l'appareil, qui son tour focalise l'énergie sur ses détecteurs convertissent l'énergie en un signal électronique cette valeur peut être visualisé sur le capteur infrarouge ou converti en tant qu'entrée numérique et transmis à un microcontrôleur pour le traitement [20].

## 2.6 Partie de puissance

### 2.6.1 L293D : [22]

Le L293D est un circuit intégré qui commande de moteur en pont en H. composé de plusieurs transistors qui permet d'activer la rotation des moteur a plusieurs direction Le L293D est aussi connu pour être une sorte de Pont-H. Typiquement, un pont H est un circuit électronique qui permet d'appliquer une tension en sortie sur une charge dans une direction ou l'autre. Par exemple, un moteur. Les opérations de moteur de deux moteurs peuvent être contrôlées par la logique d'entrée aux broches input. Et Les broches d'activation enable.

Le L293D un circuit intégré monolithique, Cela veut dire que ce circuit intégré peut être utilisé pour des moteurs DC et alimentation jusqu'à 36 Volt[22]

#### 2.6.1.1 Schéma des broches avec l'explication :

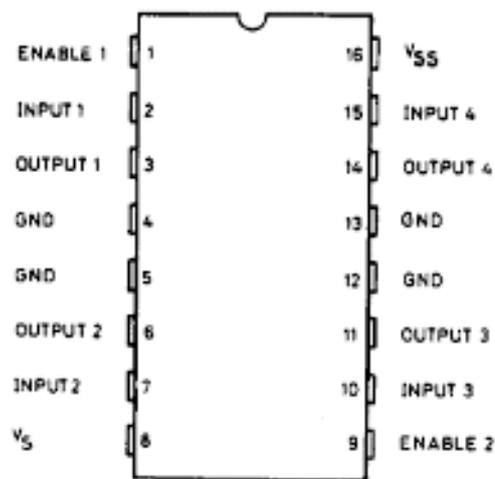


Figure 15 : les broches de l293D

**VSS** :Alimentation de la commande 5V à raccorder la broche 5V de l'arduino

**VS** : Alimentation de puissance moteurs

**GND** : Doit être raccordé à la masse (GND) de la source d'alimentation de puissance VS (donc le négatif de l'accumulateur) état la masse de la source d'alimentation de VSS (donc GND d'arduino).

**OUTPUT1, OUTPUT2** : Broches à raccorder à la charge (le moteur).

**INPUT1, INPUT2** : Broche de commande du Pont-H. Se raccorde à Arduino.

ENABLE : permet d'envoyer (ou pas) la tension sur les sorties du moteur via OUTPUT1 & OUTPUT2.

ENABLE1 commande l'activation du premier Pont-H.

ENABLE2 commande l'activation du deuxième Pont-H.

### 2.6.1.2 Le pont H – principe :

Dans son principe de base, le pont H est un assemblage de 4 transistors monté de telle façon que le courant puisse passer soit dans un sens, soit dans l'autre dans la charge (le moteur par exemple).

### 2.6.1.3 Principe de fonctionnement d'un pont h avec enable :

enable	Input 1	Input 2	FONCTION
HIGH	Low	High	Tourne dans le sens horlogique
HIGH	High	Low	Tourne dans le sens anti horlogique
HIGH	Low	Low	Stop
HIGH	High	High	Stop
Low	Non applicable	Non applicable	Stop

Tableau 3 : principe de fonctionnement d'un pont H avec enable

## 2.6.2 Moteur DC :[24]

Le moteur à courant continu est une machine électrique : convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique. En fonctionnement moteur l'énergie électrique est transformée en énergie mécanique.

### 2.6.2.1 Les éléments du moteur dc : [25]

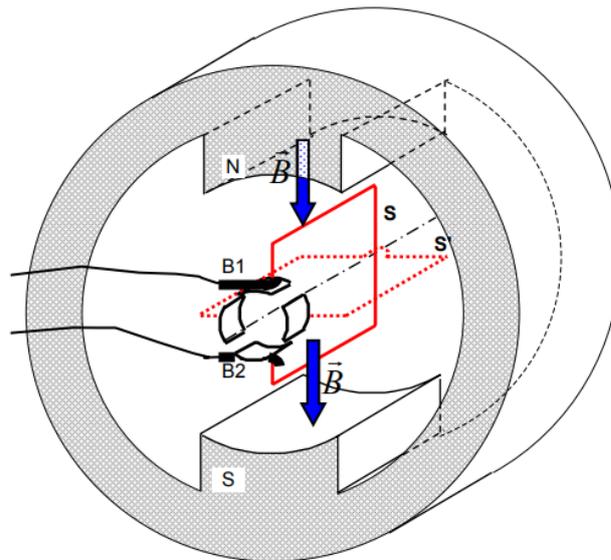
**Un stator** : (partie fixe avec des aliments ou des électro-aimants) qui possède un pôle nord et un pôle sud et qui engendre donc un champ d'induction B entre eux (pour cette raison on l'appelle inducteur).

**Un rotor** : (partie tournante) qui subit l'induction, (c'est pourquoi on l'appelle l'induit.)

Dans sa version très simplifiée ci-contre, cette partie tournante constituée de deux spires s et s'

**Un organe de liaison** : entre l'induit et l'extérieur de la machine les conducteurs de l'induit sont soudés à des lames conductrices sur lesquelles frottent des éléments conducteurs fixes appelés les balais. L'ensemble des lames constitue le collecteur.

Dans sa version simplifiée ci-contre, le collecteur possède quatre lames. Il y a deux balais fixes (B1 et B2) qui frottent sur ces lames et sont reliés à l'extérieur de la machine.



