



République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur
et de la Recherche scientifique



Université Echahid Hamma Lakhdar d'El-Oued
Faculté de Technologie

Mémoire de Fin d'Étude
En vue de l'obtention du diplôme de
MASTER ACADEMIQUE

Domaine : Technologie
Filière : Génie électrique
Spécialité : Télécommunications

Thème

Réalisation d'un robot mobile contrôlé par un smartphone

Présenté par :

HACHEM Abdelkader

SOUALAH Badr eddine

Soutenu le juin 2018

Devant le jury composé de :

Mr TOUHAMI Ridha

Mr AJGOU Riadh

Mr GHENDIR Said

Président

Encadreur

Examineur

Année universitaire :2017/2018



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciements

Nous exprimons notre profonde gratitude à Dieu Tout-Puissant qui nous a aidés à faire ce travail humble.

Nous exprimons notre profonde gratitude à nos parents pour leurs encouragements, leur soutien et leur sacrifice.

Nous vous remercions notre guide M. Ajgou Riadh pour ses efforts, pour nous aider, conseiller, encourager et corriger.

Nous aimerions remercier les membres du jury d'avoir accepté de revoir notre travail.

Nous remercions également tous les membres de la faculté de génie électrique qui ont contribué à notre formation universitaire. Sans oublié tous nos amis.

Et à tous ceux qui ont contribué directement ou indirectement à la réalisation de ce travail, nous retrouvons ici notre sincère reconnaissance.



Dédicace

Je dédie ce travail et nos profonde gratitude à :

A ma mère source de tendresse et d'amour.

A mon père lui je tant adoré Qui m'a soutenu tout au long de ma vie scolaire.

Mes frères et sœurs que j'aime beaucoup.

A toute la famille.

A mes collègues de l'université.

A département de génie électrique

Nos cher ami (e) s, et enseignants, administratif, et le personnel de la faculté de technologie.

A tous ceux qui m'ont donné la force et la détermination de compléter ce travail, notamment mes amis Thabet, Bilal.

Que dieu leur accorde santé et prospérité.

✍ ABDELKADER

Dédicace

Je dédie ce travail À la plus belle créature que Dieu a créée sur terre,,,

Pour cette source de tendresse, de patience et de générosité.,,,

Pour ma mère

À la personne qui m'a toujours donné la volonté et pour sacrifice et leur dévouement pour mon bonheur.

Pour mon père

A tous mes frères et sœurs qui ont toujours été avec moi

À tous mes amis et collègues

Pour tous les étudiants

Département de génie électrique

Pour tous ceux qui m'ont donné la force et la détermination de compléter ce travail.

✍ Badr Eddine

Résumé

Notre projet concerne la conception et la réalisation d'un robot mobile commandé par un smartphone à distance en utilisant une plate-forme disponible à deux roues, des moteurs à engrenages placés sur la plate-forme et forcent à déplacer les roues du robot. On crée une interface utilisateur graphique pour contrôler le robot par un smarte phone (le service RemoteXY).

En plus, le robot peut détecter le gaz de monoxyde de carbone par un capteur de gaz et effectuer l'alarme.

Nous sommons en besoin d'une carte Arduino, du module Bluetooth HC-(06), deux moteurs, driver L293D, la batterie d'alimentation, capteur de gaz (MQ7) et Buzzer.

Mots clés: Robot mobile, carte Arduino, moteur, module Bluetooth HC (06), driver L293D, MQ7, Buzzer.

ABSTRACT

Our project involves the design and implementation of a mobile robot controlled by a remote smartphone using a platform available on two wheels, gear motors placed on the platform and force to move the wheels of the robot. A graphical user interface is created to control the robot by a smartphone (the RemoteXY service).

In addition, the robot can detect carbon monoxide gas through a gas sensor and perform the alarm.

We need an Arduino board, Bluetooth module HC- (06), two motors, driver L293D, battery pack, gas sensor (MQ7) and Buzzer.

Key words: Mobile robot, Arduino cards, engine, Bluetooth HC (06), driver L293D, MQ7, Buzzer.

ملخص:

يتضمن هذا المشروع بناء روبوت متنقل باستخدام منصة متوفرة على عجلتين، ويتم وضع محركات التروس على المنصة لتحريك عجلات الروبوت وذلك عن طريق إنشاء واجهة مستخدم رسومية للتحكم في الروبوت عن طريق الهاتف الذكي (خدمة RemoteXY).

بالإضافة إلى ذلك، يمكن للروبوت الكشف عن غاز أول أكسيد الكربون بواسطة حساس الغاز والقيام بالإنذار.

نحتاج إلى لوحة Arduino، وحدة Bluetooth HC-06 (06)، محركان، متحكم محركات L293D، حزمة البطارية، مستشعر الغاز (MQ7)، صفارة الإنذار.

الكلمات المفتاحية: روبوت متحرك، لوحة أردوينو، محرك، وحدة بلوتوث HC-06، متحكم محركات L293D، حساس غاز أول أكسيد الكربون، صفارة إنذار.

Table de matière

Table des matières

Remerciement	i
Dédicace	ii
Résumé	iv
Table des matieres	v
Liste des figures	viii
Liste des tableaux	x
Liste des abreviations	xi
Introduction générale	1

CHAPITRE I : Généralité sur les robots

1. Introduction.....	3
2. Historique de l'évolution de la robotique et les robots	3
3. Définition de robots.....	5
4. Les types des robots	5
4.1 Les robots mobiles	5
4.1.1 Définition de robots mobiles	5
4.1.2 Architecture des robots mobiles	6
4.1.3 Classification des robots mobile.....	6
4.1.3.1 Classification selon le degré d'autonomie.....	6
4.1.3.2 Classification selon le type de locomotion.....	7
4.1.4 Caractéristiques d'un robot.....	10
4.2 Les robots manipulateurs	11
4.2.1 Les types des robots manipulateurs	11
5. Domaine d'utilisation des robots.....	12
6. Avantages et inconvénient des robots	15
7. Conclusion	16

CHAPITRE II : Le dispositif programmable Arduino

1. Introduction	17
2. Définition du module Arduino	17
3. Les différentes cartes Arduino.....	18
4. Pourquoi Arduino UNO	20
5. Principales caractéristiques Arduino UNO	21

TABLE DES MATIERES

6.	La constitution de la carte Arduino UNO	21
6.1	Partie matérielle.....	21
6.1.1	Qu'est-ce qu'un microcontrôleur	21
6.1.2	Le Microcontrôleur ATmega328	22
6.1.3	Les sources de l'alimentation de la carte	24
6.1.4	Les connecteurs	24
6.1.5	Les LED de contrôle	24
6.1.6	Entrée analogique	24
6.1.7	Entrées /sorties numérique.....	25
6.1.8	Les sorties électriques	26
6.1.9	Les ports de communications	26
6.2	Partie programme	27
6.2.1	Le langage Arduino	27
6.2.2	Description du programme	28
7.	Les Accessoires de la carte Arduino	29
7.1	Communication	29
7.1.1	Communication sans fil	29
7.1.2	Le module Arduino Bluetooth	31
7.2	Le capteur De Gaz Monoxyde de carbone MQ7	32
7.3	Le module Buzzer.....	33
7.4	Les actionneurs	34
7.4.1	Moteurs à courant continu	34
7.4.2	Carte de puissance (L293D)	34
8.	Les Accessoires utilisés	38
9.	Conclusion	39

CHAPITRE III: Réalisation et application

1.	Introduction	40
2.	La conception de robot	40
2.1	La structure de base du robot	40
3.	Comment réaliser un robot mobile contrôlé par téléphone portable par application Android.	41
3.1	La partie mécanique	41
3.2	La partie logicielle	42

TABLE DES MATIERES

3.2.1	Proteus.....	42
3.2.2	ISIS	42
3.2.3	ARES	42
3.2.4	Fritzing	42
3.3	La partie électronique	43
3.3.1	Les Cartes Electroniques	43
3.3.1.1	Carte de command	43
3.3.1.2	Carte de puissance.....	44
3.3.2	La structure du capteur MQ7 en général	45
3.4	L'application utilisée dans le contrôle	50
3.4.1	Une application Remote XY	50
3.4.2	Les Étapes pour créer l'application RemoteXY	51
3.5	Arduino IDE.....	53
3.5.1	L'environnement de la programmation	53
3.5.2	Structure générale du programme (IDE Arduino).....	53
3.5.3	Injection du programme	55
4.	Le robot et système d'alarme	60
5.	Algorithme de l'approche de commande	61
6.	L'algorithme de l'auto-calibration	63
7.	L'algorithme de détecteur de Gaz.....	64
8.	Conclusion	65
	Conclusion générale	66
	Annexe	67
	Bibliographie	69

Liste des figures

Les Figures	Nom de figure	Numéro de page
Figure (I.1)	La tortue de Grey Walter (nommée “machina speculatrix” et surnommée Elsie) Et une illustration de sa trajectoire pour rejoindre saniche.	4
Figure (I.2)	Architecture d’un robot mobile.	6
Figure (I.3)	Robot de type uni-cycle.	7
Figure (I.4)	Robot de type tricycle.	8
Figure (I.5)	Robot de type voiture.	8
Figure (I.6)	Robot mobile omnidirectionnel	9
Figure (I.7)	Assemblage de sous-ensembles soudés par robotisation en atelier de ferrage.	12
Figure (I.8)	Développement du premier robot nettoyeur à l'Université Isao Shimoyama tokyo	13
Figure (I.9)	Une table d'opération surmontée des bras d'un Da Vinci.	13
Figure (I.10)	« Projet Mars Exploration Rover » (Spirit & Opportunity) en 2003, et Projet Mars Science Laboratory" succédant au projet Rover, envoi prévu de Curiosity fin 2011.	14
Figure (I.11)	Logiciel de contrôle éthique des machines à tuer de l’armée	14
Figure (I.12)	Robots de laboratoires	15
Figure (II.1)	Schéma générale de système Arduino	17
Figure (II.2)	Les différentes cartes Arduino	19
Figure (II.3)	Carte ARDUINO	19
Figure (II.4)	Le microcontrôleur ATmega328	22
Figure (II.5)	Porte Entrée analogique	25
Figure (II.6)	Porte Entrées /sorties numérique	25
Figure (II.7)	Porte de sorties électriques	26
Figure (II.8)	Les différentes connexions de la carte arduino uno	27

LISTE DES FIGURES

Figure (II.9)	Logo Bluetooth	30
Figure (II.10)	Module Bluetooth HC-06	32
Figure (II.11)	Capteur de gaz MQ7	33
Figure (II.12)	Un Buzzer en réel	33
Figure (II.13)	Diagramme schématique.	33
Figure (II.14)	Moteurs à courant continu	34
Figure (II.15)	forme intégrée DIL du L293D.	35
Figure (II.16)	Brochage du L293D	36
Figure (II.17)	Le pont en H	37
Figure (II.18)	Sens du courant en fonction de l'état des interrupteurs d'un pont en H.	37
Figure (II.19)	carte de test	89
Figure (II.20)	Câble USB	38
Figure (III.1)	La structure de base du robot	40
Figure (III.2)	Nomenclature Chassie du Robot	42
Figure (III.3)	Schéma électronique de carte de commande et puissance et Bluetooth	44
Figure (III.4)	carte de command-cote composants	44
Figure (III.5)	carte de command-3D	44
Figure (III.6)	Schéma électronique de carte de puissance	45
Figure (III.7)	carte de puissance-cote composants	45
Figure (III.8)	carte de puissance -3D	45
Figure (III.9)	structure interne de capteur MQ7	46
Figure (III.10)	cycle de fonctionnement de capteur MQ7	47
Figure (III.11)	schéma générale de projet	48
Figure (III.12)	Vue schématique de notre circuit	48
Figure (III.13)	l'éditeur RemoteXY	51
Figure (III.14)	quelques additions sur l'interface	52

LISTE DES FIGURES

Figure (III.15)	configuration et module d'interface de l'application	52
Figure (III.16)	l'interface finale d'application	53
Figure (III.17)	Structure générale du programme (IDE Arduino)	54
Figure (III.18)	Les différents boutons de commande	54
Figure (III.19)	Choix de la carte Arduino	55
Figure (III.20)	Choix du port de connexion de la carte	56
Figure (III.21)	vérifie les erreurs dans le code source	57
Figure (III.22)	Envoi du programme	57
Figure (III.23)	Le programme est en cours de téléchargement	58
Figure (III.24)	Fin de téléchargement	58
Figure (III.25)	Etapes d'injection du code	59
Figure (III.26)	Connexion par Bluetooth	60

Liste des tableaux

Tableaux	Nom de tableaux	Numéro de page
Tableau I.1	Les avantages et des inconvénients des différents types de robots à roues	9
Tableau II.1	caractéristiques Arduino UNO	21
Tableau II.2	quelques instructions sont utilisées	28
Tableau II.3	représentation des ondes radio UHF sur une bande de fréquence de 2,4 GHz	30
Tableau II.4	Caractéristique de Bluetooth	31
Tableau II.5	Caractéristiques du moteur à courant continu	34
Tableau II.6	Caractéristiques du circuit L293D	35
Tableau II.7	Logique de la commande de L293D	36
Tableau III.1	Nomenclatureur Chassie du Robot	41

Liste des abréviations

L.A.A.S	Laboratory for Analysis and Architecture of Systems
INSA	Institut national des sciences appliquées
IDE	Integrated Development Environment
Vcc	Common Collector Voltage
GND	Ground
USB	Universal Serial Bus
RX	Receiver
TX	Transmitter
PWM	Pulse Width Modulation
ROM	Read Only Memory
RAM	Random Access Memory
EEPROM	Electrically erasable programmable read -only memory
PWM	Pulse Width Modulation
CC	Courant Continu
SPI	Serial peripheral interface
SCL	Serial clock ligne
MISO	Master Input Slave Output
UHF	Ultra hautes fréquences
ISM	Industriel, Scientifique et Médicale
LED	Light Emitting Diode
ROM	Read Only Memory
RAM	Random Access Memory
FTDI	future Technology Devices International
Ppm	parts per million
PCB	Printed Circuit Board.
ISIS	Intelligent Schematic Input System
CAO	Conception Assistée par Ordinateur
TRS/MIN	tours par minute
IP	Internet Protocol
URL	Uniform Resource Locator

LISTE DES ABREVIATIONS

LPT Line Printing Terminal

MPIDE Multi-Platform IDE

WiFi Wireless Fidelity.

Introduction Générale

1. Introduction générale

Malgré son aspect récent, la robotique tire ses origines des civilisations les plus antiques. Il est peut-être même vrai que toutes les autres sciences ont existé justes pour permettre à la robotique de se développer afin d'aider l'homme à créer ce « esclave » qui appliquera ses ordres au doigt et à l'œil et qui le libérera à jamais du travail.

De l'homme préhistorique qui s'empresse d'inventer des extensions technologiques de ses membres afin de faciliter son travail, jusqu'au robot mobile autonome le plus performant, la robotique mobile n'a cessé de connaître des progrès spectaculaires. En effet, les générations des robots mobiles offrant à chaque fois plus de performances en matière de perception, d'autonomie et de pouvoir décisionnel. Ces progrès étaient un résultat inévitable des développements qu'ont connus la mécanique, la microélectronique et l'informatique. C'est pour cela qu'en très peu de temps, les générations de robots se succèdent pour voir naître une branche de la robotique visant de plus amples horizons : c'est l'ère de la robotique mobile. [2]

D'une manière générale, on regroupe sous l'appellation robots mobiles l'ensemble des robots à base mobile, par opposition notamment aux robots manipulateurs. L'usage veut néanmoins que l'on désigne le plus souvent par ce terme les robots mobiles à roues.

On peut estimer que les robots mobiles à roues constituent le gros des robots mobiles.

Leur faible complexité en a fait de bons premiers sujets d'étude pour les roboticiens intéressés. Cependant, malgré leur simplicité apparente (mécanismes plans, à actionneurs linéaires), ces systèmes ont soulevé un grand nombre de problèmes difficiles. Nombre de ceux-ci ne sont d'ailleurs toujours pas résolus, Ainsi, alors que les robots manipulateurs se sont aujourd'hui généralisés dans l'industrie, rares sont les applications industrielles qui utilisent des robots mobiles.

L'industrialisation de ces systèmes bute sur divers problèmes délicats. Ceux-ci viennent essentiellement du fait que, 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.

Néanmoins, l'intérêt indéniable de la robotique mobile est d'avoir permis d'augmenter considérablement nos connaissances sur la localisation et la navigation de systèmes autonomes. La gamme des problèmes potentiellement soulevés par le plus simple des robots mobiles à roues en fait un sujet d'études à part entière et forme une excellente base pour l'étude de systèmes mobiles plus complexes.

2. L'Objectif du projet

Dans notre travail concernant réalisation d'un robot mobile, nous abordons le problème en comment réaliser un robot mobile contrôlé par Smartphone détecté gaz CO, et améliorer la précision du contrôle et de la vitesse.

3. Les étapes à suivie :

Dans ce projet trois objectifs ont été visés :

- Le premier est de regrouper suffisamment d'informations sur une grande catégorie de cartes d'interfaçage (Arduino) : son langage de programmation, sa construction, son principe de fonctionnement.
- Le deuxième objectif principal de notre travail est de réaliser un robot mobile ont utilisant une carte électronique Arduino.
- Le troisième est de réaliser une application sous smartphone afin de simplifier la commande.

4. Présentation du mémoire

Ce travail est alors organisé en trois chapitres de la manière suivante :

Le premier chapitre présente certaines généralités concernant la robotique et les différents types des robots et robots mobiles.

Le second chapitre est consacré aux cartes Arduino microcontrôleur et les dispositifs qui fonctionnent avec l'Arduino, les langages de programmation et en particulier l'IDE de l'Arduino.

Le troisième chapitre est consacré à la partie pratique.

En fin par une conclusion et perspective on termine ce travail.

CHAPITRE I

Généralité sur les robots mobiles

1. Introduction :

La robotique est un ensemble de disciplines (mécanique, électronique, automatique, informatique), elle se subdivise en deux types : les robots industriels et les robots mobiles. Les robots industriels sont généralement fixes, ils sont utilisés dans des nombreuses applications industrielles : l'assemblage mécanique, la soudure, la peinture...etc. Les robots mobiles ne sont pas fixes, ils sont classifiés selon la locomotion en robots marcheurs, à roues, à chenilles...etc... , comme ils peuvent être classifiés selon le domaine d'application en robots militaires, de laboratoire, industriels et de services.

Les robots mobiles présentent un cas particulier en robotique. Leur intérêt réside dans leur « mobilité », destinés à remplir des tâches pénibles (exemple : transport de charges Lourdes) et ils travaillent même en ambiance hostile (nucléaire, marine, spatiale, lutte Centre l'incendie, surveillance...).

L'aspect particulier de la mobilité impose une complexité technologique (capteur, motricité, énergie) et méthodologique tel que le traitement des informations par utilisation des techniques et des processeurs particuliers (vectoriels, cellulaires). [8]

2. Historique de la robotique :

La robotique est passée par plusieurs générations comme suit :

- Le terme de robot apparaît pour la première fois dans une pièce de Karel Capek en 1921: Rossum's Universal Robots. Il vient du tchèque 'robo9ta' (servitude) et présente une vision des Robots comme serviteurs dociles et efficaces pour réaliser les taches pénibles mais Qui déjà vont se rebeller contre leurs créateurs.
- Le terme robotique a été employé pour la première fois par Asimov en 1941. Avant définir qu'est-ce qu'un robot nous citerons les trois lois qui ont été développés par Isaac Asimov, et qui sont régissant le comportement d'un robot. [1]

➤ Les trois lois de la robotique :

- 1- Un robot ne peut blesser un humain ni, par son inaction, permettre qu'un humain soit blessé.
 - 2- Un robot doit obéir aux ordres donnés par les humains, sauf si de tels ordres se trouvent en contradiction avec la première loi.
 - 3- Un robot doit protéger sa propre existence aussi longtemps qu'une telle protection n'est pas en contradiction ni avec la première et/ou ni avec la deuxième loi. [30]
- 1947 : Premier manipulateur électrique téléopéré.

• La tortue construite par Grey Walter dans l'année 1950 (Figure I.1), est l'un des premiers robots mobiles autonomes. Grey Walter n'utilise que quelques composants analogiques, dont des tubes à vide, mais son robot est capable de se diriger vers une lumière qui marque un but, de s'arrêter face à des obstacles et de recharger ses batteries lorsqu'il arrive dans sa niche. Toutes ces fonctions sont réalisées dans un environnement entièrement préparé, mais restent des fonctions de base qui sont toujours des sujets de recherche et de développement technologiques pour les rendre de plus en plus génériques et robustes. [3]

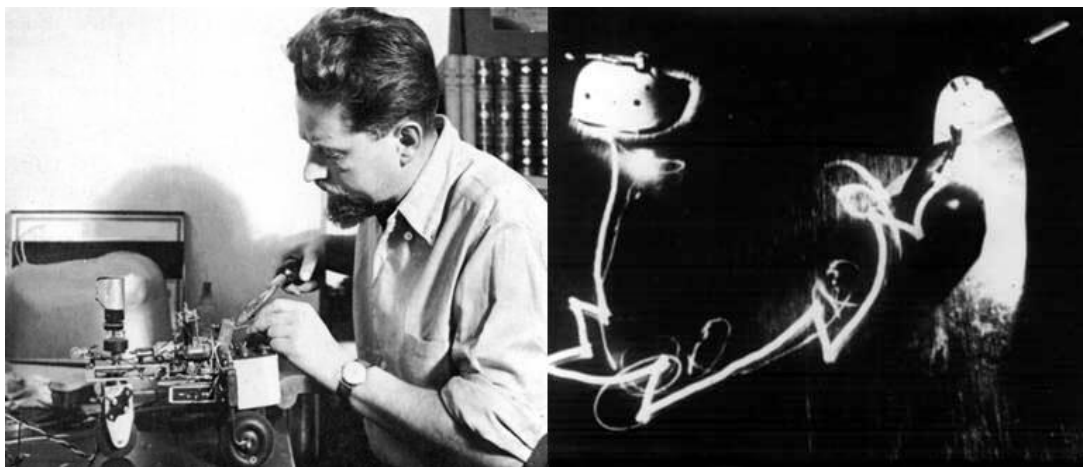


FIGURE I.1 –La tortue de Grey Walter (nommée “machina speculatrix” et surnommée Elsie)
Et une illustration de sa trajectoire pour rejoindre sa niche.

- 1954 : Premier robot programmable.
- 1961 : Utilisation d'un robot industriel, commercialisé par la société unimation (USA), sur une chaîne de montage de General Motors.
- 1961 : Premier robot avec contrôle en effort.
- 1963 : Utilisation de la vision pour commander un robot.
- 1978 : Le robot ARGOS, développé à l'université Paul Sabatier de Toulouse (France), le robot ARGOS simule la navigation d'un robot mobile équipé d'un système de vision au fur et à mesure de ses déplacements.
- 1979 : Le robot HILARE. Les chercheurs du L.A.A.S. de Toulouse (France) étudièrent la planification des trajectoires d'un robot mobile ponctuel, dans un environnement totalement connu.
- 1981 : Le robot VESA. Ce robot, construit à l'I.N.S.A (France). De Rennes, est équipée d'un arceau de sécurité pour réaliser la détection d'obstacles dans un environnement totalement inconnu.

- 1984 : Le robot FLAKEY. Ce robot, conçu et construit au Stanford Research Institut et le reflet des améliorations apporté par 14 années de développement. Le robot FLAKEY est équipé de deux roues motrices avec encodeurs, mais sa vitesse maximale est de 66 cm/s au lieu de quelques centimètres par seconde. Ce robot est capable de naviguer dans des environnements réels.
- 1993 : Les robots ERRATIC et PIONNER. Le robot ERRATIC a été conçu par Kurt Konolige, au Stanford Research Institut, comme un robot mobile de faible coût pour ses cours de robotique. [4] [5]

3. Définition de robot :

La robotique est un très bon exemple de domaine pluri-disciplinaire qui implique de nombreuses thématiques telles que la mécanique, la mécatronique, l'électronique, l'automatique, l'informatique ou l'intelligence artificielle. En fonction du domaine d'origine des auteurs, il existe donc diverses définitions du terme robot, mais elles tournent en général autour de celle-ci :

- Un robot est une machine équipée de capacités de perception, de décision et d'action qui lui permettent d'agir de manière autonome dans son environnement en fonction de la perception qu'il en a. [3]

4. Les types des robots :

Il existe deux grandes familles de robots sont :

Les robots mobiles.

