الجمهورية الجزائرية الديمقراطية الشعبية

UNIVERSITE YAHIA FARES DE MEDEA

FACULTE DES SCIENCES ET DE LA TECHNOLOGIE

Département de Génie Mécanique

Projet de fin d'études

Présenté pour l'obtention du Diplôme De Master en Génie Mécanique

Spécialité : Mécatronique

THEME

Localisation et guidage GPS d'un robot mobile dans une serre agricole

Proposé et dirigé par

Encadreur : Dr KARA Redha

Préparé et présenté par

BELKHIRAT Ayoub

HADJIDJ Amira

Année Universitaire 2019-2020

Remercîments

En préambule à ce mémoire,

Nous remercions Dieu tout puissant, Maitre des cieux et des terres, qui nous a donné la santé, la patience et la volonté afin de terminer notre étude et d'accomplir ce modeste travail.

Nos remerciements les plus sincères à notre promoteur Mr Kara Redha qui

Nous a fait l'honneur d'avoir veillé et dirigé ce labeur par son aide scientifique,

Ses précieux conseils et son aide pratique et pédagogique.

Nous exprimons également notre gratitude au corps professionnel et administratif de l'université et surtout nos enseignants qui ont collaboré à notre formation depuis notre premier cycle d'études jusqu'à la fin.

Nos plus vifs remerciements sont destinés aussi aux membres de jury d'avoir accepté d'examiner et d'évaluer notre travail.

Un grand remercîment aussi à Noureddine, Adel et Sohaib qui nous ont aidés d'élaborer et organiser ce mémoire

Dans l'impossibilité de citer tous les noms, nous adressons nos plus sincères remerciements à toute personne qui a contribué de près ou de loin à la réalisation de ce travail.

Dédicaces

Je dédie ce modeste travail :

À la souveraine de mon cœur ma Chère Mère

En vous, je vois la maman parfaite, toujours prête à se sacrifier pour le bonheur de ses enfants.

A Mon cher Père

En vous, je vois un père dévoué à sa famille. Votre présence en toute Circonstance m'a maintes fois rappelé le sens de la responsabilité.

A mes frères et mes sœurs

Qui je le sais ma réussite est très importante pour vous. Que Dieu vous paye pour tous
vos bienfaits.

A toute la famille de pré et de loin sans exception.

A tous mes camarades de promotion, cher binôme et à mes valeureux enseignants

De vous j'acquière la volonté et la persévérance d'étudier et de progresser.

A mes amis

Et spécialement à mes très chers amis

ملخص:

يهدف هذا المشروع الى التحكم في الروبوت باستعمال نضام تحديد المواقع، من اجل تحقيقه قمنا باستغلال ودمج الأجهزة الالكترونية اردوينو و راسبيري مع مستشعر تحديد المواقع لنتمكن من توجيه الروبوت عبر الاحداثيات المستقبلة.

الكلمات المفتاحية:

روبوت، نضام تحديد المواقع، اردوينو، مستشعر.

Résumé :

L’objectif de ce projet est de guider un robot mobile en utilisant un système GPS.

Pour cela nous avons exploité et intégré un Arduino, une carte Raspberry avec un capteur GPS, ce dernier nous permet de manipuler le robot a travers les cordonnées GPS.

Mots clés :

Robot mobile, système GPS Arduino, Raspberry, capteur GPS.

Abstract :

The objective of this project is to guide a mobile robot using a GPS system.

For this we have used and integrated an Arduino, a Raspberry board with a GPS sensor, to manipulate the robot using the GPS coordinates.

Keywords :

Mobile robot, GPS system, Arduino, Raspberry, sensor

SOMAIRE

I. CHAPITRE I : Généralité sur les robots mobiles 14
1. Introduction : 14
2. Historique : 15
3. Les bases mobiles 16
3.1. Holonomie 16
3.2. Les plates-formes différentielles 16
3.3. Les plates-formes omnidirectionnelles 17
3.4. Les plates-formes non holonomes 17
4. Types des robots 18
4.1. Robots manipulateurs 18
4.1.1. La structure mécanique et la motricité 19
4.1.2. Domaines d'application 20
4.2. Robots mobiles 21
4.2.1. Classification 22
4.2.2. Architecture des robots mobiles 22
5. Les robots à roues 23
5.1. Robot unicycle 23
5.2. Robot tricycle 24
5.3. Robot voiture 24
5.4. Robot omnidirectionnel 24
6. Capteurs en robotique mobile 25
7. La navigation 27
8. La modélisation de l'environnement 28
9. La planification de trajectoire 29
10. Méthodes de navigation 29
10.1. Navigation métrique : 29
10.2. Navigation géométrique : 30
10.3. Navigation topologique : 30
10.4. Navigation par action associé à un lieu : 30
10.5. Navigation par guidage : 31
10.6. Navigation par approche d'un objet : 31
11. Conclusion 31

II. CHAPITRE II Les système embarqués utilisés 34
1. Introduction 34
2. Arduino 34
2.1. Définition 34
2.2. Les applications 34
2.3. Les atouts majeurs d'un système à carte Arduino 35
2.4. Les avantages d'Arduino 35
2.5. Arduino Uno 36
2.6. Présentation 36
2.7. Choix d'Arduino 41
2.8. Le logiciel de gestion de la carte Arduino 42
3. La carte Raspberry pi 44
3.1. Pourquoi Pi 3 B ? 44
3.2. La carte mère Raspberry Pi 3 Type B 45
3.3. Les composantes du Raspberry Pi 45
3.4. Programmation 48
3.5. Système d'exploitation 48
4. L'extension (Shield) de commande des moteurs 49
5. Conclusion 51
DESCRIPTION DU 52
SYSTEME GPS 52
III. CHAPPITRE III Description du système GPS 53
1. Introduction 53
2. Comment fonction la géolocalisation : 53
3. Description du système 54
3.1. Le secteur spatial 55
3.1.1. Les différents types de satellites 55
3.1.2. Identification des satellites 60
3.1.3. Le signal satellite 60
3.1.4. L'échelle de temps GPS 62
3.2. Le secteur de contrôle 62
3.3. Le secteur utilisateur 64
3.3.1. Mode de fonctionnement GPS 64
4. Les mesures GPS 66