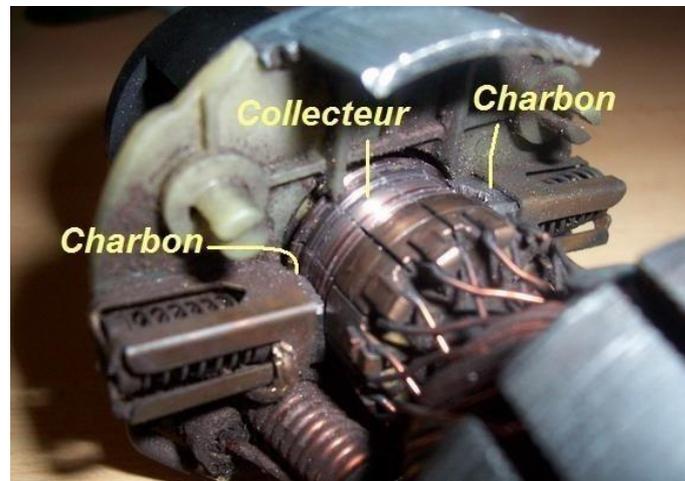
**Figure 16 : constitution du moteur DC**

### 2.6.2.2 Principe de fonctionnement d'un moteur à courant continu : [26]

Le stator d'un moteur à courant continu est la partie fixe du moteur (*statique* = qui ne bouge pas): on fait passer un courant dans le bobinage du stator et c'est lui qui crée (qui induit) un champ magnétique. Le stator pose le décor pour le rotor qui se retrouve ainsi plongé dans ce champ magnétique. Le stator (inducteur) crée un champ magnétique  $B$  appelé champ inducteur. On le note  $B_s$  (s comme... stator !)

Le stator peut être fait d'aimants permanents (comme ceux qui collent vos papiers sur la porte du frigo) ou d'électroaimants qu'il faut alimenter. Dans ce cas, la relation électrique s'écrit : La deuxième partie est le rotor alors là il est la partie en rotation du moteur. C'est lui qui tourne. Il est constitué du bobinage induit. Il faut alimenter cette bobine pour la transformer en électroaimant qui entrera en interaction avec le stator. Si on n'alimentait pas le rotor, il ne serait l'objet d'aucune force et ne tournerait pas.

Un système de frottement spécial permet d'alimenter le rotor : des balais (ou charbons montés sur des ressorts) frottent sur les contacts en rotation : le collecteur.



**Figure 17 : Un système de frottement du moteur a courant contenu**

Charbons de remplacement pour moteur à courant continu

Le collecteur est un ensemble de plages métalliques qui font contact avec les charbons. Il appartient au rotor. Le frottement des charbons font qu'ils s'usent : ils sont montés sur ressort pour garantir un contact même lorsqu'ils raccourcissent à cause de l'usure :

$$E = k \Phi \Omega$$

Le courant du rotor passe par les charbons. La tension  $E$  aux bornes du rotor est proportionnelle à la vitesse de rotation du rotor :

Oméga est la vitesse de rotation,  $k$  est une constante liée au moteur lui-même et  $\phi$  est le flux magnétique.

### 2.7.1 Algorithme de suivre la ligne :

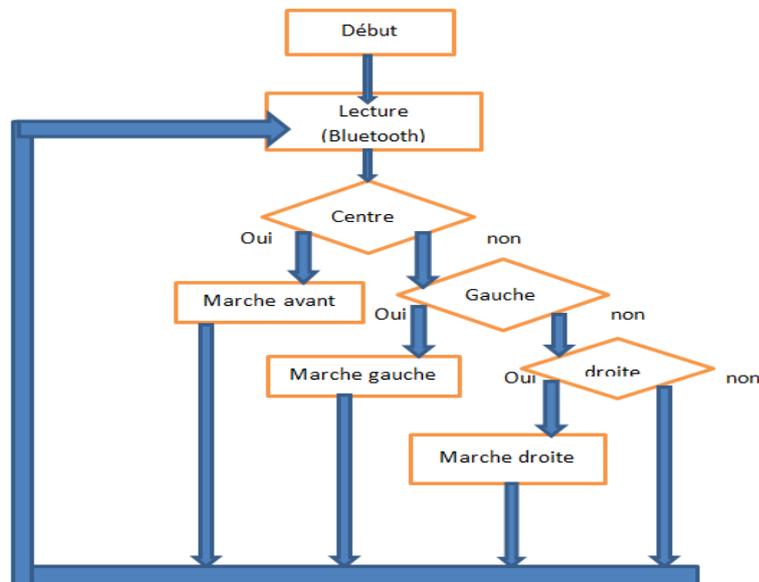


Figure 18 : algorithme d'un code pour suivre la ligne

### Conclusion :

Dans ce chapitre, nous nous sommes concentrés sur le côté matériel de notre projet alors là l'étude des composants électroniques et son principe de fonctionnement, en particulier la carte Arduino uno et tous ses composants, et nous avons étudié en profondeur le système de contrôle commande et le principe de fonctionnement de notre système, et le principe de suivre la ligne. Et la manière de navigation et de perception avec l'adaptation dans l'environnement du robot, en plus le coté de programmation. Le troisième chapitre sera consacré au la Conception et développement d'une application à distance d'un robot mobile sous Android.

## **Chapitre 3 :**

# **Conception et développement d'une application à distance d'un robot mobile sous Android**

### **3.1 Introduction :**

Dans le monde actuel, les téléphones intelligents font désormais partie intégrante de notre vie et sont utilisés dans tous les domaines, y compris les plates-formes Android, qui sont à leur tour largement utilisées comme médias sociaux, jeux, etc.

Dans ce chapitre, nous nous intéressons à l'androïde et l'application android, à son historique et à ses types et versions, ainsi qu'à ses utilisations.