Les robots manipulateurs.

4.1 Les robots mobiles :

4.1.1 Définition de robots mobiles

Un robot mobile est celui qui peut se déplacer dans son environnement de façon Indépendante. Pour ce faire, le robot doit pouvoir naviguer dont la portée et la précision de navigation requise varient en fonction de la taille du robot et du type de sa tâche.

Un robot mobile est un véhicule doté de moyens de locomotion qui lui permettent de ses déplacer. Suivant son degré d'autonomie il peut-être dote de moyens de perception et de raisonnement. [6]

4.1.2 L'architecture des robots mobiles

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. [6]

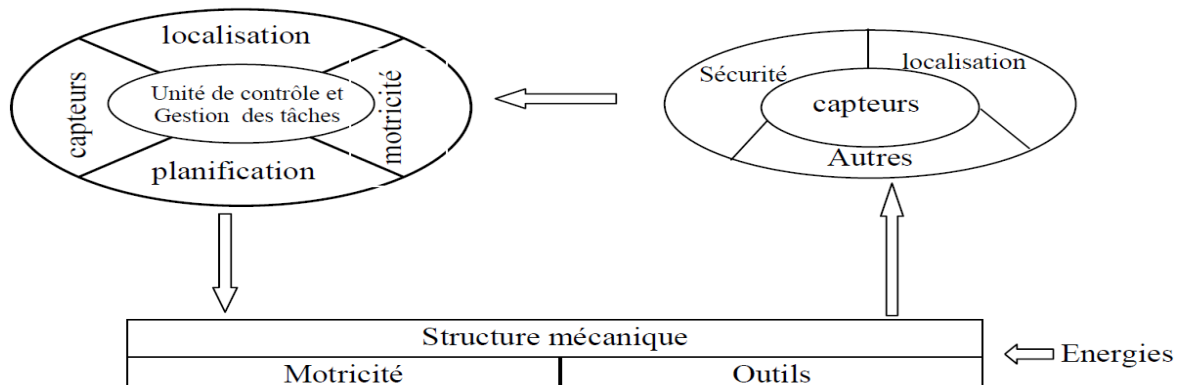


Figure I.2 : Architecture d'un robot mobile. [6]

4.1.3 Classification des Robots Mobiles :

On peut classer les robots mobiles selon leur degré d'autonomie, système de locomotion, leur domaine d'application, leur système de localisation, l'énergie utilisée...

Nous allons présenter ici deux classifications importantes qui semblent être les plus intéressantes.

4.1.3.1 Classification selon le degré d'autonomie

Un robot mobile autonome est un système automoteur doté de capacités décisionnelles et de moyens d'acquisition et de traitements de reformation 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 télécommande par un opérateur ces robots sont commandés par un opérateur qui leur impose chaque tâche élémentaire à réaliser.
- Véhicule télécommandé au sens de la tâche à réaliser le véhicule contrôle automatiquement ses actions.
- Véhicule semi-autonome ce type de véhicule réalise des tâches prédéfinies sans l'aide de l'opérateur.
- Véhicule autonome ces derniers réalisent des tâches semi-définies. [6][8][9]

4.1.3.2 Classification selon le type de locomotion

Selon le système de locomotion, on peut distinguer quatre types des robots :

A. Les robots mobiles à roues

La mobilité par roues est la structure mécanique la plus utilisée, ce type de robot assure un déplacement avec une accélération et une vitesse rapide mais nécessite un sol relativement plat. On distingue plusieurs classes de robots à roues déterminées, principalement, par la position et le nombre de roues utilisées, nous citerons ici les quatre classes principales de robots à roues.

- **Robot uni-cycle**

Un robot de type uni cycle est actionné par deux roues indépendantes, il possédant éventuellement des roues folles pour assurer sa stabilité. Son centre de rotation est situé sur l'axe reliant les deux roues motrices, c'est un robot non-holonyme, en effet il est impossible de le déplacer dans une direction perpendiculaire aux roues de locomotion, sa commande peut être très simple, il est en effet assez facile de le déplacer d'un point à un autre par une suite de rotations simples et de lignes droites. (Figure I.3). [10]



Figure I.3 : Robot de type uni-cycle.

- **Robot tricycle**

Un robot de type tricycle est constitué de deux roues fixes placées sur un même axe et d'une roue centrée orientable placée sur l'axe longitudinal. Le mouvement du robot est donné par la vitesse des deux roues fixes et par l'orientation de la roue orientable. Son centre de rotation est situé à l'intersection de l'axe contenant les roues fixes et de l'axe de la roue orientable. C'est un robot non-holonyme. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues fixes. Sa commande est plus compliquée. Il est en général impossible d'effectuer des rotations simples à cause d'un rayon de braquage limité de la roue orientable. (Figure I.4). [10]

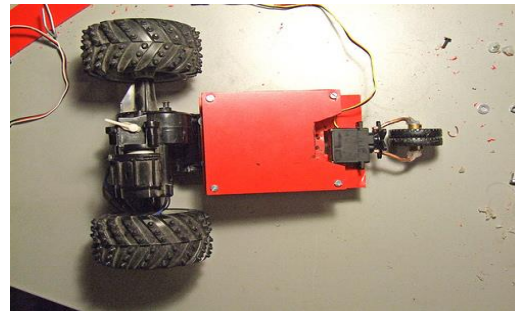
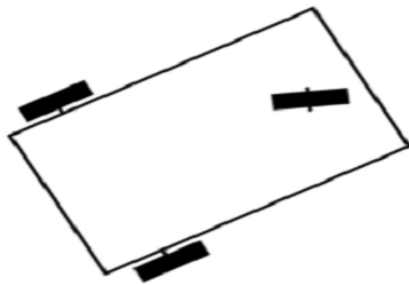


Figure I.4 : Robot de type tricycle.

- **Robot voiture**

Un robot de type voiture est semblable au tricycle, il est constitué de deux roues fixes placées sur un même axe et de deux roues centrées orientables placées elles aussi sur un même axe (Figure I.5).

Le robot de type voiture est cependant plus stable puisqu'il possède un point d'appui supplémentaire. Toutes les autres propriétés du robot voiture sont identiques au robot tricycle, le deuxième pouvant être ramené au premier en remplaçant les deux roues avant par une seule placée au centre de l'axe, et ceci de manière à laisser le centre de rotation inchangé. [10]

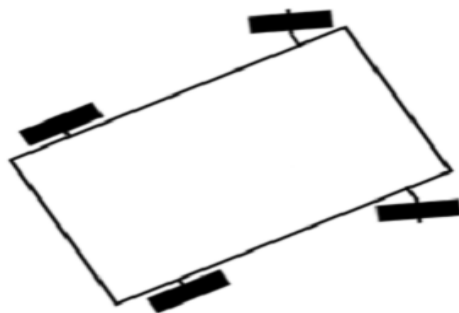


Figure I.5 : Robot de type voiture.

- **Robot omnidirectionnel**

Un robot omnidirectionnel est un robot qui peut se déplacer librement dans toutes les directions. Il est en général constitué de trois roues décentrées orientables placées en triangle équilatéral.

L'énorme avantage du robot omnidirectionnel est qu'il est holonome puis qu'il peut se déplacer dans toutes les directions. Mais ceci se fait au dépend d'une complexité mécanique bien plus grande (Figure I.6). [10]



Figure I.6 : Robot mobile omnidirectionnel.

Comparaison des différents types

Nous pouvons observer dans le tableau I.1 ci-dessous un récapitulatif des avantages et des inconvénients des différents types de robots à roues.

Type du robot	Avantage	Inconvénient
unicycle	- Stable - Rotation sur soi-même - Complexité mécanique faible	- Non-holonyme
Tricycle	-Complexité mécanique modérée	- Non-holonyme - Peu stable - Pas de rotation sur soi-même
Voiture	- Stable - Complexité mécanique modérée	- Non-holonyme - Pas de rotation sur soi-même
Omnidirectionnel	- Holonyme - Stable - Rotation sur soi-même	- Complexité mécanique important

Tableau I.1 : Les avantages et des inconvénients des différents types de robots à roues.

B. Les robots à chenilles :

L'utilisation des chenilles présente l'avantage d'une bonne adhérence au sol et d'une faculté de franchissement d'obstacles .L'utilisation est orientée vers l'emploi sur sol accidenté ou de mauvaise qualité au niveau de l'adhérence. [11]

C. Les robots marcheurs :

Les robots marcheurs sont destinés à réaliser des tâches variées dont l'accès au site est difficile. Leur anatomie à nombreux degrés de liberté permet un rapprochement avec les

robots manipulateurs. La locomotion est commandée en termes de coordonnées articulaires. Les différentes techniques étudiées se rapprochent de la marche des animaux et notamment de celle des insectes. [11]

D. Les robots rampants

La reptation est une solution de locomotion pour un environnement de type tunnel qui conduit à réaliser des structures filiformes.

Le système est composé d'un ensemble de modules ayant chacun plusieurs mobilités. Les techniques utilisées découlent des méthodes de locomotion des animaux.

- Le type scolopendre constitue une structure inextensible articulée selon deux axes orthogonaux.
- Le type lombric comprend trois articulations, deux rotations orthogonales et une translation dans le sens du mouvement principal.
- Le type péristaltique consiste à réaliser un déplacement relatif d'un module par rapport aux voisins. [11]

4.1.4 Caractéristiques d'un robot :

Un robot doit être choisi en fonction de l'application qu'on lui réserve. Voici quelques paramètres à prendre, éventuellement, en compte :

➤ La charge maximale transportable

De quelques kilos à quelques tonnes, à déterminer dans les conditions les plus défavorables (en élongation maximum).

➤ Le volume de travail

Défini comme l'ensemble des points qu'on peut atteindre par l'organe terminal.

Tous les mouvements ne sont pas possibles en tout point du volume de travail. L'espace (volume) de travail (reachable - workspace), également appelé espace de travail maximal, est le volume de l'espace que le robot peut atteindre via au moins une orientation. L'espace de travail dextre (dextrous - workspace) est le volume de l'espace que le robot peut atteindre avec toutes les orientations possibles organe terminal. Cet espace de travail est un sous-ensemble de l'espace de travail maximal,

➤ Le positionnement absolu

Correspondant à l'erreur entre un point souhaité (réel), défini par une position et une orientation dans l'espace cartésien et le point atteint. Il est calculé via le modèle géométrique

inverse du robot. Cette erreur est due au modèle utilisé, à la quantification de la mesure de position et à la flexibilité du système mécanique.

En général, l'erreur de positionnement absolu, également appelée précision, est de l'ordre de 1mm.

➤ **La répétabilité**

Ce paramètre caractérise la capacité que le robot a de retourner vers un point (position, orientation) donné. La répétabilité correspond à l'erreur maximum de positionnement sur un point prédéfini dans le cas de trajectoires répétitives.

➤ **La vitesse de déplacement**

Vitesse maximum en élongation maximum ou accélération.

Il existe d'autres caractéristiques comme :

- ✓ La masse du robot.
- ✓ Le coût du robot.
- ✓ La maintenance,[8]

4.2 Les robots manipulateurs

Un robot manipulateur est en forme d'un bras et se compose d'un certain nombre de segments qui est conçu pour manipuler ou déplacer des matériaux, outils et pièces sans contact humain direct. Ils sont des dispositifs qui permettent aux humains d'interagir avec des objets dans un environnement en toute sécurité. Les robots manipulateurs sont utilisés dans des applications industrielles pour s'effectuer efficacement des tâches telles que l'assemblage, soudage, traitement de surface, et le forage. [11]

4.2.1 Les types des robots manipulateurs

Les robots manipulateurs viennent sous plusieurs formes. Les formes se répartissent en cinq grandes catégories :

- Robots cylindriques
- Robots rectilignes
- Robots sphériques
- Robots articulés
- Robots SCARA. [11]

5. Domaine d'utilisation des robots

❖ Les robots industriels :

Robots industriels sont des robots utilisés dans un environnement de fabrication industrielle. Ils sont utilisés dans la fabrication des automobiles, des composants et des pièces électroniques, des médicaments et de nombreux produits, La figure I.9 représente Assemblage de sous-ensembles soudés par robotisation en atelier de ferrage. [2]



Figure I.7 : Assemblage de sous-ensembles soudés par robotisation en atelier de ferrage. [12]

❖ Robots domestiques ou ménagers :

Robots utilisés à la maison. Ce type de robots comprend de nombreux appareils très différents, tels que les aspirateurs robotiques, robots nettoyeurs de piscines, balayeuses, nettoyeurs gouttières et autres robots. [2]

Les robots commencent à entrer dans nos maisons, le marché de la robotique domestique est en pleine évolution. Avec la tondeuse Automower ou l'aspirateur Roomba, les robots ménagers connaissent un grand succès et plusieurs dizaines d'industriels sont aujourd'hui sur ce marché.

Les robots sont également mis au service des personnes handicapées ou âgées pour leur offrir des soins et un regain d'autonomie. Le robot d'assistance musculaire HAL peut permettre aux personnes handicapées de retrouver l'usage de leurs membres. Paro, le bébé phoque, est utilisé comme animal de compagnie dans les maisons de retraite, la, La figure I.10 représente Développement du premier robot nettoyeur à l'Université Isao Shimoyama tokyo. [12]



Figure I.8 : Développement du premier robot nettoyeur à l'Université Isao Shimoyama Tokyo.

❖ **Robots en médecine et chirurgie :**

Les robots semblent avoir de l'avenir à l'hôpital. Robodoc aide à réaliser certaines opérations de chirurgie. Le robot infirmier est encore en projet. Le cyber squelette HAL aide les personnes à se déplacer. Et le robot patient permet aux futurs chirurgiens-dentistes d'apprendre à soigner sans faire de dégâts...

Le système chirurgical Da Vinci : Le robot chirurgien permet d'opérer à distance, soit dans la même pièce avec une machine comme intermédiaire, soit d'un endroit très éloigné, ce qui peut être très utile souvent, La figure I.11 représente Une table d'opération surmontée des bras d'un Da Vinci. [2]

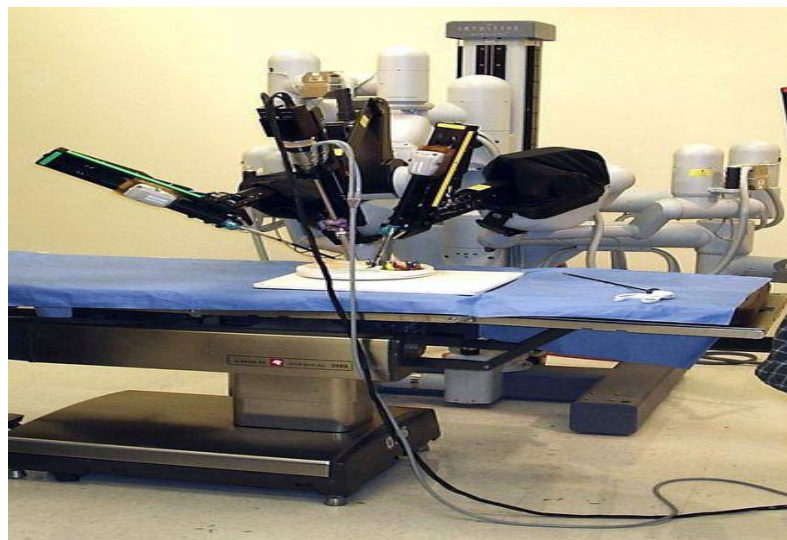


Figure I.9 : Une table d'opération surmontée des bras d'un Da Vinci. [12]

❖ **Robot d'exploration de surveillance :**

Ils explorent la planète Mars, déminent les champs de bataille, opèrent dans les zones irradiées, ou sondent les profondeurs abyssales. Les robots d'exploration en milieux

difficiles, voire tout à fait impraticables pour l'homme, se substituent à ce dernier pour sonder, secourir, opérer, en autonomie ou téléguidés, des missions à moindre coût puisque les conditions de viabilité et de sécurité sont moindres, La figure I.12, représente Laboratoire succédant au projet Rover, envoi prévu de Curiosity fin 2011. [12]

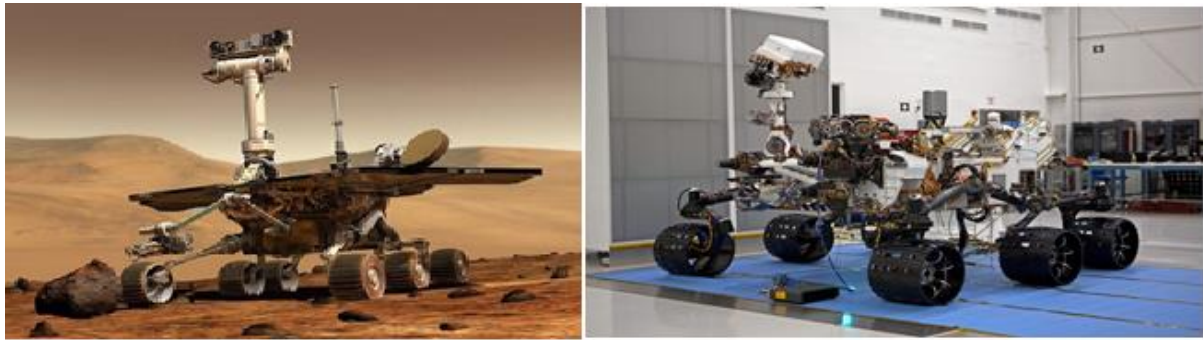


Figure I.10 : Laboratoire succédant au projet Rover, envoi prévu de Curiosity fin 2011.

❖ Robots militaires :

Les robots militaires ont des missions similaires à celles évoquées précédemment et sont utilisés dans le cadre militaire, la détection d'armes chimiques, comme engins de rapatriement, mais aussi comme armes à feu téléguidées... [12]

Les applications militaires de la robotique mobile sont nombreuses. Ce champ d'application présente l'intérêt de fournir des spécifications serrées telles que la vitesse des véhicules, leurs capacités de franchissement des obstacles et leur rapidité de réaction, La figure I.12, représente Logiciel de contrôle éthique des machines à tuer de l'armée. [8]



Figure I.11 : Logiciel de contrôle éthique des machines à tuer de l'armée.

❖ Robots de laboratoire :

De nombreux laboratoires travaillant dans le domaine de la robotique, disposent de plates-formes expérimentales pour valider des travaux théoriques en perception ou en planification de mouvement, La figure I.14, représente robots de laboratoire. [8]



Figure I.12 : Robots de laboratoires.

6. Avantages et inconvénients des robots

Un système robotique consiste non seulement des robots mais aussi d'autres dispositifs et systèmes qui sont utilisés avec le robot pour effectuer la tâche nécessaire. Les avantages des robots sont :

Robotique et automatisation peut dans de nombreuses situations d'accroître la productivité, la sécurité, l'efficacité, la qualité et la cohérence des produits.

Les robots peuvent travailler dans un environnement dangereux, sans le besoin de soutien de la vie, ou les préoccupations concernant la sécurité.

Robots n'ont pas besoin de l'éclairage, la climatisation, de ventilation et de protection contre le bruit. [2]

Robots travailler continuellement, sans ressentir une fatigue ou l'ennui, et ne nécessitent pas une assurance médicale ou de vacances.

Les robots sont de précision répétable à tous les moments, sauf si quelque chose arrive à eux ou ils s'usent.

Les robots peuvent être beaucoup plus précis que les humains. Précision linéaire d'un robot typiquement est de 20 à 10 microns.

L'inconvénient des robots est qu'ils manquent de capacité de réagir en cas d'urgence, à moins que les situations comprises et les réponses sont inclut dans le système. Les mesures de sécurité nécessaires pour s'assurer qu'ils ne lèsent pas les opérateurs et n'endommagent les machines qui travaillent avec eux. Inconvénients des robots comprennent :

- Réponse inadéquate ou mal,
- Le manque de pouvoirs prendre une décision,
- Consommation de l'énergie
- Ils peuvent causer des dommages à des autres appareils, et la blessure de l'homme

Bien que les robots ont de bonnes certaines caractéristiques mais aussi ont ces caractéristiques limités comme la capacité à degré de liberté, la dextérité, capteurs, système de vision et la réponse en temps réel. Les robots sont coûteux en raison du : coût initial de l'équipement, le coût d'installation, le besoin de périphériques, le besoin de formation et la nécessité de la programmation. [2]

7. Conclusion :

Nous avons présenté, dans ce chapitre une idée brève sur les robots, l'historique des robots, leurs définitions et les différents types de robots, leurs architectures, ainsi que leurs classifications, leurs caractéristiques, et les domaines d'utilisations.

Nous notons que les robots auront une place importante dans un futur et que la relation entre Homme et Machine se tisseront de plus en plus facilement.

Les robots ont beaucoup d'avantages, ils améliorent la vie humaine et le rendre plus facile. Tout devient réalisable avec un robot.

Il est bien de montrer que la robotique transformera nos vies, mais il faut bien prendre en compte que de telles avancées devront être contrôlées, car il existe toujours des abus, en particulier dans le domaine militaire. On nous le montre trop souvent dans les films de science-fiction où les robots prennent une place trop importante dans le monde humain, et deviennent incontrôlables. Ce n'est bien sûr que l'imagination des réalisateurs mais il ne faut pas négliger la sécurité des êtres humains. Nous devons prendre garde à toujours rester maître de nos inventions.

CHAPITRE II

Le dispositif programmable Arduino

1. Introduction :

Aujourd'hui, l'électronique est de plus en plus remplacée par de l'électronique programmée, on parle aussi de système embarquée ou d'informatique embarquée. Son but est de simplifier les schémas électroniques et par conséquent réduire l'utilisation de composants électroniques, réduisant ainsi le coût de fabrication d'un produit. Il en résulte des systèmes plus complexes et performants pour un espace réduit.

Depuis que l'électronique existe, sa croissance est fulgurante et continue encore aujourd'hui. L'électronique est devenue accessible à toutes personnes en ayant l'envie : ce que nous allons apprendre dans ce travail est un mélange d'électronique et de programmation. On va en effet parler d'électronique embarquée qui est un sous-domaine de l'électronique et qui a l'habileté d'unir la puissance de la programmation à la puissance de l'électronique. [31]

2. Définition du module Arduino

Le module Arduino est un circuit imprimé en matériel libre (plateforme de contrôle) dont les plans de la carte elle-même sont publiés en licence libre dont certains composants de la carte : comme le microcontrôleur et les composants complémentaires qui ne sont pas en licence libre. Un microcontrôleur programmé peut analyser et produire des signaux électriques de manière à effectuer des tâches très diverses. Arduino est utilisé dans beaucoup d'applications comme l'électrotechnique industrielle et embarquée ; le modélisme, la domotique mais aussi dans des domaines différents comme l'art contemporain et le pilotage d'un robot, commande des moteurs et faire des jeux de lumières, communiquer avec l'ordinateur, commander des appareils mobiles (modélisme). Chaque module d'Arduino possède un régulateur de tension +5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). Pour programmer cette carte, on utilise l'logiciel IDE Arduino. [13]



Figure II.1 : Schéma générale de système Arduino.

3. Les différentes cartes Arduino

Il existe plusieurs différentes cartes Arduino qui ont les caractéristiques et spécifications différentes. Citer ci-dessous est quelque uns :

- Le NG d'Arduino, avec une interface d'USB pour programmer et usage d'un ATmega8.
- L'extrémité d'Arduino, avec une interface d'USB pour programmer et usage d'un Microcontrôleur ATmega8.
- L'Arduino Mini, une version miniature de l'Arduino en utilisant un microcontrôleur ATmega168.
- L'Arduino Nano, une petite carte programme l'aide porte USB cette version utilisant un microcontrôleur ATmega168 (ATmega328 pour une plus nouvelle version).
- Le NG d'Arduino plus, avec une interface d'USB pour programmer et usage d'un ATmega168.
- L'Arduino Bluetooth, avec une interface de Bluetooth pour programmer en utilisant un microcontrôleur ATmega168.
- L'Arduino Diecimila avec une interface d'USB et utilise un microcontrôleur ATmega168.
- L'Arduino Duemilanove ("2009"), en utilisant un microcontrôleur ATmega168 (ATmega328 pour une plus nouvelle version) et actionné par l'intermédiaire de la puissance d'USB/DC.
- L'Arduino Méga, en utilisant un microcontrôleur ATmega1280 pour I/O additionnel et mémoire.
- L'Arduino UNO, utilisations microcontrôleur ATmega328.
- L'Arduino Mega2560, utilisations un microcontrôleur ATmega2560, et possède toute la mémoire à 256 KBS. Elle incorpore également le nouvel ATmega8U2 (ATmega16U2 dans le jeu de puces d'USB de révision 3).
- L'Arduino Leonardo, avec un morceau ATmega3U4 qui élimine le besoin de raccordement d'USB et peut être employé comme clavier. [13]

La figure II.2 représente les différentes cartes Arduino.