4.1. La pseudo-distance 67
4.2. Mesure de phase : 68
5. Positionnement statique et positionnement cinématique 69
6. Sources d'erreur 70
7. Conclusion 73
IV. CHAPPITRE IV DESCRIPTION DE LA PLATEFORME ROBOT MOBILE 75
1. Introduction 75
2. Robot mobile Arduino 76
2.1. Description 76
2.2. Modélisation cinématique du robot mobile 77
3. L'architecture matérielle de gestion et de commande de la plateforme robot mobile 79
3.1. L'extension (Shield) de commande des moteurs 79
3.2. Le capteur GPS 82
3.2.1. Les différents types de GPS 82
3.2.2. Principes de fonctionnement et d'architecture de récepteur GPS 83
3.2.3. MODULE GPS utilisé (SIM 28) : 84
3.2.4. Caractéristiques 84
3.2.5. Schéma de câblage 84
3.2.6. Résultats moniteur série 85
3.2.7. Vérification du résultat sur « Google Maps » 86
4. La carte Raspberry Pi : le cerveau 87
5. L'organigramme du contrôleur de robot mobile 88
6. Description du programme d'asservissement par navigation : 89
7. Résultat 90
8. Conclusion 93

LISTE DES FIGURES

Figure 1 : La tortue de Grey Walter. 15
Figure 2 : Exemple de plate-forme différentielle. Pioneer 2 DX de la société Mobile Robots. 16
Figure 3 : Exemple de plate-forme omnidirectionnelle 17
Figure 4 : Exemple de plate-forme non holonome de type Ackerman. 18
Figure 5 : structure générale d'un robot industriel. [3] 19
Figure 6 : Architecture d'un robot mobile. 23
Figure 7 : Robot de type unicycle. 23
Figure 8 : Robot de type tricycle. 24
Figure 9 : Robot de type voiture. 24
Figure 10 : Robot de type omnidirectionnel Comparaison des différents types. 25
Figure 11 : Navigation de robot mobile en environnement encombré 28
Figure 12 : Carte Arduino UNO 36
Figure 13 : Les compositions de la carte arduino UNO 37
Figure 14 : Structure de la carte ArduinoUno. 40
Figure 15 : logiciel Arduino. 42
Figure 16 : Présentation des boutons. 43
Figure 17 : Raspberry pi 3 B 44
Figure 18 : les composants de raspberry pi 45
Figure 19 : Adafruit Motor Shield 49
Figure 20 : Description Shield l293D 50
Figure 21 : Principe de la triangulation pour géolocaliser un récepteur a la surface du globe 54
Figure 22 : Les trois segments du GPS 54
Figure 1 : Constellation des satellites du GPS.....55

Figure 24 : satellite bloc IIIA en orbite 59
Figure 25 : satellite Navstar-2F 59
Figure 26 : Structure du signal GPS 61
Figure 27 : Structure du signal GPSD Deux frequence porteuse L1 et L2 modulé sur des codes C/A des codes P et des codes de navigation.....61
Figure 28 : Dislocation géographique mondiale de la station contrôle 63
Figure 29 : Équipement GPS 64
Figure 30 : positionnement absolu 65
Figure 31 : positionnement relatif 66
Figure 32 : Comparaison entre le signal reçu et le signal propre du récepteur sur les codes 67
Figure 33 : Comparaison entre le signal reçu et le signal du récepteur sur la porteuse 68
Figure 34 : Positionnements statique et cinématique 69
Figure 35 : Principales sources d'erreur impactant la précision du calcul de la géolocalisation 70
Figure 36 : Schéma bloc du projet. 75
Figure 37 : Robot mobile de type tricycle 76
Figure 38 : Paramétrage du robot mobile 77
Figure 39 : Shield contrôle de moteurs L293D 79
Figure 40 : fonctionnement de L293D. 80

Figure 41 : montage L293D avec Shield..... 81
Figure 42 : IC 74HCT595N 81
Figure 43 : fonctionnement de IC 74HCT595N..... 82
Figure 44 : Schéma de fonctionnement des récepteurs GPS..... 83
Figure 45 : Capteur GPS SIM 28 84
Figure 46 : Schéma de câblage..... 85
Figure 47 : Résultats moniteur série..... 85
Figure 48 : Résultats sur « Google Maps 86
Figure 49 : la carte Raspberry Pi utilisé. 87
Figure 50 : L'organigramme du contrôleur de robot mobile 88
Figure 51 : simulation dc moteur Shield.....90

Figure 51 : Synoptique du système d'alimentation en pont des moteurs du robot.....90

Figure 52 : graphe de la tension d'alimentation du moteur en fonction de PWM……91

Figure 53 : : graphe du courant d'excitation du moteur en fonction de PWM……..……92

LISTE DES TABLEAUX

Tableau 1 :Les avantages et des inconvénients des différents types de robots à roues. 25
Tableau 2 : Les différents éléments de la carte de commande. 41
Tableau 3 : les déférentes erreurs GPS..... 72
Tableau 4 : brochage d'IC 74HCT595N 82
Tableau 5 : Les caractéristiques du capteur GPS SIM28 84
Tableau 6 : Le sens de rotation des moteurs Shield l293d.....91

LISTE DES ABREVIATIONS

GPS : GLOBAL POSITIONING SYSTEM

3D : signifie trois dimensions. Ces dimensions permettent de définir un objet possédant trois axes x, y et z

W : Le watt est l'unité internationale de puissance

Mhz : mégahertz unité de mesure de fréquence

UHF : Ultra Haut Frequency

PRN : Pseudo Random Noise

C/A : Coarse Acquisition code

SPS : Standard Positioning Service

P-code : précise positioning code

PPS : Précise Positioning Service

UTC : Universal Time Coordinats.