Nous avons conçu et développé l'application Android pour contrôler notre robot. Nous sommes appuyés sur le programme (MIT App inventor) qui permet aux débutants d'apprendre et de développer des applications pour Android, l'objectif principal étant de montrer comment l'application est complétée et comment elle est développée, en plus de fonctionner avec Android, ce qui permet d'apprendre pendant une courte période. La programmation dépend plusieurs étapes de conception, nous allons expliquer tout cela en contexte.

### **3.2 Définition de smartphone :[33]**

Le smartphone ou «téléphone intelligent» désigne un téléphone mobile doté de fonctionnalités évoluées qui s'apparentent à celles d'un ordinateur : navigation sur Internet, lecture de vidéos, jeux vidéo, courrier électronique, vidéoconférence, bureautique légère...

Muni d'un processeur puissant, souvent multi cœur, il embarque une série de capteurs (boussole, GPS ...) qui lui permettent de faire fonctionner des applications dédiées à l'activité physique, de navigation assistée ainsi que des jeux que l'on peut contrôler d'un simple mouvement. Les smartphones sont généralement dotés d'un appareil photo-vidéo et d'une caméra frontale dont les performances ne cessent de progresser.

#### **3.2.1 Caractéristiques : [34]**

- Prix
- Écran
- Système d'exploitation
- Autonomie
- Appareil photo
- Mémoire
- Puissance

### 3.2.2 Systèmes d'exploitation de smart phone :[34]

Il existe trois principaux systèmes d'exploitation sur les téléphones:

#### 3.2.2.1 Google Android : [34]

Est le système d'exploitation le plus populaire sur les téléphones mobiles, et il y a des majorité de téléphones modèles de dizaines de fabricants qui l'utilisent. Il ressemble et fonctionne très similaire à l'iPhone OS (iOS) -pour lancer une application ou démarrer une fonction du téléphone, vous soit appuyez sur une icône sur l'écran ou appuyez sur un matériel bouton sur le côté. La plupart des téléphones Android sont personnalisés par leur fabricant, Les fabricants de téléphones se réfèrent à cela comme une interface,

. En conséquence, Android les téléphones sont semblables les uns aux autres, mais pas identiques. Différents modèles de téléphone viennent également avec différents applications préinstallées.



Figure 19 : le logo officielle de l'android

#### 3.2.2.2 Interface Apple OS :

Apple iOS vient sur iPhone est le même système qui est utilisé sur l'iPad d'Apple. C'est très Système simple - il suffit de toucher un icône pour lancer un téléphone particulier Fonctionnalité ou application.



Figure 20 : le logo Apple iOS

### 3.2.2.3 Windows Phone

Windows phone est fabriqué par Microsoft et il ressemble beaucoup à la Nouvelle interface Windows 8. Émouvant'un des blocs à l'écran commence Up différentes applications et téléphone fonctionnalités.



Figure 21 : le logo de Windows phone

## 3.3 Histoire d'Android : [27]

L'histoire commence bien avant l'arrivée de la firme de MountainView sur le système d'exploitation. En effet, tout a débuté avec une société américaine du nom d'Android justement, fondée en 2003, Android est racheté par Google en 2005, Création du consortium Open Alliance Handset en 2007 Regroupement de 84 acteurs industriels en télécom, composants électroniques et logiciels Google, Samsung, LG, T-Mobile, ARM, Intel, Toshiba, Acer... Première smartphone équipée du système Android – T-Mobile



Figure 22 : Première smartphone équipée du système Android – T-Mobile G1

### 3.3.1 Définition de l'Android : [28]

Android est un système d'exploitation mobile fondé sur le noyau Linux et développé actuellement par Google, conçu pour les smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs comme les télévisions, les voitures (Android Auto), et les smart Watch.

### 3.3.2 Les versions d'Android : [29]

Version Android	date
Android 1.0	09/2008
Petit Four 1.1	02/2009
Cupcake 1.5	02/2009
Donut 1.6	09/2009
Gingerbread 2.3	12/2010
Honeycomb 3.0	02/2011
Ice Cream Sandwich 4.0.1	10/2011
Jelly Bean 4.1	07/2012
KitKat 4.4	10/2013
Lollipop 5.0	10/2014
Marshmallow 6.0	05/2015
Nougat 7.0	09/2016
Oreo 8.0	08/201

Tableau 2 : les versions android



Figure 23 : les versions android

### 3.4 Définition de MIT inventor : [30]

APP INVENTOR 2 pour Android est un IDE (environnement de développement intégré) qui permet la création d'applications destinées à des systèmes équipés de plateformes Android (Smartphones ou tablettes). Il est basé sur l'assemblage de blocs : on manipule des « briques » et on les agence comme avec Scratch pour former une application. La prise en main est rapide par des élèves, notamment par ceux qui ont été initiés à Scratch. Il a été développé par Google : il vous faudra donc un compte Gmail pour l'utiliser. Il est actuellement géré par le MIT. App Inventor est un outil facile à utiliser pour créer des applications Android simples et complexes, Les applications peuvent facilement être portées sur votre téléphone, partagées avec d'autres, Pour ceux qui souhaitent apprendre un langage de programmation, MIT App Inventor peut servir de excellent pont pour acquérir des compétences de codage plus complexes. Qui utilise les méthodes de glisser, déposer et connecter les pièces du puzzle au programme applications



Figure 24 : L'interface officielle de MIT APP Inventor

### 3.4.1 Les avantages d'App Inventor sont nombreux :

- la prise en main est extrêmement simple : en quelques minutes, l'utilisateur entre déjà en création
- l'application est exécutée immédiatement, soit sur un émulateur, soit sur son propre téléphone ou/et ordinateur
- rien n'est à installer sur son ordinateur : tout se fait sur le cloud
- au niveau mobile, il suffit d'installer une application
- l'interface simplifiée en seulement deux parties : une page de conception de l'écran et une page de programmation.