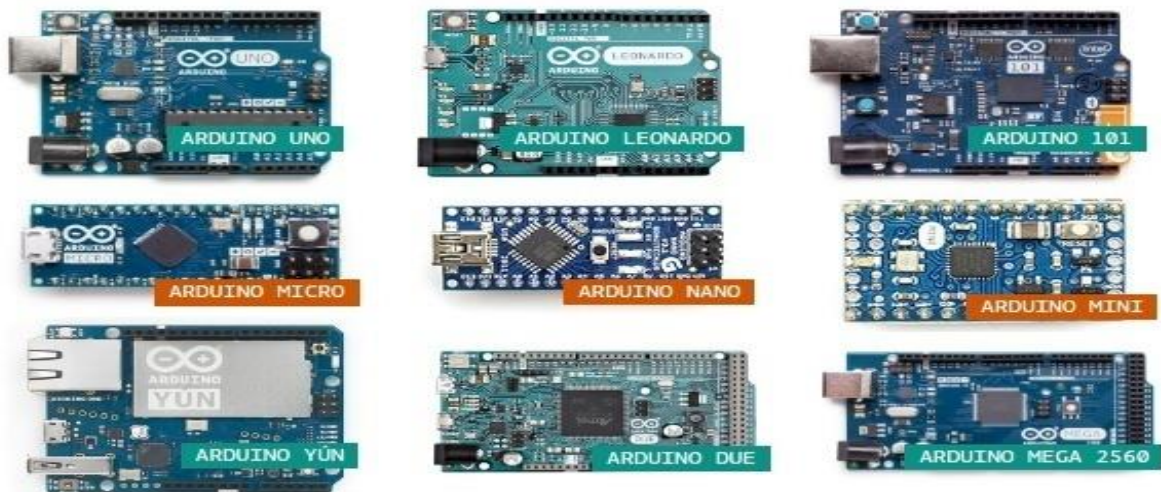


Figure II.2 : Les différentes cartes Arduino.

Parmi ces types, nous avons choisi une carte Arduino UNO (carte Basique). L'intérêt principal de cette carte est de faciliter la mise en œuvre d'une telle commande qui sera détaillée par la suite.

L'Arduino fournit un environnement de développement s'appuyant sur des outils open source comme interface de programmation. L'injection du programme déjà converti par l'environnement sous forme un code dans la mémoire du microcontrôleur se fait d'une façon très simple par la liaison USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées-sorties. Cette carte est basée sur un microcontrôleur ATmega 328 et des composants complémentaires. La carte Arduino contient une mémoire morte de 1 kilo. Elle est dotée de 14 entrées/sorties digitales (dont 6 peuvent être utilisées en tant que sortie PWM), 6 entrées analogiques et un cristal à 16 MHz, une connexion USB et possède un bouton de remise à zéro et une prise jack d'alimentation. [13]

La figure II.3 représente Carte Arduino UNO.

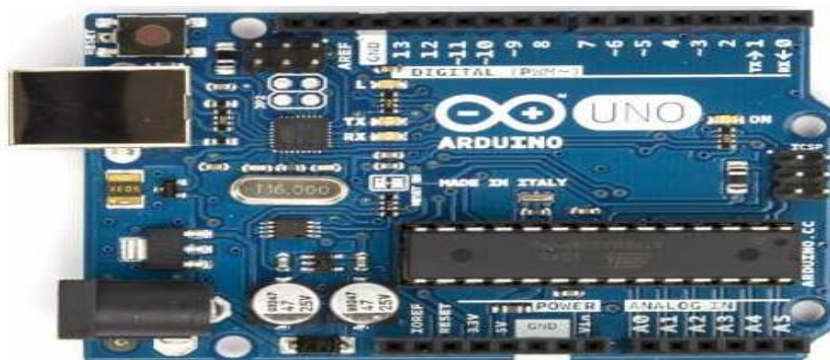


Figure II.3 : Carte Arduino UNO.

4. Pourquoi Arduino UNO

Il y a de nombreuses cartes électroniques qui possèdent des plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée, tous ces outils prennent en charge les détails compliqués de la programmation et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs tout en offrant à personnes intéressées plusieurs avantages cités comme suit :

- **Le prix (réduits) :** les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes. La moins chère des versions du module Arduino peut être assemblée à la main, (les cartes Arduino pré-assemblées coûtent moins de 4500 Dinars).
- **Multi plateforme :** le logiciel Arduino, écrit en JAVA, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.
- **Un environnement de programmation clair et simple :** l'environnement de programmation Arduino (le logiciel Arduino IDE) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Logiciel Open Source et extensible :** le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponible pour être complété par des programmeurs expérimentés. Le logiciel de programmation des modules Arduino est une application JAVA multi plateformes (fonctionnant sur tout système d'exploitation), servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison série (RS232, Bluetooth ou USB selon le module).
- **Matériel Open source et extensible :** les cartes Arduino sont basées sur les microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA 328, les schémas des modules sont publiés sous une licence créative commune, et les concepteurs des circuits expérimentés peuvent réaliser leur propre versions des cartes Arduino, en les complétant et en les améliorants. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser le coût. [14]

5. principales caractéristiques Arduino UNO

Le tableau II.1 représente les caractéristiques Arduino UNO

Microcontrôleur	Atmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7V-12V
Tension d'alimentation (limites)	6V-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches Entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité max disponible par broche E/S (5 V)	40 mA
Mémoire programme Flash	32 Kb dont 0,5 Kb sont utilisés par le bootloader
Mémoire SRAM (mémoire volatile)	2 Kb
Mémoire EEPROM (mémoire non volatile)	1 Kb
Vitesse d'horloge	16MHz

Tableau II.1 : Les Caractéristiques Arduino UNO .[13]

6. La constitution de la carte Arduino UNO

Un module Arduino est généralement construit autour d'un microcontrôleur ATMEL AVR, et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits. Chaque module possède au moins un régulateur linéaire 5V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). Le microcontrôleur est préprogrammé avec un bootloader de façon à ce qu'un programmeur dédié ne soit pas nécessaire.

6.1 Partie matérielle

Généralement tout module électronique qui possède une interface de programmation est basé toujours dans sa construction sur un circuit programmable ou plus.

6.1.1 Qu'est-ce qu'un microcontrôleur

Les cartes Arduino font partie de la famille des microcontrôleurs. Un microcontrôleur est une petite unité de calcul accompagné de mémoire, de ports d'entrée/sortie et de périphériques permettant d'interagir avec son environnement. Parmi les périphériques, on recense généralement des timers, des convertisseurs analogique-numérique, des liaisons séries, etc. On peut comparer un micro contrôleur à un ordinateur classique, mais son système d'exploitation à une puissance de calcul considérablement plus faible.

Les microcontrôleurs sont inévitables dans les domaines de l'informatique embarquée, de l'automatique et de l'informatique industrielle. Ils permettent de réduire le nombre de composant et de simplifier la création de cartes électroniques logiques. [15]

6.1.2 Le Microcontrôleur ATmega328

Un microcontrôleur ATmega328 est un circuit intégré qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit au temps des pionniers de l'électronique. Aujourd'hui, en soudant un grand nombre de composants encombrants ; tels que les transistors; les résistances et les condensateurs tout peut être logé dans un petit boîtier en plastique noir muni d'un certain nombre de broches dont la programmation peut être réalisée en langage C.[13]

La figure II.4 montre un microcontrôleur ATmega 328, qu'on trouve sur la carte Arduino.

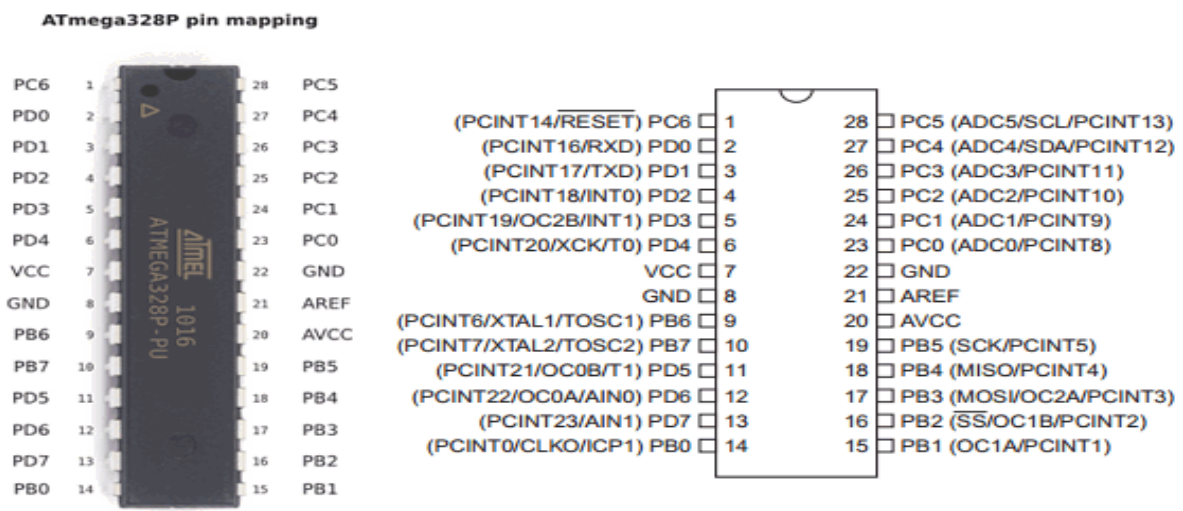


Figure II.4 : Le microcontrôleur ATmega328. [13]

Le microcontrôleur ATmega328 est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur. Globalement, l'architecture interne de ce circuit programmable se compose essentiellement sur :

- **La mémoire Flash :** c'est celle qui contiendra le programme à exécuter. cette mémoire est effaçable et réinscriptible mémoire programme de 32Ko (dont bootloader de 0.5 ko).
- **RAM :** c'est la mémoire dite "vive", elle va contenir les variables du programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur. Sa capacité est 2 ko.

- **EEPROM** : c'est le disque dur du microcontrôleur. on y enregistre des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme.
Arduino pour bien commencer en électronique et en programmation Par Astalaseven, Eskimon et olyte. [3]
- **Digital I/O (entrées-sorties Tout Ou Rien)** : 3 ports : B, port port C, port D (soit 23 broches en tout I/O).
- **Timers/Counters** : timer 0 et timer 2 (comptage 8 bits), timer 1 (comptage 16 bits) chaque timer peut être utilisé pour générer deux signaux PWM. (6 broches OCxA/OCxB).
- **Plusieurs broches multifonctions** : certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation.
- **PWM** = 6 broches OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB2), OC2A(PB3), OC2B (PD3)
- **Analog to Digital Converter** (resolution 10 bits) = 6 entrées multiplexées ADC0 (PC0) à ADC5 (PC5).
- **Gestion bus I2C** (TWI Two Wire Interface) = le bus est exploité via les broches SDA (PC5)/SCL (PC4).
- **Port série (USART)** = émission/réception série via les broches TXD(PD1)/RXD(PD0)
- **Comparateur Analogique** = broches AIN0(PD6) et AIN1 (PD7) peut déclencher interruption
- **Watchdog Timer programmable.**
- **Gestion d'interruptions (24 sources possibles):** en résumé
 - Interruptions liées aux entrées INT0 (PD2) et INT1 (PD3).
 - Interruptions sur changement d'état des broches PCINT0 à PCINT23.
 - Interruptions liées aux timers 0, 1 et 2 (plusieurs causes configurables).
 - Interruption liée au comparateur analogique.
 - Interruption de fin de conversion ADC.
 - Interruptions du port série USART.
 - Interruption du bus TWI (I2C). [13]

6.1.3 Les sources de l'alimentation de la carte

On peut distinguer deux genres de sources d'alimentation (Entrée Sortie) et cela comme suit :

- **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). on peut 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 toute 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. L'intensité maximale disponible sur cette broche est de 50mA. [16]

6.1.4 Les connecteurs

Les ranges de connecteur qui se nomment aussi les pins, ont plusieurs fonctions, certains sont des entrées et d'autres peuvent agir comme des entrées ou des sorties, certains sont analogiques et d'autres numériques, ces caractéristiques dépendent de modèle de la carte, les descriptions qui suivent s'applique à la carte d'Arduino UNO. [17]

6.1.5 Les LED de contrôle

Les petites LED TX et RX permettent de vérifier que les informations sont bien transmises entre la carte et l'ordinateur en expédition (TX) et en réception (RX). La LED ON, qu'on elle est allumé lorsque la carte Arduino est alimenté en électricité est fonctionné, le deuxième présent sur la carte indique lorsque pin 13 est parcouru par un courant électronique. [17]

6.1.6 Entrée analogique

Les connecteurs de A0 à A5 sont des entrées analogiques dont nous pouvons connaître la valeur codée sur 10 bits (de 1 à 1024), grâce à la commande analogRead (). Ce sont ces ports que l'on utilisera, par exemple pour un potentiomètre ou un capteur de luminosité par défaut, ces

connecteurs sont prévus pour mesurer une tension allant de 0 à 5 volt mais cet écart peut être modifier à l'aide de la commande `analogReference()`.

Les ports A5 et A4 peuvent être utilisé comme interface de communication à deux fils TWI (TWI Wire interface aussi appelé I2C), dans ce cas, le port A4 est utilisé pour la ligne SDA (Serial data ligne) et le port A5 en SCL (serial clock ligne). [17]



Figure II.5 : Porte Entrée analogique.

6.1.7 Entrées /sorties numérique

Les connecteurs numériques de 0 à 13 peuvent être réglés en entrée ou en sortie, grâce à la fonction **`pinMode()`**, selon le mode choisi. On accèdera à leur valeur à l'aide de la fonction **`digitalRead()`**, ou en y ouvrira une valeur à l'aide de la fonction **`digitalWrite()`**.

Certains de ces ports ont en outre des propriétés particulières :

- Le port « 0 » est capable d'agir comme un port série en réception (RX) ;
- Le port « 1 » est capable d'agir comme un port série en émission (TX) ;
- Les ports « 3, 5, 6, 9, 10 et 11 » permettent d'envoyer PWM (Pulse Width Modulation) codée sur 8 bits (0 à 255) à l'aide de la fonction **`analogwrite()`**.
- Les ports 10, 11, 12 et 13 peuvent être utilisés pour communiquer selon le protocole SPI (serial peripheral interface), de MOTOROLA qui est intéressant pour sa vélocité, dans ce mode le pin 10 gèrera du MOSI (Master Output Slave Input), le pin 12 servira pour le master MISO (Master Input Slave Output) et le pin 13 du serial clock (sck). [17]



Figure II.6 : Porte Entrées /sorties numérique.

6.1.8 Les sorties électriques

La carte Arduino peut envoyer de l'électricité afin d'alimenter les appareils qui y sont connectés. Un cerveau moteur par exemple est connecté à la carte d'Arduino par trois fils : une alimentation de 5V et enfin un câble cerveau pour envoyer les l'ordre à l'appareil.

Celui-ci pourrait être alimenté indépendamment de la carte Arduino, mais il est pratique que cette dernière soit capable d'envoyer de l'électricité.

- Les trois connecteurs GND sont des masses. tout appareil relie à un port GND.
- Le port 3.3 volts, comme son nom le suggère, envoie une tension 3.3 volts
- Le port 5 volts, de la même manière envoie une tension de 5 volts.
- Le port Vin (9 volts) est en revanche prévu pour recevoir une alimentation électrique. [17]



Figure II.7 : Porte de sorties électriques

6.1.9 Les ports de communications

La carte Arduino UNO a de nombreuses possibilités de communications avec l'extérieur. L'Atmega328 possède une communication série UART TTL (5V), grâce aux broches numériques 0 (RX) et 1 (TX).

On utilise (RX) pour recevoir et (TX) transmettre (les données séries de niveau TTL). Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328 programmé en convertisseur USB – vers – série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

Comme un port de communication virtuel pour le logiciel sur l'ordinateur, La connexion série de l'Arduino est très pratique pour communiquer avec un PC, mais son inconvénient est le câble USB, pour éviter cela, il existe différentes méthodes pour utiliser ce dernier sans fil : La Figure II.8 représente les différentes connexions de la carte Arduino UNO.

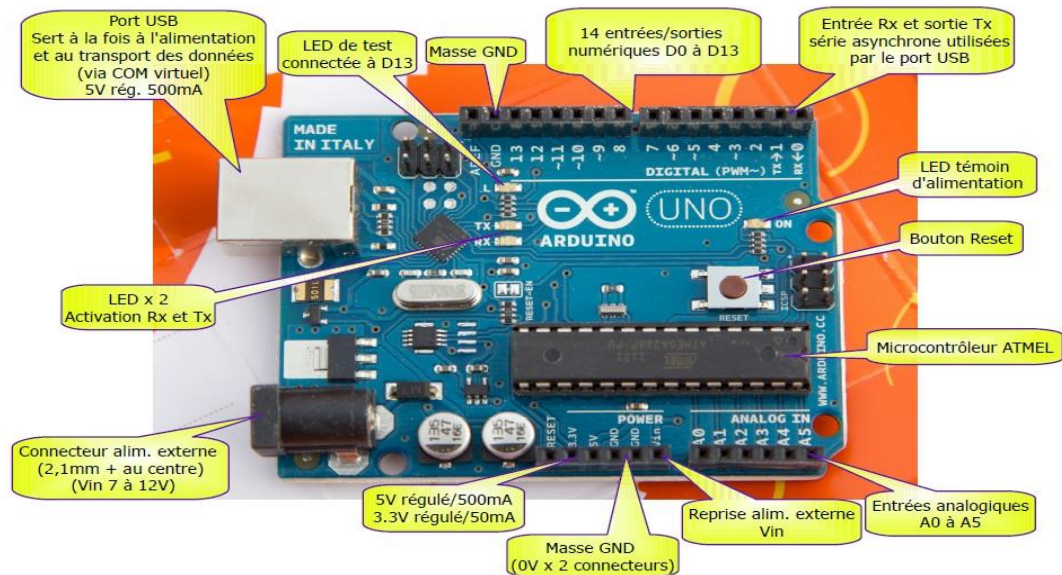


Figure II.8 : Les différentes connexions de la carte Arduino UNO.

6.2 Partie programme

6.2.1 Le langage Arduino

Le projet Arduino était destiné à l'origine principalement à la programmation multimédia interactive en vue de spectacle ou d'animations artistiques. C'est une partie de l'explication de la descendance de son interface de programmation et de traitement.

L'interface de programmation et de traitement contient est une librairie java et un environnement de développement libre. Le logiciel fonctionne sur Macintosh, Windows, Linux, et Android.

Référence :

- Le langage Java.
- Le langage C.
- L'algorithmique.

Cependant, le projet Arduino a développé des fonctions spécifiques à l'utilisation de la carte qui ont été listées ci-dessous. Vous obtiendrez la description de chacune d'elles dans le manuel de référence. [18]

Le tableau II.2 représente quelques instructions sont utilisées.

Structure	Constants	Functions
• setup() • loop()	• HIGH, LOW • INPUT, OUTPUT • INPUT_PULLUP • LED_BUILTIN	E/S numérique • pinMode () • digitalWrite () • digitalRead () E/S analogique • analogReference () • analogRead () • analogWrite () - PWM E/S avancée • tone () • noTone () • shiftOut () • shiftIn () • pulseIn () Temps • millis () • micros () • delay () • delayMicroseconds ()

Tableau II.2 : quelques instructions sont utilisées. [18]

6.2.2 Description du programme

On va commencer à mettre les mains dans le cambouis numérique. Pour donner des instructions à notre Arduino il faut lui parler, ou plutôt lui écrire dans un langage bien spécifique car, de base l'Arduino n'en connaît qu'un seul. Le but va donc être de lui rédiger un message (le programme), de l'envoyer dans le microcontrôleur pour qu'il l'interprète et qu'il agisse. [19]

Le programme Arduino (ou une séquence - sketch en anglais) est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne. Une fois téléversé dans le microcontrôleur Arduino, notre carte lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code. [19]

Le programme est structuré en 4 parties distinctes :

- La première partie est une zone de commentaire (qui ne sera pas interprété par le microcontrôleur) dans lequel on renseigne en général, le nom du créateur du programme, la date de création, d'éventuels liens pointant vers une page web, les révisions (s'il y en a) et la description du programme. Cette partie n'a pas une grande importance mais elle peut vous aider à comprendre ce qu'a voulu faire le codeur de ce programme. Elle est donc aussi

utile pour les autres. Cette zone de commentaire (pour ne pas être lue par le microcontrôleur et donc générer des erreurs) commencera par un `/*` et se terminera par un `*/`.

- La deuxième partie est faite pour déclarer des variables et les représentées par la fonction ***int***, les variables sont des expressions que vous pouvez utiliser dans les programmes pour stocker des valeurs numériques. Les constantes prédéfinies du langage Arduino sont des valeurs particulières ayant une signification spécifique.
- La troisième partie permet de configurer les entrées et sorties. La syntaxe employée est ***void setup ()*** dans laquelle on donnera le détail entre accolade par exemple : `{fonction (nom, état)}`
- La quatrième et dernière partie concerne la programmation des interactions et des comportements que l'on souhaite donner à un composant électronique par exemple. La syntaxe employée est ***void loop ()*** dans laquelle on donnera le détail entre accolade comme pour la configuration des entrées/sorties. [19]

7. Les Accessoires de la carte Arduino

La carte Arduino généralement est associée aux accessoires qui simplifient les réalisations.

7.1 Communication :

Le constructeur a suggéré qu'une telle carte doit être dotée de plusieurs ports de communications ; on peut éclaircir actuellement quelques types.

7.1.1 Communication sans fil :

Principe de la Communication sans fil :

La communication sans fil utilise plutôt les ondes électromagnétiques pour transmettre des données en utilisant l'air comme canal de transfert.

- L'émetteur applique une certaine variation de courant à son antenne.
- La variation de courant induit une onde électromagnétique.
- L'onde électromagnétique se propage à une vitesse proche de celle de la lumière dans l'air.
- Un courant électrique est induit dans l'antenne du récepteur par la variation de champs magnétique.
- Le récepteur lit la variation de courant et l'interprète selon le protocole de communication.

Bluetooth :

Bluetooth est un standard de communication permettant l'échange bidirectionnel de données à très courte distance en utilisant des ondes radio UHF 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, la technologie Bluetooth utilise une fréquence ISM (Industriel, Scientifique et Médicale) 2.4-GHz, cette technologie est utilisable sans License par les utilisateurs, la distance entre les périphériques Bluetooth est 10 mètre mais cette distance varie selon la classe du périphérique Bluetooth (puissance de transmission).

• **C'est quoi les ondes radio UHF :** La bande des Ultra hautes fréquences (UHF) est la bande du spectre radioélectrique comprise entre 300 MHz et 3 000 MHz, soit les longueurs d'onde de 1 m à 0,1 m. le tableau II.3 représente les ondes radio UHF sur une bande de fréquence de 2,4 GHz.

Longueur d'onde	Fréquence	Nom	Usage
1-10 mètres	30-300 MHz	VHF : Very High Frequency	Radio FM Télévision
0.1-1 mètres	300 MHz – 3 GHz	UHF : Ultra High Frequency	Télévision Téléphonie mobile Réseaux sans fil
10-100 mm	3-30 GHz	SHF : Super High Frequency	satellites
1-10 mm	30-300 GHz	EHF : Extra High Frequency	Satellites Radars

Tableau II.3 : Représentation des ondes radio UHF sur une bande de fréquence de 2,4 GHz.



