

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUDE MAMMERI DE TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT D'ELECTRONIQUE

**Mémoire de Fin d'Etudes de  
MASTER ACADEMIQUE**

Domaine : **Sciences et Technologies**

Filière : **Génie Electrique**

Spécialité : **Télécommunication et Réseaux**

*Présenté par*

**Omar ICHEROUFENE**

Thème

**Conception et réalisation d'un Robot  
mobile télécommandé à base de la  
PCDUINO V3**

*Mémoire soutenu publiquement le 19/07/2016 :*

**Mr M. LAGHROUCHE**

Promoteur

**Mr M.NACHEF**

Co-promoteur

**Année universitaire 2015/2016**

# *Remerciements*



*Nous remercions d'abord le bon Dieu qui nous a donné le courage, la patience, la santé et la volonté d'arriver à la fin de ce travail.*

*Nous tenons à remercier notre promoteur Monsieur **M. LAGHROUCHE** pour son aide, sa disponibilité et les conseils qu'il n'a cessé de nous prodiguer pour l'aboutissement de ce modeste travail, qu'il trouve ici notre profonde gratitude et toute notre reconnaissance et notre respect. Nous tenant aussi à remercier Mr : **M. NACHEF** et **H. Achour** pour leurs précieux conseils*

*Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner ce travail et de l'enrichir par leurs propositions.*

*Enfin à toutes et tous ceux qui ont contribué au déroulement de ce travail. Que tous les enseignants ayant contribués et participés a notre formation trouvent ici notre profond respect.*

# *Dédicaces*



*Ce travail, et bien au-delà, je le dois à mes très chers parents qui m'ont fourni au quotidien un soutien et une confiance sans faille et de ce fait, je ne saurais exprimer ma gratitude seulement par des mots. Que dieu vous protège et vous garde pour nous.*

*A mes sœurs*

*A mes frères*

*A mes beaux frères*

*A mes nièces*

*Enfin a tout(e)s mes ami(e)s*

## Sommaire

Introduction générale :	1
Cahier des charges :	3

### Chapitre I : Généralité sur la robotique

I- 1 Introduction :	5
I- 2 Présentation du robot mobile :	6
I- 2 - 1 Classification des robots mobiles :	6
a – Classification selon degré d'autonomie :	6
a - 1 Robot télécommandé :	6
a - 2 Robot semi-autonome :	7
a - 3 Robot autonome :	7
b – Classification selon le domaine d'application :	7
c – Classification selon le type de locomotion :	8
c - 1 Les robot mobiles a roues :	8
c - 1 - 1 Robots unicycle :	8
c - 1 - 2 Robots tricycle :	9
c - 1 - 3 Robots mobiles omnidirectionnels :	9
c - 1 - 4 Robots voiture quatre roues :	10
c - 2 Les robot à chenilles :	10
c -3 les robot marcheurs :	10
d – Les dates marquantes de la robotique :	11
I-3 Avantage et inconvenient de quelques types de robots :	11
I-4 Conclusion :	12

### Chapitre II : Conception du système

II – 1 Introduction :	13
II – 2 Structure du système :	13
II - 3 Description de la partie matérielle :	15
II – 3 – 1 la carte pcDuino V3 :	15
a - Présentation de la carte pcDuino V3:	15

b - Les caractéristiques de l'ARM A20 Dual-Core :	16
c - Les broches numériques de la pcDuino V3 :	18
d - Programmation de la carte pcDuino :	19
II - 3 - 2 Description de la carte Arduino Uno :	19
a - Caractéristiques de la carte Arduino Uno :	20
II - 3 - 3 Description de la carte Arduino Nano :	20
a - Caractéristiques de la carte Arduino Nano :	21
II - 3 - 4 Le Châssis Rover :	22
a - Caractéristiques de Rover 5 :	22
II - 3 - 5 Shields DK electronics :	23
a - Caractéristiques de shield DK électronique :	24
II - 3 - 6 Moteur à courant continu :	24
II - 3 - 7 Module joystick :	25
a - Caractéristiques techniques :	25
b - Principe de fonctionnement de la joystick :	26
II - 3 - 8 Module XBee :	26
a - Les différents Versions du XBee :	27
b - Caractéristique XBee Version Normale et Pro :	27
c - Modes de XBee :	28
II - 3 - 9 Shield XBee pour Arduino :	28
II - 3 - 10 description des différent capteur utilisée :	29
a- Capteur infrarouge :	29
b - Capteur de gaz MQ-2 :	29
b-1Caractéristique de capteur de gaz :	30
c- Capteur de température (LM 335) :	30
d- Capteur de Flammes :	31
d-1Caractéristique techniques :	32
II-3-11 Webcam PC810 :	32
a- Caractéristique de JVC PC810 :	32
II-3-12 Alimentation MB 102 :	33
II-4 description de la partie logicielle :	34
II-4-1 VNC (Virtual Network Computing – réseau virtuel de calcul) :	34
II-4-2 IDE Arduino :	34
II-4-3 IDE PyDuino :	36
II-4-4 Fritzing :	37
II - 5 Conclusion :	37

## Chapitre III : Réalisation pratique

III-1 Introduction :	38
III-2 Réalisation matérielle et configuration logicielle :	38
III-2-1 Réalisation logicielle :	38
a- Mise en marche de la pcDuino V3 :	38
b- Liaison entre pcDuino et le poste fixe avec contrôle à distance :	39
c- Acquisition des images à travers une caméra :	41
c-1 Installation 'mjpeg streamer' sur la pcDuino :	41
III-2-2 Réalisation Matérielle :	43
a- Structure globale de notre système :	43
b- Capteur de gaz MQ-2 :	44
c- Capteur de flamme :	44
d- Capteur de température :	45
e- Schéma globale des différents capteurs avec la pcDuino :	46
f- Capteur infrarouge :	46
g- Contrôle global de robot :	47
h- Organigramme générale du système :	49
i- Vue globale du robot :	50
III-3 Conclusion :	50
Conclusion générale :	51
Annexe	
Bibliographie	

## Liste des figures :

Figure 1 : Schéma bloc détaillés de cahier des charges .....	4
Figure 2 : Robot unicycle .....	8
Figure 3 : Robot voiture .....	9
Figure 4 : Robot omnidirectionnel .....	10
Figure 5 : Robot de type omnidirectionnel .....	10
Figure 6 : Structure globale du Système .....	14
Figure 7 : Description de la carte PcDuino V3 .....	15
Figure 8 : Carte PcDuino V3 .....	18
Figure 9 : Carte Arduino Uno .....	20
Figure 10 : Carte Arduino Nano .....	21
Figure 11 : Châssis Rover 5 .....	22
Figure 12 : Shield DK électronique .....	23
Figure 13 : Moteur à courant continu .....	25
Figure 14 : Joystick .....	26
Figure 15 : Fonctionnement du joystick .....	26
Figure 16 : Module XBee normale (Antenne céramique) .....	27
Figure 17 : XBee version Pro (Antenne fouet) .....	27
Figure 18 : Shield XBee Arduino .....	28
Figure 19 : Capteur infrarouge .....	29
Figure 20 : Capteur de gaz MQ-2 .....	29
Figure 21 : Capteur de température .....	30
Figure 22 : Capteur de Flammes .....	31
Figure 23 : Webcam PC810 .....	32
Figure 24 : Alimentation MB 102 .....	33
Figure 25: Interface de serveur VNC .....	34
Figure 26 : IDE Arduino .....	35
Figure 27 : IDE pyDuino.....	36
Figure 28 : Interface de Fritzing .....	37
Figure 29 : matériel nécessaire pour la mise en marche de la pcDuino.....	38
Figure 30 : Mise en réseau de la pcDuino.....	39
Figure 31 : Utilisation VNC avec WIFI.....	40
Figure 32 : Procédure pour se connecter au mini-pc à partir du poste fixe.....	40
Figure 33 : le bureau de la pcDuino visible sur le poste fixe.....	41
Figure 34 : pcDuino avec Webcam.....	41
Figure 35: Interface de site embarque 'ROVER 2016'.....	42

Figure 36 : Schéma global du système.....	43
Figure 37 : schéma de branchement électrique MQ-2.....	44
Figure 38 : schéma de branchement électrique de capteur de flamme.....	45
Figure 39 : schéma de branchement électrique de capteur de température.....	45
Figure 40 : schéma globale des différents capteurs avec la pcDuino.....	46
Figure 41 : schéma de branchement électrique de capteur infrarouge.....	46
Figure 42 : schéma électrique permettant le contrôle des moteurs avec DK electronics.....	47
Figure 43 : Sens de déplacement de notre châssis.....	48
Figure 44 : Organigramme du Système .....	49
Figure 45 : Vue globale du robot.....	50

## Liste des tableaux

Tableau 1 : Domaine d'application des robots .....	8
Tableau 2 : Avantages et inconvénients des robots mobiles .....	11
Tableau 3 : caractéristiques de la PcDuinoV3 .....	17
Tableau 4 : caractéristique Arduino Uno .....	20
Tableau 5 : Caractéristiques Arduino Nano .....	21
Tableau 6 : Caractéristique de Rover 5 .....	23
Tableau 7 : Caractéristique XBee Version Normale et Pro .....	27
Tableau 8 : Fiche technique de MQ-2 .....	30
Tableau 9 : Caractéristique de JVC PC810 .....	32

## Glossair

**PC** : Personal Computers.

**HDMI** : High Definition Multimedia Interface.

**HDCP** : High-Bandwidth Digital Content Protection : Protection des contenus numériques haute définition.

**LVDS** : Low Voltage Differential Signaling : signalisation différentielle à basse tension.

**LCD** : Liquid Crystal Display : écran à cristaux liquides.

**CSI** : Camera Serial Interface.

**USB** : Universal Serial Bus : Bus universel en série.

**USB OTG** : Universal Serial Bus On-The-Go.

**VNC** : Virtual Network Computing : réseau virtuel de calcul.

**API** : Application Programming Interface.

**SBC** : Single Board Computer : Ordinateur à carte unique.

**GPU** : Graphics Processing Unit.

**HD** : Haute Définition.

**IDE** : Integrated Development Environment : environnement de développement libre.

**IR** : Infrarouge.

**SATA** : Serial Advanced Technology Attachment.

**UART** : Universal Asynchronous Receiver Transmitter : émetteur-récepteur asynchrone universel.

**ADC** : Analog to Digital Converter : Convertisseur analogique-numérique.

**PWM** : Pulse Width Modulation : modulation de largeur d'impulsions.

**GPIO** : General Purpose Input/Output : littéralement Entrée/Sortie pour un Usage Général.

**I<sup>2</sup>C** : Inter-Integrated Circuit.

**ISM** : Industrie Science et Médical.

**SPI** : Serial Peripheral Interface.

**E/S** : Entrées/Sorties.

**SCL** : Serial Clock.

**SDA** : Serial Daata.

**Tx-Rx** : Transmission et Réception.

**ICSP** : In Circuit Serial Programming.

**DRAM** : Dynamic Random Access Memory : une type de mémoire vive.

**SRAM** : Static Random Access Memory.

**EEPROM** : Electrically-Erasable Programmable Read-Only Memory : mémoire morte effaçable électriquement et programmable.

**SD** : Secure Digital.

**DC** : Direct Current : courant continu.

**MCC** : Moteur à Courant Continue.

**IN** : INPUT (Entrées).

**OUT** :OUTPUT (Sorties).

**GND** : Ground.

**VCC** : Alimentation Tension Continue.

**S-X, S-Y** : système de coordonnées cartésiennes

**ISM** : Industrie Science et Médical.

**LED** : Light-Emitting Diode : (DEL) diode électroluminescente.

**ON /OFF** : commande marche-arrêt.

**LAAS** : Laboratoire d'Analyse et d'Architecture des Systèmes.

**CNRS** : Centre National de la Recherche Scientifique.

**HTTP**: HyperText Transfer Protocol.

## **Résumé**

Un robot mobile est un système mécanique, électronique et informatique agissant physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné. Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditions de fonctionnement. Elle est dotée de fonctions de perception, de décision et d'action. Ainsi, le robot devrait être capable d'effectuer des tâches diverses, de plusieurs manières, et accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues.

Ce projet concerne la conception, la réalisation et la commande d'un robot mobile à chenille à l'aide des cartes électroniques " Arduino et pcDuino".

## **Mots clés**

Robot mobile, pcDuino, pyDuino, Arduino Uno, Arduino Nano, serveur streamer, VNC, Xbee, Caméra embarquée.

# Introduction générale

## Introduction générale

Depuis quelques années, on assiste à des progrès scientifiques et technologiques dans les différents domaines d'application de la robotique : mécanique, électronique, automatique et informatique. La robotique est essentiellement cantonnée au domaine manufacturier pour réaliser des tâches complexes, pénibles et/ou répétitives ou pour opérer dans des milieux inaccessibles à l'homme (exploration planétaire, nucléaire...). Citons l'exemple de l'accident de Tchernobyl, le 26 avril 1986 dans la centrale nucléaire Lénine, où les membres des équipes d'intervention, retrouvés face à un problème rapide et d'extrême urgence, se sont intervenus sans prendre de précautions liées à leur santé.

Une solution technique à ce problème serait d'évaluer au préalable les dégâts à l'aide d'un système télécommandé afin, par exemple, de débayer des voies d'accès au site d'intervention ou de trouver le chemin le plus court avec le moins de risques, mais la meilleure solution est de résoudre le problème sans l'intervention sur place de l'homme. Pour cela, il faut mettre dans le système une caméra pour visualiser la zone dans laquelle il faut intervenir et placer les différents capteurs pour avoir l'idée sur l'environnement, et pour une utilisation large et efficace, un bras robotisé pour manipuler des objets sera agréable. Cette solution présente d'ailleurs le contexte dans lequel s'inscrit notre travail.

Notre projet consiste à concevoir et réaliser un robot télécommandé qui répond au cahier des charges. Notre premier objectif est de donner au robot la capacité de se mouvoir dans son environnement. Pour cela, on a choisi l'utilisation de châssis 'rover 5' à chenilles. Le robot (châssis + bras) sera téléguidé avec des joysticks en utilisant une transmission sans fil basée sur le module XBee.

Pour pouvoir faire circuler le robot facilement dans une zone donnée, une caméra embarquée sur le robot est utilisé pour faire l'acquisition d'images. Cette caméra est implémentée sur la carte pcDuino qui transmet l'image via le wifi vers l'ordinateur où va se faire la visualisation. La carte pcDuino nous permet aussi d'obtenir plus d'informations sur l'environnement où se trouve le système grâce à des capteurs.

Ce mémoire est structuré en trois chapitres dont l'agencement reflète la démarche utilisée et les étapes de conception.

## Introduction générale

Le premier chapitre est consacré à la présentation du cadre général de la robotique. En accord avec les objectifs de mémoire, le deuxième chapitre est consacré à la description des différents logiciels et matériels utilisés dans notre réalisation.

Le troisième chapitre est réservé exclusivement à la description de la réalisation pratique.