### 3.4.2 Comment se connecter à MIT inventor et l'utiliser : [31]

Il vous faudra donc un compte Gmail pour l'utiliser. Il est

Actuellement géré par le MIT (Massachusetts Institute of Technology) et reste gratuit.

- Accéder au site <http://ai2.appinventor.mit.edu>

- Se connecter avec son adresse Gmail.

- on clique sur créeApps!

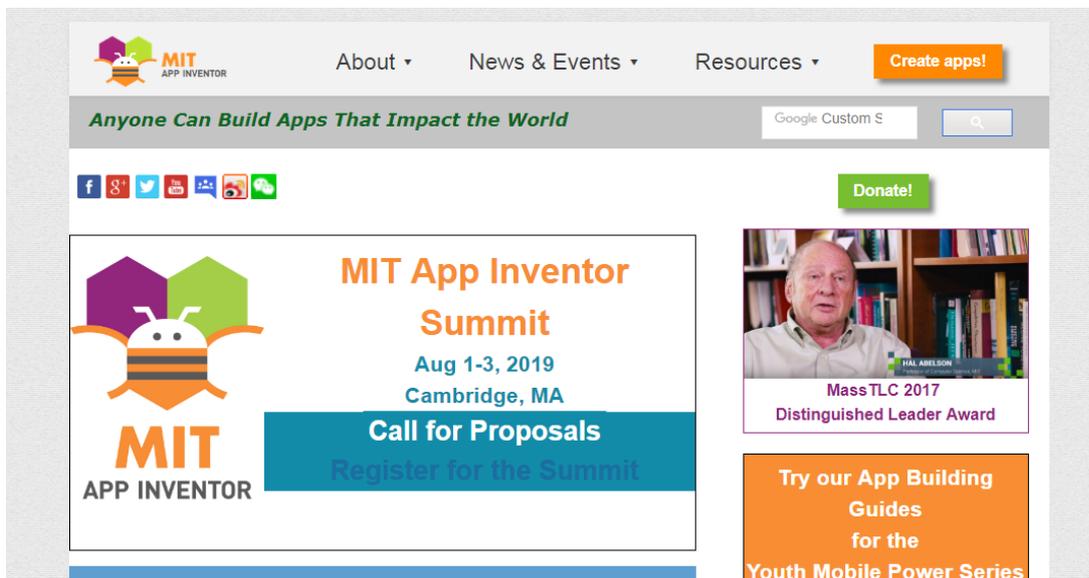


Figure 25 : Page officielle de mit inventor

- Il est possible de changer la langue de l'interface



Figure 26 : le menu pour choisir la langue

Cliquer sur « Commencer nouveau projet » :



Figure 27 : Commencer nouveau projet

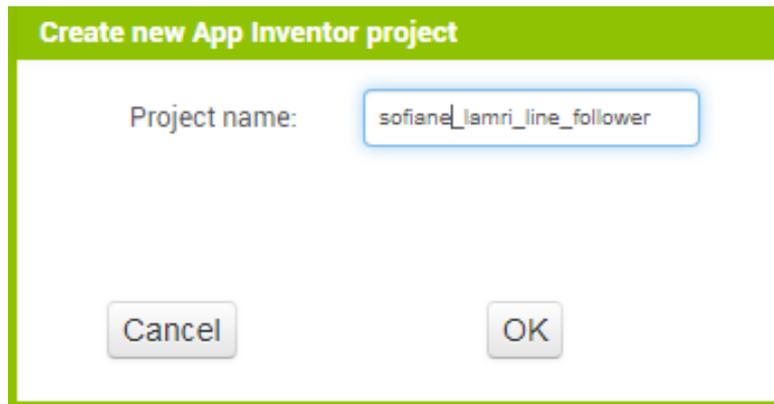


Figure 28 : créations un nouveau projet dans mit App inventor

### 3.4.2.1 L'interface graphique :

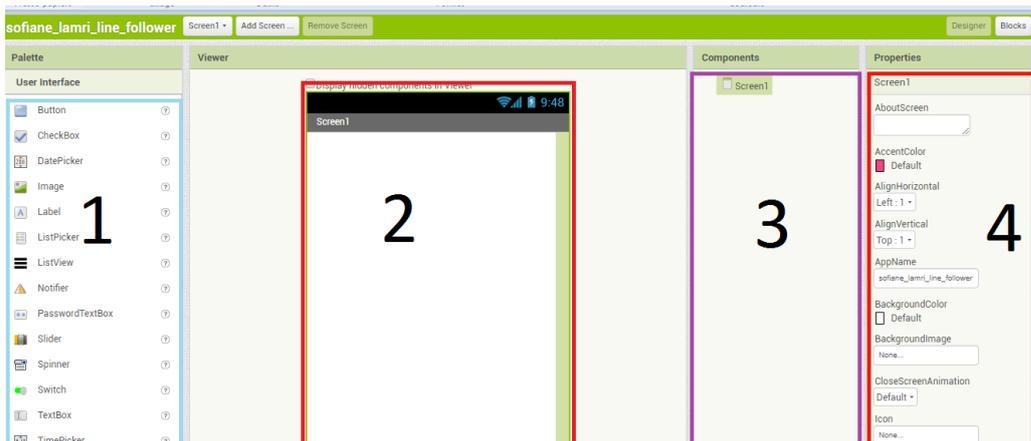


Figure 29 : l'interface graphique de mit App inventor

On obtient la fenêtre suivante en mode « Designer » :

**Cadre 1** : palette des objets disponibles on glisse l'objet disponibles sur la zone de travail.

**Cadre 2** : Zone de travail elle représente ce qui sera affiché sur l'écran du smartphone.

**Cadre 3** : fenêtre d'exploration on clique sur l'objet permet on utilisant les fenêtres des propriétés pour la modification.