Figure II.9 : Logo Bluetooth.

Caractéristique de Bluetooth :

Caractéristique	Bluetooth
Débit	< 3Mb/s
Portée	- 100m
Autonomie	Jours
Nb.de nœuds	7
Besoin mémoire	+250 Kb
IEEE	802.15.1

Tableau II.4 : Caractéristique de Bluetooth.

7.1.2 Le module Arduino Bluetooth :

Le module microcontrôleur Arduino Bluetooth est la plateforme populaire Arduino avec une connexion série Bluetooth à la place d'une connexion USB, très faible consommation d'énergie, très faible portée (sur un rayon de l'ordre d'une dizaine de mètres), faible débit, très bon marché et peu encombrant. [20]

Dans cette partie il consiste à commander un robot entre intervalle par Android, pour ce faire on a besoin d'une communication sans fil (Bluetooth) afin de transférer les ordres vers la carte Arduino UNO qui s'occupe dans la suite par le déplacement de robot.

Le HC-06 dispose de 6 broches : WAKEUP (réveil), VCC, GND, TXD, RXD et STATE (de l'état). En ce moment je ne traiterai que quatre épingle, qui sont VCC, GND, TXD et RXD.

Voici comment vous devez connecter le module Bluetooth à votre Arduino :

- GND (Bluetooth) $\Rightarrow$ GND (Arduino)
- VCC (Bluetooth) $\Rightarrow$ 3.3V (Arduino)
- TX-O (Bluetooth) $\Rightarrow$ RX (0) (Arduino)
- RX-I (Bluetooth) $\Rightarrow$ TX (1) (Arduino) .[20]

Caractéristique JY-MCU (HC-06)

- port série : 9600 baudrate.
- Alimentation (VCC) entre 3.3V et 5V.
- Bluetooth de bande 2.4 GHz.
- Entrée de puissance, 3.6V ~ 5V.
- GND Connectez-vous à la terre.
- TX Connectez-vous à dispositifs RX, 3.3V niveau.
- RX Connectez-vous à dispositifs TX, 3.3V niveau.
- Pour mot de passe 1 2 3 4. [20]

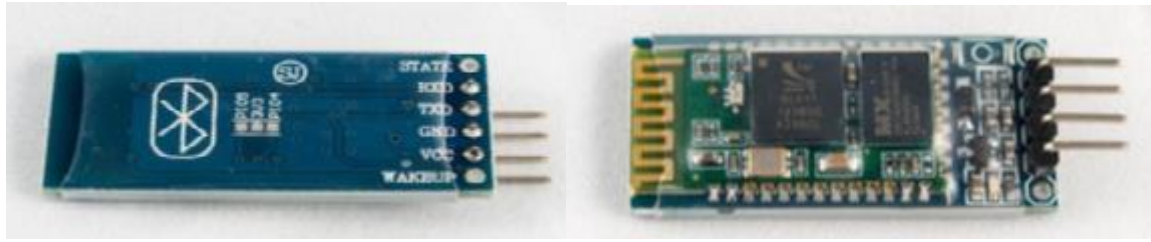


Figure II.10: Module Bluetooth HC-06. [20]

7.2 Le capteur De Gaz Monoxyde de carbone MQ7 :

Le module qu'on a choisi est basé sur le capteur MQ7 permettant de détecter la présence de monoxyde de carbone CO de 20 à 2000 ppm. Haute sensibilité et temps de réponse rapide. La sensibilité est réglable par potentiomètre.

Ce module se raccorde sur une entrée analogique ou digitale d'une carte compatible arduino ou directement sur le shield d'expansion E/S via le cordon inclus. [21]

Principe de fonctionnement

Le MQ-7 est capteur sensible au CO avec une conductivité plus faible dans l'air pur. Il fait la détection par méthode de cycle haute et basse température, et détecter le CO à basse température (chauffé de 1,5 V). La conductivité du capteur augmente avec la concentration du gaz CO à haute température (chauffée Par 5.0 V), il ignore complètement les autres gaz absorbés à faible température. Notre choix s'orienter vers ce capteur grâce à ses avantages comme le variateur de sensibilité qui permet le changement de la conductivité correspondante au signal de sortie par rapport la concentration du gaz. [21]

Caractéristiques :

- Alimentation : 5 Vcc
- Sortie analogique
- Sortie digitale
- Temps de réponse rapide et haute sensibilité
- Longue durée de vie et bonne stabilité
- Dimensions : 37 x 27 x 14 mm
- La plage de concentration de détection : 20-2000 ppm



Figure II.11 : Capteur de gaz MQ7.

7.3 Module Buzzer :

Un buzzer ou beeper est un dispositif de signalisation audio, qui peut être mécanique, électromécanique ou piézoélectrique. Utilisations typiques des buzzers et Les beepers comprennent des dispositifs d'alarme, des minuteurs et la confirmation de l'entrée de l'utilisateur, comme un clic de souris ou une frappe.

Buzzer est une structure intégrée de transducteurs électroniques, alimentation DC, largement utilisé dans les ordinateurs, imprimantes, copieurs, alarmes, jouets électroniques, automobile équipement électronique, téléphones, minuteurs et autres produits électroniques pour le son dispositifs. Buzzer actif 5V Puissance nominale peut être directement connecté à un continu son, cette section dédiée module d'expansion de capteur et la carte en combinaison, peut compléter une conception de circuit simple, à "plug and play". [22]

🌈 Caractéristiques :

- Buzzer passif intégré
- Lecteur de triode 8550 intégré
- Peut contrôler avec un microcontrôleur monopuce IO directement
- tension de fonctionnement : 5 V
- Taille de la carte : 22 (mm) x12 (mm). [22]

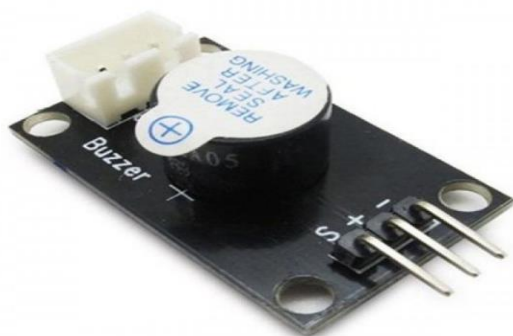


Figure II.12 : Un buzzer en réel.

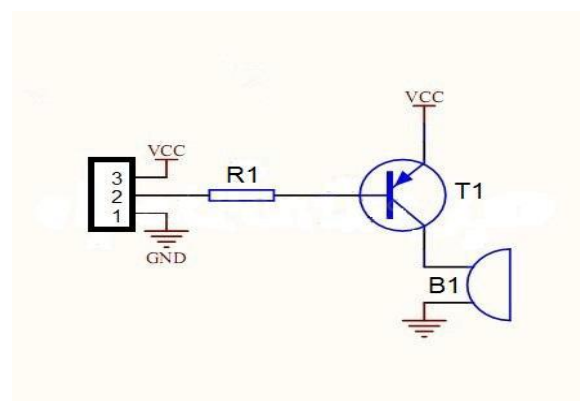


Figure II.13 : Diagramme schématique.

7.4 Les actionneurs :

7.4.1 Moteurs à courant continu :

C'est une machine électrique. Il s'agit d'un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant Continu et un dispositif mécanique. Elle est aussi appelée dynamo.

- En fonctionnement moteur, l'énergie électrique est transformée en énergie mécanique.
- En fonctionnement générateur, l'énergie mécanique est transformée en énergie électrique.

[23]

✚ Caractéristiques du moteur :

Micro moteurs à courant continu	12V
Diamètre	38.5mm
Longueur	57mm
Arbre	3.17mm.

Tableau II.5 : Caractéristiques du moteur à courant continu.

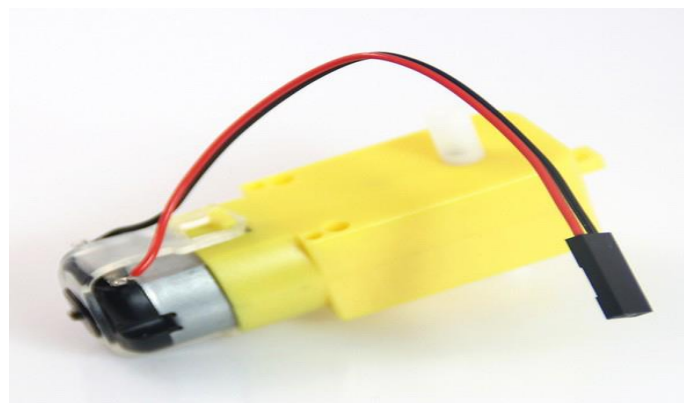


Figure II.14 : Moteurs à courant continu. [23]

7.4.2 Carte de puissance (L293D) :

L293D est un circuit intégré (IC) de pilote de moteur à double pont en H. Les pilotes de moteur agissent comme des amplificateurs de courant car ils prennent un signal de commande à faible courant et fournissent un signal de courant plus élevé. Ce signal de courant plus élevé est utilisé pour entraîner les moteurs.

Le L293D contient deux circuits d'attaque de pont en H intégrés. Dans son mode de fonctionnement commun, deux moteurs CC peuvent être entraînés simultanément, à la fois en marche avant et en marche arrière. Le fonctionnement du moteur de deux moteurs peut être contrôlé par une logique d'entrée sur les broches 2 et 7 et 10 et 15. La logique d'entrées 00 ou 11 arrête le moteur correspondant. Les logiques 01 et 10 le font pivoter dans le sens des aiguilles d'une montre et dans le sens contraire des aiguilles d'une montre, respectivement. [23]



Figure II.15 : Forme intégrée DIL du L293D.

✚ Caractéristiques du circuit L293D :

Caractéristique	Valeur
Nombre de pont-H	2
Courant Max Régime continue	600mA (x2)
Courant de pointe Max < 2ms	1200mA
VS Max Alim moteur	36v
VSS Max Alim logique	7v
Nombre de Broche	16 DIP
Perte de tension	1.3 à 1.4v

Tableau II.6 : Caractéristiques du circuit L293D. [23]

🔧 Brochage :

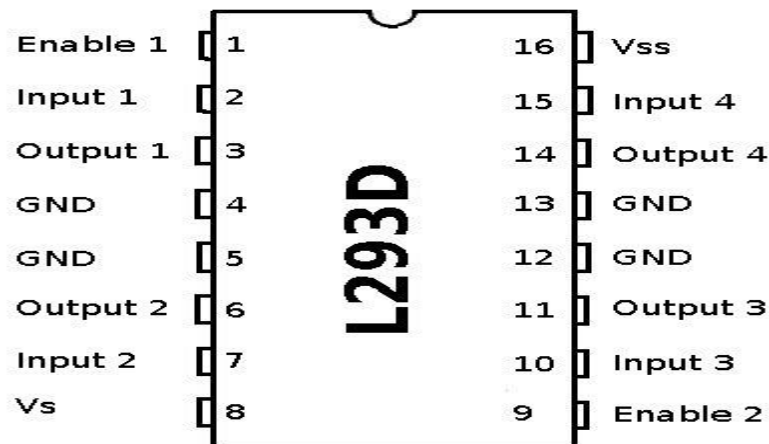


Figure II.16 : Brochage du L293D. [23]

🔧 La logique de commande de L293D :

Le tableau II.7 représente décrivez le nome de chaque broche de L293D.

Broche	Nome	Description
1	ENABLE1	L'activation/désactivation du premier Pont-H.
2	Input1	La commande du Pont-H Out1/Out2
3	Output 1	Raccorder à la charge (le moteur).
4	GND	La masse (GND) de l'alimentation de puissance VS
5	GND	Raccorder à la masse (GND).
6	Output2	Raccorder à la charge (le moteur).
7	Input2	La commande du Pont-H Out1/Out2.
8	VS	Alimentation de puissance des moteurs.
9	ENABLE2	Commande l'activation du second pont-H Out3/Out4.
10	Input 3	Avec Input 4 pour commander le pont-H Out3/Out4.
11	Output 3	La sortie 1 du second pont-H (Out3/Out4).
12	GND	Doit être raccordé à la masse (GND).
13	GND	Doit être raccordé à la masse (GND).
14	Output 3	La sortie 2 du second pont-H (Out3/Out4).
15	Input 4	Avec Input 3 pour commander le pont-H Out3/Out4.
16	VSS	Alimentation de la logique de commande (5V).

Tableau II.7 : Logique de la commande de L293D. [23]

Le pont en H

Le pont en H est une structure électronique servant à contrôler la polarité aux bornes d'un dipôle.

Il est composé de quatre éléments de commutation généralement disposés schématiquement en une forme de H comme l'indique la figure II.17. Les commutateurs peuvent être des relais, des transistors, ou autres éléments de commutation en fonction de l'application visée. [23]

Principe de fonctionnement du pont en H

Les interrupteurs fonctionnent deux par deux. Le A est associé au D et le B est associé au C. Dans le schéma ci-dessous, rien ne se passe car tous les interrupteurs sont ouverts (ils ne laissent pas passer le courant), donc le moteur est arrêté.

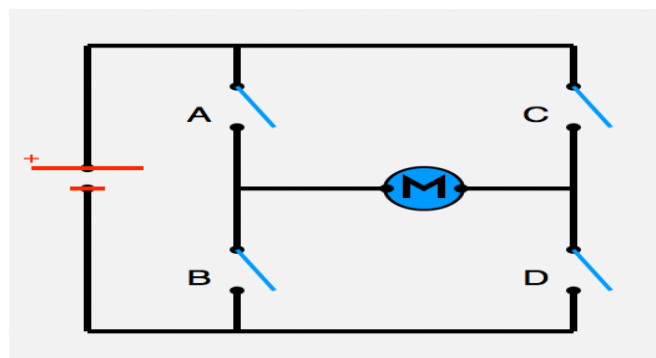


Figure II.17 : Le pont en H.

Voyons maintenant ce qui arrive lorsqu'on actionne en même temps les interrupteurs A et D (schéma de gauche), ou les interrupteurs B et C (schéma de droite) selon la figure (II.18).

Sur le schéma de gauche, les interrupteurs A et D sont fermés, donc le courant entre par la patte gauche du moteur et sort par sa droite et par conséquent le moteur tourne.

Sur le schéma de droite, les interrupteurs B et C sont fermés, donc le courant entre par la patte droite du moteur et sort par sa gauche ce qui fait tourner le moteur dans le sens inverse. [23]

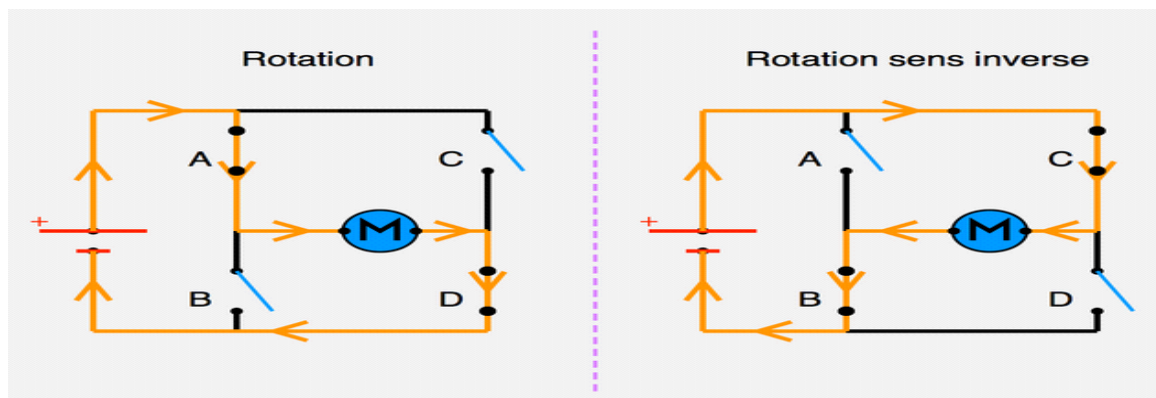


Figure II.18 : Sens du courant en fonction de l'état des interrupteurs d'un pont en H. [23]

8. Les accessoires utilisés :

➤ Carte de test (Breadboard) :

Le meilleur outil pour pratiquer les circuits électroniques est la planche à pain. Une planche à pain est un outil utilisé pour créer des prototypes ou tester des circuits électroniques. Le principal avantage de ce système est qu'il est totalement réutilisable car il ne nécessite aucune soudure. Il vient dans différentes tailles et formes, mais le design principal est toujours le même.

Ces planches sont vraiment pratiques. Ils sont parfaits pour expérimenter avec des circuits électroniques parce que nous avons juste besoin de brancher les composants et les fils. Cela signifie qu'il sera impossible de brancher un composant avec deux broches sur la même ligne, car cela provoquer un court-circuit. Par conséquent, le composant doit être branché sur deux lignes parallèles différentes. Cela peut sembler assez difficile, mais une fois que la planche à pain a été utilisée pendant un certain temps, elle est facile à utiliser. [24]

La figure II.19 représente la carte de teste ou Breadboard.

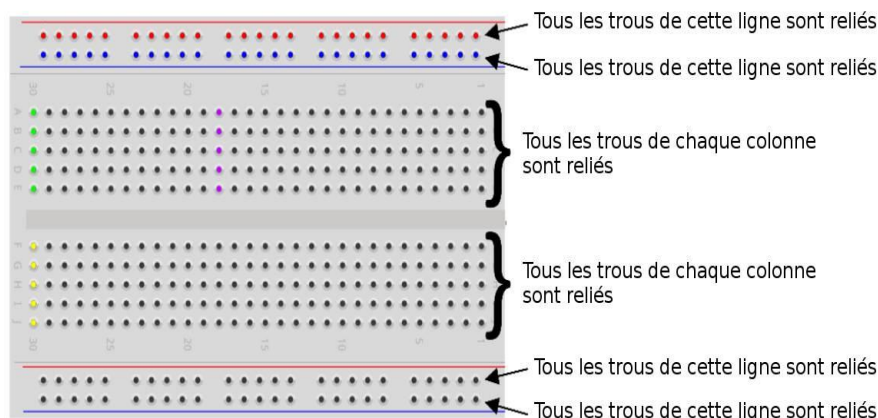


Figure II.19 : carte de test. [24]

➤ Câble USB :

Cela vous permet de connecter votre Arduino UNO à votre ordinateur personnel pour la programmation. Il fournit également de l'électricité à l'Arduino pour la plupart des projets du kit.

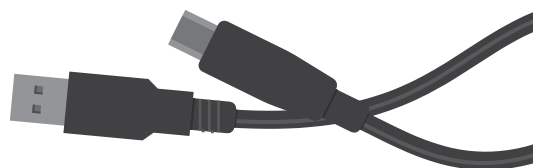


Figure II.20 : Câble USB.

9. Conclusion :

Dans ce chapitre, nous avons mis en évidence la carte d'acquisition Arduino et donné ainsi les raisons qui ont été choisies, puis nous en avons mentionné les différents types. Nous avons ensuite expliqué la partie de base d'Arduino (la partie matérielle) dans laquelle nous avons expliqué plus précisément tous les dispositifs utilisés dans le projet. Nous avons également expliqué le principe de fonctionnement de la carte Arduino sans oublier ses propriétés.

CHAPITRE III

Réalisation et Application

1. Introduction :

Dans ce chapitre on présente notre projet qui est un robot mobile commandé à distance par un smart phone sous Android 4.4 via Bluetooth ainsi équipé d'un capteur de GAZ CO qui génère une sirène dans le cas d'une fuite de gaz. A ce fin, nous devons construire un prototype pour ce robot qui a un petit cadre en plastique et trois roues, et doit être capable de contrôler cette boîte et être connecté à des circuits de puissance, de contrôle et capteur de GAZ et carte Arduino qui contient un programme qui nous permet de gérer tous les Applications de cet Android.

2. La conception de robot

2.1 La structure de base de notre robot :

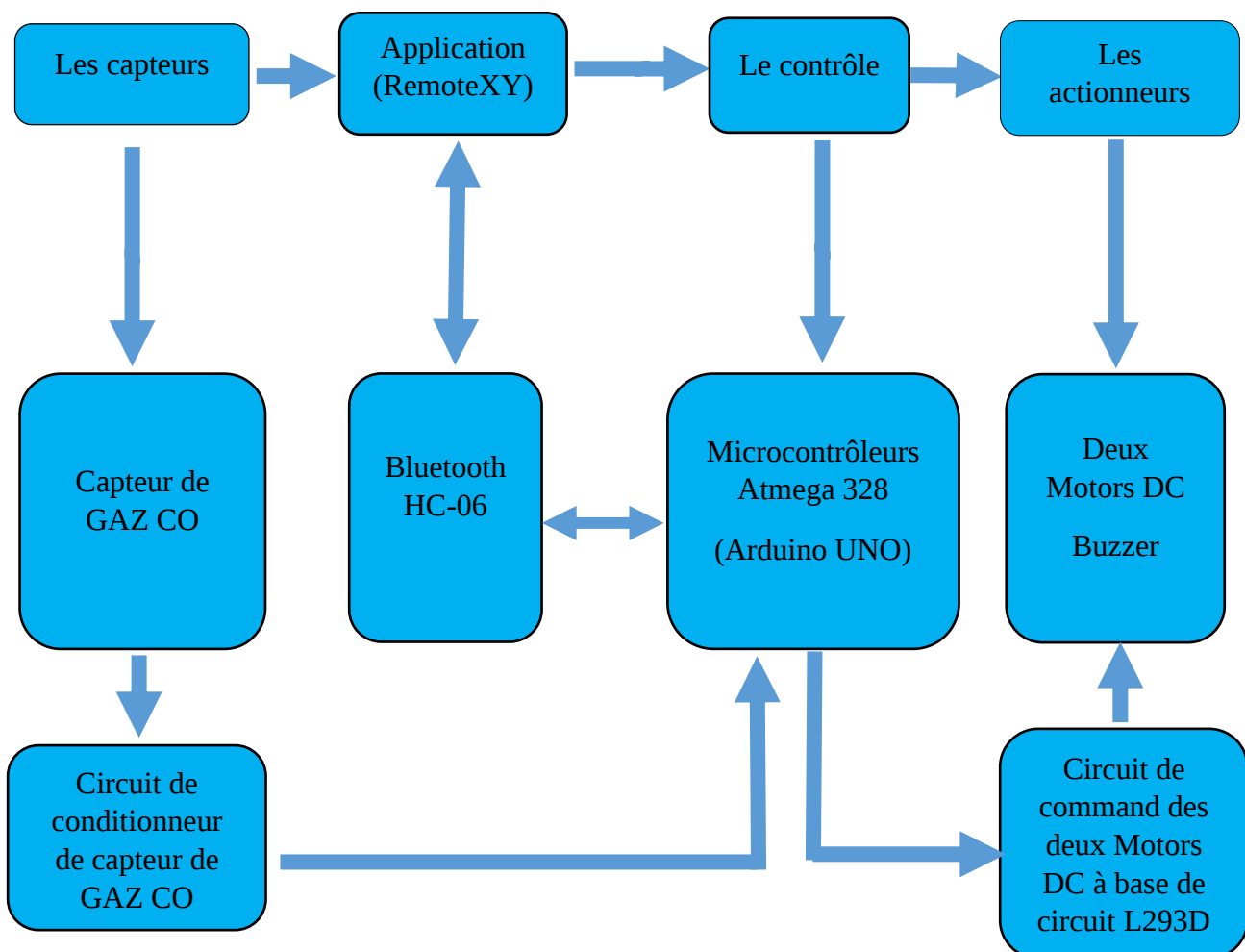


Figure III.1 : La structure de base du robot.

Les signaux sont envoyés via la fonction Bluetooth du smartphone via l'application RemoteXY au module Bluetooth HC 06, Il est ensuite envoyé au carte Arduino pour lire, traiter et décoder les signaux dans les microcontrôleurs, cette information est envoyée au driver L293D qui contrôle le sens de rotation des deux moteurs, donc le robot prêt à naviguer.

En même temps, Nous avons un capteur de gaz monoxyde de carbone installé sur le robot, il capte les molécules de gaz au milieu, Et l'envoie au microcontrôleur qui est au niveau du traitement et de la mesure du volume de gaz, Si le volume de gaz au-dessus du seuil envoie un signal (5V) au buzzer Si moins, il n'envoie pas (0V), En outre, affiche la taille du gaz mesurée sur l'application.