Et on termine avec une conclusion générale

# Cahier des charges

## **Cahier des charges**

Nous avons fixé notre propre cahier des charges en proposant ce sujet. L'idée était donc de créer notre propre robot mobile idéalement commandé à distance. Nous avons donc pu distinguer les différentes fonctions de notre cahier des charges comme suit :

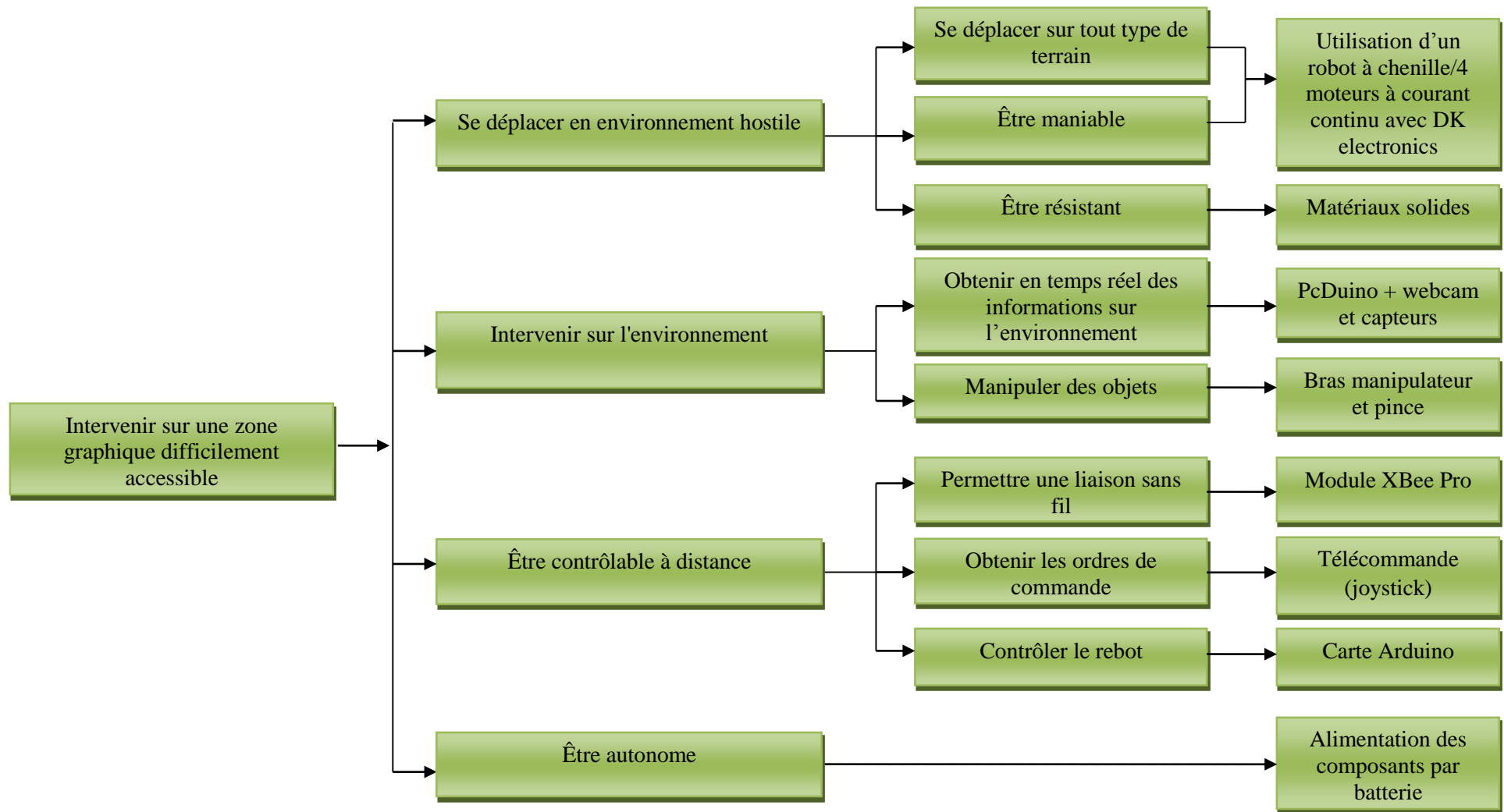
Fonction principale : Permettre une intervention de l'homme sur une zone géographique difficilement accessible.

Fonction secondaire 1 : Être contrôlable et pilotable à distance.

Fonction secondaire 2 : Acquérir une vidéo et des informations.

Fonction secondaire 3 : Disposer d'une autonomie énergétique suffisante.

Ce diagramme se lit de gauche à droite en posant la question "Comment ?" et de droite à gauche en posant la question "Pourquoi ?".



**Figure 1** : Schéma bloc détaillés de cahier des charges

( Nous allons donc voir par la suite, le processus de développement de toutes ces fonctionnalités. )

# Chapitre I : Généralité sur les robots

## I-1 Introduction

Le terme « robot » a été introduit en 1920 par l'écrivain tchèque Karel Èapek. Ce terme, provenant du tchèque *robota*, « travail forcé », désigne à l'origine une machine androïde capable de remplacer l'homme dans toutes ses tâches. Après, pendant les années 70 et 80, un nouveau domaine émergea dans l'automatisme : la robotique. Ce dernier a connu un développement considérable profitant des avancées technologiques des autres domaines tels que la microélectronique, les microprocesseurs et les capteurs. [1]

Depuis une vingtaine d'années, un effort particulier a été fait dans les domaines de la recherche et de l'industrie pour construire des robots mobiles évoluant avec un minimum d'intervention humain. Une première génération des robots a consisté en des machines capables d'évoluer dans des environnements parfaitement connus : celles-ci réalisent des missions planifiées à partir d'une modélisation complète de l'environnement (laboratoire) ou se contentent de suivre une trajectoire par un mécanisme de filo-guidage (robot de manutention). Le point commun de ces robots est qu'ils évoluent dans un environnement qui leur est totalement dédié. [2]

Cependant, lorsque l'environnement devient plus complexe (c.-à-d. : partiellement connu, dynamique, etc.), il apparaît indispensable que le robot soit doté de capacités décisionnelles aptes à le faire réagir aux aléas qui peuvent contrarier ses mouvements (pannes partielles, obstacles imprévus). Cela peut être le cas lorsque le robot mobile évolue dans des environnements hostiles à l'homme (milieu radioactif) ou trop éloignés (exploration spatiale).

Pour cela, le robot doit suivre le schéma correspondant au paradigme (Percevoir, Décider, Agir) au sein d'une architecture de contrôle. L'activité d'un tel robot se ramène aux tâches énoncées ci-après :

### - Percevoir :

Le robot doit acquérir des informations sur l'environnement dans lequel il évolue par l'intermédiaire de capteurs ; ces informations permettent de mettre à jour un modèle de l'environnement (architecture hiérarchique) ou peuvent être directement utilisées comme entrées de comportements de bas niveau (architectures purement réactives).

## **- Décider :**

Le robot doit définir des séquences d'actions résultant d'un raisonnement appliqué sur un modèle de l'environnement (architecture hiérarchique) ou répondant de manière réflexe à des stimuli étroitement liés aux capteurs (architectures purement réactives).

**- Agir :** Le robot doit enfin exécuter les séquences d'actions en envoyant des consignes aux actionneurs par l'intermédiaire des asservissements.

Au sein de la partie décision, il est nécessaire de contrôler l'exécution de ces actions afin que le robot s'adapte rapidement à des événements imprévus. [3]

## **I-2 Présentation du robot mobile**

Un robot mobile est un système mécanique, électronique et informatique agissant physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné, ce robot doté de moyens de locomotion qui lui permettent de se déplacer suivant son degré d'autonomie, il peut être doté de moyens de perception et de raisonnement. [4]

### **I-2-1 Classification des robots mobiles**

La classification des robots mobiles se fait suivant plusieurs critères (degré d'autonomie, système de locomotion, énergie utilisée...). La classification la plus intéressante et la plus utilisée est selon leur degré d'autonomie.

#### **a- Classification selon le degré d'autonomie**

C'est la classification la plus intéressante. Un robot autonome est un système doté de capacités décisionnelles et de moyens d'acquisition et de traitements de données lui permettant d'accomplir, sous contrôle humain réduit ou voire même absent, un certain nombre de tâches, dans un environnement inconnu. Selon ce critère, on peut classer les robots mobiles selon les catégories suivantes : [5]

##### **a-1 Robot télécommandé**

Ce sont des robots commandés par un opérateur (machine ou être humain), qui leurs dicte chaque tâche élémentaire à faire (avancer, reculer, tourner à droite, etc.)

## a-2 Robot semi-autonome

Ce type de robots effectuent un certain nombre de tâche par eux-mêmes d'une façon complètement autonome mais peuvent être interrompus pour recevoir des commandes de contrôle par un opérateur.

## a-3 Robot autonome

On considère qu'un robot est autonome s'il est capable d'adapter son comportement à l'environnement.

L'autonomie est la capacité propre d'un système sans équipage, à capter, percevoir, analyser, communiquer, planifier, prendre des décisions et agir afin d'atteindre les buts qui lui ont été assignés par un opérateur humain à l'aide d'une interface homme /machine dédiée.

## b- Classification selon le domaine d'application

Les domaines d'application des robots est vaste, à titre d'exemple :

Domaines	Applications
Industrie nucléaire	- surveillance de sites - manipulation de matériaux radioactifs - démantèlement de centrales
Sécurité civile	- neutralisation d'activité terroriste - déminage - pose d'explosif - surveillance de munitions
Chimique	- surveillance de sites - manipulation de matériaux toxiques
Mine	- assistance d'urgence
Agricole	- cueillette de fruits - traite, moisson, traitement des vignes.
Nettoyage	- coque de navire - nettoyage industriel
Espace	- exploration

Industrie	- convoyage - surveillance
Militaire	- surveillance - pose d'explosif - manipulation de munition

**Tableau 1** : domaine d'application des robots [4]

## c- Classification selon le type de locomotion

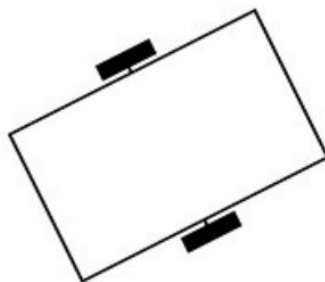
### c-1 Les robots mobiles à roues

La mobilité par roues est la structure mécanique la plus utilisée. Ce type de robot assure un déplacement aisé, 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. [4]

#### c-1-1 Robots unicycle

Un robot de type unicycle est actionné soit par une seule roue ou deux roues indépendantes. On utilise des capteurs d'attitude et de déplacement pour assurer sa stabilité. Son centre de rotation est situé sur l'axe reliant les deux roues motrices.

Un robot unicycle est un robot qui bouge dans un plan 2D ayant une certaine vitesse allant d'avant, mais zéro mouvement latéral instantané, car les robots unicycle sont des systèmes non-holonome. Il est impossible de le déplacer dans une direction perpendiculaire aux roues de locomotion. [4]



**Figure 2** : unicycle

## c-1-2 Robots 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. [4]

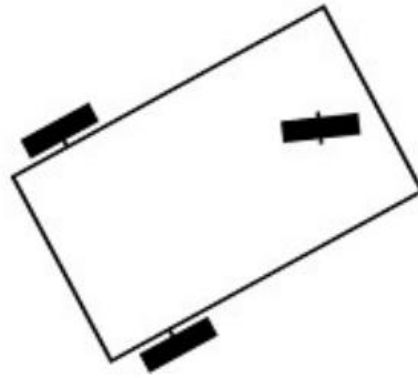


Figure 3 : robot voiture

## c-1-3 Robots mobiles omnidirectionnels

Un robot mobile est dit omnidirectionnel si l'on peut agir indépendamment sur les vitesses : vitesse de translation selon les axes  $x$  et  $y$  et vitesse de rotation autour de  $z$ . Il peut se déplacer librement dans toutes les directions. Il est généralement constitué de trois roues décentrées orientables placées en triangle équilatéral.

L'énorme avantage du robot omnidirectionnel est qu'il est holonome puisqu'il peut se déplacer dans toutes les directions. Mais ceci se fait au dépend d'une complexité mécanique bien plus grande. [4]

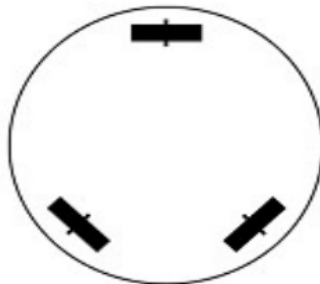


Figure 4 : robot omnidirectionnel

## c-1-4 Robot voiture quatre roues

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. 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. [4]

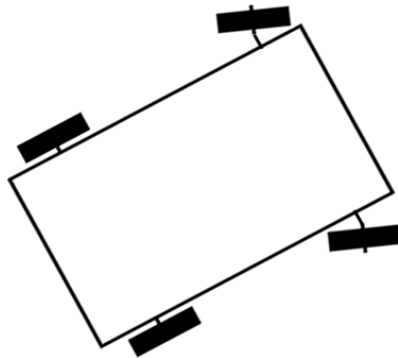


Figure 5 : Robot de type omnidirectionnel

## c-2 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. [4]

## c-3 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.

## d- Les dates marquantes de la robotique

1947 : premier manipulateur électrique télé-opéré.

1954 : premier robot programmable.

1961 : apparition d'un robot sur une chaîne de montage de General Motors, qui a fait le premier robot avec un contrôle en effort.

1963 : utilisation de la vision pour commander un robot.

1973 : premier robot mobile à roues. [6]

## I-3 Avantage et inconvénient de quelques types de robots

Type de robot	Avantages	Inconvénients
Robot unicycle	-stable -rotation sur soi-même -complexité mécanique faible	-non holonome
Robot tricycle	-complexité mécanique modérée	-non holonome -peu stable -pas de rotation sur soi-même
Robot voiture	-complexité mécanique modérée -stable	-non holonome
Robot omnidirectionnel	-holonome -stable Rotation sur soi-même	-complexité mécanique importante

Tableau 2 : Avantages et inconvénients des robots mobiles [7]

## I-4 Conclusion

Dans ce chapitre, nous avons pu définir certaines notions fondamentales liées à la robotique en général.

Un robot est en fait une machine programmable qui imite des actions d'une créature intelligente. L'homme a tenté de concevoir des machines qui puissent nous remplacer sous formes de robots, plus le temps avance plus la robotique évolue.

La robotique se développe actuellement en deux grandes branches :

- La robotique manufacturière : utilisée comme un outil de la productique, destinée à la production de biens. Ce domaine de la robotique regroupe tous les robots industriels.
- La robotique non-manufacturière : utilisée comme outil pour réaliser des tâches difficiles ou impossibles pour l'humain seul. Ce domaine regroupe essentiellement les robots de télémanipulation, les robots d'exploration, les robots mobiles et les simulateurs.

# Chapitre II :

# Conception

# du système

## II-1 Introduction

L'objectif de ce chapitre est de mettre l'accent sur la description des différentes parties utilisées dans notre projet.

## II-2 Structure du système

Le système conçu comporte un bloc principal constitué d'une carte pcDuinoV3 pour l'acquisition d'informations sur l'environnement de robot ainsi qu'une carte Arduino Uno pour la commande des actionneurs (moteur et servomoteur).