MC : master clock

USNO : United States Naval Observatory

SA : selective availability

AS : Authorized Service

PVT : position, vitesse, temps

DGPS : GPS différentiel

DOP : Dilution of précision

x_r, y_r, z_r **:** Coordonnées du récepteur

x_s, y_s, z_s **:** Coordonnées du satellite

As : anti spoofing

r : rayent de roue

v : Vitesse de véhicule

l : distance entre les roues.

CCD : Charge Coupled Device

Pc : personale computer

LED : light-emitting diode

PWM : Modulation de la largeur des pics

USB : Universal Serial Bus

ICSP : In-Circuit Serial Programming ou (Programmation Série Dans le circuit

GND : Ground (la masse)

VIN : Voltage Input

DC : Direct Current.

IDE : Integrated Development Environment.

SPI : Serial Peripheral Interface

TX : émission data

RX : Réception data

PWM : Pulse Width Modulation (eng)

AREF : Analogique Référence

CPU : Central processing unit.

RAM : Random Access Memory (mémoire vive).

WIFI : Wireless Internet for Frequent Interface

SD : Secure Digita

HDMI : High Definition Multimedia Interface

GPIO : (General Purpose Input/Output)

DSI : Display Serial Interface

CSI : Camera Serial Interfac

Introduction générale

La robotique permet d'aider l'homme dans les tâches difficiles, répétitives ou Pénibles, de plus elle constitue le rêve de substituer la machine à l'homme dans ces tâches, les facultés de perception et de raisonnement des robots progressent chaque jour actuellement et plus encore dans l'avenir, ils sont appelés à jouer un rôle de plus en plus important dans notre vie.

La robotique comporte deux grands pôles d'intérêt : la robotique de manipulation (robotique industrielle) et la robotique mobile. Un des problèmes majeurs de la robotique mobile est la planification de mouvement. Autour de ce problème de planification de mouvement de nombreuses études ont été réalisées dans le but de développer des méthodes générales pour guider les robots.

Dans ce contexte, notre travail s'articule autour de l'objectif majeur qui est la réalisation de la localisation et le guidage d'un robot mobile en utilisant une carte Arduino et une carte Raspberry.

Ce document est organisé en quatre chapitres qui portent sur les aspects suivants :

- Le premier chapitre présente une introduction à la robotique mobile, des descriptions, des différents éléments constituants les robots mobiles ainsi que les différentes structures mécaniques existantes ;
- Le deuxième chapitre est consacré à la description des systèmes embarqués utilisés dans notre projet, la carte Raspberry Pi3, la carte Shield L293D et la carte Arduino ;
- Le troisième chapitre est consacré à la présentation du système GPS, ses principales caractéristique et son fonctionnement ;
- Le quatrième chapitre sera consacré au détail de notre réalisation.

  Nous présenterons l'agencement et le fonctionnement de l'architecture matérielle que nous avons mise on ouvre. Par ailleurs, nous validerons a partir d'expérimentation de navigation, notre algorithme de commande base sur l'exploitation de donnes de géolocalisation

  Enfin, la conclusion nous permettra de résumer l'ensemble des travaux réalises.

  Nous terminerons ce mémoire par quelques perspectives envisageables dans un futur proche ou plus lointain.

CHAPITRE I :

GENERALITE

SUR LES ROBOTS MOBILES

I. 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... Les robots mobiles ne sont pas fixes, ils sont classifiés selon la locomotion en robots marcheurs, à roues, à chenilles... comme ils peuvent être classifié selon le domaine d'application en robots militaires, de laboratoire, industriels et de services.

Les robots mobiles sont largement utilisés dans les environnements industriels, le plus souvent pour des tâches répétitives en suivant un chemin bien défini matérialisé parfois par des lignes sur le sol ou par l'utilisation d'amers artificiels. Cependant actuellement il y'a une forte tendance à élargir les milieux où évoluent les robots à des environnements domestiques. Les types d'applications possibles sont innombrables. Cela peut aller à des tâches de nettoyage et d'entretien, à une assistance à une personne handicapée dans des tâches d'exploration et de préhension. On parle alors, de façon générale, de robotique d'intérieur.

Un tel cadre d'utilisation requiert que le système robotisé dispose d'un niveau minimum d'autonomie et de facilités de navigation. Pour cela, le système doit obligatoirement accomplir trois tâches de base qui sont la localisation, la planification et la navigation. Il faut toutefois noter que les deux dernières tâches sont tributaires de la bonne exécution de la première.

En effet la localisation est l'une des fonctions essentielles qui permet aux robots mobiles de se mouvoir en respectant les règles élémentaires de sécurité et d'évoluer, de façon générale, vers une autonomie totale. C'est pourquoi, depuis l'origine des travaux entrepris en matière de robotique mobile, la localisation de l'engin dans son environnement a constitué une voie de recherche privilégiée qui conditionne les résultats expérimentaux et les applications en milieu industriel et/ou domestique. Elle consiste à calculer et à maintenir à jour la connaissance de la position et de l'orientation du robot dans un repère absolu lié à l'environnement. On se restreint dans notre travail au cas de systèmes navigant sur un plan.