3. Comment réaliser un robot mobile contrôlé par téléphone portable par une application sous Android

3.1 La Partie mécanique :

Le robot est constitué de deux roues motrices indépendantes qui permettent d'avancer, de reculer, de tourner à gauche ou à droite. Ainsi une roue folle permettant de maintenir l'équilibre du robot.

Notre projet base sur une chassie avec des pièces partantes les caractéristiques suivantes :

Désignation	Caractéristiques	Quantité
Châssis	Dimension 21x15cm	01
Moteur avec réducteur	- Tension : 3V ~ 12V DC - Vitesse sans charge : 170 TRS/MIN (3V) - Courant Max 250mA	02
Roue	- Diamètre 6.5cm - Hauteur 2.5cm	02
Encodeur de Vitesse	- Diamètre 2.5cm	02
Boîte de Batterie (Pile)	/	01
Interrupteur	/	01
Kit vis écrou	/	01

Tableau III.1 : Nomenclature chassie du robot.



Figure III.2 : Nomenclature Chassie du Robot.

3.2 La Partie logiciel :

3.2.1 Proteus

Proteus est un logiciel regroupant ISIS, ARES. Tous ces modules sont destinés à l'électronique. Grâce à ce logiciel, nous pouvons réaliser des schémas structuraux et les simuler. Développe par L'absenter Electronics, ces logiciels présents dans Proteus, permettent la CAO (Conception Assistée par Ordinateur) dans le domaine de l'électronique. Ce logiciel est bien connu et utilisé dans de nombreuses entreprises et organismes de formation (universités). [25]

3.2.2 ISIS

ISIS (Intelligent Schematic Input System) est principalement connue pour éditer des Schémas électriques. Il permet aussi de simuler les schémas réalisés auparavant. Grâce à lui, Nous pouvons également contrôler la majorité de l'aspect graphique du circuit. [25]

3.2.3 ARES

Le logiciel ARES est un outil d'Édition et de routage qui complète parfaitement ISIS. Un schéma électrique réalisé sur ISIS peut alors être importé facilement sur ARES pour réaliser le PCB (Printed Circuit Board). Bien que l'édition d'un circuit imprimé soit plus Efficace lorsqu'elle est réalisée manuellement par routage, ce logiciel permet de placer Automatiquement les composants et de réaliser le routage automatiquement. [25]

3.2.4 Fritzing

On a créé le schéma de montage avec le programme « fritzing », Fritzing est Un logiciel libre de conception de circuit imprimé permettant de concevoir de façon Entièrement graphique

le circuit et d'en imprimer le typon. Le logiciel conçu par cette Association, également nommé Fritzing, est un logiciel d'édition de circuit imprimé. Il est Disponible dans 16 langues dont le français. Il est adapté aux débutants ou confirmés en Électronique pour faire rapidement des circuits simples.

Le logiciel comporte trois vues principales :

- La « platine d'essai », où l'on voit les composants tels qu'ils sont dans la réalité et où l'on Construit le montage.
- La « vue schématique », représentant le schéma fonctionnel du circuit.
- Le « circuit imprimé », représentant la vue du circuit imprimé tel qu'il sera sorti en PDF Pour être imprimé. [26]

3.3 Partie électronique

3.3.1 Les Cartes Electroniques

Pour la partie électronique nous avons réalisé deux circuits électroniques avec les composants nécessaires, le premier c'est la carte de command, le deuxième est la carte de puissance.

3.3.1.1 Carte de command

Le cœur de notre carte est un microcontrôleur de la famille Atmel « Atmega 328 p » Ce dernier va contenir les algorithmes responsables de l'intelligence du robot. En effet Atmel permet de faire la liaison entre les capteurs et les moteurs de mouvements, par les bits des ports d'entrées sorties.

- Schéma électronique de la carte de commande, de puissance et Bluetooth réalisé par le logiciel Proteus.

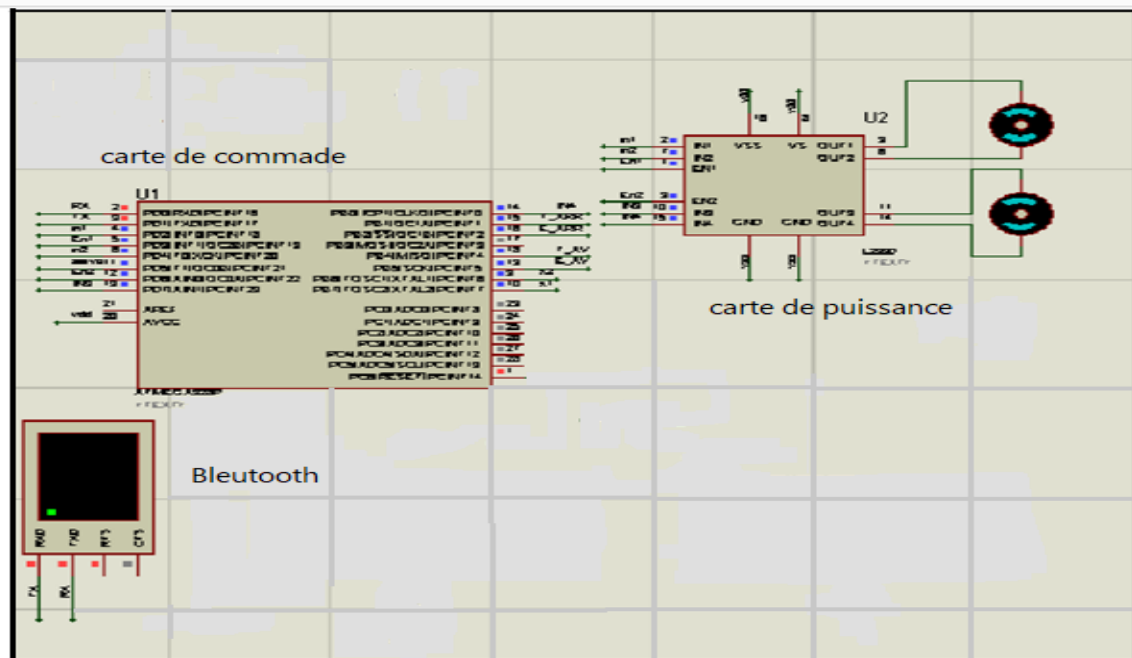


Figure III.3 : Schéma électronique de carte de commande et puissance et Bluetooth.

• Réalisation Pratique

Le circuit imprimé simple face dont l'Arduino y installé est illustré par la figure III.4, une représentation 3D de cette carte est représentée par la figure III.5

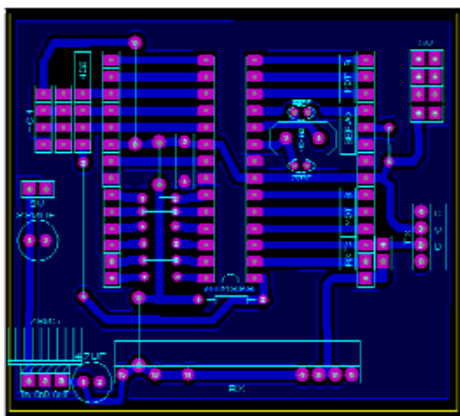


Figure III.4 : carte de command-cote composants.

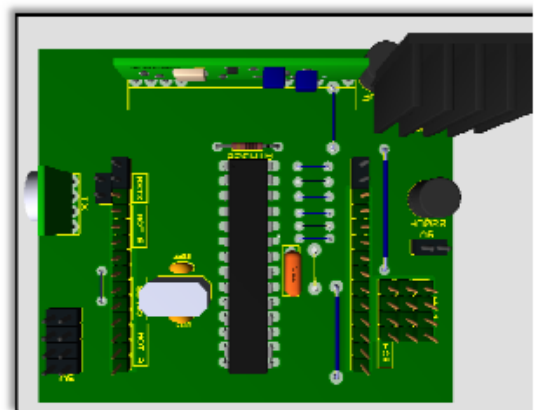


Figure III.5 : carte de command-3D.

3.3.1.2 Carte de puissance

Le cœur de notre carte est un hacheur « L293D » ce dernier va commander les deux moteurs avec la puissance nécessaire de 9V, une représentation de schéma électrique de cette carte de puissance est représentée par la figure III.6.

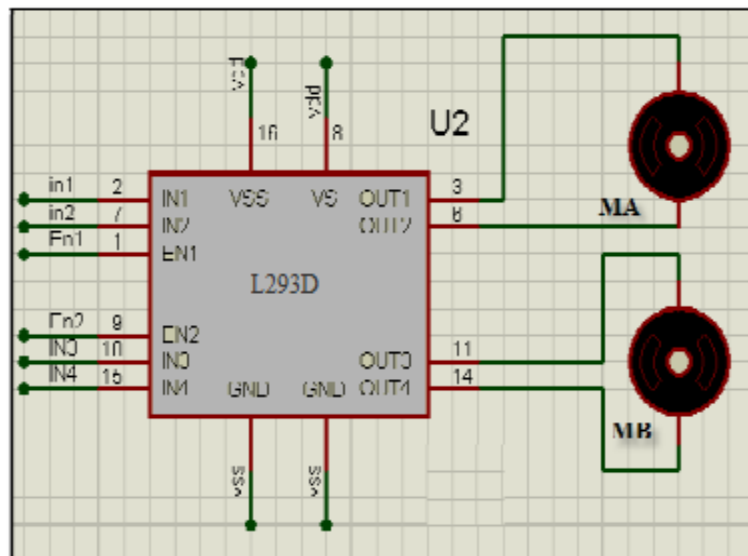


Figure III.6 : Schéma électronique de carte de puissance L293D.

Le circuit de puissance imprimé du circuit intègre L293D est illustré par la figure III.7, la figure III.8 illustre la représentation de 3D

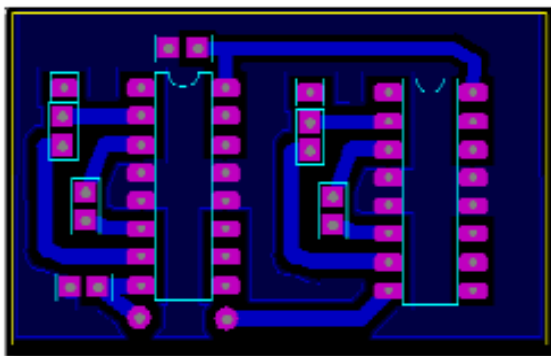


Figure III.7 : carte de puissance-cote composants.

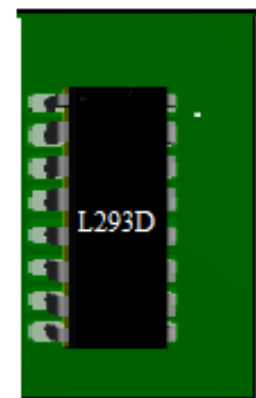


Figure III.8 : carte de puissance -3D.

3.3.2 La structure du capteur MQ7 en général :

- L'élément sensible (matériau fritté ou dépôt de couches minces),
- Un substrat chauffant (résistance interne du capteur),
- Adaptation de l'élément sensible sur le substrat chauffant (connectique).

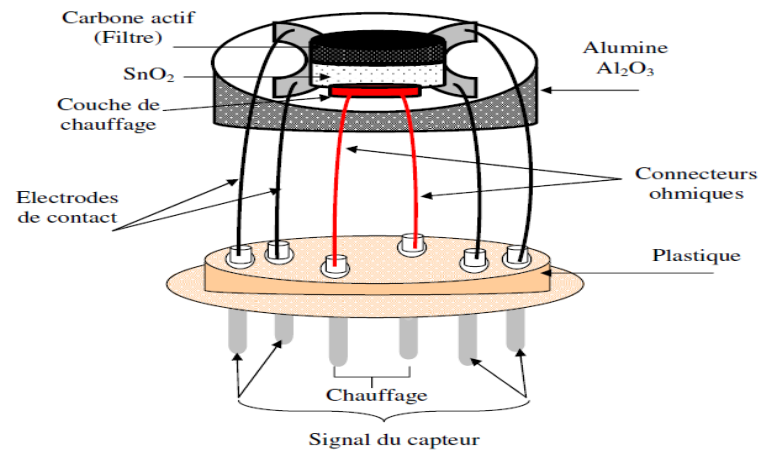


Figure III.9 : structure interne de capteur MQ7.

➤ Élément sensible

L'élément sensible (Fig. III.9) du capteur MQ7 est constitué de :

- Une couche semi-conductrice poreuse : couche sensible au gaz CO, à base de L'oxyde d'étain SnO_2 pur (non dopée).
- Dépôt d'une couche filtrante de carbone actif, sur la surface de la couche sensible SnO_2 , pour filtrer et réduire les interférences d'autres gaz tels que les oxydes d'azotes NO_x , les alcanes (méthane, etc.) [32].
- Un micro tube céramique en alumine Al_2O_3 , qui sert comme un support à l'élément sensible et un séparateur thermique entre le système de chauffage et le plastique qui couvre la partie inférieure du capteur (Fig. III.9),
- Un ensemble d'électrodes de contact, pour faire sortir le signal électrique.

➤ Résistance chauffante

On trouve dans le capteur une couche résistive (résistance chauffante), destinée à jouer le rôle de système de chauffage et de contrôle de la température, car l'interaction chimique entre le gaz CO à détecter et la couche sensible (semi - conducteur SnO_2), nécessite une température, pour exciter les électrons de la couche sensible et modifier sa conductivité électrique, pour donner à fin un courant électrique. Pour chauffer la couche résistive, le courant d'alimentation passe à travers cette couche, en utilisant les connecteurs ohmiques (Fig. III.9).

➤ Principe de fonctionnement :

Une famille fait appel à l'adsorption physico-chimique sur la surface du capteur, et la modification d'une grandeur physique en résultant est alors aisément mesurée. Le plus souvent

cette grandeur sera la résistivité du matériau, ce qui modifie la conductibilité du matériau. Son principe est de mettre en jeu un semi-conducteur dont la conductivité est influencée par l'injection ou le piégeage de charges depuis la surface, sous l'influence de gaz à détecter.

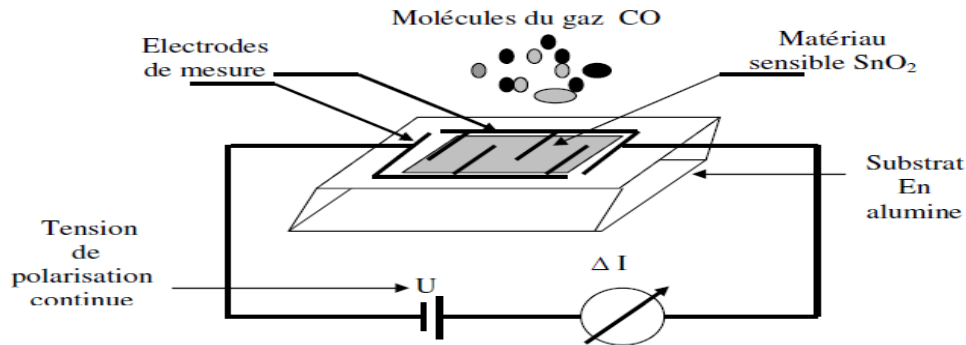


Figure III.10 : cycle de fonctionnement de capteur MQ7.

Dans ce capteur de gaz, on a l'intervention des phénomènes physico-chimique, l'adsorption du gaz à la surface du capteur (sur le matériau sensible) et électroniques (Fig.III.10). Après l'adsorption du gaz sur le matériau sensible, on aura une interaction du gaz à détecter avec le matériau, en présence de la température (l'effet de substrat chauffant), et ceci va modifier la conductivité du matériau, en donnant courant électrique (Fig.III.10). Par conséquent, la mesure du courant permet de nous donner, outre l'information de l'existence du gaz, la quantité de gaz détecté (Fig.III.10).

Il y a aussi un type redresseur. Alors le gaz adsorbé modifie la hauteur de la barrière de potentiel existant au niveau du contact. Mais aussi parfois, tout simplement, la variation de masse de l'élément sensible. Si on applique le principe de l'isotherme d'adsorption de Langmuir, la surface du capteur est représentée par un nombre de sites définis. Les sites sont occupés au fur et à mesure de l'augmentation de la quantité de gaz présent, jusqu'à la formation d'une monocouche de gaz occupant tous les sites. Par conséquent, le calcul de la superficie spécifique de la masse de la monocouche formée sur la surface du capteur peut nous donner la quantité du gaz. [32]

- Schéma du montage de notre projet

La figure III.9 représente schéma général de notre projet illustré Comment installer les deux moteurs, Bluetooth, L293D, capteur CO, buzzer avec la carte Arduino.

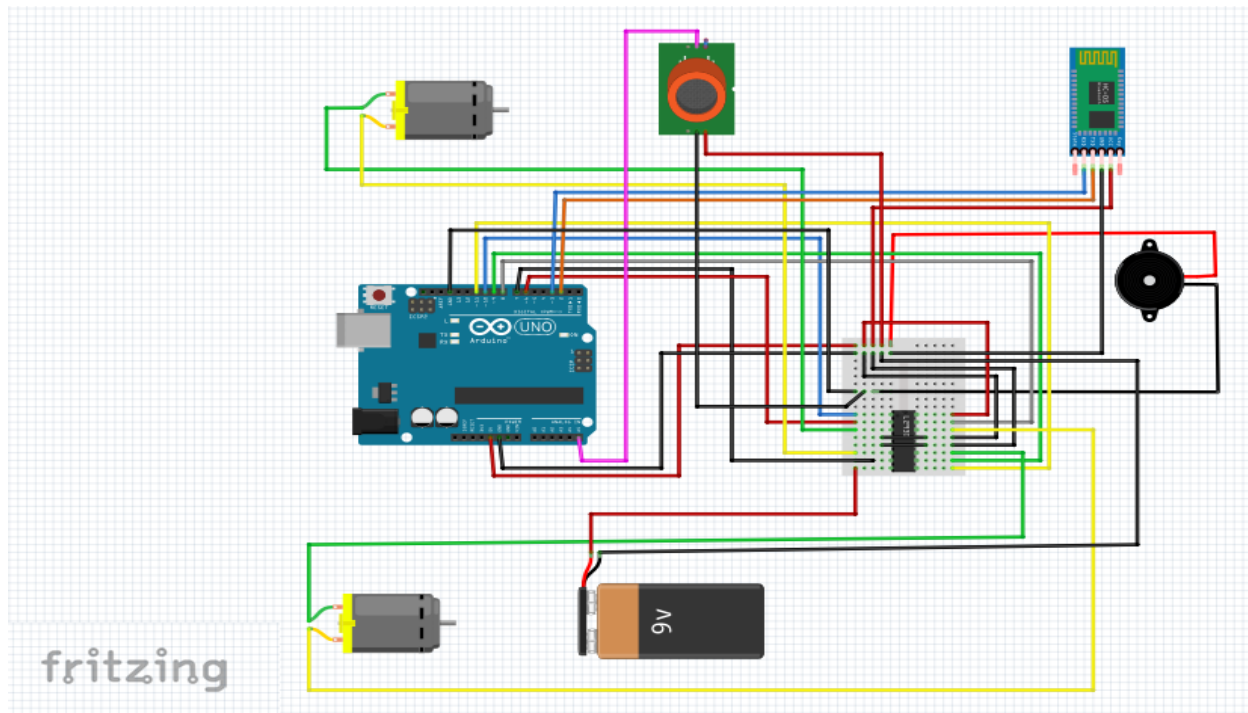


Figure III.11 : schéma générale de projet.

- Le circuit avec une vue schématique

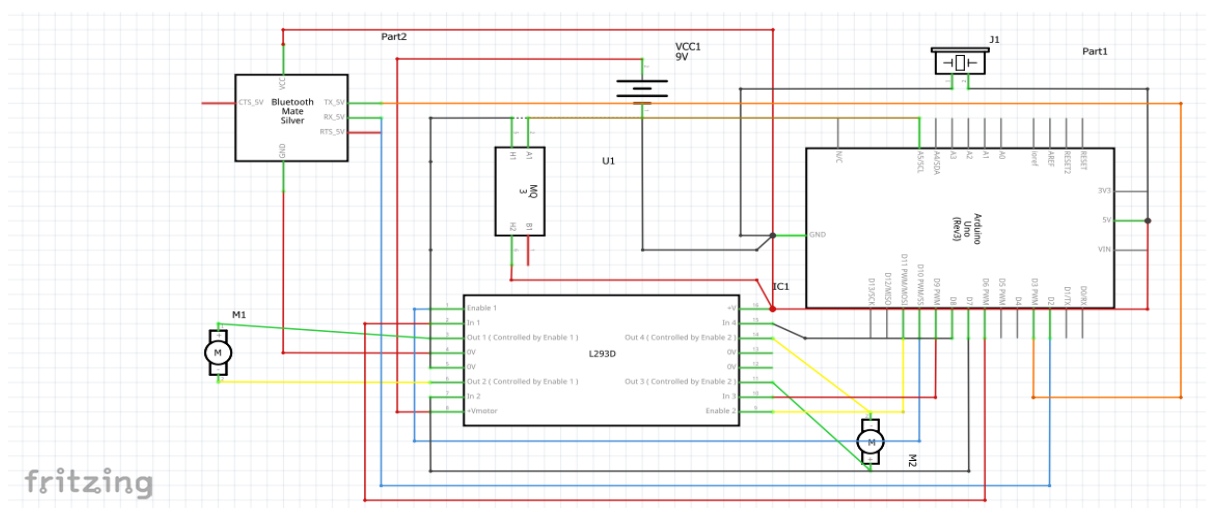


Figure III.12 : Vue schématique de notre circuit.

- **Les détails du montage :**

Au premier lieu on a branché le driver L293D avec la carte Arduino UNO, donc on a raccordé les fils comme suit :

Les fils du signal pour les deux moteurs, les fils de signal des moteurs doivent être branchés avec des pins qui génèrent un signal PWM.

 Brancher les fils de signal avec le driver L293D :

PIN 10 avec ENABLE 1

PIN 6 avec INPUT 1

PIN 7 avec INPUT 2

PIN 11 avec ENABLE 2

PIN 8 avec INPUT 4

PIN 9 avec INPUT 3

 Maintenant raccorde les fils d'alimentation avec le driver L293D :

Vcc d'Arduino avec Vcc1

GND d'Arduino avec GND (les quatre GND).

 Brancher les deux moteurs avec le drive L293D :

Pôle positif de moteur droit avec OUTPUT3

Pôle négatif de moteur droit avec OUTPUT4

Pôle positif de moteur gauche avec OUTPUT1

Pôle négatif de moteur gauche avec OUTPUT2

 Brancher le module Bluetooth HC06 avec la carte Arduino :

On raccorde le VCC du module avec le 5v de l'Arduino.

Le GND avec le GND de la carte

La patte TX du module avec le PIN 2 de la carte Arduino.

La patte RX du module avec le PIN 3 de la carte Arduino.

 Brancher le module buzzer :

On raccorde le VCC du module avec le 5v de l'Arduino.

Le GND avec le GND de la carte.

La patte signal branché avec le PIN 12.

 Brancher le capteur MQ7 :

On raccorde le VCC du module avec le 5v de l'Arduino.

Le GND avec le GND de la carte.

La patte de sortie (A) avec le PIN analogique A5.

🔧 Finalement ajouter alimentation externe (batterie 9v) :

Vcc de Batterie avec Vcc2 de driver L293D.

GND de batterie avec tous les GND.

Batterie pour alimenter l'Arduino.

3.4 L'application utilisée dans le contrôle :

3.4.1 Une application RemoteXY :

Une application RemoteXY sous Android est utilisée par notre smartphone pour contrôler notre robot mobile. L'application RemoteXY est un moyen facile utilisée comme une interface graphique d'utilisateur pour un objectif de contrôle via un smartphone ou tablette.

✓ **Le système comprend :**

- ✓ Editeur d'interfaces graphiques mobiles pour les cartes de couplage.
- ✓ Application mobile RemoteXY qui permet de se connecter au contrôleur et de le contrôler via une interface graphique.

✓ **Caractéristiques distinctives :**

- ✓ La structure d'interface est stockée dans le contrôleur. Une fois connecté, il n'y a pas d'interaction avec les serveurs pour télécharger l'interface. La structure d'interface est téléchargée vers l'application mobile à partir du contrôleur.
- ✓ Une application mobile peut gérer tous vos appareils. Le nombre d'appareils n'est pas limité.