Le schéma de la figure 6 illustre la structure globale de notre système il est composée essentiellement :

- D'une carte de contrôle de type pcDuino.
- D'une carte de commande des actionneurs de type Arduino Uno
- D'une carte de commande distance à base d'une Arduino Nano

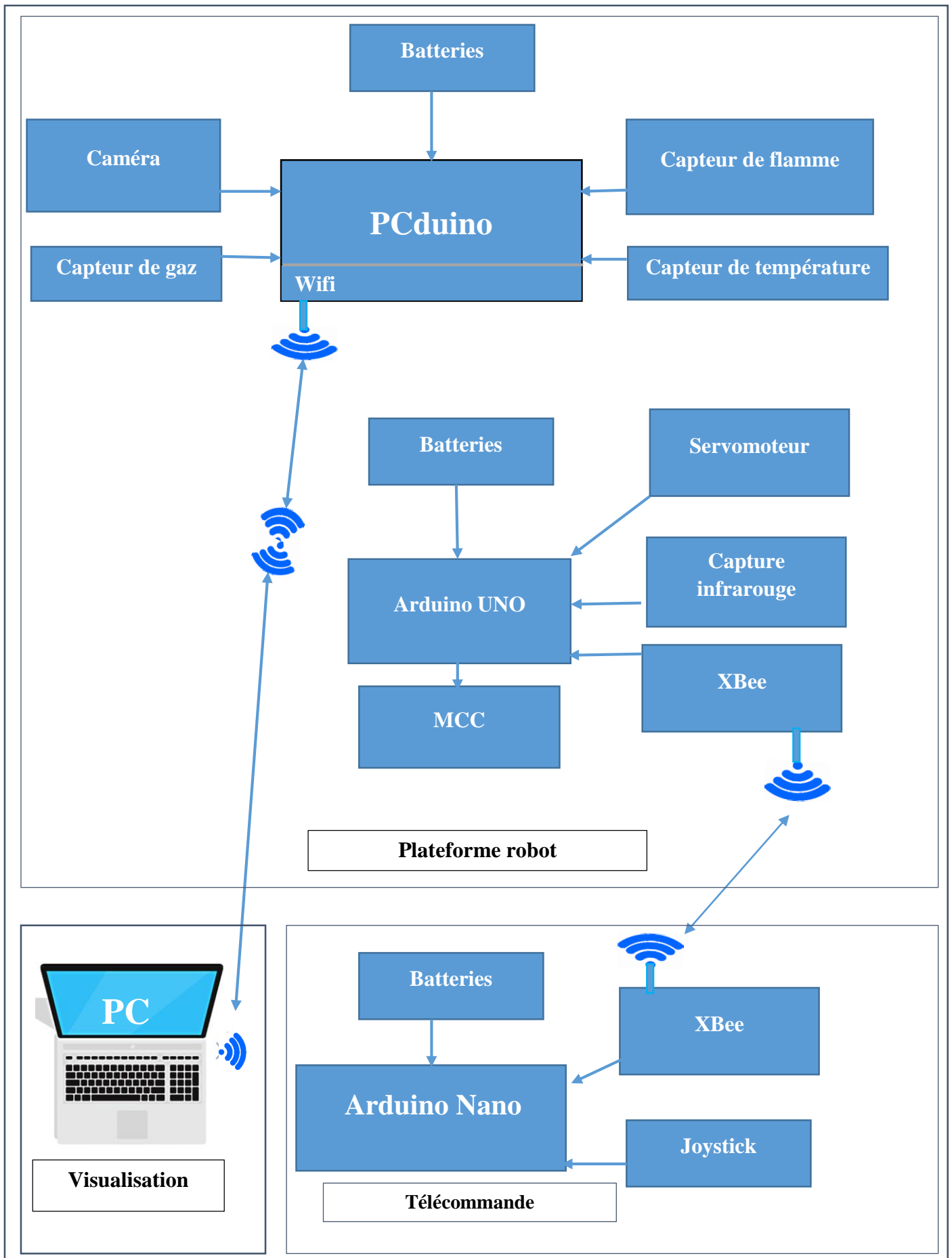


Figure 6 : Structure globale du système

## II-3 Description de la partie matérielle

### II-3-1 La carte pcDuino V3

#### a- Présentation de la carte pcDuino

PcDuino V3 (figure 7) est un mini PC à hautes performances, équipé d'un module wifi et supportant les systèmes d'exploitation tels que Linux Ubuntu et Android. PcDuino possède une interface vidéo HDMI pour la sortie de son écran de bureau graphique. Il pourrait supporter le multi-format 1080p 60fps décodeur vidéo ou 1080p 30fps et MPEG4 encodeur vidéo avec son moteur de traitement vidéo matériel intégré.

Ce mini PC dispose d'un lecteur de carte mémoire microSD et peut être également démarré à partir de cette dernière (par exemple Ubuntu 12.04). En général, le dispositif va démarrer d'abord à partir de la mémoire flash. Avec l'interface LVDS existante, les moniteurs LCD peuvent être exploités directement. En outre, le PcDuinoV3 dispose d'un port CSI pour se connecter à une caméra CSI, comme une option supplémentaire à côté des caméras USB.

Une autre caractéristique intéressante de PcDuinoV3 est l'interface USB OTG intégrée. Il fournit une option pour être accessible à partir d'un PC utilisant la plateforme Windows connecté avec une interface réseau USB via le logiciel de bureau à distance VNC installé sur le PcDuinoV3. Ainsi, le PcDuinoV3 peut être aussi utilisé sans écran, ce qui est une fonction très utile pour certaines applications, par exemple, les tâches de contrôle. Il est très approprié pour les applications professionnelles et industrielles.

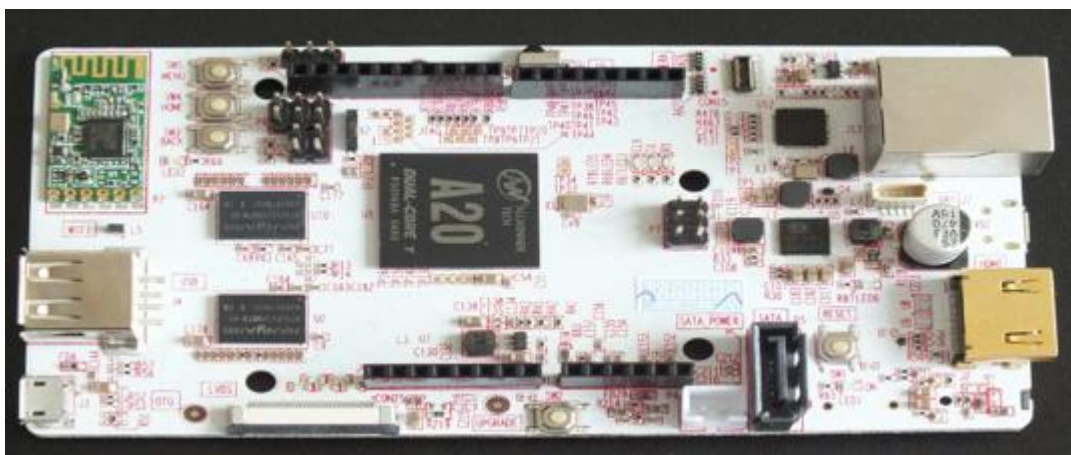


Figure 7 : Description de la carte PcDuino V3

Cette carte a été conçue pour faciliter le développement des projets pour la communauté open source et peut utiliser la plupart des shields (module) compatibles avec Arduino 3.3Volt. Une API a été développée et permet aux utilisateurs de la PcDuino d'utiliser le langage de programmation Arduino. PcDuinoV3 peut être utilisé pour enseigner Python, C et pour d'autres fonctions plus intéressantes.

Le PcDuinoV3 est le génie universel parmi les SBC Le but des développeurs était de créer un dispositif de matériel libre et puissant qui peut être largement utilisé dans diverses applications. Tous ces objectifs sont atteints avec la PcDuinoV3. [8]

### **b- Les caractéristiques de 'ARM A20 Dual-Core'**

Le processeur Allwinner-A20 en conjonction avec le Mali-400 GPU offre assez de puissance pour gérer le multimédia (Full HD) ou des applications de bureau en douceur et rapidement. Ainsi, le PcDuinoV3 peut accepter les mêmes tâches comme un PC classique sous Linux ou Android et peut être utilisé comme un media center avec télécommande infrarouge.

Il convient de noter que :

- Linux Ubuntu 12.04 ;
- XBMC Media Center ;
- Scratch ;
- Arduino IDE ;

Sont déjà installés sur la mémoire flash intégrée de 4 Go.

**Voici un tableau qui explique les différentes caractéristiques de PcDuinoV3 :**

Article	PcDuino3
CPU	AllWinner A20 SoC, 1GHz ARM Cortex A7 Dual Core
GPU	OpenGL ES2.0, OpenVG 1.1, Mali 400 Dual Core
Mémoire	4 Go de mémoire Flash, carte micro SD jusqu'à 32 Go, DRAM 1 GB
Sortie vidéo	HDMI avec support HDCP (Protection des contenus numériques haute définition)
Open Source	Ubuntu 12.04 et Android 4.2
Interface d'extension Arduino	même que Arduino Uno
Interface réseau	WiFi intégré Ethernet 10M / 100Mbps
Sortie Audio	<ul style="list-style-type: none"> <li>• 3.5mm analog audio interface</li> <li>• I2S stereo digital audio interface</li> </ul>
LCD	LVDS
IR	IR récepteur
SATA	SATA host socket
Caméra	CSI
Batterie	Interface Li-Poly Batterie
USB	1 x USB host, 1xUSB OTG
Alimentation	5V, 2A
Taille globale	121mm x 65mm
Langage	<ul style="list-style-type: none"> <li>• English</li> </ul>
API	<ul style="list-style-type: none"> <li>• Toutes les broches de blindage Arduino sont accessibles avec l'API fournie</li> <li>• Il se compose d'API pour accéder à des interfaces suivantes: <ul style="list-style-type: none"> <li>○ UART</li> <li>○ ADC</li> <li>○ PWM</li> <li>○ GPIO</li> <li>○ I2C</li> <li>○ SPI</li> </ul> </li> </ul>
Programmation avec les langages	<ul style="list-style-type: none"> <li>• C, C++ with GNU tool chain</li> <li>• Java with standard Android SDK</li> <li>• Python</li> </ul>

**Tableau 3 :** caractéristiques de PcDuinoV3 [8]

## c- Les broches numériques de la PcduinoV3

Le brochage de la pcduino est semblable à celui de l'Arduino sauf pour :

- Les E/S 14 à 17 qui sont sur bornier séparé
- SCL(serial clock) et SDA(serial data) (liaison série I<sup>2</sup>C ) qui ne sont pas entrées sur A4 et A5 mais séparées (peuvent être mises sur A4/A5 par soudure d'un pont, mais on perd 2 broches analogiques dans ce cas)

Voir la figure 8

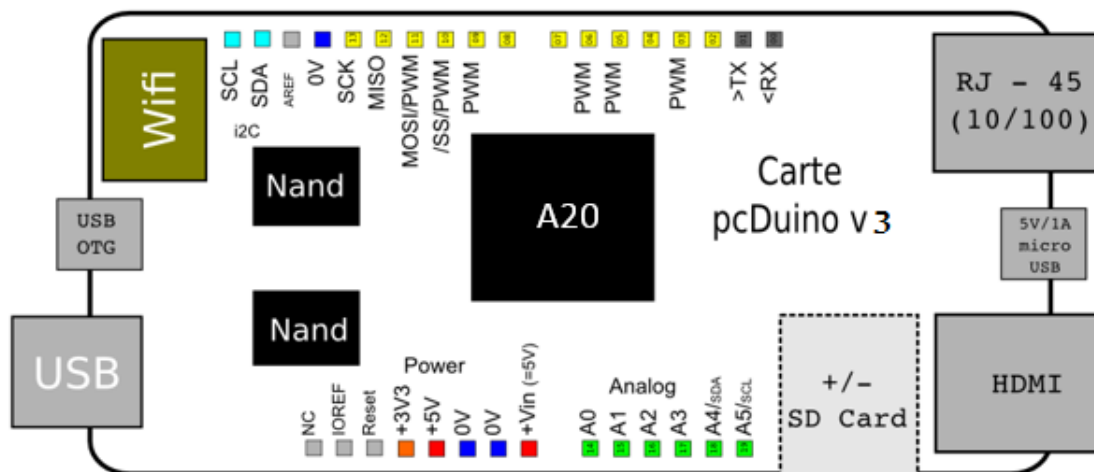


Figure 8 : Carte Pcduino V3

La Pcduino est un miniPC qui dispose des broches de la carte Arduino de base (la UNO, la Duemilanove, etc.) :

- 18 broches d'Entrée/Sortie numérique numérotées de 0 à 17,
- 6 broches analogiques « pures » nommées A0 à A5 dont 4 sont de résolution 12 bits,
- 2 broches de communication I2C.

Parmi les 18 broches d'Entrée/Sortie numérique numérotées de 0 à 17 certaines broches ont des « fonctions spéciales » :

- les broches 0 et 1 sont utilisées par la communication UART (Tx, Rx) donc, son utilisation fait perdre cette possibilité de communication ;
- les broches 3, 5, 6, 9, 10, 11 qui disposent de la génération d'impulsion PWM(Pulse Width Modulation);
- les broches 9, 10, 11, 12 pour la communication SPI (Serial Peripheral Interface). [9]

### Remarque :

A noter que les broches numériques de la PcDuino, donne le niveau HAUT en 3.3V et non pas 5V. Les circuits numériques étant «tolérants» en entrée jusqu'à 2.8V pour un niveau HAUT.

### d- Programmation de la carte PcDuino

Pour les possibilités de programmation des broches E/S, on peut distinguer deux langages :

**Langage C/C++** : (langage compilé – chaîne de compilation présente nativement sur le système) :

- soit en C/C++ dans un éditeur (Geany)
- soit en Arduino-like dans l'éditeur Arduino adapté, qui repose sur Java. En pratique, le code utilisé est tout de même assez différent du code Arduino original.

**Langage Python** : (langage interprété – présent nativement sur le système)

- soit en Python dans un éditeur Python (ex : DreamPie) permettant l'exécution des codes « à la volée » par simple clic sur bouton *exécuter* ;
- soit en Python « Arduino-like » dans un éditeur Python (IDE PyDuino).

Ce qu'il faut savoir, c'est que les broches E/S du pcDuino sont accessibles par l'intermédiaire de fichiers système, en lecture ou écriture. Dès lors, travailler avec les broches E/S et analogiques du pcDuino va essentiellement consister à travailler sur des fichiers. [10]

### II-3-2 Description de la carte Arduino Uno

La carte Arduino Uno (figure 9) est basée sur le microcontrôleur ATmega328. Elle dispose de 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, d'un oscillateur à quartz de 16 MHz, d'une connexion USB, d'une prise d'alimentation, un connecteur ICSP (In Circuit Serial Programming), et un bouton de reset.

Elle contient tout le nécessaire pour piloter le microcontrôleur, il suffit simplement de le connecter à un ordinateur avec un câble USB pour l'utiliser simplement. [11]



**Figure 9 :** Carte Arduino Uno

## ➤ Caractéristique de la carte Arduino Uno

Microcontrôleur	ATmega328
Fréquences Horloge	16 Mhz
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM représenté par le symbole tilde )
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA ( Attention 200mA max total entrée/sortie cumulées)
Intensité maxi disponible pour la sortie 3.3V	50 mA
Intensité maxi disponible pour la sortie 5V	500 mA max Sur port USB
Mémoire Programme Flash	32 KB (ATmega328) dont 0.5 KB sont utilisés par le bootloader (chargeur d'amorçage)
Mémoire SRAM (mémoire volatile)	2 KB (ATmega328)
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Interface	USB (permet d'alimenté la carte et de transféré les programmes vers la carte)
Alimentation externe	Jack
Dimensions	6,86 cm x 5,3 cm

**Tableau 4 :** caractéristique Arduino Uno

### II-3-3 Description de la carte Arduino Nano

La carte Arduino Nano (figure 10) est basée sur le microcontrôleur ATmega328 en version 3.0, c'est une version compacte et réduite de l'Arduino Uno.

Il dispose de 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme sorties PWM), 8 entrées analogiques, d'un oscillateur à quartz de 16 MHz, d'une connexion USB Mini-B, un connecteur ICSP, et un bouton de reset.