Actuellement la localisation des robots mobiles est un thème de recherche ouvert puisque aucune méthode globale n'est susceptible de générer des algorithmes suffisamment robustes, rapides et fiables pour être appliqués à tous types de problèmes

2. Historique :

Ce n'est qu'au tout début du XXe siècle que les robots firent leur apparition, suite aux travaux d'ingénieurs qui voulaient tester des hypothèses émises par des biologistes et des psychologues. Le chien électrique conçu par Hammond et Miessner en 1915 était attiré par une lumière, selon le phototropisme animal mis en évidence par Loeb en 1918. Les machines de Russell (1913) et de Stephens (1929), les tortues cybernétiques de Grey Walter (Figure 1) (1950), le renard électronique de Ducrocq (1953) ou l'homéostat d'Ashby (1952) étaient, elles, dotées de capacités d'apprentissage directement issues des travaux des psychologues Thorndike (1911), Hull (1943) et du physiologiste Pavlov (1903) sur l'Homme et l'animal.

Ces réalisations sont des robots, car elles ne se comportent plus comme de simples automates dont les organes moteurs - leurs mécanismes - obéissent à un programme préétabli.

À la différence des automates, ils ont des organes sensoriels - les capteurs - qui recueillent des informations de l'environnement qui vont, elles, influencer l'activité de leurs organes moteurs les actionneurs. **[1]**



Figure 2 : La tortue de Grey Walter.

3. Les bases mobiles

Nous présentons rapidement les différents types de bases mobiles utilisées en robotique, en nous focalisant sur les plateformes mobiles terrestres pour le milieu intérieur.

3.1.Holonomie

En robotique, une plateforme est dite holonome lorsque que le nombre de degrés de libertés contrôlables est égal au nombre total de degrés de liberté. Pour un robot se déplaçant sur un plan, il y a 3 degrés de liberté (deux translations et une rotation).

A partir d'une position donnée, une plateforme holonome devra donc pouvoir se déplacer en avant, sur le côté et tourner sur elle-même. Cette capacité permet de contrôler très simplement le robot car tous les déplacements imaginables sont réalisables, ce qui simplifie le problème de planification de trajectoire. Nous allons cependant voir quelques mécanismes permettant d'obtenir des plateformes holonomes, ou s'en approchant. **[2]**

3.2.Les plates-formes différentielles

Une des configurations les plus utilisées pour les robots mobiles d'intérieur est la configuration différentielle qui comporte deux roues commandées indépendamment. Une ou plusieurs roues folles sont ajoutées à l'avant ou à l'arrière du robot pour assurer sa stabilité (Figure 2). Cette plate-forme est très simple à commander, puisqu'il suffit de spécifier les vitesses des deux roues, et permet de plus au robot de tourner sur place. Cette possibilité permet de traiter dans certains cas le robot comme un robot holonome, ce qui va simplifier la planification de déplacement et la commande du robot. **[2]**



Figure 3 : Exemple de plate-forme différentielle. Pioneer 2 DX de la société Mobile Robots.

Urban Robot de la société iRobot**.[2]** L'estimation du déplacement par odométrie est également très simple à partir de la mesure des vitesses de rotation des deux roues w1

et w2. Les vitesses de translation v et de rotation w sont en effet données par : $v = \frac{\omega_1 r}{2}$

$\omega = \frac{\omega_1 r - \omega_2 r}{2l}$

3.3.Les plates-formes omnidirectionnelles

Les plates-formes omnidirectionnelles permettent de découpler de manière encore plus nette le contrôle de la rotation et de la translation d'un robot et en font une plateforme réellement holonome. Elles utilisent pour cela 3 ou quatre roues qui tournent à la même vitesse pour fournir une translation et un mécanisme qui permet d'orienter simultanément ces roues dans la direction du déplacement souhaitée.

Le corps du robot lui-même n'effectue pas de rotation mais uniquement des translations. Ce système permet un contrôle très simple et relativement rapide car les changements de direction ne concernent que les roues et peuvent donc se faire très vite. Par contre ces plates-formes sont relativement limitées en capacité de franchissement et requièrent un sol très plan.

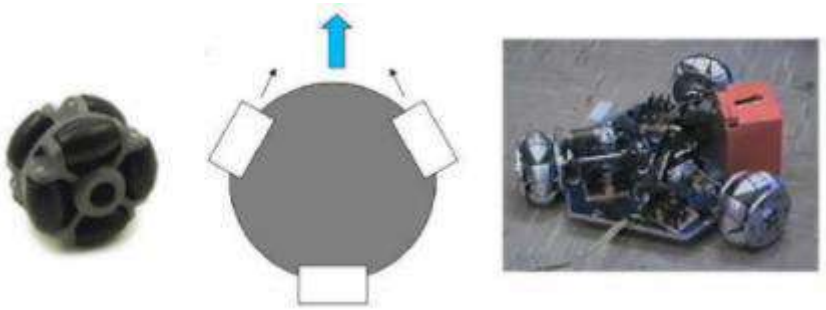


Figure 4 : Exemple de plate-forme omnidirectionnelle

3.4.Les plates-formes non holonomes

Des plates-formes non holonomes, telles que les voitures, sont également utilisées en robotique mobile (Figure 4). Ces plates-formes sont toutefois plus difficiles à commander car elles ne peuvent pas tourner sur place et doivent manœuvrer, ce qui peut être difficile dans des environnements encombrés.