**Cadre 4** : fenêtre de propriété.

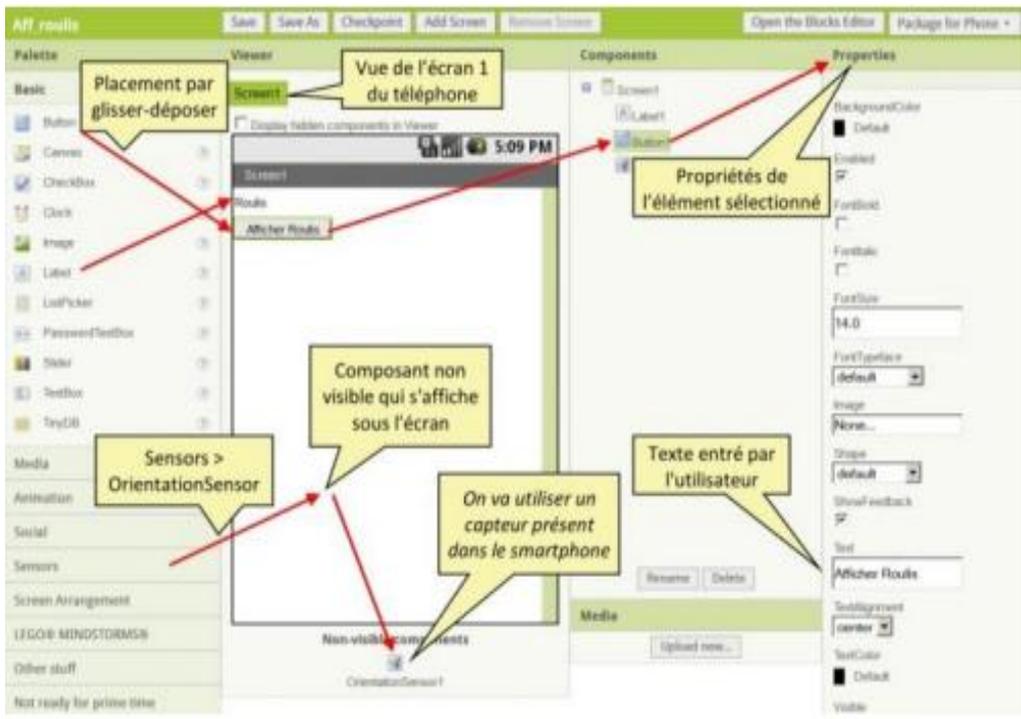


Figure 30 : les éléments de l'aspect graphique du mit App inventor

3.4.2.2 l'aspect des blocs :

Une fois l'allure de notre désigne est créée, il est nécessaire de décrire son comportement : Pour cela, il faut cliquer sur « Blocks » en haut et à droite de la page :

3.4.3 Les étapes de créations l'application :

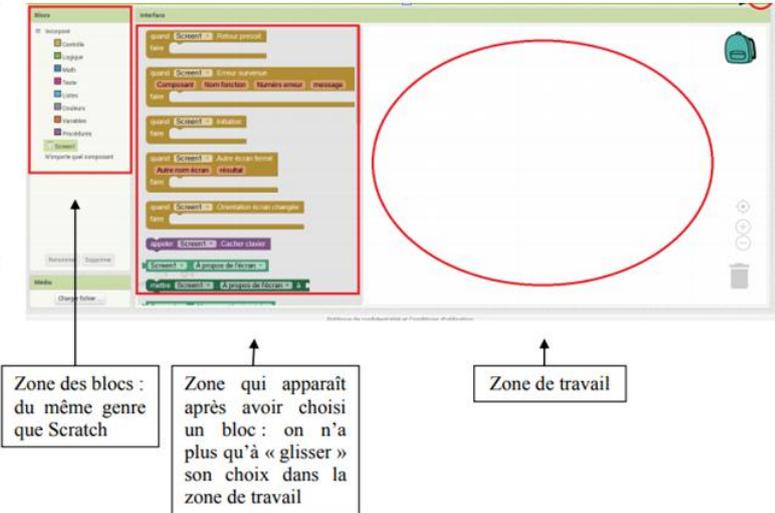


Figure 31 : interface de bloc (coté de programmation)

### 3.4.3.1 Labels :

Sont des composants utilisés pour afficher du texte. **Label** affiche le texte spécifié par la propriété Texte. D'autres propriétés, qui peuvent toutes être définies dans Designer ou dans l'éditeur de blocs, contrôlent l'apparence et l'emplacement du texte[32].



Figure 32 : le label

### 3.4.3.2 Horizontal arrangement :

Utilisez un composant de disposition horizontale pour afficher un groupe de composants disposés de gauche à droite.

Ce composant est un élément de mise en forme dans lequel vous placez les composants à afficher de gauche à droite. Si vous souhaitez afficher les composants les uns sur les autres utilisez plutôt Vertical Arrangement



Figure 33 : l'horizontal arrangement

### 3.4.3.3 Listpicker :

On a utilisé le bouton **listpicker** c'est Un bouton qui, lorsque l'utilisateur clique dessus, affiche une liste de textes parmi lesquels l'utilisateur peut choisir. Les textes peuvent être spécifiés via le concepteur ou l'éditeur de blocs, par exemple (choix 1, choix 2, choix 3...)