Il contient tout le nécessaire pour piloter le microcontrôleur, il suffit simplement de le connecter à un ordinateur avec un câble USB pour l'utiliser. [12]



Figure 10 : Carte Arduino Nano

## a- Caractéristiques de la carte Arduino Nano

caractéristique	NANO version 3
Microcontrôleur	ATmega328
Tension d'alimentation nominale	5V
Tension d'alimentation (recommandé)	7-12V
Entrées/sorties digitales	14 (dont 6 pouvant être utilisées comme sorties PWN)
Entrées Analogiques	8
Courant continu pour les E/S	40 mA
Courant continu pour la pin 3.3V	50 mA
Mémoire Flash	32 KB (ATMega328)
SRAM	2 KB
EEPROM	1 KB
Fréquence d'horloge	16 MHz

Tableau 5 : Caractéristiques Arduino Nano

### II-3-4 Le Châssis Rover 5

Le châssis Rover 5 (figure 11) est un dispositif sur laquelle sont posés tous les composants de notre robot, cette petite plateforme à chenilles avec garde au sol réglable dispose de 4 moteurs à courant continu avec encodeur permettant de réaliser facilement un robot mobile. Ce châssis est conçu pour s'adapter facilement avec toutes les extensions telles que la carte de contrôle et les capteurs. [13]

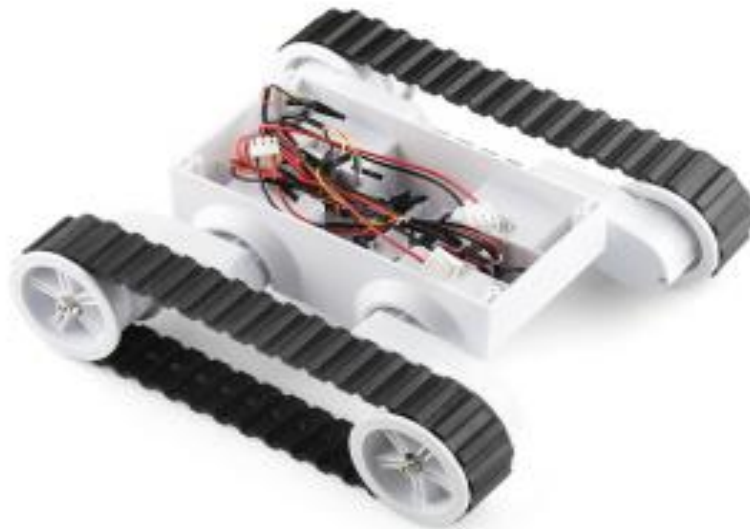


Figure 11 : Châssis Rover 5

#### a- Caractéristique de Rover 5

caractéristique	Rover 5
Motorisation :	4 moteurs à courant continu avec réducteur et encodeurs
-Résolution des encodeurs :	100 pas pour 3 tours
-Tension d'alimentation :	7.2V
-Consommation :	2.5A
-Couple pour 4 moteurs :	10Kg-cm

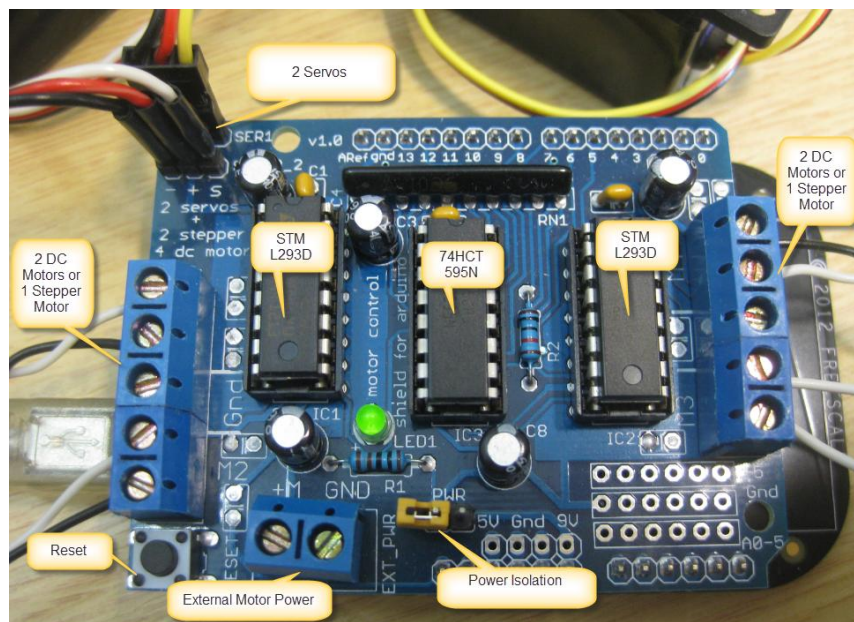
-Réduction :	86%
-Vitesse max :	1 Km/h
-Garde au sol :	Ajustable
-Dimensions du berceau :	170x90mm
-Dimensions totales :	240x230x90mm
-Alimentation :	6 piles de type LR6

**Tableau 6 :** Caractéristique de Rover 5

## II-3-5 Shields DK electronics

DK electronics (figure 12) est un module de contrôle de moteur à courant continu par 4 ponts de H (Motor Drive Shields L293D). C'est un module compatible avec carte Arduino Uno, le circuit permet de contrôler des moteurs à courant continu (DC) dont l'alimentation peut atteindre 36 Volt. Ce circuit intégré pilote 4 canaux de haut tension, haut intensité (courant maximale et de 600mA par canal).

Le pont en H de ce circuit permet de piloter une tension aux bornes d'une charge dans les deux sens, donc faire varier la vitesse et le sens de rotation d'un moteur DC. [11]



**Figure 12 :** Shield DK électronique

### a- Caractéristiques de Shield DK électronique

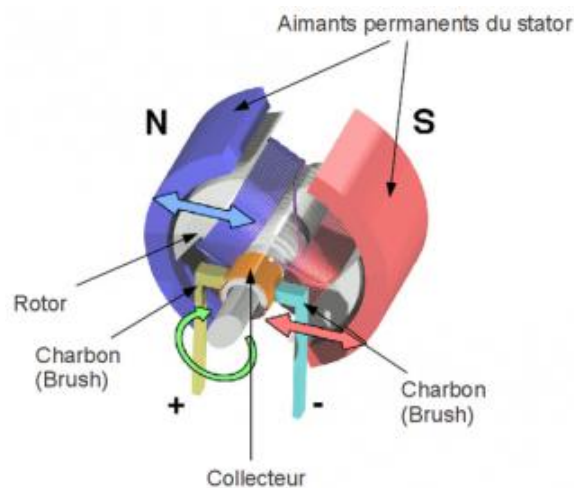
- 2 interfaces pour servomoteurs 5V ;
- Peut piloter 4 moteurs à courant continu DC, ou 2 moteurs pas à pas, et 2 servo à la fois ;
- Jusqu'à 4 moteurs DC bidirectionnels avec sélection de la vitesse individuelle (sur 8 bit) ;
- Jusqu'à 2 moteurs pas à pas (unipolaire ou bipolaire) avec une seule bobine, double bobine, ou demi-pas ;
- 4 ponts en H (H-Bridges) ;
- Il peut fournir 0,6 A par pont (1.2A en courant de crête) avec protection thermique ;
- Pilotage des moteurs à courant continu de 4.5V à 36V ;
- Des résistances pull down désactivant les moteurs au cours de la mise sous tension ;
- Bouton de réinitialisation (Reset) ;
- 2 interfaces d'alimentation pour séparer la partie logique de la partie puissance (moteurs) ;
- Compatible avec les cartes Arduino Uno, Mega, Diecimila & Duemilanove.

### II-3-6 Moteur à courant continu (MCC)

Un moteur est un composant de conversion d'énergie électrique en énergie mécanique. Les moteurs à courant continu (figure 13) transforment l'énergie électrique en énergie mécanique de rotation, pour être précis. Mais ils peuvent également servir de générateur d'électricité en convertissant une énergie mécanique de rotation en énergie électrique. C'est le cas par exemple de la dynamo.

Le moteur à courant continu est composé de deux parties principales :

Le rotor (partie qui tourne) et le stator (partie qui ne tourne pas, statique). En électrotechnique le stator s'appelle aussi inducteur et le rotor s'appelle l'induit. Sur la (figure 13), on peut l'observer au milieu entouré par les aimants bleu et rouge qui constituent le stator le rotor composé de fils de cuivre enroulés sur un support lui-même monté sur un axe. Cet axe, c'est l'arbre de sortie du moteur. C'est lui qui va transmettre le mouvement à l'ensemble mécanique (pignons, chaîne, actionneur...) qui lui est associé en aval. [11]



**Figure 13 :** Moteur à courant continu

## II-3-7 Module joystick

Les joysticks (figure 14) sont composés de deux résistances variables linéaires insérées sur deux axes différents qui vont faire varier la tension de 5V délivrée en sortie par la carte. Cette tension va alors varier entre 0 et 5V, et cette variation sera la première donnée analogique nécessaire pour positionner le moteur à courant continu à la vitesse voulu

En effet cette tension (située entre 0 et 5 V) va être appliquée sur une entrée analogique de la carte dite « ANALOG IN » (ex : A0, A1, A2, ...) et sera pris en charge comme variable analogique par le microcontrôleur.

Un seul joystick peut alors émettre deux variables analogiques qui à leurs tours peuvent contrôler deux moteur.

Ces résistances variables doivent impérativement être linéaires pour faciliter le contrôle des moteurs. [14]

### a- Caractéristiques techniques

- 2 axes par potentiomètre 10 k $\Omega$ .
- Ressorts de rappel au centre.
- Bouton de commande de forme champignon, qui se déclenche lors d'une pression au milieu du joystick.

On trouve aussi plusieurs indications :

- GND : ira à la masse ;
- VCC : +5V ;
- S-X : ira sur une entrée analogique. Ceci est l'axe avant/arrière ;

S-Y : ira sur une entrée analogique. Ceci est l'axe gauche/droite.



Figure 14 : Joystick

## b- Principe de fonctionnement du joystick

Le principe de fonctionnement du joystick est présenté sur la figure 15 :

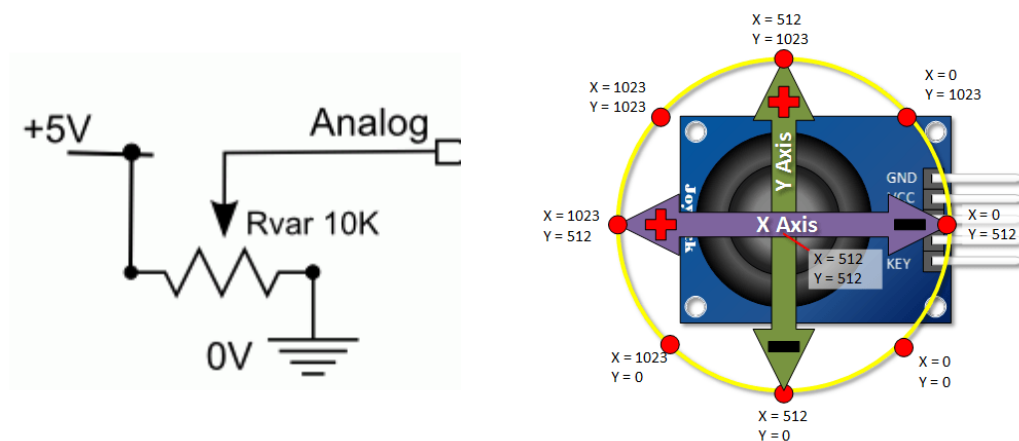


Figure 15 : Fonctionnement du joystick

Quand le joystick est au centre, les moteurs sont arrêtés donc le robot est en repos, vers « le haut » ou vers « le bas » le robot avance ou recule respectivement, vers « la gauche » ou vers « la droite », le robot marche à gauche ou à droite respectivement.

## II-3-8 Module XBee

Les modules XBee sont des circuits de communication sans-fil utilisant les protocoles 802.15.4 et Zigbee. Proposés depuis quelques années par la société 'Digi International', les modules XBee sont des modems radio très élaborés, fonctionnant dans ce que l'on appelle

aujourd'hui la bande ISM (Industrie Science et Médical), c'est-à-dire sur une fréquence de 2.4 GHz. Il existe deux version de module de communication dans ce site : [15]

### a- Les différents Versions du XBee

Le XBee dispose de deux versions :

- une version normale XBee (figure 16) avec une portée d'une centaine de mètres (30/100m) en terrain découvert ;
- une version XBee-PRO (figure 17) d'une puissance supérieure et destinée à couvrir des distances dépassant le kilomètre (100/1600m).

Les deux versions sont totalement compatibles entre eux. [15]

#### ❖ Version normale et pro



Figure 16 : Module XBee normale (Antenne céramique)



Figure 17 : XBee version Pro (Antenne fouet).

### b- Caractéristique XBee Version Normale et Pro

caractéristique	Version normale	Version pro
Portée intérieur / urbain :	jusqu'à 30 m.	jusqu'à 100 m.
Portée extérieur champs libre :	jusqu'à 100m.	jusqu'à 1.6Km.
Puissance d'émission RF :	1 mW (0 dBm).	100 mW (20 dBm).
Courant de repos :	< 10 $\mu$ A.	< 10 $\mu$ A.
Fréquence de fonctionnement :	2.4 GHz.	2.4 GHz.
Vitesse de transmission RF :	250Kbps	250Kbps

Tableau 7 : Caractéristique XBee Version Normale et Pro

Il existe aussi des XBee Shield parmi lesquels on a le Shield XBee USB (platine d'interface USB XBee) et le Shield XBee pour Arduino.

## c-Modes de XBee

Le XBee possède trois modes de communication : TRANSPARENT, COMMAND et API.

**Le mode TRANSPARENT ou AT :** est le mode par défaut à la mise en marche du module, celui qui reçoit et envoie les données.

**Le mode COMMAND :** permet de configurer le module, les entrées, les sorties, l'adresse, l'adresse de destination des messages.

**Le mode API :** est un peu plus compliqué. Une API (Application programming interface) est un terme bien connu en informatique. Il désigne une interface fournie par un programme informatique, c'est-à-dire un ensemble de fonctions qui facilitent la programmation d'un côté et qui de l'autre communique en langage binaire pour le XBee, sous forme de paquets. Ce mode devient utile quand il s'agit de construire des messages au format XBee à partir d'un ordinateur ou d'un microcontrôleur comme Arduino. Le mode API n'est possible qu'avec une connexion locale en série et filaire avec l'ordinateur ou l'Arduino mais pas entre modules XBee.

[15]

### I-3-9 Shield XBee pour Arduino

Carte d'extension qui se branche directement sur la carte Arduino voir (figure 18).

La connexion vers Arduino est simple :

- pin1 = 3.3 Volt
- pin2 = Digital OUT —> vers RX Arduino (récepteur)
- pin3 = Digital IN —> vers TX Arduino (émetteur)
- pin10 = GND.