✓ **Connexion entre le contrôleur et l'appareil mobile en utilisant :**

- ✓ Bluetooth ;
- ✓ Client Wifi et point d'accès.
- ✓ Ethernet par IP ou URL.
- ✓ Internet depuis n'importe où via le serveur cloud.

✓ **Le générateur de code source supporte les contrôleurs suivants :**

- ✓ Arduino UNO, Arduino MEGA, Arduino Leonardo, Arduino Pro Mini, Arduino Nano, Arduino MICRO.

✓ **Modules de communication supportés :**

- ✓ Bluetooth HC-05, HC-06 ou compatible.
- ✓ WiFi ESP8266.
- ✓ Ethernet Shield W5100.

- ✓ IDE pris en charge :
- ✓ IDE Arduino.
- ✓ MPIDE.
- ✓ Système d'exploitation mobile pris en charge :
- ✓ Android.
- ✓ IOS. [27]

3.4.2 Les Étapes pour créer l'application RemoteXY :

Étape 1. Créez un projet d'interface utilisateur graphique

Enter dans [l'éditeur RemoteXY](#) créer l'interface définissez les boutons dans le champ du smartphone.

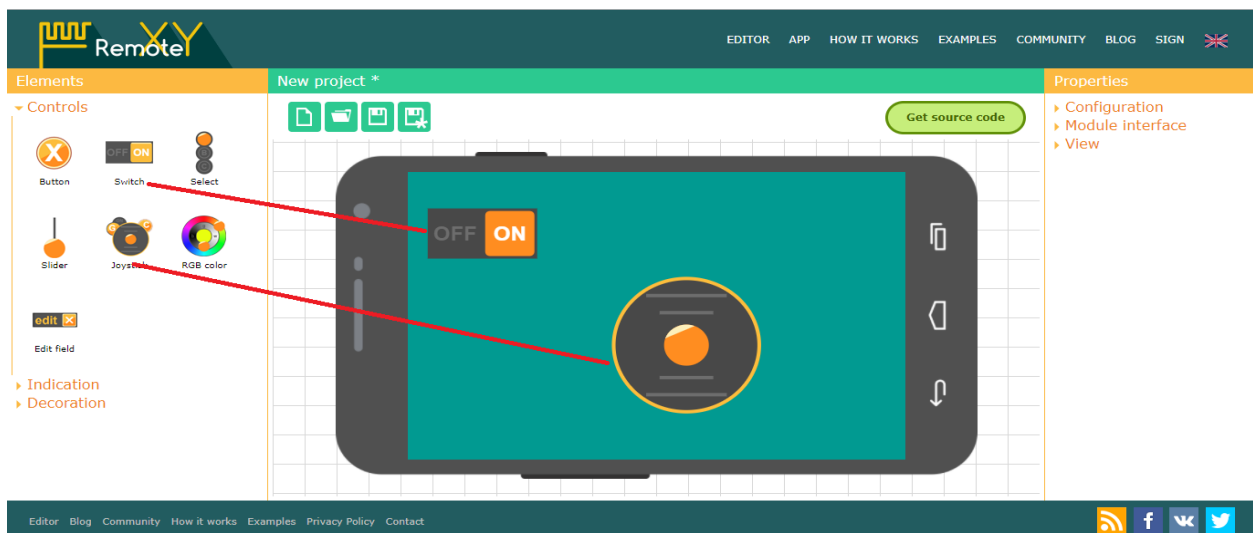


Figure III.13 : l'éditeur RemoteXY.

Nous Faisons quelques additions contrôles additionnels « centering button » et « G-sensor button ».

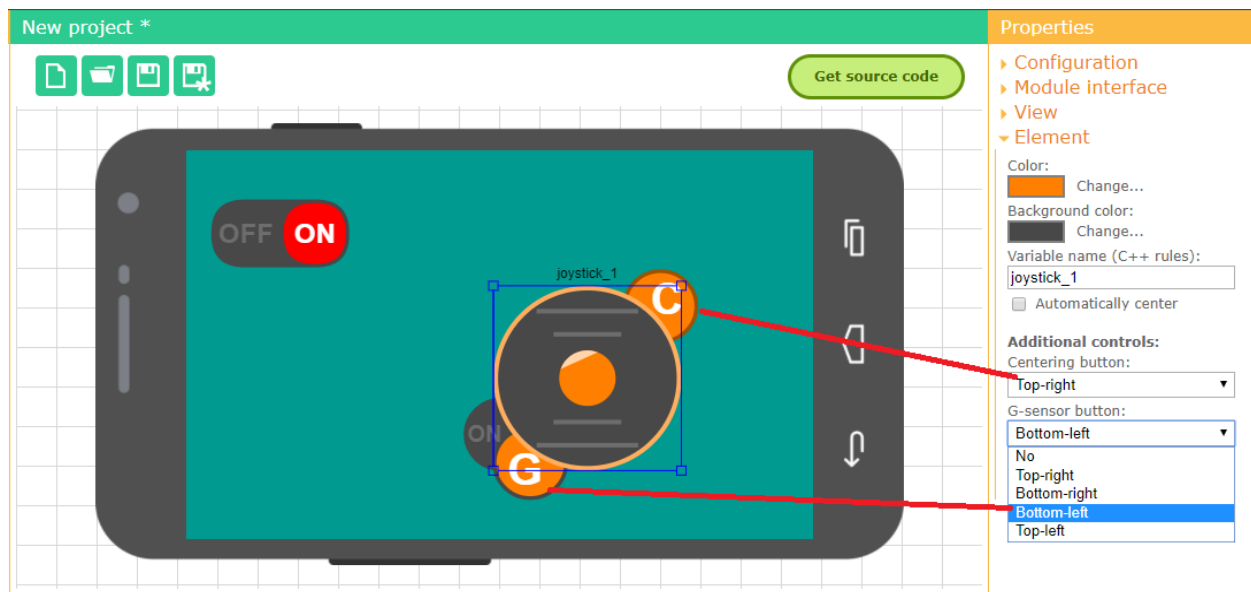


Figure III.14 : quelques additions sur l'interface.

Étape 2. Configurer le projet :

Dans le volet de droite, sélectionnez les paramètres suivants sous l'onglet "Configuration" et "Module interface".

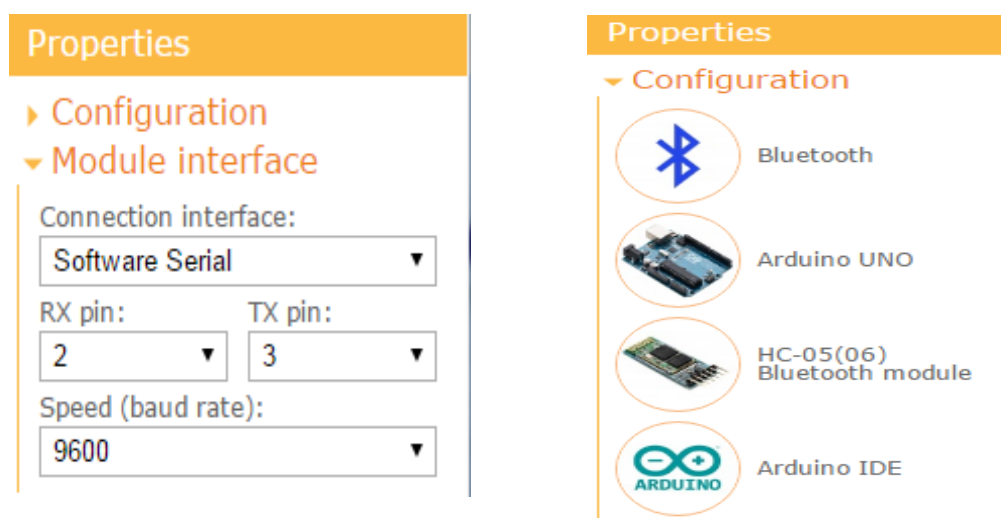


Figure III.15 : configuration et module d'interface de l'application.

Les paramètres indiquent que le module HC-05 (06) est connecté à l'Arduino via le port série logiciel software Serial en utilisant les contacts 2 et 3 à 9600 vitesses. C'est la vitesse standard pour les modules HC-05 (06).

Étape 3. Obtenez l'esquisse pour l'Arduino

- Appuyez sur le bouton "Obtenir le code source" dans l'éditeur.
- Dans la page ouverte avec le code source de l'esquisse, téléchargez-la sur votre ordinateur et ouvrez-la dans l'IDE Arduino.
- Aussi, téléchargez la bibliothèque RemoteXY.
- La bibliothèque étant correctement installée, le code source de l'esquisse doit être compilé sans erreur.

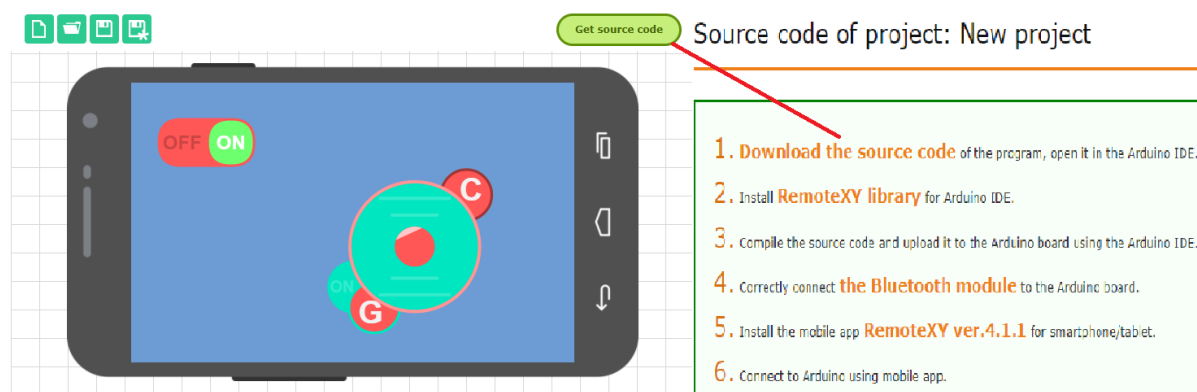


Figure III.16 : l'interface finale d'application.

3.5 Arduino IDE

Une telle carte d'acquisition qui se base sur sa construction sur un microcontrôleur doit être dotée d'une interface de programmation comme est le cas de notre carte.

L'environnement de programmation open source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux). [6]

3.5.1 L'environnement de la programmation

Le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information se programme appelé IDE Arduino.

3.5.2 Structure générale du programme (IDE Arduino)

Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation Arduino basé sur la programmation en C.

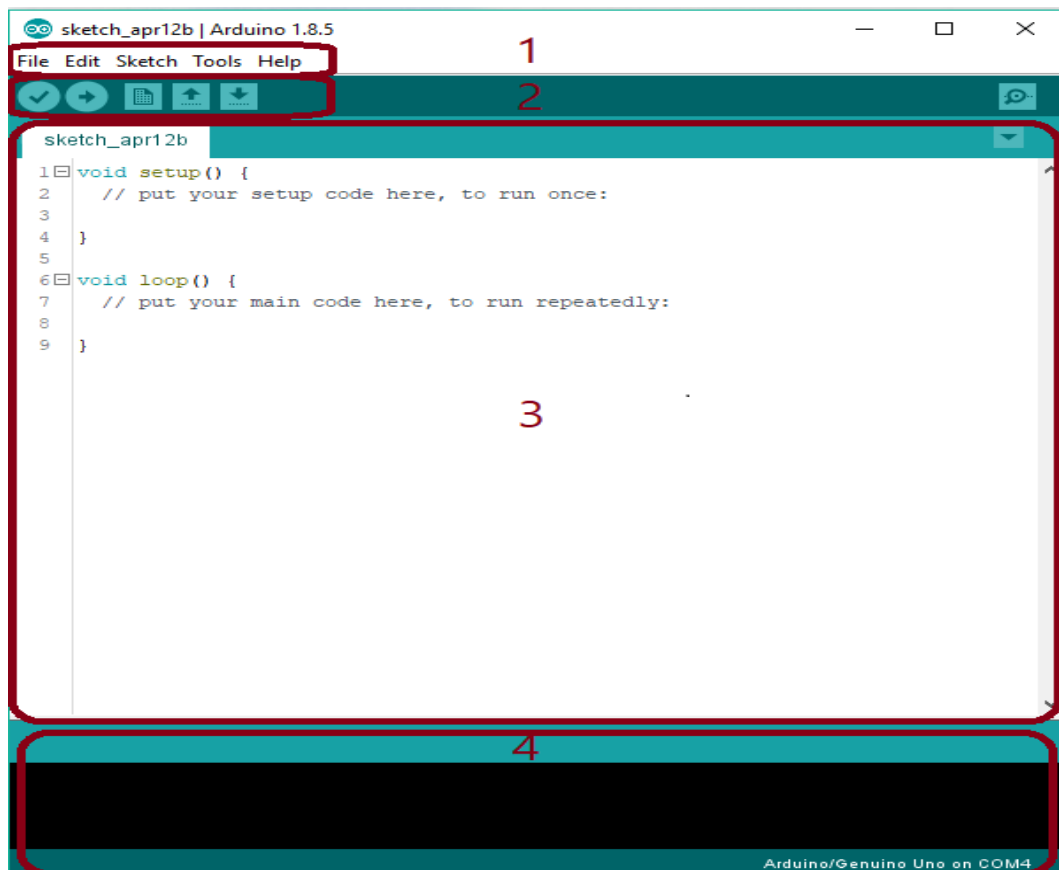


Figure III.17 : Structure générale du programme (IDE Arduino).

- **Correspondance**

Le cadre numéro 1 : ce sont les options de configuration du logiciel

Le cadre numéro 2 : il contient les boutons qui vont nous servir lorsque l'on va programmer nos cartes.

Le cadre numéro 3 : ce bloc va contenir le programme que nous allons créer

Le cadre numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur. [28]

- **Les différents boutons de commande :**



Figure III.18 : Les différents boutons de commande.

- Bouton 1 (verify) : Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
- Bouton 2 (upload) : Compiler et Transférer le programme vers la carte Arduino : permet d'envoyer le programme vers le microcontrôleur de la carte Arduino.
- Bouton 3 (new) : Créer un nouveau fichier.
- Bouton 4 (open) : Charger un programme existant.
- Bouton 5 (Save) : Sauvegarder le programme en cours.
- Bouton 6 (serial monitor) : Permet d'accéder au port série (en RX/TX). [28]

3.5.3 Injection du programme

Avant d'envoyer le programme vers la carte, il faut dire au logiciel quel est le nom de la carte et sur quel port elle est branchée. Choisir la carte que l'on va programmer : ce n'est pas très compliqué, le nom de votre carte est indiqué sur elle. Pour nous, il s'agit de la carte "Uno". Allez dans le menu "**Tools**" ("outils" en français) puis dans "**Board**" ("carte" en français). Vérifiez que c'est bien le nom "arduino /genuino Uno" qui est coché. Si ce n'est pas le cas, cochez-le. [29]

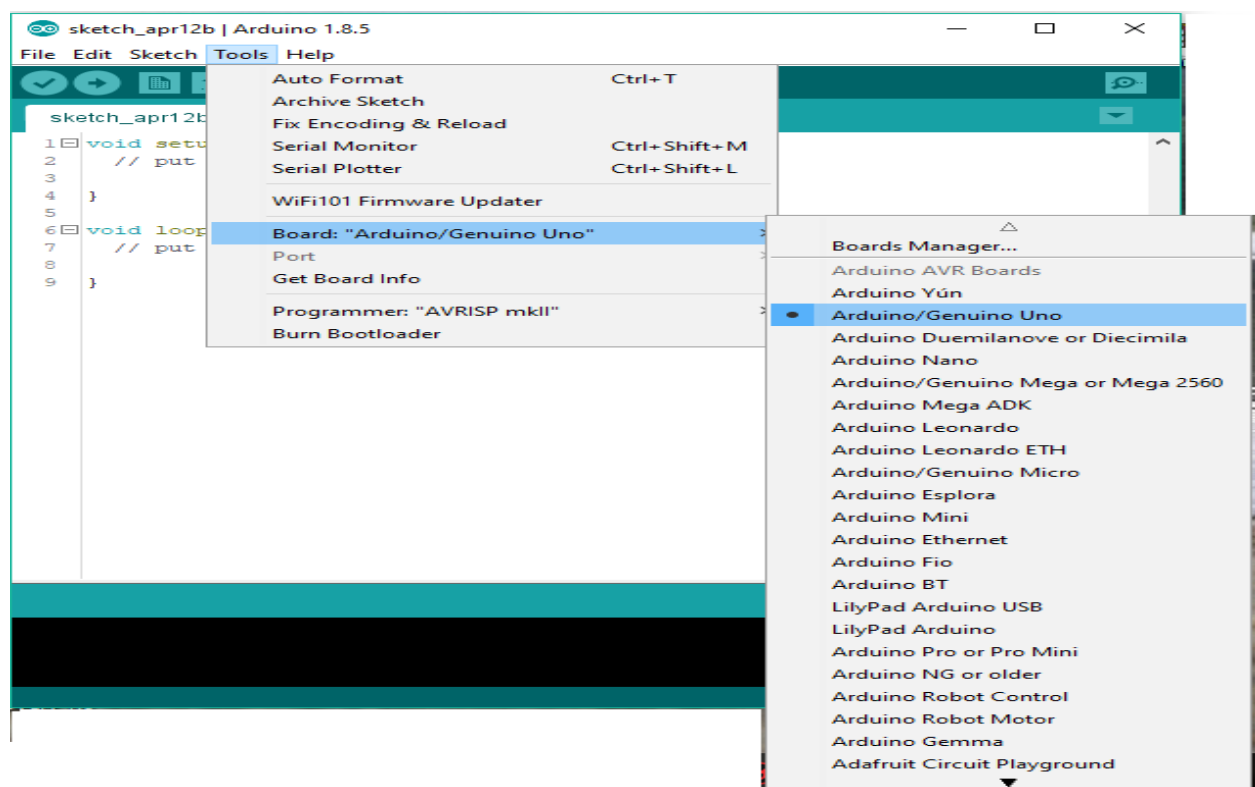


Figure III.19 : Choix de la carte Arduino.

✓ **Choisissez le port de connexion de la carte :**

Allez dans le menu *Tools*, puis *Serial port*. Là, vous choisissez le port COMX, X étant le numéro du port qui est affiché. Ne choisissez pas COM3 car il n'est quasiment jamais connecté à la carte.

Pour trouver le port de connexion de la carte, vous pouvez aller dans le gestionnaire de périphérique qui se trouve dans le panneau de configuration. Regardez à la ligne Ports (COM et LPT) et là, vous devriez avoir Arduino UNO (COMX).

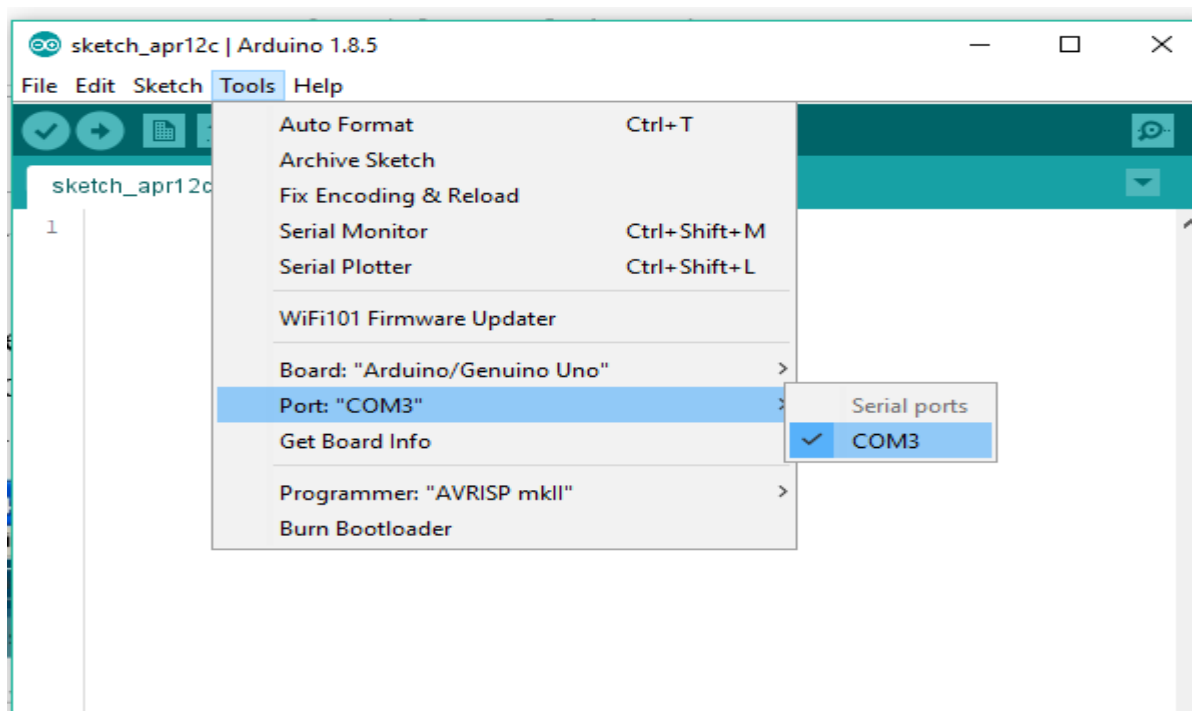


Figure III.20 : Choix du port de connexion de la carte.

Maintenant, il va falloir vérifier le programme. Pour ce faire, il suffit de cliquer sur le bouton "verify" (ou "vérifie" en Français), Montré dans la photo suivante :

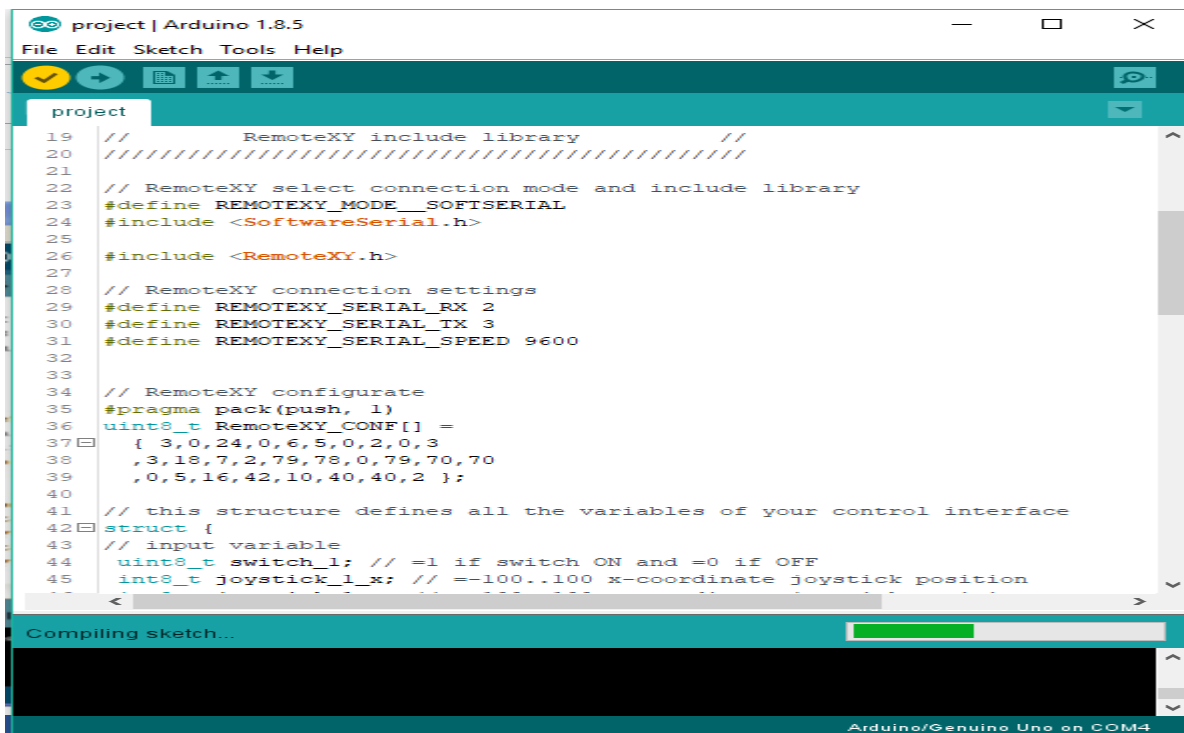


Figure III.21 : vérificaion les erreurs.