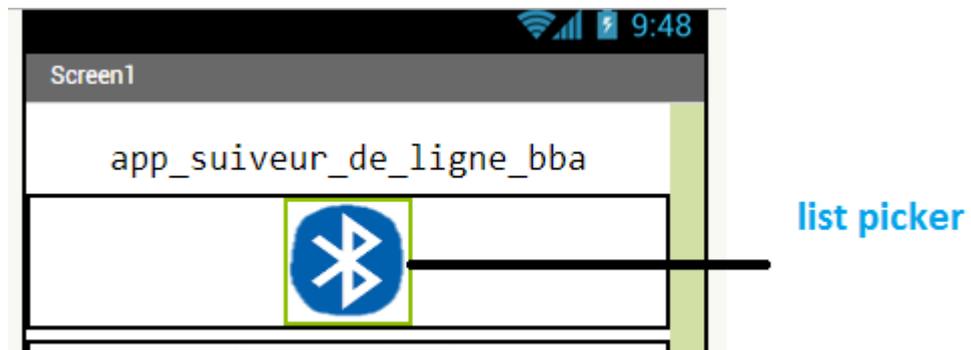


Figure 34 : List picker

[32]

### 3.4.3.4 Table Arrangement :

Utilisez un composant d'agencement de table pour afficher un groupe de composants sous forme de tableau. Ce composant est un élément de mise en forme dans lequel vous placez les composants à afficher sous forme de tableau. [32]

Nous pouvons mettre des boutons à l'intérieur de table arrangement, Nous allons

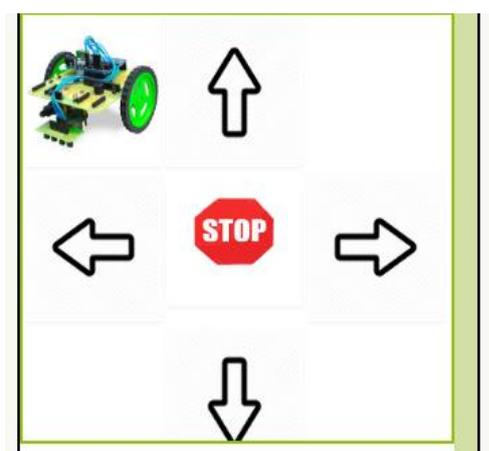


Figure 35 : table arrangement

Télécharger des images pour les directions et une image pour suivre la ligne à et un bouton stop Pour désactiver la fonction de robot Android à l'intérieur de table arrangement.

### 3.4.3.5 Non visible –component :

#### 3.4.3.5.1 Bluetooth client : [32]

La connectivité à partir de la palette, se sélectionnez client Bluetooth client et faites le glisser vers le panneau de visualisation, notez que ceci est le composant caché, la seule différence entre le client bluetooth et le service Bluetooth est qu'il existe un bouton de connexion au périphérique (List picker).



**Figure 36 : Bluetooth client  
(composant non visible )**

#### 3.4.3.5.2Le Clock :

C'est un Composant non visible qui fournit l'instant dans le temps en utilisant l'horloge interne du téléphone. Il peut déclencher une minuterie à intervalles réguliers et effectuer des calculs, des manipulations et des conversions de temps [32].

Après avoir suivi toutes les étapes mentionnées précédemment, nous avons lancé notre interface graphique de notre application, qui consiste se connecter avec le module Bluetooth



**Figure 37 : le clock  
(Composant non visible) [32]**



**Figure 38 : l'aspect finale de notre interface graphique**

### **3.5 Conclusion :**

Dans ce chapitre nous avons fait un aperçu général sur les smartphones et les systèmes d'exploitation, surtout les applications Android et les différentes méthodes pour concevoir et développer une application Android à partir d'un logiciel comme MIT App Inventor, ainsi que les divers composants d'une application et les différentes étapes et les différentes phases de développement de l'application Android.

# **Conclusion générale**

## Conclusion générale :

En robotique, on distingue deux principaux types de robots : les robots manipulateurs et les robots mobiles. Les robots manipulateurs ont une base fixe contrairement aux robots mobiles qui peuvent se déplacer. Les robots mobiles ont alors une place particulière en robotique. Leur intérêt réside dans leur mobilité qui ouvre des applications dans de nombreux domaines. Comme les robots manipulateurs, ils sont destinés à assister l'homme dans les tâches pénibles (transport de charges lourdes), monotones ou en ambiance hostile (nucléaire, marine, spatiale, lutte contre l'incendie, surveillance...).

Dans ce travail nous nous sommes principalement intéressés à la conception et à la réalisation d'un robot mobile capable de suivre un trajet bien déterminé en toute autonomie. Il s'agit plus précisément d'un robot suiveur de ligne. De ce fait, nous avons développé dans un premier temps une application Android qui permet de commander à distance, via une liaison sans fil (Bluetooth), un robot mobile à deux roues motorisées et une roue folle. Cette application permet, outre la mise en marche et l'arrêt du robot, de positionner celui-ci sur la ligne noire désirée (trajet choisi). D'autre part, nous avons réalisé une carte de commande d'un robot mobile à roues motorisées. En effet, nous avons développé un système de commande autour d'un microcontrôleur ATMEGA à base d'Arduino UNO. Celui-ci permet de générer les commandes appropriées à un pont H qui entrainera à son tour deux moteurs pour commander un robot dans deux sens de marche (avant et arrière). Aussi, la carte de commande réalisée permet de faire tourner le robot à droite ou à gauche en contrôlant le sens de rotation des moteurs droite et gauche. De plus, notre robot a été équipé de trois capteurs IR type KY-033 (placés à l'avant du robot) pour identifier la ligne noire (tracée sur le sol) afin d'obliger le robot de suivre cette ligne.