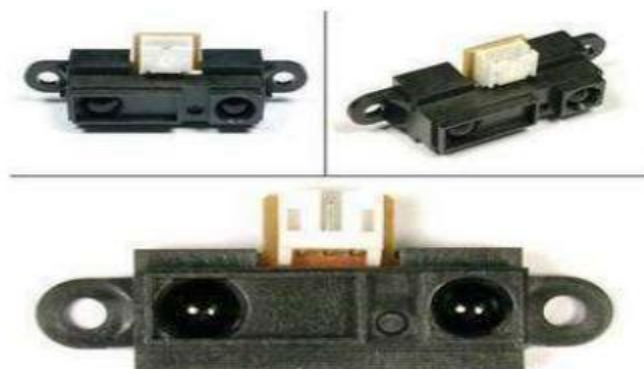


**Figure 18** : Shield XBee Arduino

### II-3-10 description des different capteur utilisée

#### a- Capteur infrarouge

Les capteurs infrarouges (figure 19) fonctionnant avec des radiations non visibles, dont la longueur d'onde est juste inférieure à celle du rouge visible, la simplicité de mise en œuvre, font de ces capteurs un composant incontournable à coût raisonnable. Néanmoins ces capteurs sont après test, très sensibles aux bruits, c'est pourquoi il faut filtrer (Passe-bas) leurs données renvoyées afin de ne pas détecter un obstacle inexistant ce qui engendre une erreur grave sur la trajectoire qui suivra le robot. On les rencontrera le plus souvent comme détecteurs de proximité. En utilisant un couple émetteur/récepteur, comme le montre la figure 19. [17]



**Figure 19** : Capteur infrarouge

## b- Capteur de gaz MQ-2

Le capteur de gaz MQ2 (figure 20) est utile pour la détection de fuites de gaz (dans la maison et dans un environnement industrie). Il est adapté pour détecter les gaz H<sub>2</sub>, GPL, CH<sub>4</sub>, CO, l'alcool, la fumée. En raison de sa grande sensibilité et de temps de réponse rapide, les mesures peuvent être prises dès que possible. La sensibilité de ce capteur peut être réglée à l'aide d'un potentiomètre intégré sur la carte de conditionnement de ce capteur. [18]



Figure 20 : Capteur de gaz MQ-2

### b - 1 Caractéristique de capteur de gaz

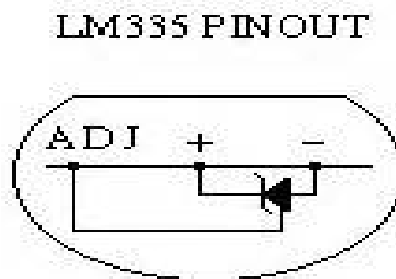
Caractéristique	Valeur
Largeur (mm) x Longuer (mm)	22 x 35
Tension d'entrée	5V
Courant (A)	0.16A
Pins	4
Model	MQ2
Tension de sortie	0 V a 5V DC
Puissance (W)	0 .8

Tableau 8 : Fiche technique de MQ-2

### c- Capteur de température (LM 335)

LM335 (figure 21) est un capteur de température de haute précision et facile à calibré. Il fonctionne comme une diode Zener dont la tension de claquage est directement proportionnelle à la température absolue avec une sensibilité de +10mV/°K .et d'une impédance dynamique inférieure à 1Ω.

Il peut fonctionner dans une plage de température allant de - 40°C à 100°C sous un courant constant pouvant varier de 400µA à 5mA. La tension à ces bornes est de 2,98V à 25°C, le montage le plus utilisé pour ce composant est donnée par la (figure 21). [19]

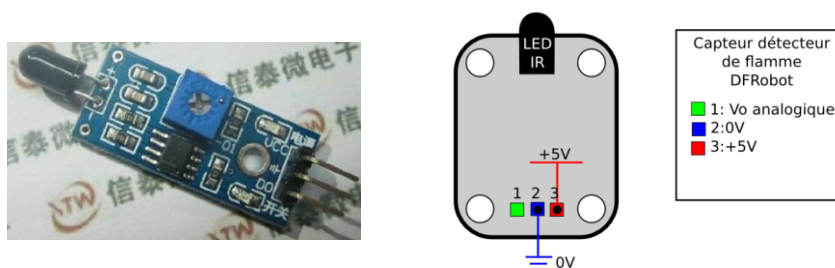


**Figure 21 :** Capteur de température

### d- Capteur de Flammes

Le Capteur de Flammes (figure 22) peut être utilisé pour détecter un feu ou d'autres longueurs d'onde de 760 nm à 1 100 nm. Dans le jeu robotique de lutte contre l'incendie, la flamme joue un rôle important lors de la recherche, celle-ci peut être utilisée au travers des yeux du robot afin de trouver la source de l'incendie. Il peut être utilisé par des robots de lutte contre l'incendie, des robots footballeurs. Angle de recherche du capteur de flamme de 60 degrés.

La température opérationnelle du détecteur de flamme est de -25 degrés Celsius à +85 degrés Celsius, dans le cadre de la flamme il convient de noter que la distance de recherche de la flamme ne doit pas être trop proche afin d'éviter des dommages. [20]



**Figure 22 :** Capteur de Flammes

## d -1 Caractéristique techniques

- Il détecte le feu et d'autres longueurs d'onde : entre 760 à 1100 nm (dans l'infrarouge)
- Portée de détection : 20 cm (4,8 V) - 100 cm (1 V)
- Tension d'alimentation : + 5 V
- Angle de recherche du capteur de flamme de 60 degrés.

## II-3-11 Webcam PC810

La webcam PC 810 (figure 23) est dotée d'un microphone et une caméra, conçu pour être utilisé comme un périphérique d'un ordinateur qui produit une vidéo dont la finalité n'est pas d'atteindre une haute qualité, mais de pouvoir être transmis en direct au travers d'un réseau.(voir figure 23) [22]



Figure 23 : Webcam PC810

### a- Caractéristique de JVC PC810

caractéristique	JVC PC810
Résolution	5 MEGA PIXEL
Poids	0.1 kg
Eclairage	Quatre Leds
Entre	USB

Tableau 9 : Caractéristique de JVC PC810

### II-3-12 Alimentation MB 102

L'alimentation MB102 (figure 24) est un module d'alimentation de breadboard fabriqué par la YwRobot. Ce module fournit 5 V et 3,3 V doubles alimentation, et dispose d'une prise femelle USB.

Il y a une LED verte pour indiquer la présence de la puissance, et une ON / OFF de verrouillage interrupteur pour contrôler le pouvoir au conseil d'administration.

La tension d'entrée à travers la prise de canon doit être comprise entre **6,5 V** et **12 V**. Par conséquent, si vous souhaitez l'utiliser à sa capacité maximale, vous aurez besoin de rester dans cette fourchette. Ce qui est assez bon pour la plupart des applications. Le courant de sortie maximale des états manuels soit 700 mA. Cependant, il est probablement préférable d'utiliser des tensions et des courants beaucoup plus faibles pour être sur le côté sûr au cas où vous faites une erreur sur votre circuit de breadboard. [21]



Figure 24 : Alimentation MB 102

### II-4 Description de la partie logicielle

**I-4-1 Le logiciel VNC** (Virtual Network Computing – réseau virtuel de calcul)

VNC (figure 25) est un système de visualisation et de contrôle, initialement développé à l'Université de Lancaster (R.U) qui a ensuite été intégré à la plate-forme collaborative Platine développée au LAAS-CNRS. Le logiciel VNC permet de visualiser le fonctionnement d'un

ordinateur et d'interagir avec lui à partir de n'importe quel autre ordinateur raccordé à internet. Ainsi, si un utilisateur effectue une navigation HTTP (HyperText Transfer Protocol) sur cet ordinateur, l'ensemble des utilisateurs raccordés à ce dernier via l'application VNC pourront également voir le résultat de cette navigation.

Ce n'est pas seulement l'application « navigateur Web » qui s'exécute dans le serveur qui est partagée mais tout le bureau de l'utilisateur utilisant cet ordinateur. [23]



**Figure 25:** Interface de logiciel VNC

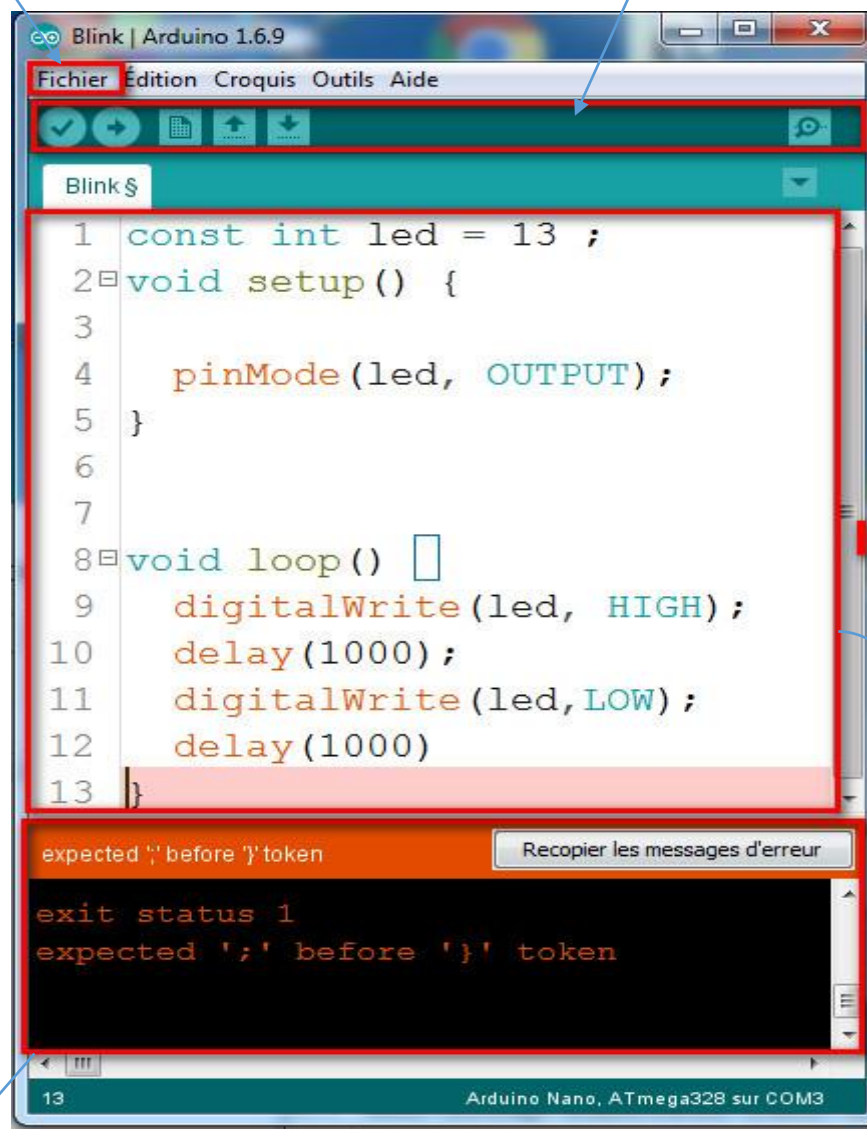
### II-4-2 L'environnement de programmation Arduino

L'environnement de programmation Arduino appelé IDE est distribué sur le site d'Arduino (compatible avec Windows, Linux et mac), A noté qu'on peut associe au compilateur Arduino d'autre programme saisie avec Arduino (extensions pour code blocks, visuel studio, Eclipse, etc.). Et la carte Arduino utilise le langage de programmation C/C++ parce qu'il est facile à utiliser et très efficace.

L'interface de l'IDE Arduino est plutôt simple (voir la figure 26), il offre une interface minimale et épurée pour développer un programme pour les cartes Arduino. [11]

Les options de configuration du logiciel





les boutons qui facilitent l'utilisation de l'IDE



Console d'affichage des messages de compilation

Fenêtre d'édition du programme

Figure 26 : L'IDE Arduino

-  *New* : pour créer un nouveau programme (sketch) ;
-  *Open* : ouvrir un programme existant. Le menu n'est pas déroulant à cause d'un bug...pour obtenir un menu déroulant passer par file/open ;
-  *Save* : sauvegarde le programme, si vous voulez le sauvegarder sous un autre nom, passer par file/save as ;
-  *Serial Monitor* : pour ouvrir la fenêtre qui permet de visualiser les données transmises par le port série de l'Arduino (très utile pour le débogage...)

## II-4-3 L'IDE PyDuino

Le PyDuino est un portage en langage Python des instructions du langage Arduino pour les nouvelles plateformes de type mini-pc : en un mot, vous accédez aux broches E/S numériques et analogiques de votre mini-pc comme vous le feriez avec Arduino, sauf que vous codez en Python et donc vous pouvez au sein d'un même code programmer les broches E/S, faire des mesures, etc. Et interagir avec le système, utiliser les périphériques système (vidéo, réseau, fichiers,...), etc. De plus, PyDuino est indépendant du matériel utilisé : un même code PyDuino pourra être exécuté sur des plateformes différentes (pcDuino, RaspberryPi,...) ! (figure 27).

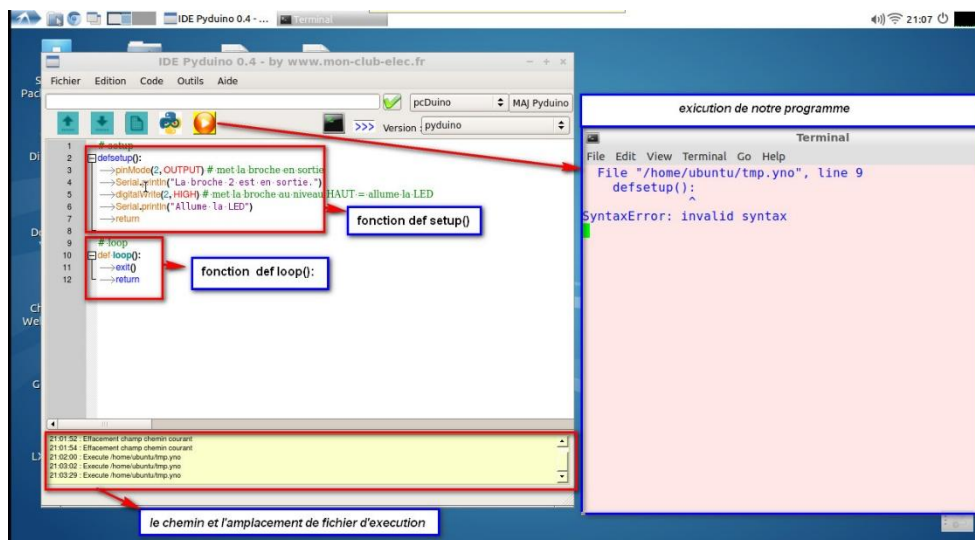


Figure 27 : L'IDE pyDuino



*New* : pour créer un nouveau programme ;



*Open* : ouvrir un programme existant. Le menu n'est pas déroulant à cause d'un bug...pour obtenir un menu déroulant passer par file/open ;



*Save* : sauvegarde le programme, si vous voulez le sauvegarder sous un autre nom, passer par file/save as ; [24]

## II-4-4 Le logiciel Fritzing :

Fritzing est un logiciel open-source multiplateforme permettant de construire des schémas des circuits que nous utilisons avec Arduino. Plusieurs vues sont disponibles : platine d'essai, schémas électriques et circuit imprimé. Il permet aussi l'export en image voir la (figure28).

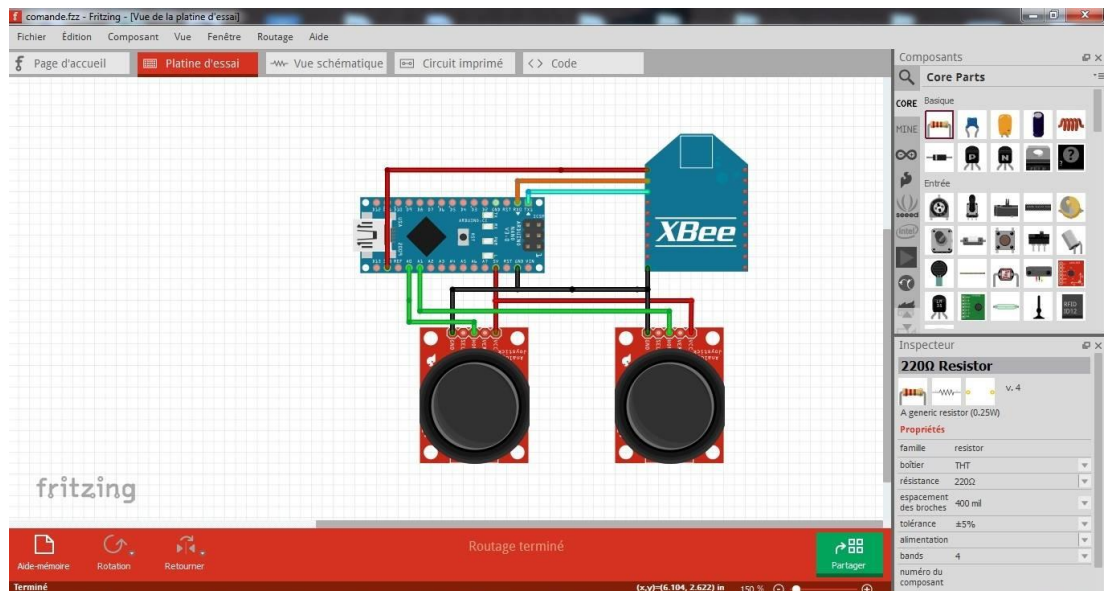


Figure 28 : Interface de Fritzing

## II - 5 Conclusion :

Dans ce chapitre l'utilisation des différents logiciels et matériels, nous a permis de concevoir notre système qui est basé sur des diverses cartes (pcDuino, Arduino Uno, Arduino Nano).