La commande de ces plates-formes pour réaliser un déplacement particulier est un problème à part entière que nous n'aborderons pas dans ce cours. Par contre, il est possible

de prendre en compte ces contraintes de manière relativement simple dans la planification. **[2]**

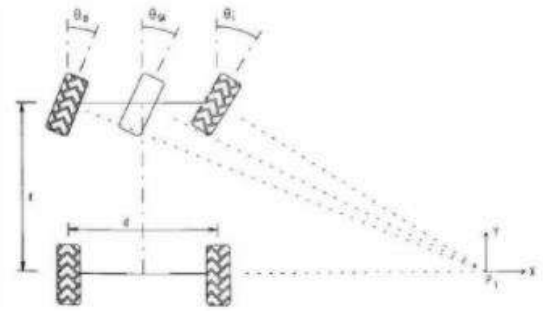


Figure 5 : Exemple de plate-forme non holonome de type Ackerman.

4. Types des robots

Au cours de l'histoire, 2 types différents de robots sont apparus attestant chacun de nouvelles évolutions : les robots manipulateurs et les robots mobiles.

4.1. Robots manipulateurs

Un robot manipulateur, quelle que soit la fonction qui lui est attribuée (transfert d'objet, soudage, assemblage), est un mécanisme capable de déplacer et de situer un objet appelé «organe terminal» ou «point outil» dans une partie de l'espace appeler « volume de travail ».

Un robot industriel est un manipulateur à plusieurs degrés de liberté contrôlé automatiquement, reprogrammable et multitâche qui peut être fixe ou mobile pour une application en automatisation industrielle.

Le robot industriel se compose d'une structure mécanique animée par des actionneurs, à partir d'ordres élaborés par un calculateur. Ces ordres dépendent des informations délivrées par les capteurs. L'utilisation de capteurs externes, capteurs'' extéroceptifs'', pour évaluer et mesurer l'interaction du robot avec l'environnement directement depuis son organe terminal devient une pratique de plus en plus courante dans les applications robotiques de haute précision**.[3]**

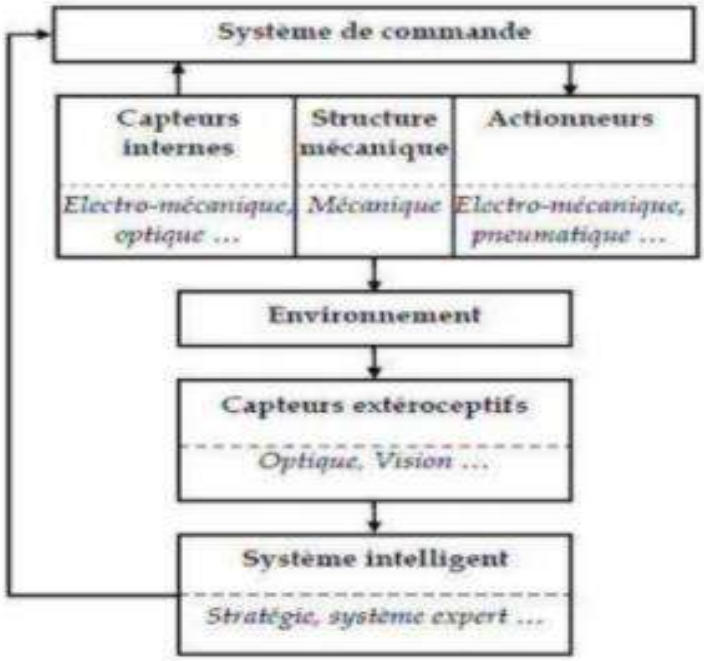


Figure 6 : structure générale d'un robot industriel. [3]

4.1.1. La structure mécanique et la motricité

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

A -Les mobiles à roues :

La mobilité par roues est la structure mécanique la plus communément appliquée. Cette technique assure selon l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importante. Le franchissement d'obstacles ou l'escalade de marches d'escalier est possible dans une certaine mesure. Toutes les configurations (nombre, agencement, fonction) des roues sont appliquées.

B- Les mobiles 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 (présence de boue, herbe…).

C- Les mobiles marcheurs :

Les robots mobiles marcheurs sont destinés à réaliser des tâches variées dont l'accès au site est difficile, dangereux ou impossible à l'homme. Leur anatomie à nombreux degrés de liberté permet un rapprochement avec les robots manipulateurs. La locomotion est commandée en termes de coordonnées articulaires. Les méthodes de commande des articulations définissent le concept d'allure qui assure le déplacement stable de l'ensemble. Les différentes techniques étudiées se rapprochent de la marche des animaux et notamment de celle des insectes**. [4]**

L'adaptation au support est un problème spécifique aux marcheurs. Il consiste à choisir le meilleur emplacement de contact en alliant l'avance et la stabilité avec l'aide de capteurs de proximité, de contact ou de vision.

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.

4.1.2. Domaines d'application

La robotique est un domaine en plein essor depuis quelques années. Les évolutions technologiques, dépassant sans cesse nos espérances, permettent maintenant de réaliser des solutions technologiques s'adaptant au moindre problème.

Par conséquent, la robotique est utilisée dans des domaines extrêmement rigoureux et exigeants. Nous allons explorer ces différents domaines. **[3]**

*** L'industrie**. Le but premier des robots est de remplacer l'homme dans des activités fastidieuses ou onéreuses pour l'employeur. Les robots ont donc commencé à être utilisés dans les chaînes d'assemblage industrielles. Dans ces chaînes d'assemblage, on retrouve des robots soudeurs, manipulateurs, peintres ...

*** Le domaine militaire**. Les robots sont de plus en plus utilisés dans le domaine militaire. En effet, la miniaturisation permet aujourd'hui de créer des robots discrets mais dotés de nombreux capteurs, ce qui est idéal pour des missions d'espionnage ou d'éclairement.