Il est à noter que notre système ainsi conçu a été réalisé et testé sur un petit robot (à deux roues motorisées) dans les locaux de nos laboratoires où le fonctionnement a été bel et bien comme prévu. Néanmoins, nous envisagerions de conjuguer le concept suiveur de ligne et l'algorithme Q-Learning pour faire l'auto-apprentissage de trajectoire d'un robot, afin d'optimiser un trajet donné, en évitant ainsi que le robot dévie de son trajet à cause de ses roues non parfaitement équilibrées. Ceci constitue en fait une perspective à notre présent travail.

# Bibliographie

[1] <https://fr.wikipedia.org/wiki/Robot>

[1] :<https://sites.google.com/site/technondba3e2gr112013/ci-2-recherche-de-solutions-techniques/historique-des-robots>

[2] DavidFilliat. Robotique Mobile. Engineering school.Robotique Mobile, ENSTA Paris Tech, 2011, pp.175.

[3] Mr Abdelmadjid BENMACHICHE <<Reconnaissance des Formes et Intelligence Artificielle (RFIA)>> Présenté en vue de l'obtention du Diplôme de Doctorat 3ème Cycle LMD

[4] Chapitre I Généralités et Architecture des robots.

[5] Robot différentiel suiveur de ligne Master ASE – Projet S1 - Pierre Bonnet, John Klein - Sept/Déc 2011 –université Lille

[6] SLIMANE Noureddine <<SYSTEME DE LOCALISATION POUR ROBOTS MOBILES>> SPECIALITE : CONTROLE Discipline : Robotique Pour l'obtention du grade de DOCTEUR D'ETAT DE L'UNIVERSITE DE BATNA Soutenu le : 23 / 11 / 2005

[7] Mr Abdelmadjid BENMACHICHE << Approche de Navigation Coopérative et Autonome des Robots Mobiles (Application sur un chantier de construction)>> en vue de l'obtention du Diplôme de Doctorat 3ème Cycle LMD BADJI MOKHTAR UNIVERSITY -ANNABA 2016

[8] Michel DEVY, directeur de recherche CNRS, et Florent LAMIRAUX, chargé de recherche CNRS, du Laboratoire d'Analyse et d'Architecture des Systèmes (unité CNRS, associée à l'UPS).michel@laas.fr et [florent@laas.fr](mailto:florent@laas.fr)

[9]DOCTORANT : CYRIL ROUSSILLON cyril.roussillon@laas.fr Directeur de thèse : Simon Lacroix - [simon.lacroix@laas.fr](mailto:simon.lacroix@laas.fr)<<ROBOT TERRESTRE Au profit du combattant, en répondant au besoin des 3 D (dull, dirty, Dangerous) >>.

[10] Bernard BAYLE <<Robotique mobile >> Télécom Physique Strasbourg Université de Strasbourg

[11] : <<Généralité sur le robot mobile>>[https://www.univ-saida.dz/butec/doc\\_num.php?explnum\\_id=367](https://www.univ-saida.dz/butec/doc_num.php?explnum_id=367)

[12] Bernard BAYLE <<**Robotique mobile**>>Télécom Physique Strasbourg Université de Strasbourg

[13] : Olivier KERMORGANT Thèse soutenue à Rennes le 28 novembre 2011 <<Fusion d'informations multi-capteurs en asservissement visuel>>

<https://images.google.com/>

[14] : johnnussey<< arduino pour les nulles>>.

[15] La carte Arduino UNO PDF

- [16] :Mr. DJAFRI Menad Mr. HADJI Slimane << Etude et Réalisation d'une Carte Arduino>>.
- [17] : Erik Bartmann<< LE GRAND LIVRE D'ARDUINO >>.
- [18] :LECHALUPÉ Julien <<Cours d'initiation à Arduino>> Université Paul Sabatier Mai 2014
- [19] :B. Cottenceau B311 ISTIA< bertrand.cottenceau@univ-angers.fr<<Carte ARDUINO UNO Microcontrôleur ATmega328>>
- [20] :[www.didel.com/doc/sens/DocIr.pdf](http://www.didel.com/doc/sens/DocIr.pdf)
- [21] <<KY-033 Module suiveur de ligne>>  
[http://sensorkit.fr.jov-it.net/index.php?title=KY-033 Module suiveur de ligne](http://sensorkit.fr.jov-it.net/index.php?title=KY-033+Module+suiveur+de+ligne)
- [22] :<<CONTRÔLE MOTEUR DC VIA L293D (H BRIDGE DRIVER, CIRCUIT INTÉGRÉ PONT H)>>  
<https://arduino103.blogspot.com/2011/06/controle-moteur-dc-via-l293d-h-bridge.html>
- [23] :<https://fr.wikipedia.org/wiki/Bluetooth>
- [24] :<https://www.techno-science.net/definition/3190.html>
- [25] :<< machine a courant contenu michelpiou>>Edition 20/10/2015.
- [26] :<<le moteur a courant contenu principe>>  
<https://www.astuces-pratiques.fr/electronique/le-moteur-a-courant-continu-principe>
- [27 ]:James Douglass Lefruit Ingénieur de Recherche, INRIA Grenoble <<Android Introduction générale >>.
- [28]: <https://fr.wikipedia.org/wiki/Android>
- [29]:Jean-François Lalande - Novembre 2017 <<Développement Android>>.
- [30] :Derek Walter Mark Sherman <<Learning MIT App Inventor >>.
- [31] : Jean-François THULLIER – Novembre 2018 <<APP INVENTOR 2 EN PHYSIQUE-CHIMIE CRÉER DES APPLICATIONS ANDROID>>
- [32] :<http://ai2.appinventor.mit.edu>
- Un véhicule automoteur est un véhicule assurant sa propre propulsion et le transport de sa charge utile.
- [33] : <https://www.futura-sciences.com/tech/definitions/smartphone-smartphone-1954/>
- [34] :<<les smartphones PDF >>pmtic\_env\_num\_systexpl\_mobile\_smartphones.pdf
- [35] : <https://sites.google.com/site/technondba3e2gr112013/ci-2-recherche-de-solutions-techniques/historique-des-robots>