La conception est une étape indispensable dans le but de réaliser un projet et lui assurer un bon fonctionnement.

# Chapitre III :

## Réalisation pratique

## III-1 Introduction

Cette partie nécessite la création d'un protocole série pour que les différentes cartes communiquent entre elles. Dans cette partie, nous allons voir comment sont programmées les différentes cartes présentes sur le robot ou dans la télécommande. Cette partie permet de contrôler tous les mouvements que peut faire le robot, que ce soit l'utilisation du bras manipulateur ou des déplacements.

## III-2 configuration logicielle du matériel

### III-2-1 Configuration des différents logiciels

#### a- Mise en marche de la PcDuinoV3

Pour commencer à travailler sur le mini-pc (pcDuinoV3), il faut disposer du matériel cité ci-dessous afin de procéder à l'installation du logiciel client/serveur (VNC) dont on a parlé dans le chapitre précédent. Au final, nous travaillerons depuis notre poste fixe.

- pcDuino ;
- une alimentation 5V/1000mA (micro-USB type A) ;
- un câble HDMI ;
- un écran avec une entrée HDMI ;
- câble RJ 45 ;
- modem wifi ;
- une souris USB et d'un clavier USB.



Figure 29 : Matériel nécessaire pour la mise en marche de la pcDuino

Grâce à ses périphériques, la carte mini-pc est opérationnelle, on peut travailler facilement sur cette carte, mais pour faire des applications en utilisant la pcDuino comme une carte embarquée, cela me paraît impossible avec ses périphériques, donc il faut faire un moyen d'accéder à distance avec notre poste fixe.

### b- Liaison entre pcDuino et le poste fixe avec contrôle à distance

Il faut connecter la carte pcDuino avec le matériel ci-dessous et créer un réseau local pour installer les logiciels nécessaires qui nous permettent d'obtenir l'accès à distance (voir Figure 30).

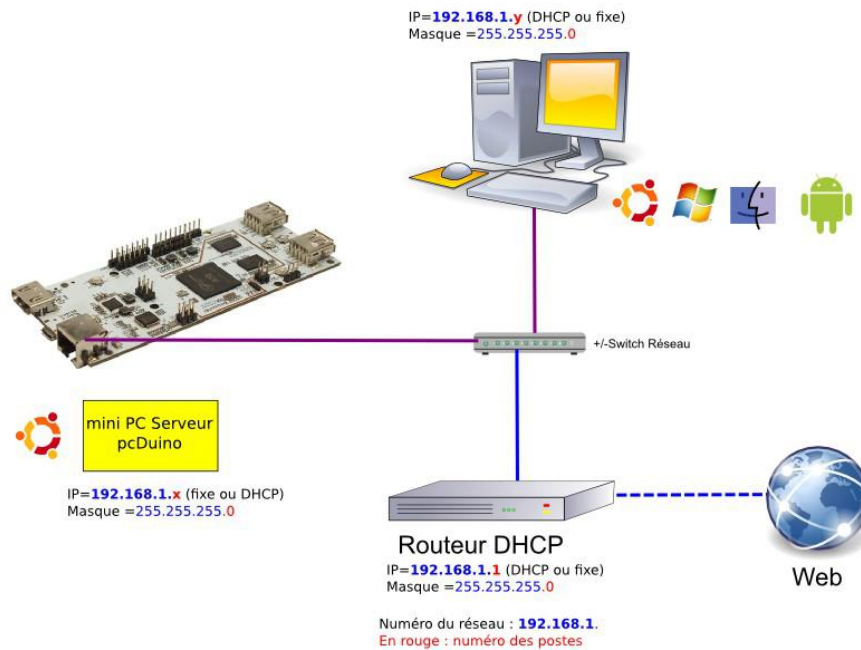


Figure 30 : Mise en réseau de la pcDuino

- Le 'serveur VNC' est installé sur la carte pcDuino, il faut télécharger donc le 'client VNC' et l'installer sur le poste fixe.
- Sur la figure 30, le mini-pc et le poste fixe sont sur le même réseau local, cela nous permet de fixer le serveur DHCP (adresse IP statique) afin d'utiliser ces adresses dans un autre redémarrage de système.
- Une fois qu'on a installé le 'client VNC' sur notre pc et fixé le serveur DHCP, il ne reste qu'à débrancher tous les périphériques (clavier, souris, écran...) et de se connecter directement au mini-pc à travers le réseau wifi par l'accès VNC.

## Réalisation pratique

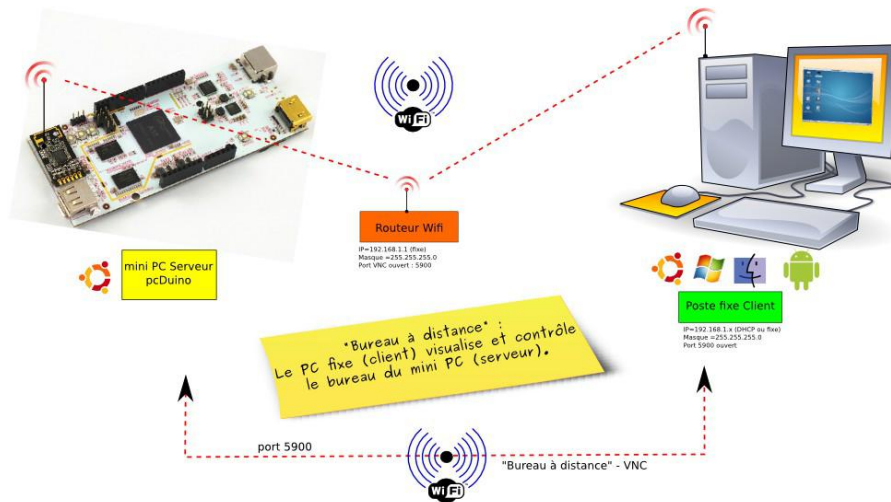


Figure 31 : Utilisation VNC avec WIFI

- pour communiquer à la pcDuino via le 'client VNC', il faut saisir l'adresse IP du mini-pc suivie du numéro port de communication et le mot de passe, comme expliqué sur la figure 32 :



Figure 32 : Procédure pour se connecter au mini-pc à partir du poste fixe

- Après quelques instants, le bureau de la pcDuino sera affiché sur notre poste fixe comme le montre la figure 33 :



**Figure 33 :** Le bureau de la pcDuino visible sur le poste fixe

### c- Obtention des images à travers une webcam

#### c-1 Installer 'mjpeg streamer' sur la pcDuino

Après la mise en marche et la configuration du mini-pc, on a procédé à l'installation de 'MJPEG-streamer' pour l'obtention de flux vidéo diffusé par la webcam de notre pcDuino.

'MJPEG-streamer' est utilisé pour capturer des images d'une caméra webcam, on les met sous forme de flux dans un navigateur, comme Firefox, Google Chrome. 'MJPEG-streamer' peut utiliser une fonction de compression matérielle de webcams pour réduire la surcharge du CPU (Central Processing Unit), offrant une consommation d'énergie légère et faible pour les appareils embarqués, voir figure 34.



**Figure 34 :** pcDuinoV3 avec Webcam

## Réalisation pratique

Afin d'obtenir le flux vidéo de la caméra webcam, nous avons installé le logiciel "mjpg Streamer" sur la pcDuino. Ce logiciel permet à tout ordinateur connecté au même réseau que le mini-pc d'obtenir un flux vidéo via un navigateur. Pour cela, on doit saisir dans la barre d'adresse de navigateur "192.168.8.193:8080/?action=stream" tel que "192.168.8.193" qui est l'adresse IP de la pcDuino et 8080 qui est le port dédié à 'MJPEG Streamer' installé sur la carte. Ainsi, pour obtenir le flux vidéo tiré par la webcam, il suffit d'avoir un navigateur Web installé sur notre pc et d'une connexion wifi dans notre réseau local.

Par la suite, on a besoin d'afficher les informations liées au capteur ainsi que la vidéo de la webcam sur la même interface. Pour cela, on doit créer une page web statique qui va nous permettre de visualiser tous les informations de notre robot à cette adresse (192.168.1.193 : 8080), cette page illustré par la figure 35.



Figure 35 : Interface de site embarqué 'ROVER 2016'

## III-2-2 Réalisation Matérielle

### a- Structure globale de notre système

Après la mise en marche et la configuration de la carte pcDuino, on est passé à la réalisation matérielle ; chaque partie réalisée est testée séparément sur un lab d'essai. Puis on a regroupé et réalisé les différentes maquettes sur des plaques perforantes, et le schéma global du système est représenté par la figure 36.

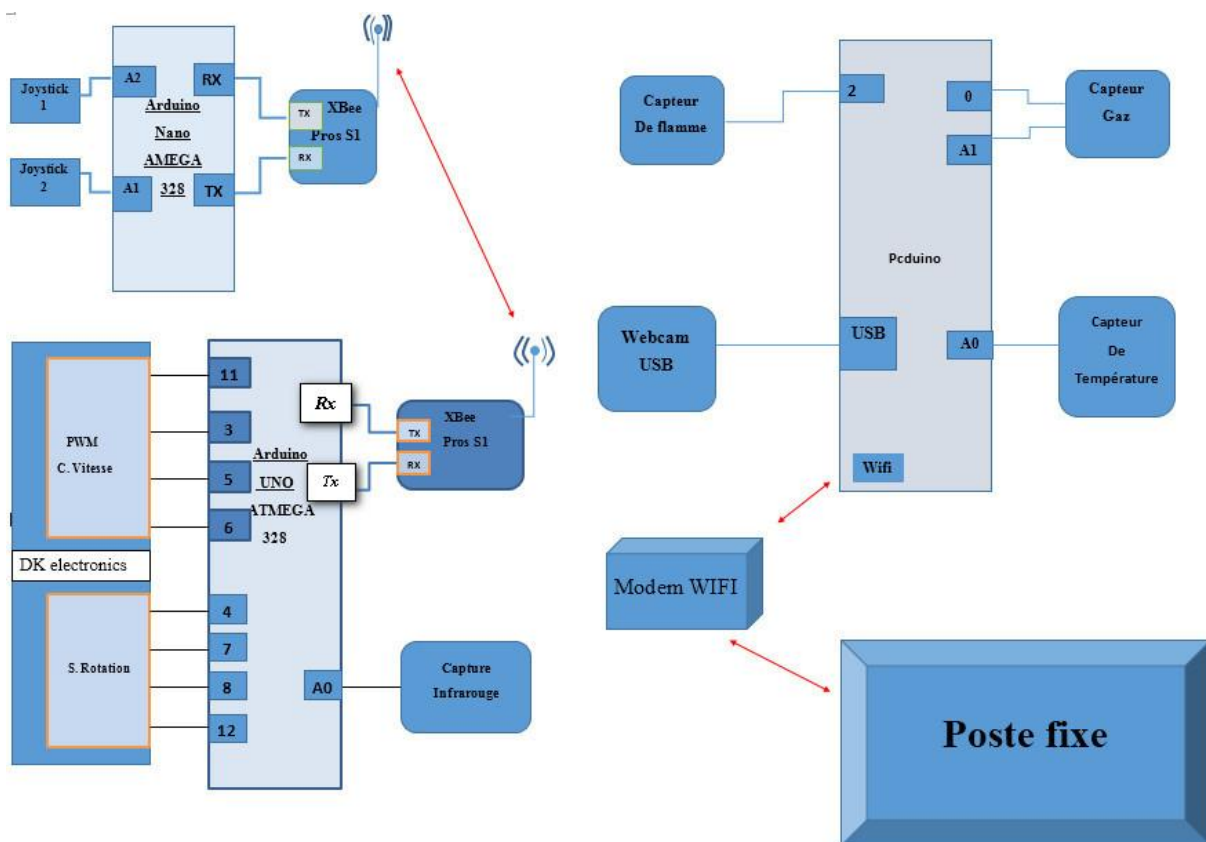


Figure 36 : Schéma global du système

#### ❖ Les broches utilisées de chaque carte de notre système :

##### 1- pcDuino

- Capteur de gaz : vers les broches 0 et A1 ;
- Capteur de flamme : vers la broche 1 ;
- Capteur de température : vers la broche A0 ;

## 2- Arduino Uno

- **Module XBee** : vers les broches 0 et 1 ;
- **Shield DK-electronics** : vers les broches 3, 4, 5, 6, 7, 8, 11, 12 ;
- **Capteur infrarouge** : vers la broche A0 ;

## 3- Arduino Nano

- **Joystick** : vers les broches A0 et A1 ;
- **Module XBee** : vers les broches 0 et 1 ;

## b- Capteur de gaz MQ-2

Le MQ2 est un capteur de gaz et de fumée. On va équiper notre robot avec le MQ-2, afin de nous donner des informations liées à l'emplacement du robot, l'information sera envoyée toutes les 100 millisecondes au site embarqué de 'rover 5', et son schéma de branchement électrique illustré dans la figure 37 :

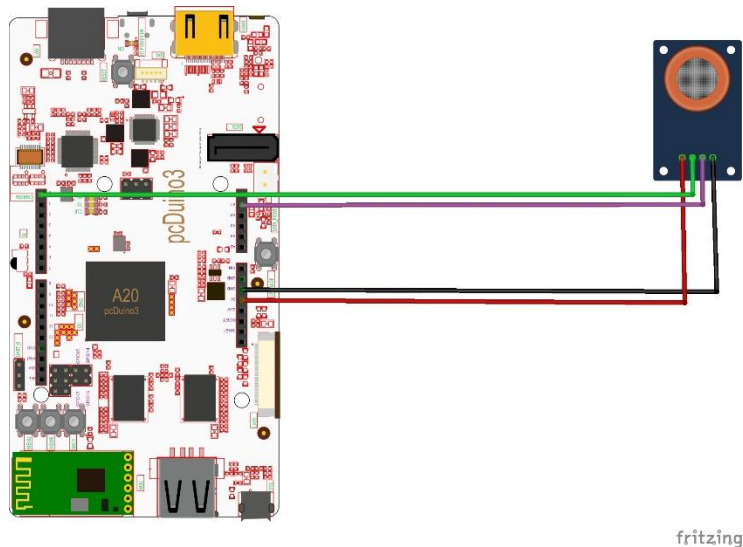
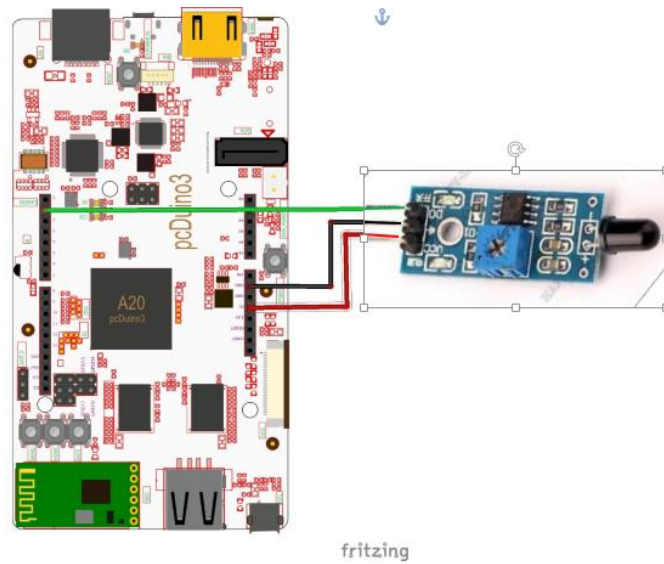