*** La santé**. Les robots commencent à être de plus en plus utilises dans le domaine médical, qu'il s'agisse de « simples » échographies ou d'opérations chirurgicales plus délicates. En fait ces robots ne sont pas complètement autonomes mais ils assistent les médecins ou chirurgiens, jusqu'à permettre des opérations médicales à distance (télémédecine). On parle de surgétique (mot né de l'anglais « surgyré » : chirurgie) c'est -à-dire tout ce qui consiste à introduire les derniers outils des technologies informatiques et robotiques dans la pratique médico-chirurgicale. Cette pratique de « chirurgie assistée » est émergente donc bien que peu répandue, elle est en phase de devenir la chirurgie du futur.

4.2.Robots mobiles

Les robots mobiles ont une place particulière en robotique. Leur intérêt réside dans leur mobilité qui ouvre des applications dans de nombreux domaines. Comme les robots manipulateurs, ils sont destinés à assister l'homme dans les tâches pénibles (transport de charges lourdes), monotones ou en ambiance hostile (nucléaire, marine, spatiale, lutte contre l'incendie, surveillance...).

L'aspect particulier de la mobilité impose une complexité technologique et méthodologique qui s'ajoute en général aux problèmes rencontrés par les robots manipulateurs. La résolution de ces problèmes passe par l'emploi de toutes les ressources disponibles tant au niveau technologique (capteurs, motricité, énergie) qu'à celui du traitement des informations par l'utilisation des techniques de l'intelligence artificielle ou de processeurs particuliers (vectoriel, cellulaires**}.[3]**

4.2.1. Classification

Une classification est proposée dans la littérature qui définit le degré d'autonomie du robot mobile.

- *Véhicule télécommandé par un opérateur qui lui impose chaque tâche élémentaire à réaliser.
- Véhicule télécommandé au sens de la tâche à réaliser. Le véhicule contrôle automatiquement ses actions.
- Véhicule semi-autonome réalisant sans l'aide de l'opérateur des tâches prédéfinies.
- Véhicule autonome qui réalise des tâches semi-définies. Ce type de véhicule
- Pose des problèmes d'un niveau de complexité élevé de représentation des connaissances, de capacité décisionnelle et de génération de plans qui sont résolus à bord dans la mesure du possible.
- L'ensemble des problèmes particuliers liés à la conception de tels robots sont :
- La conception mécanique liée à la mobilité
- La détermination de la position et de la latitude (orientation)
- La détermination du chemin optimal pour atteindre le lieu de la tâche.

4.2.2. 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

L'architecture des robots mobiles est représentée sur la figure 6.

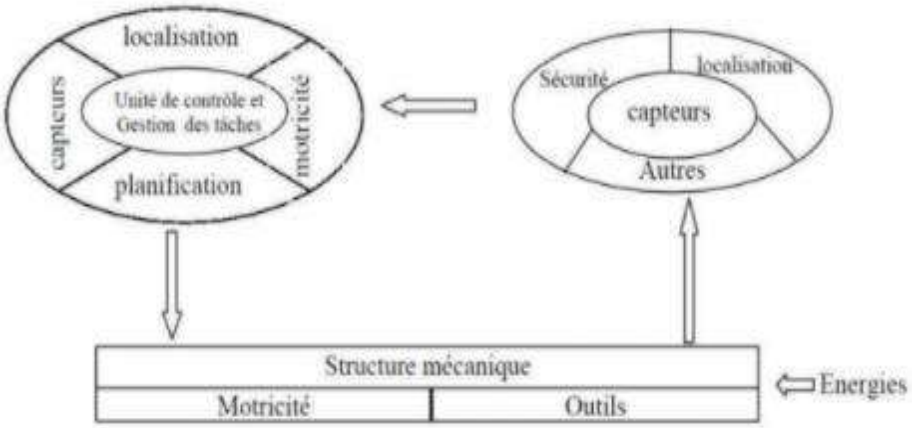


Figure 7 : Architecture d'un robot mobile.

5. Les robots à roues

Il existe 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.

5.1.Robot unicycle

Un robot mobile est dit unicycle s'il est actionné par deux roues indépendantes et qu'il possède un certain nombre de roues folles qui assurent la stabilité. La figure (7) montre le schéma d'un robot de type unicycle. On y a omis les roues folles, qui n'interviennent pas dans la cinématique, dans la mesure où elles ont été judicieusement placées. Ce type de robot est très répandu en raison de sa simplicité de construction et de propriétés cinématiques intéressantes.

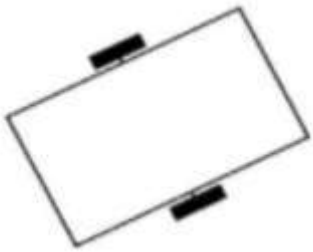


Figure 8 : Robot de type unicycle.

5.2.Robot tricycle

Le robot à configuration tricycle illustré dans la figure (8), se compose de deux roues passives arrière et une roue motrice avant et vice versa.

Le mouvement du robot conféré par deux actions : la vitesse longitudinale et l'orientation de la roue orientable. Pour la localisation de ce type de robot, un capteur d'orientation est associé à la roue orientable, et deux codeurs sont associés aux roues motrices.

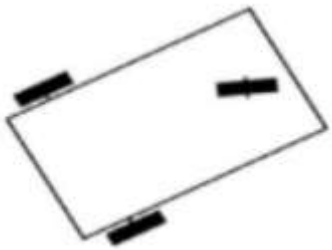


Figure 9 : Robot de type tricycle.

5.3.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. 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.



Figure 10 : Robot de type voiture.