# Annexe

## Le code source ARDUINO :

```
#include <SoftwareSerial.h>

SoftwareSerialbluetooth(11,10); //RX,TX

char data=0;//char used to store the input data

int IR=13;

int IN1=7;

int IN2=6;

int IN3=5;

int IN4=4;

int IR2=9;

int IR3=8;

intval=255;

int val2=155;

voidarriere(){

analogWrite(IN1,0);

analogWrite(IN2,val);

analogWrite(IN3,0);

analogWrite(IN4,val);}

voidstp(){

analogWrite(IN1,0);

analogWrite(IN2,0);

analogWrite(IN3,0);

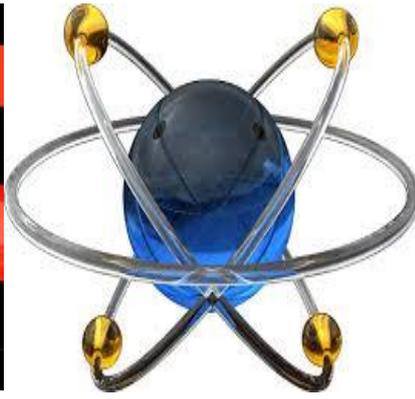
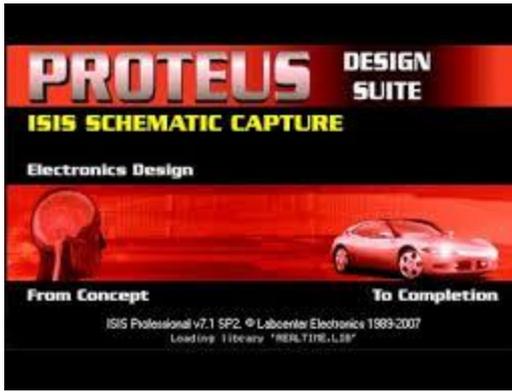
analogWrite(IN4,0);
```

```
}  
  
void av(){  
  analogWrite(IN1,val);  
  analogWrite(IN2,0);  
  analogWrite(IN3,val);  
  analogWrite(IN4,0);  
}  
  
void droite(){  
  analogWrite(IN1,0);  
  analogWrite(IN2,0);  
  analogWrite(IN3,val);  
  analogWrite(IN4,0);  
}  
  
void gauche(){  
  analogWrite(IN1,val);  
  analogWrite(IN2,0);  
  analogWrite(IN3,0);  
  analogWrite(IN4,0);  
}  
  
void lin(){  
  while(data=='f'){  
    if (!digitalRead(IR) &&digitalRead(IR2) &&digitalRead(IR3) ){  
      char data2=bluetooth.read();  
      analogWrite(IN1,val);  
      analogWrite(IN2,0);  
      analogWrite(IN3,val);  
      analogWrite(IN4,0);  
    }  
  }  
}
```

```
if(data2=='e'){
return 0;}}
if (digitalRead(IR) &&digitalRead(IR2) &&digitalRead(IR3)){
char data2=bluetooth.read();
analogWrite(IN1,0);
analogWrite(IN2,0);
analogWrite(IN3,0);
analogWrite(IN4,0);
if(data2=='e'){
return 0;}}
else if (digitalRead(IR) && !digitalRead(IR2) &&digitalRead(IR3) ){
char data2=bluetooth.read();
analogWrite(IN1,val);
analogWrite(IN2,0);
analogWrite(IN3,0);
analogWrite(IN4,0);
if(data2=='e'){
return 0;}}
else if (digitalRead(IR) &&digitalRead(IR2) && !digitalRead(IR3)){
char data2=bluetooth.read();
analogWrite(IN1,0);
analogWrite(IN2,0);
analogWrite(IN3,val);
analogWrite(IN4,0);
if(data2=='e'){
return 0;}}
else if (!digitalRead(IR) && !digitalRead(IR2) &&digitalRead(IR3) ){
```

```
char data2=bluetooth.read();
analogWrite(IN1,val);
analogWrite(IN2,0);
analogWrite(IN3,0);
analogWrite(IN4,0);
if(data2=='e'){ return 0;}}
else if (!digitalRead(IR) &&digitalRead(IR2) && !digitalRead(IR3)){
char data2=bluetooth.read();
analogWrite(IN1,0);
analogWrite(IN2,0);
analogWrite(IN3,val);
analogWrite(IN4,0);
if(data2=='e'){
return 0;}}
else if (!digitalRead(IR) && !digitalRead(IR2) && !digitalRead(IR3)){
char data2=bluetooth.read();
analogWrite(IN1,0);
analogWrite(IN2,val2);
analogWrite(IN3,0);
analogWrite(IN4,val2);
if(data2=='e'){
return 0;}}}}
void setup(){
Serial.begin(9600);
bluetooth.begin(9600);
//define the input and the output for the arduino
pinMode(IR,INPUT);
```

```
pinMode(IR2,INPUT);
pinMode(IR3,INPUT);
pinMode(IN1,OUTPUT);
pinMode(IN2,OUTPUT);
pinMode(IN3,OUTPUT);
pinMode(IN4,OUTPUT);
}
void loop() {
data=bluetooth.read();
if (data=='a'){
av(); }
else if(data=='c'){
droite();}
else if(data=='d'){
arriere();}
else if(data=='e'){
stp();}
else if(data=='b'){
gauche();
}
else if(data=='f'){
lin();}}
```



**MIT**  
APP INVENTOR