Figure 37 : Schéma de branchement électrique MQ-2

## c- Capteur de flamme

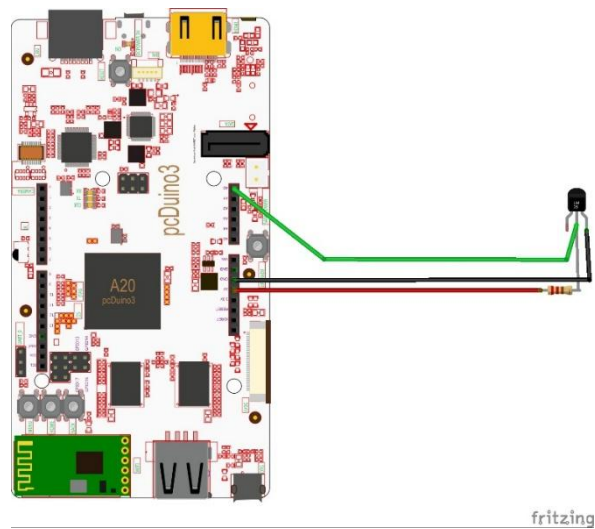
Le but de ce capteur est de pouvoir détecter une flamme. Un code a été créé afin d'indiquer la distance de la flamme par rapport au capteur. En cas de marche arrière, il servira à éviter ces flammes pour ne pas endommager le robot. Cette information sera affichée sur le site embarqué, son schéma électrique est présenté sur la figure 38 :



**Figure 38** : Schéma de branchement électrique de capteur de flamme

## d- Capteur de température

Le but de ce capteur est de nous informer sur la température ambiante de l'enceinte où se trouve notre robot mobile, de plus que cette information sera affichée sur le site embarqué. Son schéma de branchement électrique est présenté sur la figure 39 :



**Figure 39** : Schéma de branchement électrique de capteur de température

## e- Schéma globale des différents capteurs avec la pcDuino

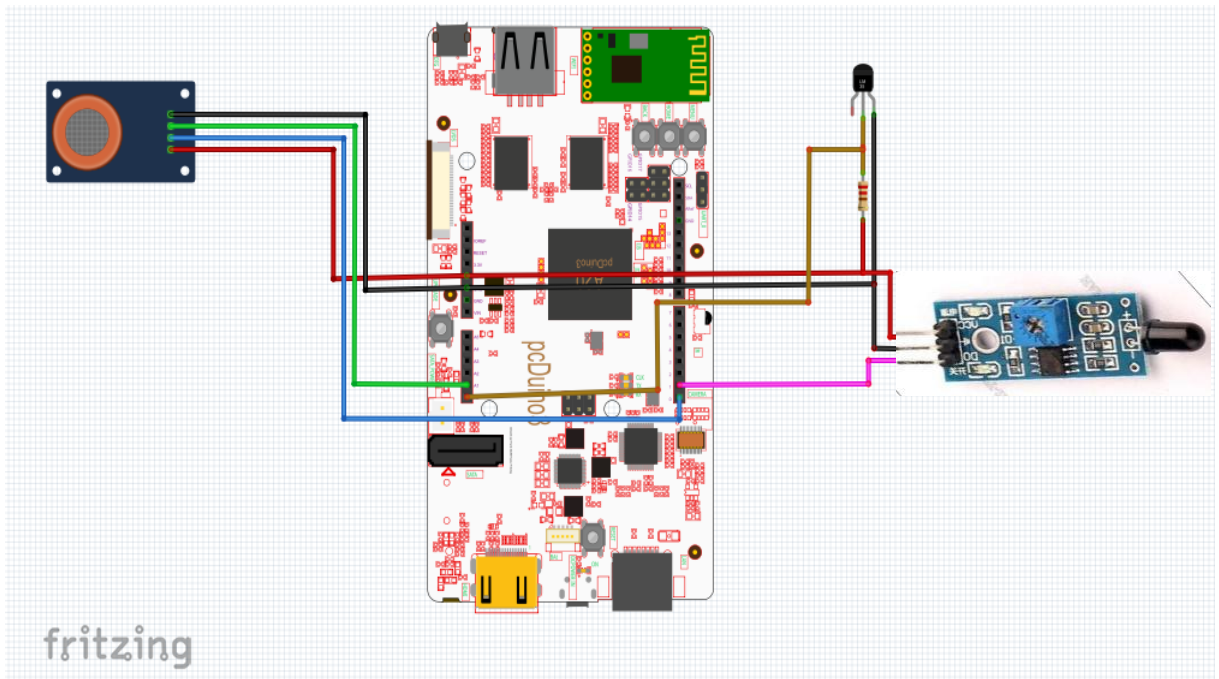


Figure 40 : Schéma globale des différents capteurs avec la pcDuino

## f- Capteur infrarouge

Le but de ce capteur est de pouvoir détecter la distance, quand le robot est en marche arrière, il servira à éviter les obstacles pour ne pas bloquer le robot. Son schéma de branchement électrique est présenté par la figure 41 :

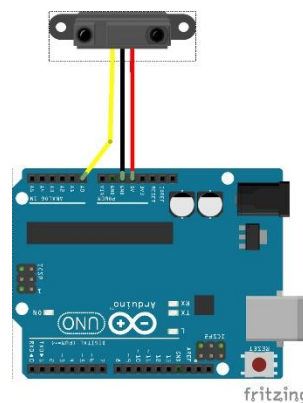


Figure 41 : Schéma de branchement électrique de capteur infrarouge

## g- Contrôle global de robot

### 1- Le déplacement

Pour faire déplacer notre robot ‘rover 5’, il faut faire actionner les moteurs à courant continu, on peut avoir les sens de directions suivantes :

#### 1-a En Avant

- Déplacement en avant : il faut que tous les moteurs soient alimentés dans un seul sens ;
- Avant-droite : il faut actionner les moteurs du côté gauche à une vitesse supérieure à celle des moteurs du côté droit ;
- Avant-gauche : il faut actionner les moteurs du côté droit à une vitesse supérieure à celle des moteurs du côté gauche.

#### 1-b En Arrière

- Pour le recul du robot, il faut que tous les moteurs actionner dans des sens opposé.
- Arrière-droit, il faut actionner les moteurs de côté droits à une vitesse supérieure à celle des moteurs de côté gauche.
- Arrière-gauche : il faut actionner les moteurs du côté droit à une vitesse supérieure à celle des moteurs du côté gauche.

#### 1-c Tourner sur l’axe

- Pour tourner sur l’axe, il faut actionner les moteurs des côtés gauche et droite avec la même vitesse mais dans des sens différents.

Les quatre moteurs sont branchés sur un ‘Shield DK electronics’ qui nous permet de mieux commander et actionner les moteurs dans les deux sens avec contrôle de vitesse grâce au pin PWM de Arduino et un Shield XBee pour la commande à distance et tout cela est placé sur une carte Arduino Uno ; son schéma électrique est représenté par la figure 42 :

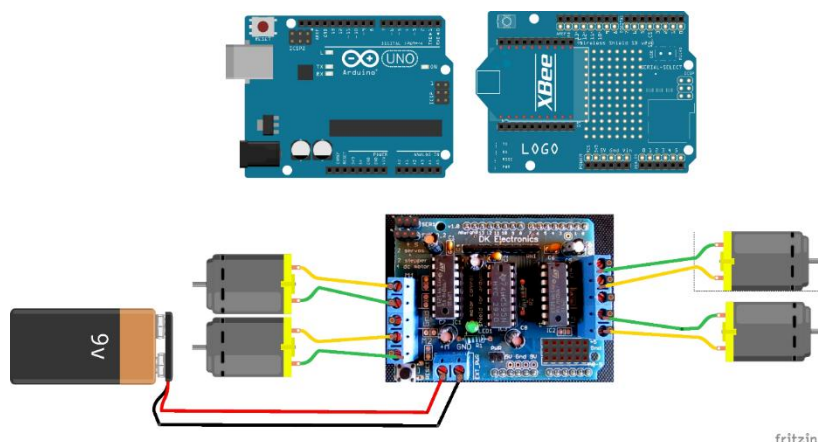


Figure 42 : Schéma électrique permettant le contrôle des moteurs avec DK electronics

## 2- La télécommande

Nous avons créé notre propre télécommande à l'aide de 2 joysticks, Arduino Nano et le module XBee. Cette télécommande permet un meilleur contrôle pour les moteurs ainsi que pour les servomoteurs, la télécommande est composée de 4 axes, mais Pour contrôler les déplacements nous n'utilisons que 2 axes comme le montre la figure 43 :

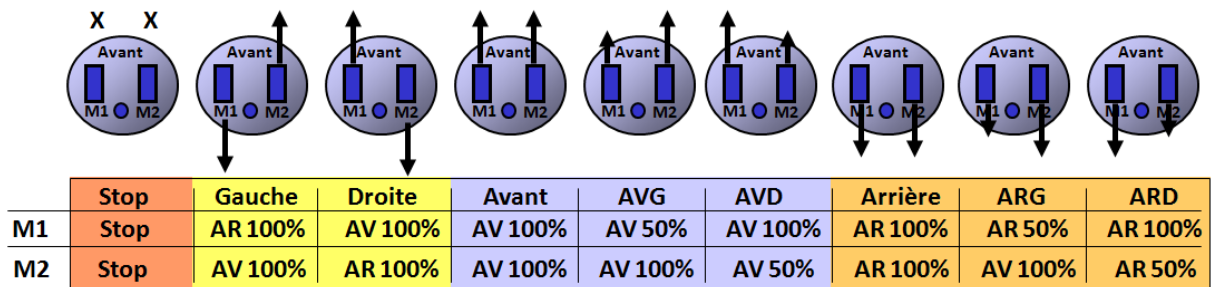


Figure 43 : Sens de déplacement de notre châssis

## h- Organigramme général du système

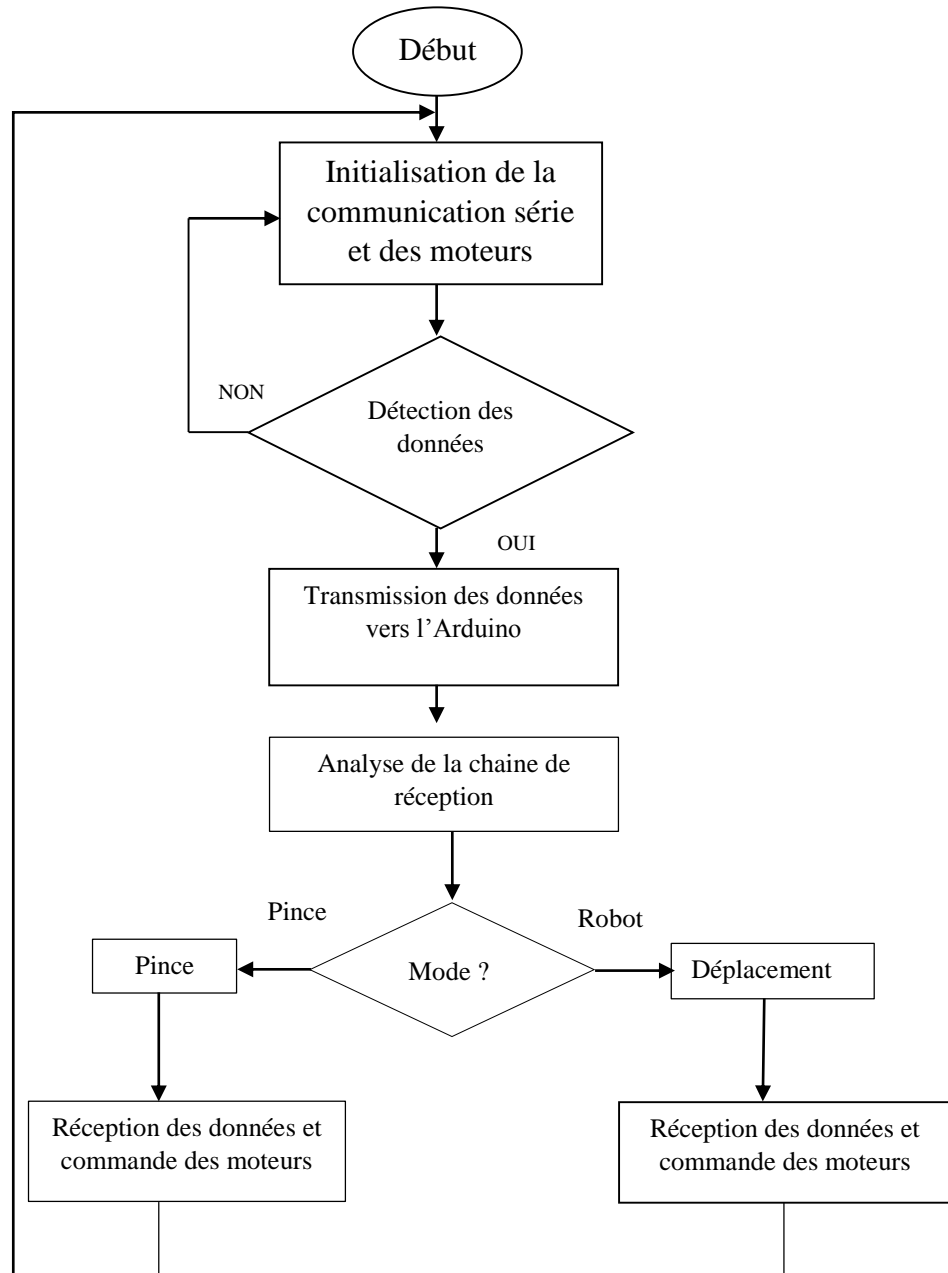
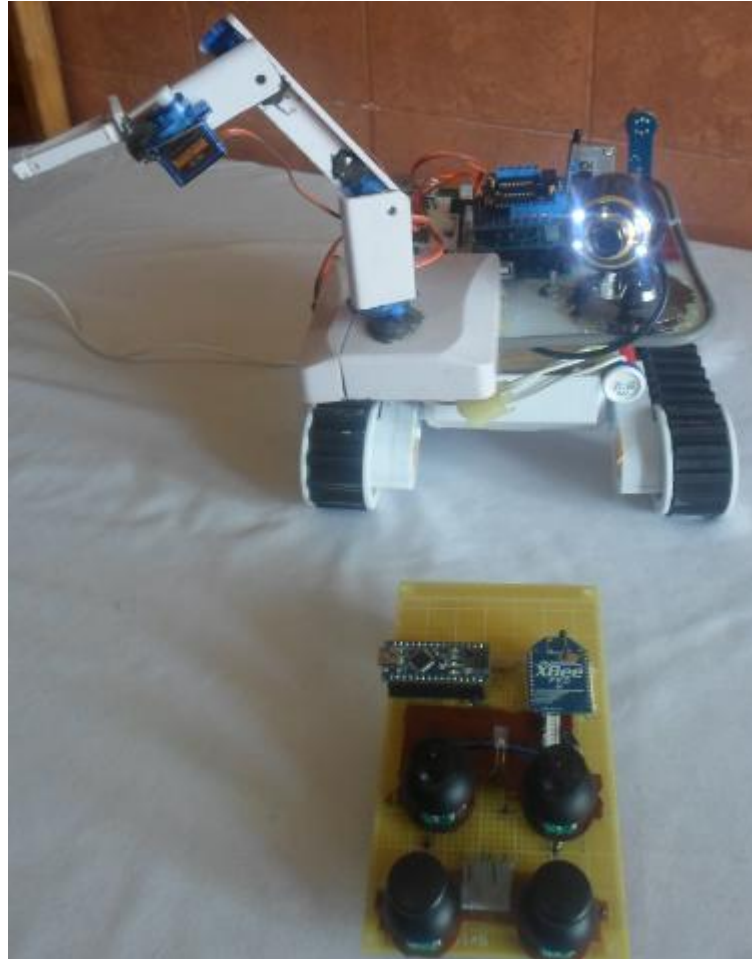


Figure 44 : Organigramme du Système

## i- Vue globale du robot

Voici notre système global que nous avons obtenu après assemblage de ses différents constituants (voir figure 45) :



**Figure 45** : Vue globale du robot

## III-3 Conclusion

Dans ce chapitre, nous avons illustré les étapes suivies pendant la réalisation de notre prototype mobile par des figures et des schémas de câblages électroniques.