5.4.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 puisqu'il peut se déplacer dans toutes les directions. Mais ceci se fait au dépend d'une complexité mécanique bien plus grand. **[5]**



Figure 11 : Robot de type omnidirectionnel Comparaison des différents types.

.

TYPE DE ROBOT	AVANTAGES	INCONVENIENTS
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

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

6. Capteurs en robotique mobile

En robotique mobile, on classe traditionnellement les capteurs en deux catégories selon qu'ils mesurent l'état du robot lui-même ou l'état de son environnement. Dans le premier cas, a l'image de la perception chez les ` êtres vivants, on parle de proprioception et donc de capteurs proprioceptifs. On trouve par exemple dans cette catégorie les capteurs de position ou de vitesse des roues et les capteurs de charge de la batterie. Les capteurs renseignant sur l'état de l'environnement, donc de ce qui est extérieur au robot lui-même, sont eux appel ˆ es´ capteurs extéroceptifs. Il s'agit de capteurs donnant la distance du robot a l'environnement, la température, signalant la mise en contact du robot avec l'environnement, etc. L'étude d´entaillée des capteurs, qui relève à la fois de la

physique, de l'électronique et du traitement du signal, ne sera pas vue ici. Nous nous contenterons d'expliquer les principes de fonctionnement des capteurs présent es. On tachera simplement de garder a` l'esprit que les défauts inhérents aux différents systèmes de mesure utilisées (bruit, erreurs ou échecs de mesures, difficulté de modélisation) influent fortement sur la perception que le robot a de l'environnement.

Les principaux capteurs utilisés en robotique mobile sont : les capteurs télémétriques (les ultrasons, les lasers et les infrarouges), le GPS et les caméras. **[6]**

a. Les capteurs télémétriques

On appelle télémétrie toute technique de mesure de distance par des procédés acoustiques, optiques ou radioélectriques. L'appareil permettant de mesurer les distances est appelé télémètre. Il existe différentes technologies pour réaliser un télémètre. Nous présentons dans ce qui suit les télémètres les plus utilisés dans les systèmes de robotique mobile, en donnant une idée de leur gamme de mesure et d'application.

b. Les GPS

Un des systèmes les plus connus de ces dernières années est le Système de Positionnement Global (GPS) du département de la défense des Etats-Unis qui exploite une constellation de vingt-quatre satellites en orbite à haute altitude.

Chaque satellite détecte les récepteurs terrestres fixes ou mobiles qui se trouvent dans sa zone de visibilité et transmet à chacun, à période fixe, un signal contenant des informations telles que son identification, l'heure d'émission du signal et les paramètres définissant son orbite donc sa position. Théoriquement, un récepteur doit être visible d'au moins trois satellites, pour pouvoir calculer sa position dans un repère centré sur la terre, en calculant l'intersection de trois sphères (trois cercles pour un système mobile au sol) ; de fait, pour tenir compte des décalages temporels entre les mesures fournies par les satellites, quatre mesures de distance entre le satellite et le récepteur sont requises pour disposer d'une précision satisfaisante **[6].**

Ce système est cependant loin de résoudre tous les problèmes de localisation des systèmes mobiles. Il fonctionne en effet difficilement dans des environnements urbains, et n'est pas utilisable à l'intérieur des bâtiments car les ondes employées sont fortement atténuées en traversant les murs. Sa précision est de plus souvent trop faible pour qu'un

système mobile terrestre puisse utiliser ces informations seules. En pratique, il est souvent couplé à un système inertiel qui permet de pallier aux pertes du signal GPS et il ne remplace de toute façon pas les capteurs du système mobile qui lui permettent de percevoir son environnement immédiat.

Mis à part son utilisation pour la localisation, ce système est aussi utilisé pour l'aide à la navigation d'avions et de bateaux lors de l'approche d'aéroports ou de ports, ainsi que dans la surveillance de camionnettes de livraison et de véhicules d'urgence, en vue de choisir au mieux leurs itinéraires.

c. Les caméras

Depuis les années 70, les caméras CCD (Charge Coupled Device) ont été déterminantes dans l'évolution de la vision : la rapidité d'acquisition, la robustesse et la miniaturisation sont autant d'avantages qui ont facilité leur intégration. Elles sont très performantes en termes de portée, précision et quantité d'informations exploitables.

Elles sont de plus les seules capables de restituer une image sensorielle de l'environnement la plus proche de celle perçue par l'être humain. C'est sans doute le capteur fournissant la plus grande variété d'informations à partir d'une seule acquisition. En effet, en appliquant divers algorithmes de traitement d'images, il est possible d'extraire des éléments caractéristiques de l'environnement comme des segments de droite, des plans, mais aussi des balises et des couleurs.

7. La navigation

La navigation est une étape très importante en robotique mobile. Bien entreprise, elle permet une large autonomie à un robot mobile. Le système de navigation comporte plusieurs modules qui peuvent être traités différemment et parmi lesquels on distingue celui de la localisation et celui de l'évitement d'obstacles. La détection et l'évitement des obstacles est l'étape fondamentale de l'évolution d'un robot en territoire inconnu. On dispose à cet effet de plusieurs types de capteurs : caméras ; un programme d'analyse des images étant alors nécessaire, capteurs laser, capteurs infrarouge et capteurs à ultrasons. On utilise en général un capteur à ultrasons qui permet de renseigner sur la présence d'un obstacle sur le chemin d'évolution. Une fois les obstacles repérés, le robot peut effectuer plusieurs actions, par exemple : cartographier le site sur lequel il évolue, vérifier si sa

distance à l'obstacle est supérieure ou non à une distance limite, et dans le cas contraire, éviter l'obstacle. Un exemple d'évitement d'obstacles est présenté sur la figure 11.