Maintenant, il va falloir envoyer le programme dans la carte. Pour ce faire, il suffit de cliquer sur le bouton Upload (ou "Télécharger" en Français), en jaune-orangé sur la photo :

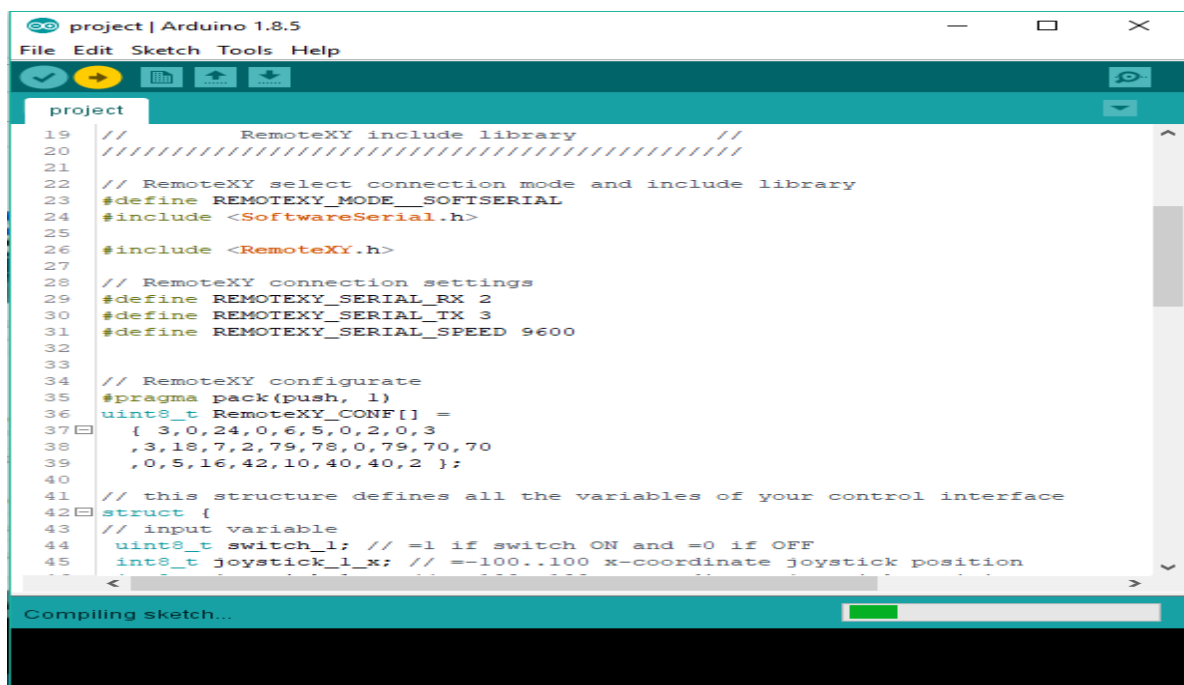


Figure III.22 : début de téléchargement .

En bas dans l'image, vous voyez le texte : "Uploading to I/O Board...", cela signifie que le logiciel est en train d'envoyer le programme dans la carte.

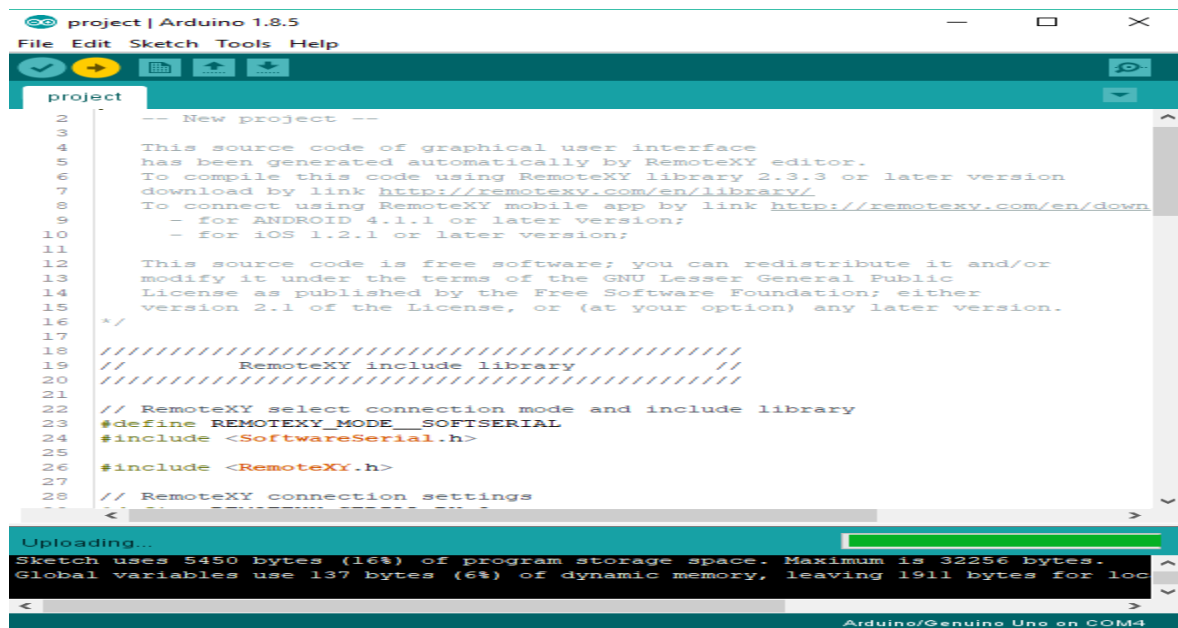


Figure III.23 : Le programme est en cours de téléchargement.

Une fois qu'il a fini, il affiche un autre message : le message affiché : "Done Uploading" signale que le programme a bien été chargé dans la carte.

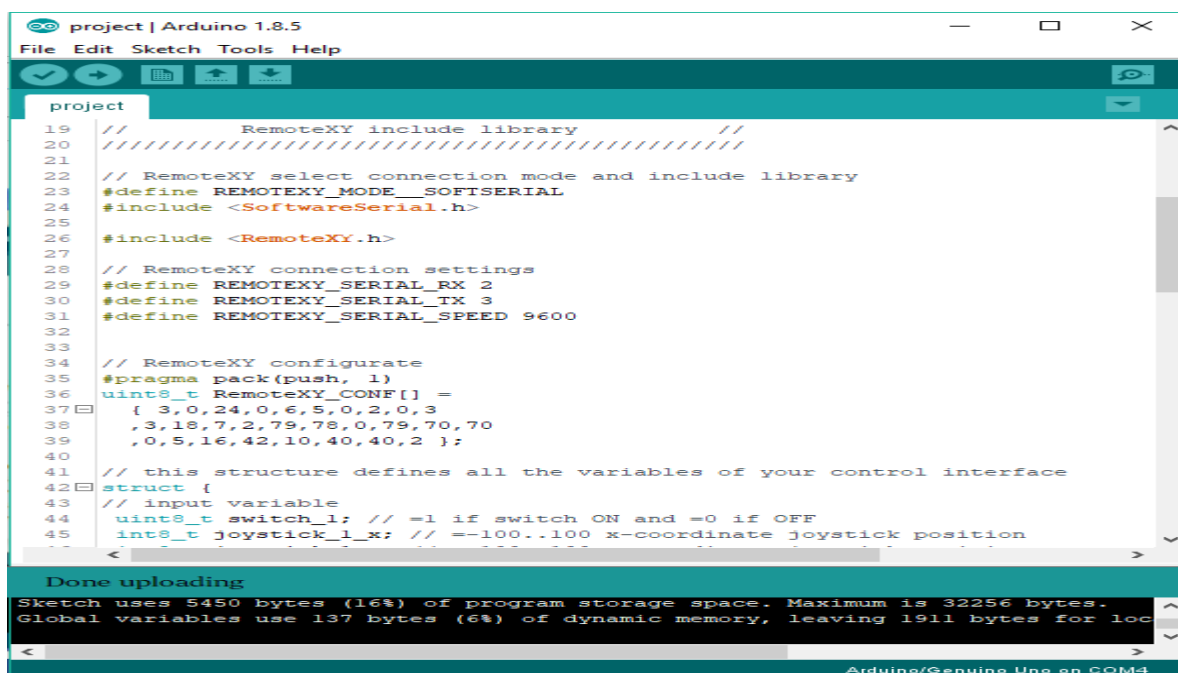


Figure III.24 : Fin de téléchargement.

✓ En résumé, l'injection du code vers la carte Arduino UNO via le port USB passe par les étapes suivantes :

1. On conçoit ou on ouvre un programme existant avec le logiciel IDE Arduino.
2. On vérifie ce programme avec le logiciel Arduino (compilation).

3. Si des erreurs sont signalées, on modifie le programme.
4. On charge le programme sur la carte.
5. On câble le montage électronique.
6. L'exécution du programme est automatique après quelques secondes.
7. On alimente la carte soit par le port USB, soit par une source d'alimentation autonome.
8. On vérifie que notre montage fonctionne.

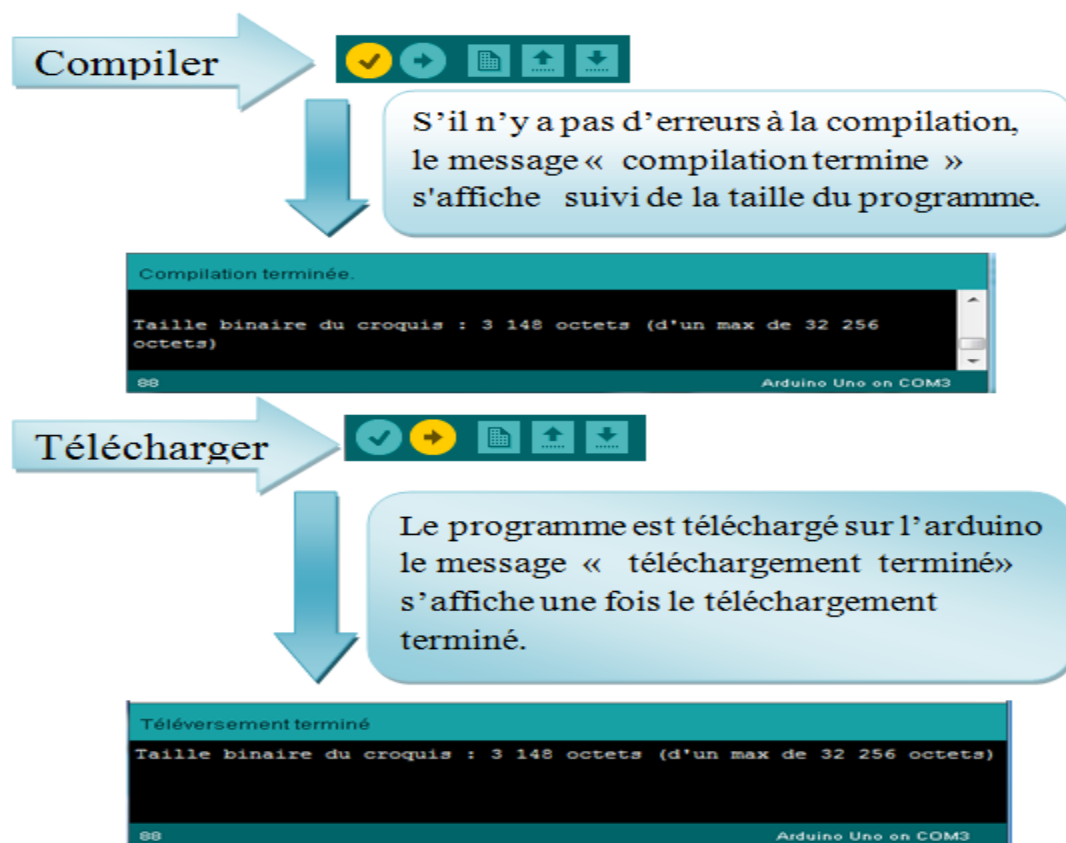


Figure III.25 : Etapes d'injection du code.

Étape 4. Connectez-vous depuis l'application mobile :

Installez l'application mobile RemoteXY sur votre smartphone / tablette.

Nous avons utilisé un smartphone de marque Samsung Galaxy -J2, avec un processeur dual core 1,3 GHz, et 1Go de RAM.

Appuyez sur le nouveau bouton de connexion "+" dans le panneau supérieur de l'application. Dans la fenêtre qui s'ouvre, sélectionnez la connexion "Bluetooth".

Dans l'application, activez le Bluetooth et appuyez sur le bouton pour mettre à jour la liste des appareils disponibles. Le nom de votre appareil est : "HC-06". Sélectionnez-le.

Une fenêtre pour entrer le mot de passe d'appariement Bluetooth s'ouvrira. Le mot de passe par défaut pour HC-06 peut être 1234 ou 0000. Entrez le mot de passe et la connexion commencera.

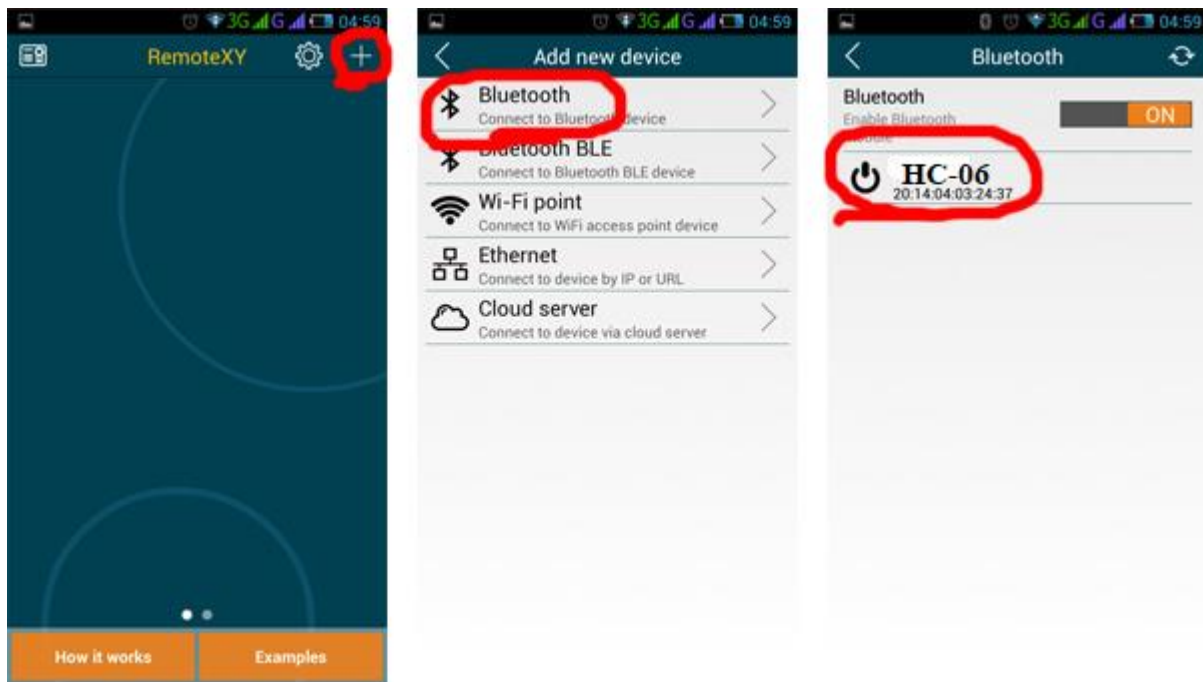


Figure III.26 : Connexion par Bluetooth.

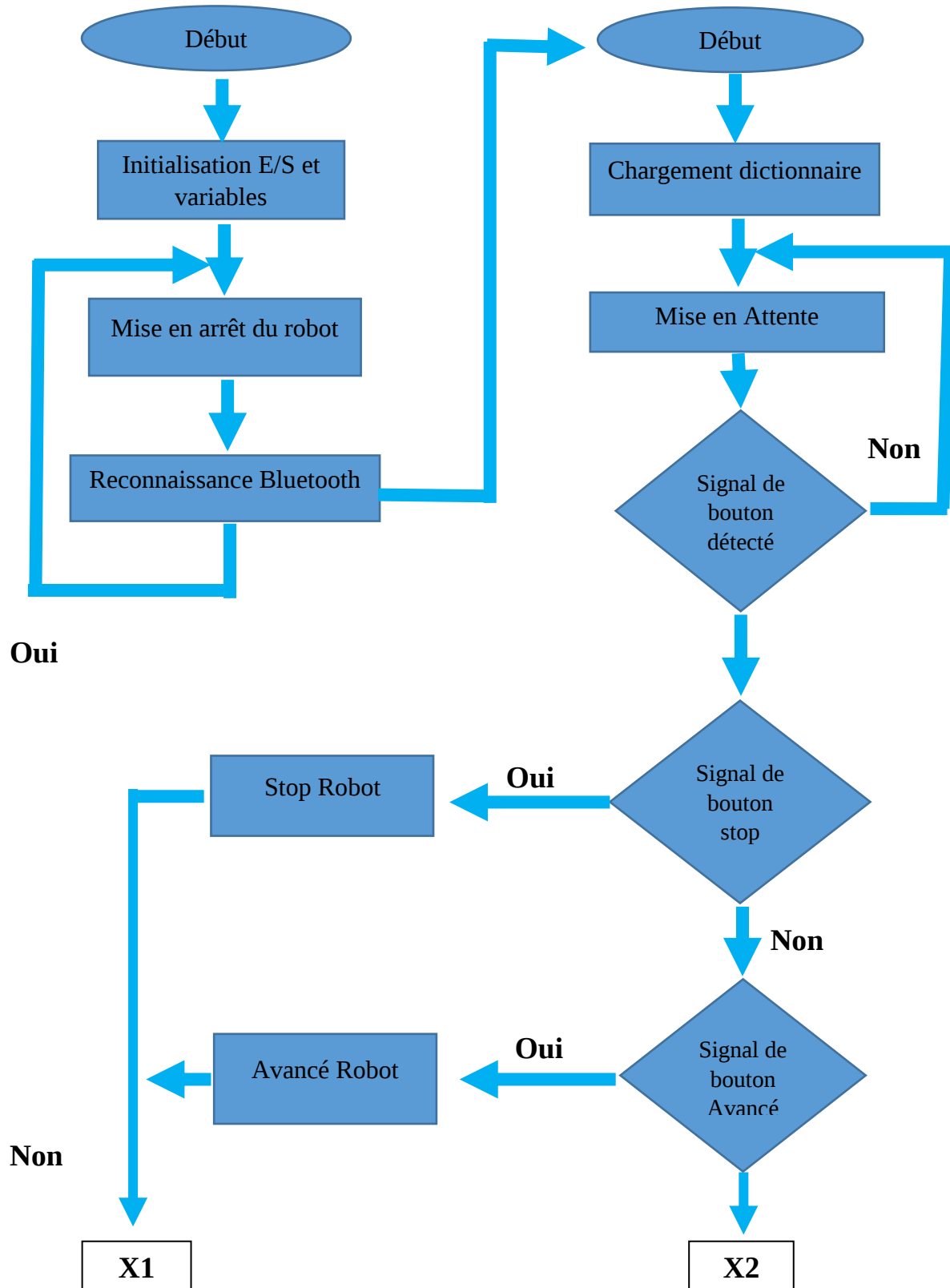
4. Le robot et système d'alarme :

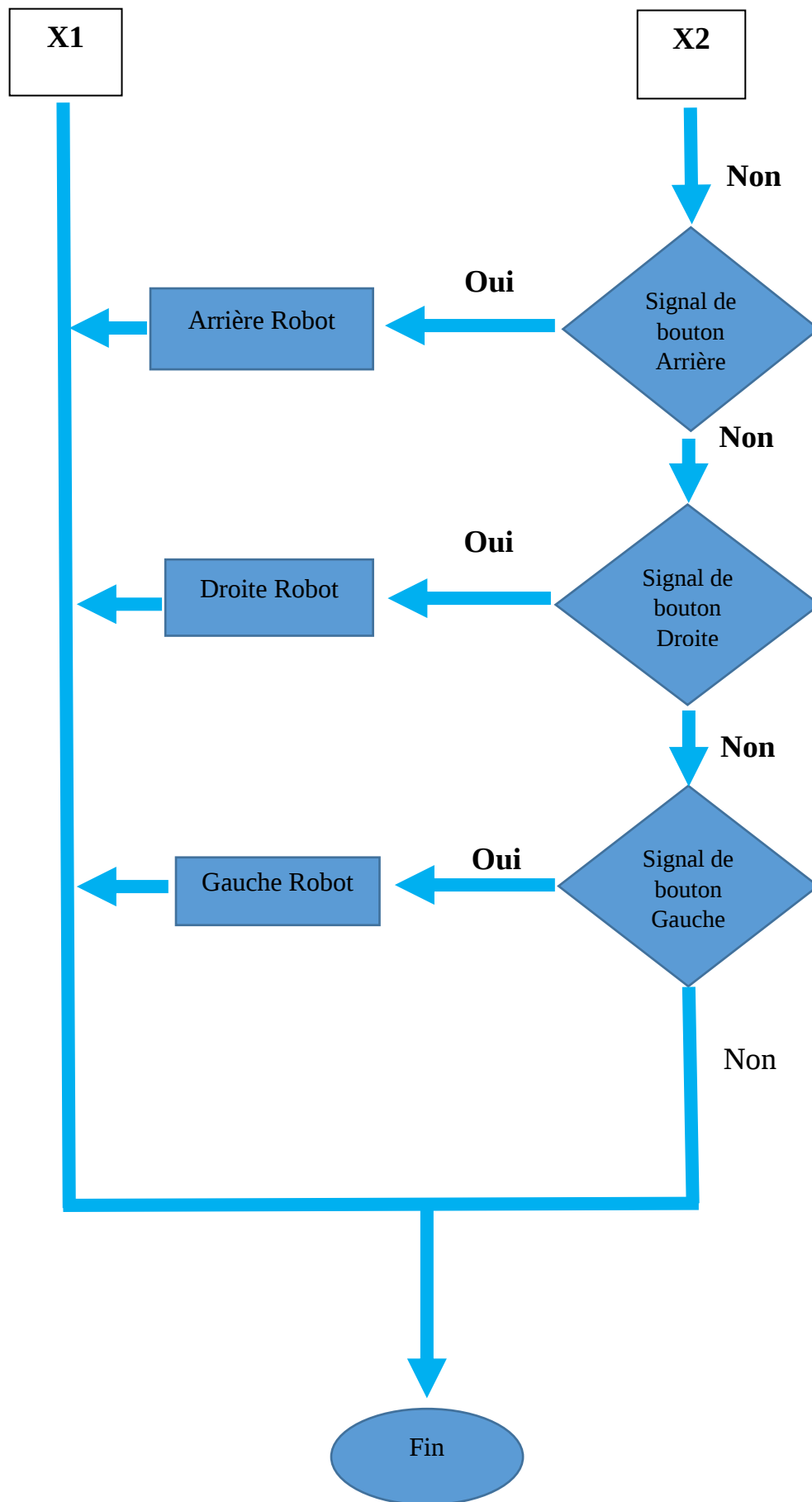
Après toutes ces étapes, le robot peut naviguer en contrôlant le smartphone et en détectant le monoxyde de carbone dangereux qui tue en silence de nombreuses vies, en particulier à l'intérieur avec l'aide du robot, nous pouvons sauver la vie de nombreuses personnes en les alertant de la présence d'une forte proportion de monoxydes de carbone qui constitue un danger pour eux.