Enfin, nous avons pu réaliser notre robot mobile avec sa télécommande pour pouvoir communiquer avec lui à distance.

# Conclusion générale

### Conclusion générale

Dans notre travail mené tout au long de ce mémoire, nous avons conçu et réalisé un système robotisé à base des cartes : pcDuino, Arduino Uno et Arduino Nano avec une transmission des données via le wifi vers un ordinateur où la visualisation des données a été faite en utilisant un site embarqué créé pour cette application, ensuite la commande du robot a été faite en utilisant un module XBee associé à une télécommande.

Le premier chapitre de ce mémoire a été consacré sur les généralités de la robotique et ses principes de base.

Dans le second chapitre nous avons fait une description détaillée des différents étages constituant notre système, commençant par la carte pcDuino et les différents capteurs connectés à cette carte, ensuite on a donné un aperçu sur les cartes Arduino Uno et Arduino Nano ainsi que les modules XBee placés sur la télécommande et en fin par une petite description des différents logiciels utilisés.

La dernière partie de ce mémoire est consacrée pour décrire les différentes étapes de la configuration logiciels et de la réalisation pratique avec illustration des schémas de branchement des différents capteurs avec les cartes, ainsi que les résultats des tests effectués avec notre système.

Nous souhaitons que ce modeste travail servira d'avantage pour les promotions qui viennent et de leur donner une image réelle sur l'importance de la partie pratique de l'électronique.

En réalisant notre projet, nous avons pris conscience de ses imperfections et avons donc pensé à des perspectives d'améliorations et à d'autres idées permettant de le compléter. Par exemple, ajouter une pince solide fonctionnant à l'aide des servomoteurs, sur laquelle nous aurions implanté une caméra supplémentaire pour plus de précision. On pourra aussi ajouter un capteur de lumière LDR pour allumer automatiquement l'éclairage du robot dans des lieux sombres.

# Bibliographie

## Bibliographie

- [1] Nicole Merle-Lamoot, Gilles Pannetier, Le développement industriel futur de la robotique personnelle et de service en France, 2012.
- [2] Brandt, Mark Van Den, US Technological Innovation Systems for Service Robotics, 2010.
- [3] [http://www.memoireonline.com/10/08/1600/m\\_mise-en-oeuvre-partie-perception-d-un-robot1.html](http://www.memoireonline.com/10/08/1600/m_mise-en-oeuvre-partie-perception-d-un-robot1.html)
- [4] Stéphane Lens « locomotion d'un robot mobile » mémoire de fin d'études à l'université de Liège, faculté des sciences appliquées, institut Montefiore, MAI 2008
- [5] B. Bayle, « Robotique Mobile », Ecole Nationale Supérieure de Physique de Strasbourg, Université Louis Pasteur, 2007.
- [6] Boimond Jean-Louis, Cours-Robotique.
- [7] Hocine TAKHI, « Conception et réalisation d'un robot mobile à base d'arduino », Mémoire de fin d'étude, Université Amar Telidji, 2014.
- [8] <http://www.linksprite.com/linksprite-pcduino3/>
- [9] [http://www.mon-club-elec.fr/pmwiki\\_mon\\_club\\_elec/pmwiki.php?n=MAIN.PCDUINO](http://www.mon-club-elec.fr/pmwiki_mon_club_elec/pmwiki.php?n=MAIN.PCDUINO)
- [10] Xavier HINAULT, Ateliers Pyduino, 2013
- [11] Simon Landrault (Eskimon), Hippolyte Weisslinger (olyte); Arduino : Premiers pas en informatique embarquée Édition du 19 juin 2014
- [12] <http://letmeknow.fr/shop/arduino/14-arduino-nano-v30-4894479490198.html>
- [13] <https://www.sparkfun.com/products/10336>
- [14] Jules Sery - Jordan Vocisano – Florian Bekaert – Cyril Gervais – Leo Abraham, Le Bras Robotisé, PPE 2012
- [15] Aina Andriamampianina RANDRIANARISAINA, « MODÉLISATION DE LA CONSOMMATION D'ÉNERGIE EN VUE DE LA CONCEPTION CONJOINTE (MATÉRIEL/LOGICIEL) DES APPLICATIONS EMBARQUÉES. APPLICATION AUX RÉSEAUX DE CAPTEURS SANS FIL (WSN) », thèse pour obtention de doctorat de l'université de Nantes, Soutenue le 6 février 2015
- [16] <http://www.instructables.com/id/Arduino-Wireless-SD-Shield-Tutorial/>
- [17] <http://arduino103.blogspot.com/2011/06/detecteur-de-proximite-infrarouge-sharp.html>
- [18] T.Karvinen, K.Karvinen, V.Valtokari « Les capteurs pour Arduino et Raspberry Pi » DUNOD, 2014.

## Bibliographie

[19] YACINE ABDSSALEMAMKASSOU, « THERMOMETR A BASE DU ARDUINO », mini projet à l'université Hassan 1<sup>er</sup> Settat école nationale des sciences appliquées Khouribga, DR.N.ELBARBRI, 2013.

[20][http://fr.aliexpress.com/store/product/Flame-Sensor-Fire-Detection-Module-For-Arduino-Learner-Projects/718232\\_32287182034.html#extend](http://fr.aliexpress.com/store/product/Flame-Sensor-Fire-Detection-Module-For-Arduino-Learner-Projects/718232_32287182034.html#extend)

[21][http://www.petervis.com/Raspberry\\_PI/Breadboard\\_Power\\_Supply/YwRobot\\_Breadboard\\_Power\\_Supply.html](http://www.petervis.com/Raspberry_PI/Breadboard_Power_Supply/YwRobot_Breadboard_Power_Supply.html)

[22] [https://www.google.fr/?gws\\_rd=ssl#q=Webcam+PC+810](https://www.google.fr/?gws_rd=ssl#q=Webcam+PC+810)

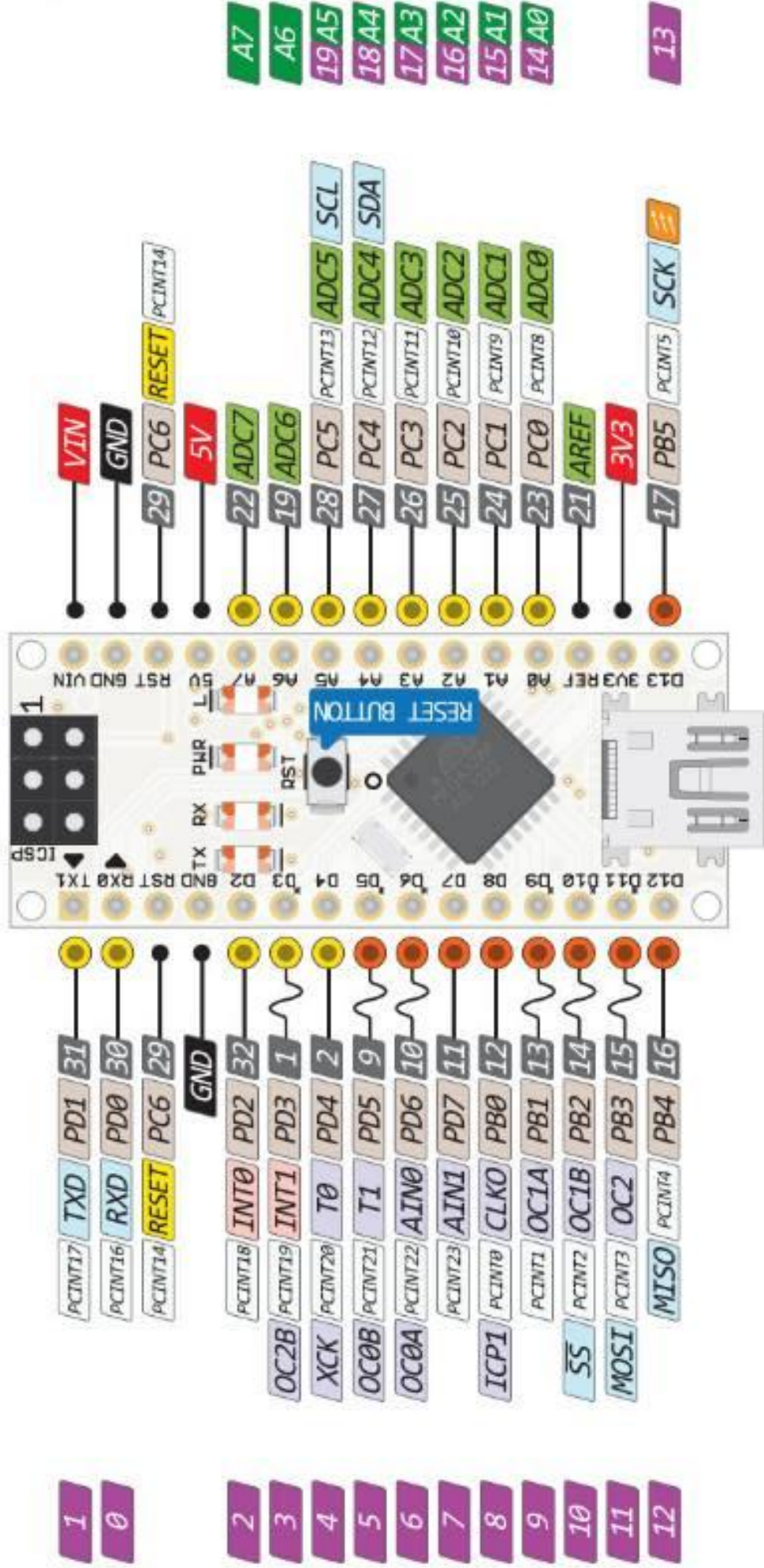
[23] Guillermo de Jesús HOYOS RIVERA, « Conception et mise en œuvre d'un outil pour la navigation coopérative sur le web », thèse pour l'obtention du Doctorat de l'Université Paul Sabatier, spécialité informatique, Soutenue le 10 juin 2005

[24][http://www.mon-club-elec.fr/pmwiki\\_reference\\_pyduino/pmwiki.php](http://www.mon-club-elec.fr/pmwiki_reference_pyduino/pmwiki.php)

[25] <http://fab.fritzing.org/fritzing-fab>

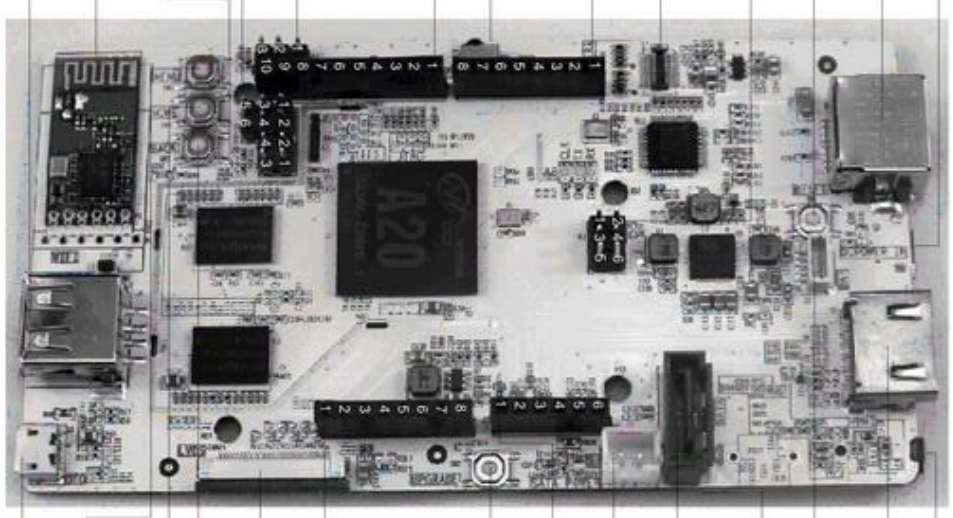
# Annexe





J11	
1	GPIO0 / UART2 RX
2	GPIO1 / UART2 TX
3	GPIO2
4	GPIO3 / PWM3
5	GPIO4
6	GPIO5 / PWM5
7	GPIO6 / PWM6
8	GPIO7
<b>J8</b>	
1	GPIO8
2	GPIO9 / PWM9
3	SPI_SS (Subordinate / Chip Select) / GPIO10 / PWM10
4	SPI0_MOSI (Master Output, Subordinate Input) / GPIO11 / PWM11
5	SPI0_MISO (Master Input, Subordinate Output) / GPIO12
6	SPI0_CLK (Clock) / GPIO13
7	GND
8	AREF
9	TW12_SDA (Data)
10	TW12_SCL (Clock)
<b>P6</b>	
1	SPI1_MISO (Master Input, Subordinate Output)
2	+5V DC
3	SPI1_CLK (Clock)
4	SPI1_MOSI (Master Output, Subordinate Input)
5	RESET
6	GND
<b>P7</b>	
1	SPI0_MISO (Master Input, Subordinate Output)
2	+5V DC
3	SPI0_CLK (Clock)
4	SPI0_MOSI (Master Output, Subordinate Input)
5	RESET
6	GND
<b>P3 (Debug Port)</b>	
1	TX (Square Pad)
2	GND
3	RX
<b>P10</b>	
1	GPIO14
2	GPIO15
3	GPIO16
4	GPIO17

<b>J1</b>	DC Power IN
<b>J13</b>	Network RJ45
<b>SW1</b>	Reset
<b>P7</b>	SPI0 Port
<b>CON15</b>	Camera
<b>J11</b>	Digital I/O
<b>U16</b>	IR
<b>J8</b>	Digital I/O
<b>P3</b>	Debug Port
<b>P6</b>	SPI1 Port
<b>P10</b>	GPIO
<b>P2</b>	Wireless Connection Module Interface
<b>J4</b>	Host USB(USB2.0)



<b>CN8</b>	Audio
<b>J2</b>	HDMI
<b>J7</b>	Battery Connector
<b>J5</b>	TF_card(bottom)
<b>P5</b>	Sata
<b>P4</b>	Sata Power
<b>J12</b>	Analog Input
<b>SW2</b>	Upgrade (not currently implemented ; reserved for future expansion)
<b>J9</b>	
1	(NO)
2	IOREF
3	RESET
4	3.3V DC output
5	5V DC output
6	GND
7	GND
8	+5V <sub>IN</sub>
<b>J3</b>	
1	OTG