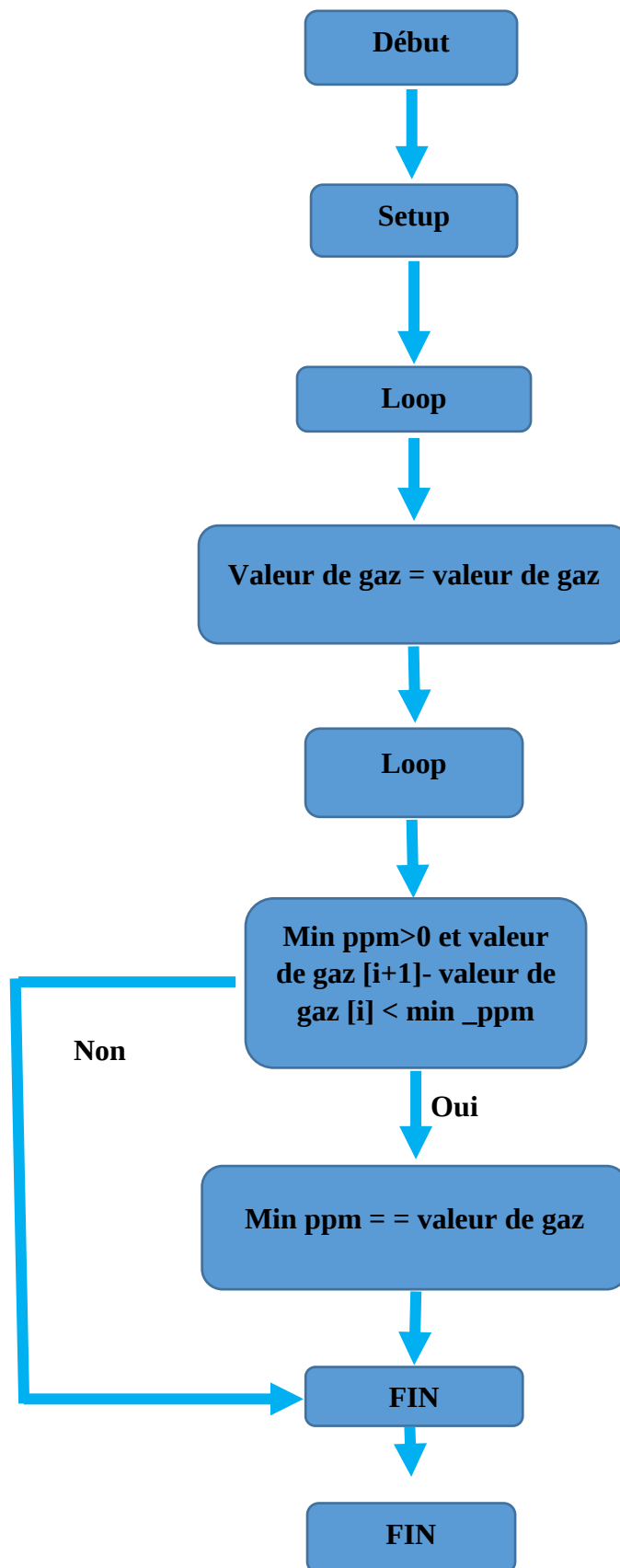
5. Algorithme de l'approche de commande :

Programme principale :

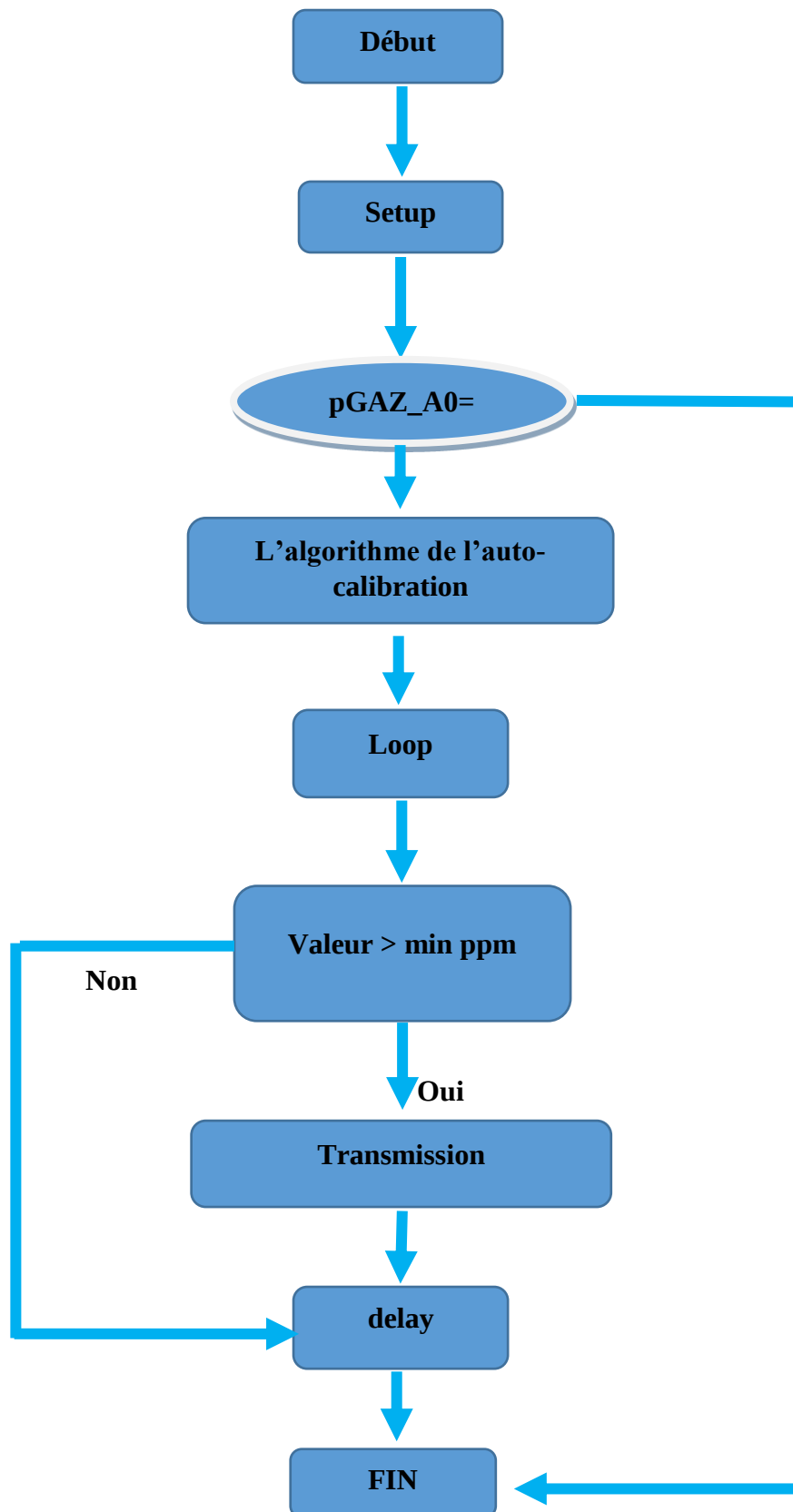
Sous-programme : Reconnaissance Bluetooth





6. L'algorithme de l'auto-calibration :

7. L'algorithme de détecteur de Gaz :



8. Conclusion :

Dans ce chapitre, nous avons montré la structure de notre robot mobile avec toutes les dimensions, les composants et les programmes utilisés pour montrer les circuits et comment installer les composants, ainsi que les étapes nécessaires pour créer une application qui contrôle le robot mobile et le détail de montage de notre circuit étape par étape jusqu'au l'état final de notre circuit.

Enfin, nous avons un robot mobile capable de mobilité et de détecter la présence de monoxydes de carbone et le lancement d'une alarme.

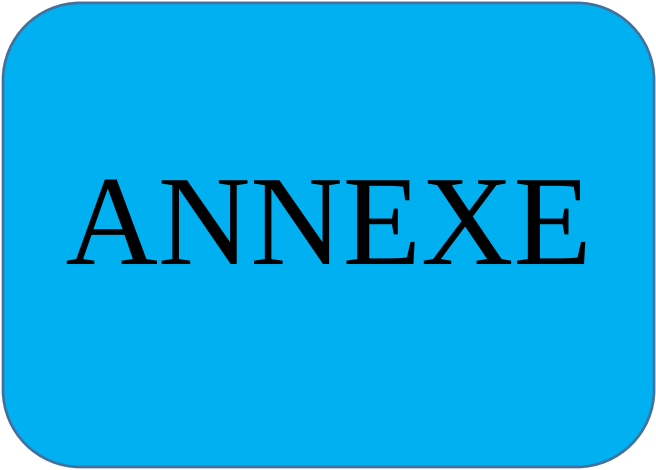
Conclusion générale

Conclusion générale

Nous avons donc pu atteint l'objectif de notre projet qui est l'utilisation de logiciel et carte électronique Arduino pour manipuler un robot mobile a deux roues. Nous avons étudié tout d'abord les principales caractéristiques d'un robot et en particulier le robot mobile. Ensuite On a abordé des différents composants utilisés pour construire ce robot: moteurs vers les drivers jusqu' Arduino et microcontrôleurs pour mieux comprendre son fonctionnement et pouvoir ensuite générer les signaux de commandes qui seront envoyés à partir de la carte électronique. Ce projet pourra plutôt servir de base pour commander d'autres systèmes ou d'autres robots qu'ils ont des différents taches et applications selon le besoin.

Une telle réalisation n'est pas dénuée de difficultés. Il est à noter que nous nous sommes confrontés à plusieurs problèmes surtout la difficulté à obtenir du matériel ainsi le prix élevé. Cependant, malgré ces difficultés, nous avons réalisé ce modeste travail.

Comme perspective nous souhaitons vivement que ce projet puisse servir comme élément de base pour d'autres études plus approfondies pour le faire intégrer sous des systèmes plus complexes. Nous espérons que nous allions le développer pour détecter et extinction les incendies, fournir des algorithmes qui donne au robot une autonomie via son programme à appréhender l'environnement physique à partir des données reçues des différents capteurs - infrarouges, ultrasons- disponibles sur le robot autonome.



ANNEXE

1. Le programme

```
////////////////////////////////////
// RemoteXY include library //
////////////////////////////////////

int buzzer = 12;

int smokeA0 = A5;

// Valeur de seuil. Vous pourriez avoir besoin de le changer.
int sensorThres = 400;

// RemoteXY sélectionner le mode de connexion et inclure la bibliothèque
#define REMOTEXY_MODE__SOFTSERIAL
#include <SoftwareSerial.h>
#include <RemoteXY.h>

// Paramètres de connexion RemoteXY
#define REMOTEXY_SERIAL_RX 2
#define REMOTEXY_SERIAL_TX 3
#define REMOTEXY_SERIAL_SPEED 9600

// Configuration de RemoteXY
#pragma pack (push, 1)
uint8_t RemoteXY_CONF [] =
    { 255,3,0,0,0,29,0,8,5,0,
      2,1,6,8,22,11,137,38,31,31,
      79,78,0,79,70,70,0,5,16,49,
      22,30,30,38,163,16 };

// Cette structure définit toutes les variables de votre interface de contrôle
struct {

    // variable d'entrée
    uint8_t switch_1; // =1 if switch ON and =0 if OFF
    int8_t joystick_1_x; // =-100..100 x-coordinate joystick position
    int8_t joystick_1_y ; // =-100..100 y-coordinate joystick position

// Autre variable
```

```
uint8_t connect_flag; // =1 if wire connected, else =0
} RemoteXY;
#pragma pack (pop)
////////////////////////////////////
//  END RemoteXY include  //
////////////////////////////////////
#define PIN_SWITCH_1 13
#define PIN_SWITCH_1 13

// Définir les broches de commande du moteur droit
#define right_motor_A 8
#define right_motor_B 9
#define right_motor_speed 11 // enable pin

// Définir les broches de commande du moteur gauche
#define left_motor_A 6
#define left_motor_B 7
#define left_motor_speed 10 // enable pin

// Définir deux tableaux avec une liste de broches pour chaque moteur
uint8_t RightMotor[3] = {right_motor_A, right_motor_B, right_motor_speed};
uint8_t LeftMotor[3] = {left_motor_A, left_motor_B, left_motor_speed};

// Contrôle de la vitesse des moteurs

void Wheel (uint8_t * motor, int v) // v = motor speed ,motor = pointer to an array of pins
{
    if (v>100) v=100;
    if (v<-100) v=-100;
    if (v>0) {
        digitalWrite(motor[0], HIGH);
        digitalWrite(motor[1], LOW);
        analogWrite(motor[2], v*2.55);
    }
    else if (v<0) {
        digitalWrite(motor[0], LOW);
```

```
    digitalWrite(motor[1], HIGH);
    analogWrite(motor[2], (-v)*2.55);
}

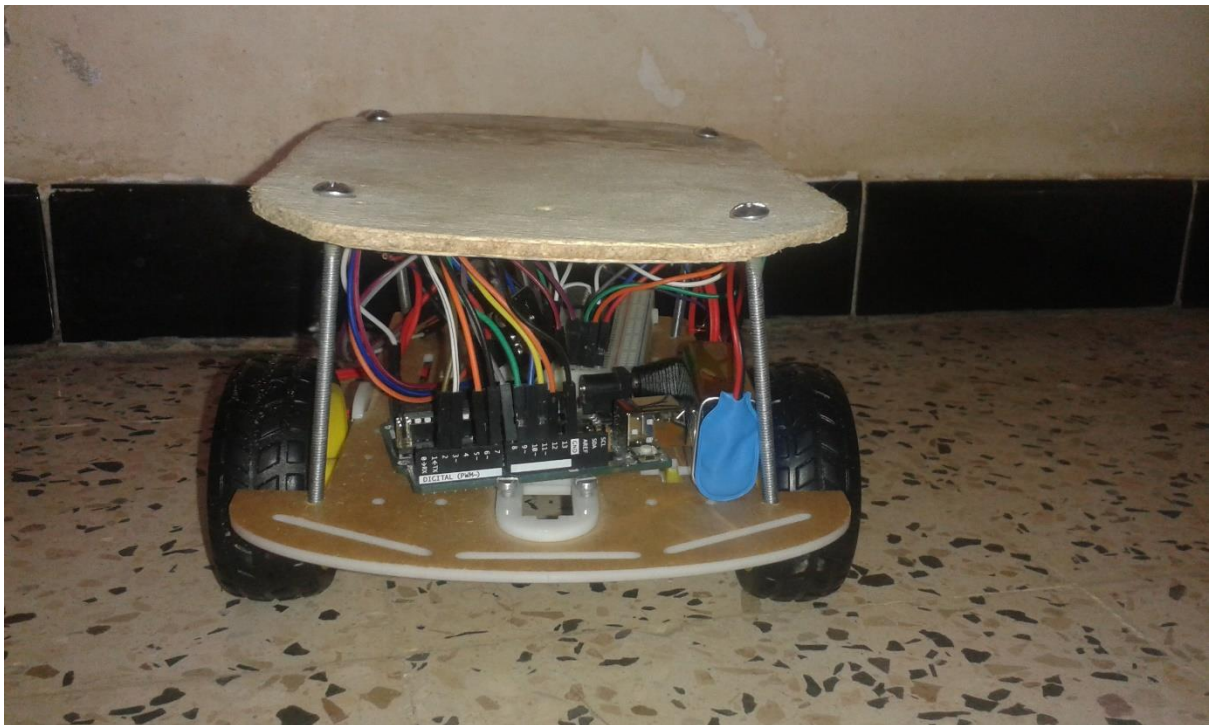
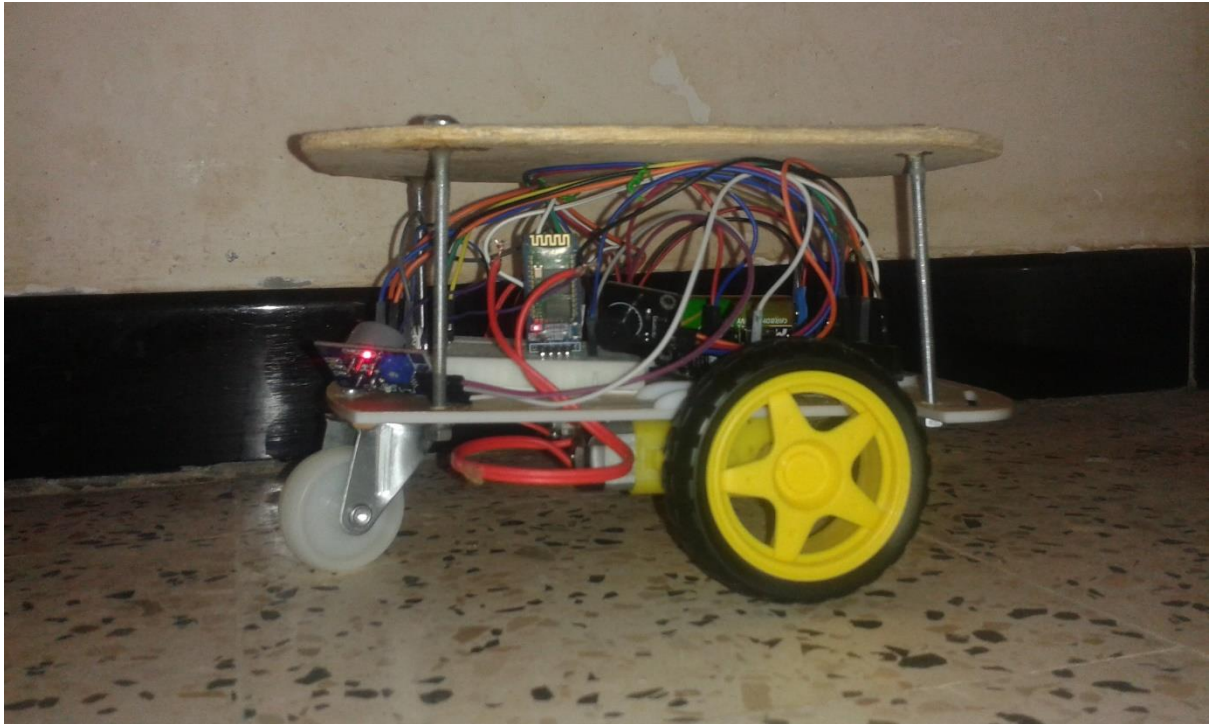
else {
    digitalWrite(motor[0], LOW);
    digitalWrite(motor[1], LOW);
    analogWrite(motor[2], 0);
}
}

void setup()
{
    RemoteXY_Init ();
    pinMode (PIN_SWITCH_1, OUTPUT);
    // TODO you setup code
    pinMode (right_motor_A , OUTPUT);
    pinMode (right_motor_B , OUTPUT);
    pinMode (left_motor_A , OUTPUT);
    pinMode (left_motor_B , OUTPUT);
    pinMode(buzzer, OUTPUT);
    pinMode(smokeA0, INPUT);
    Serial.begin(9600);
}

void loop()
{
    RemoteXY_Handler ();
    digitalWrite(PIN_SWITCH_1, (RemoteXY.switch_1==0)?LOW:HIGH);
    // TODO you loop code
    // utiliser la structure RemoteXY pour le transfert de données
    Wheel (RightMotor, RemoteXY.joystick_1_y - RemoteXY.joystick_1_x);
```

```
Wheel (LeftMotor, RemoteXY.joystick_1_y + RemoteXY.joystick_1_x);  
    int analogSensor = analogRead(smokeA0);  
Serial.print("Pin A0: ");  
Serial.println(analogSensor);  
// Vérifie s'il a atteint la valeur de seuil  
if (analogSensor > sensorThres)  
{  
    tone(buzzer, 1000, 200);  
}  
else  
{  
    noTone(buzzer);  
}  
delay(100);  
}
```

Photos de robot mobile réalisé



Bibliographie

Bibliographie

- [1] **Laetitia Matignon**, "Introduction à la robotique Licence 1ère année". Université de Caen France, 2011/2012.
- [2] **TAKHI. Hocine, ATTACHI. Redouane Cherif**, "Conception et réalisation d'un robot mobile à base d'Arduino", mémoire de master option : Instrumentation .université Ammar Telidji Laghouat ,2014 /2015.
- [3] **David FILLIAT**, "Robotique Mobile".École Nationale Supérieure de Techniques Avancées Paris Tech, 17 octobre 2016.
- [4] **A. ALLOUI, A. HAJ Brahim**. "Proposition d'une solution multi-agent pour la commande et la coopération multi -robot mobile". Mémoire d'ingénieur d'état en automatique. Université Biskra, Juin 2007.
- [5] **M. GHAOUI**. " Planification d'un mouvement pour un robot mobile " ,Thèse de magister. Université de Batna, année 1997.
- [6] **SLIMANE. Noureddine** "système de localisation pour robots mobiles", Pour l'obtention du grade de docteur d'état spécialité : contrôle, l'université de Batna, 23 / 11 / 2005.
- [7] **Nicolas. Morette**, "Contribution à la navigation de robots mobiles : approche par modèle direct et commande prédictive". Université d'Orléans, 2009.
- [8] **BALI. Chaher eddine, ABAIDI. Hakim**, "RÉALISATION D'UN ROBOT MOBILE AVEC ÉVITEMENT D'OBSTACLE ET TRAJECTOIRE PROGRAMMÉ", Mémoire de Fin d'études En vue de l'obtention du diplôme : MASTER. Université Biskra ,2011-2012
- [9] **Rehaïem.Abdelmoumen**, "Conception, Réalisation et Command D'un Robot Mobile" mémoire MASTER ACADEMIQUE Spécialité : Contrôle Industriel .université Ouargla, 2015-2016.
- [10] **Bernard. BAYLE**, "Robotique mobile Télécom Physique Strasbourg Université de Strasbourg".
- [11] **Bruno. Siciliano, Lorenzo. Sciavicco, Luigi .Villani, Giuseppe. Oriolo**, "Robotics: Modelling, Planning and Control.2009.
- [12] <http://www.cite-sciences.fr/fr/ressources/bibliotheque-en-ligne/dossiers-documentaires/des-robots-pour-tout-des-robots-pour-tous/applications/>

Bibliographie

- [13] **Mr. GUERAICHI Mohamed**, "Etude et programmation d'une carte ARDUINO UNO pour le comptage d'entrées / sorties d'un parking.
- [14] X.HINAULT. www.mon-club-elec.fr.
- [15] **LECHALUPÉ Julien**, "Cours d'initiation à Arduino ", Université Paul Sabatier, Mai 2014.
- [16] **Brechet .Thomas, Tribino.Julien**, "Le Pendule inversé et son application en robotique", 2008.
- [17] **Frédéric Bouquet, Julien Bobroff**, "MICROCONTROLEUR ARDUINO" Projets de physique statistique Magistère de physique fondamentale, Université Paris Sud, 2015.
- [18] **Astalaseven.Eskimon** "Arduino pour bien commencer en électronique et en programmation". Licence Créatives Commons BY-NC-SA 2.0, Dernière mise à jour le 31/03/2012.
- [19] "À la découverte de l'Arduino Uno " 9 juin 2015.
- [20] Datasheet HC-06 Bluetooth module.
- [21] <https://snootlab.com/sparkfun/668-capteur-de-monoxyde-de-carbone-mq-7-fr.html>.
- [22] Datasheet 5v Buzzer Module.
- [23] **A Khemissat**, "Plateforme de prototypage rapide pour la robotique mobile", Application à un robot Auto-balancé" mémoire de Master, UMBB, Boumerdes, 2016.
- [24] **Bachelor's Thesis** "Degree Program in Information Technology", 2013.
- [25] <http://blewando.dlinkddns.com/elv/Promo2016/th4/pag1.html>
- [26] **Yves .MERGY** "PRISE EN MAIN DE FRITZING (3) CONCEVIR UN CIRCUIT IMPRIME (PCB)". university of Applied Science Potsdam, 2007-2010.
- [27] <http://remotexy.com/en/help/>
- [28] **Simon. Landrault, Hippolyte Weisslinger**, "Arduino : Premiers pas en informatique embarquée", livre. Édition du 19 juin 2014.
- [29] Classes de 2nde SI-CIT et de première SI. [arduino.odt](#)
- [30] **Jean-Jacques Wittezaele**, "La double contrainte: L'influence des paradoxes de Bateson en Sciences humaines".
- [31] **KRAMA Abdelbasset, GOUGUI Abdelmoumen**, "Etude et réalisation d'une carte de Contrôle par Arduino via le système Androïde" Mémoire MASTER ACADEMIQUE. Spécialité : Electrotechnique Industrielle. Université Ouargla, 2014/2015.
- [32] Henan Hanwei Electronics Co., Ltd. "TECHNICAL DATA MQ-7 GAS SENSOR".