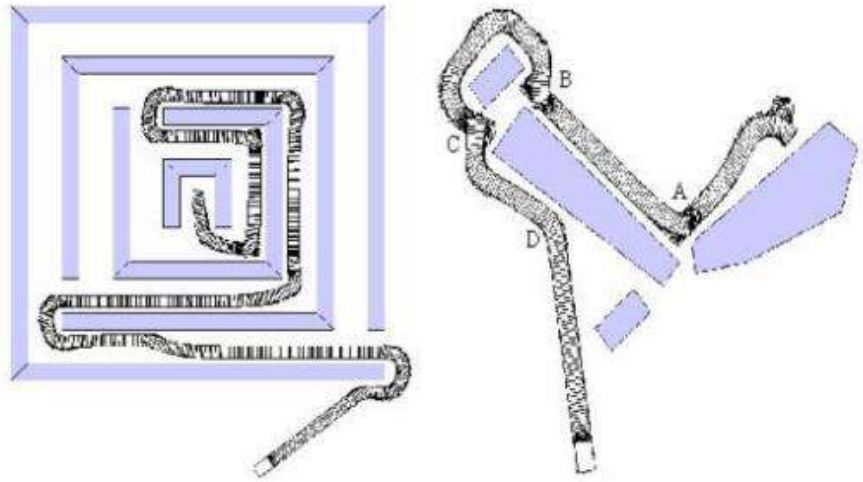


Figure 12 : Navigation de robot mobile en environnement encombré

8. La modélisation de l'environnement

La connaissance du milieu dans lequel évolue le robot mobile n'est établie en général qu'après avoir effectué une campagne de mesure de l'ensemble des éléments constituant l'environnement. Cette procédure fastidieuse peut être évitée si le robot construit lui-même son modèle d'environnement de manière dynamique. Par contre, la planification de trajectoire n'est pas utilisable tant que le robot ne dispose pas d'un modèle de l'espace d'évolution ce qui handicape très fortement l'utilisation du robot. à partir de cette base d'informations et d'une loi évaluant les erreurs de représentation, le planificateur peut générer des sous-trajectoires faisables dans certaines parties et modifier les sous trajectoires dans d'autres parties à l'aide des informations locales issues de la mesure des capteurs d'environnement. Lors de l'exécution d'une trajectoire, le robot acquiert des informations qui vont permettre de reconstituer le plus fidèlement possible le modèle de l'environnement de manière récursive à l'aide d'un algorithme approprié. **[7]**

9. La planification de trajectoire

On voit ainsi au travers de cette première approche assez théorique apparaître un problème essentiel : la planification de la trajectoire. Différentes approches sont envisageables selon que le robot évolue en milieu connu ou inconnu :

• L'évolution en territoire cartographié simplifie évidemment la tâche des concepteurs : une fois la carte de la zone d'évolution rentrée dans la mémoire d'un ordinateur communiquant avec le robot ou bien dans une mémoire intégrée au robot lui-même, des algorithmes de routage permettent de diriger le robot.

• Il en va tout autrement dans le cas de l'évolution en territoire inconnu. Le robot doit alors analyser son environnement au moyen de différents capteurs, détecter sa position par rapport à son but, et décider de sa trajectoire. Cette localisation peut s'effectuer par différentes méthodes : triangulation de signaux émis par des balises déposées au cours du déplacement ou/et repérage d'obstacles à distance et construction d'une carte du site, mesures odométriques et estimation de la position. On applique ensuite des algorithmes complexes pour diriger le robot. Ceux-ci peuvent amener des résultats plus ou moins heureux, le principal problème étant la non convergence de certaines boucles de déplacement. Si aucun algorithme de secours n'a été prévu, l'intervention humaine est alors nécessaire **[8].**

10. Méthodes de navigation

Il est possible au robot de se situer dans l'environnement d'opération et de planifier un trajet pour se rendre d'un point A à un point B. Pour y arriver, il est besoin de suivre l'une des stratégies de navigation permettant à un robot mobile de se déplacer pour rejoindre son but. Les approches utilisables actuellement cherchent plutôt à exploiter des cartes préétablies. Pour être en mesure de localiser le robot sur l'environnement, il faut choisir entre différentes approches suivantes : **[9]**

10.1. Navigation métrique :

Cette stratégie permet de calculer le chemin le plus court entre deux lieux mémorisés, en devisant l'environnement en plusieurs cellules de dimensions identiques. Une valeur est attribuée à chaque case en fonction de la probabilité qu'un obstacle

empêche le robot de se déplacer au centre de celle-ci, ça permet même de planifier des raccourcis au sein de zones inexplorées de l'environnement. Cette technique permet au robot de planifier des chemins au sein de zones inexplorées de son environnement. Elle mémorise pour cela les positions métriques relatives des différents lieux, en plus de la possibilité de passer de l'un à l'autre. Ces positions relatives permettent, par simple composition de vecteurs, de calculer une trajectoire allant d'un lieu à un autre, même si la possibilité de ce déplacement n'a pas été mémorisée sous forme d'un lien.

10.2. Navigation géométrique :

L'environnement est alors représenté par une série de formes géométriques simples comme des points ou des droites. Le localisateur prend en compte les lectures des capteurs de proximité et il essaie de retirer les formes géométriques qui sont présentes autour du robot. La position des différentes formes et la position de celles-ci les unes par rapport aux autres sont les critères pour identifier la position du robot. Le problème majeur est qu'il est parfois difficile ou même impossible de créer une carte précise due à la présence de formes trop complexes à modéliser. De plus, les différents obstacles sont définis par leur forme ainsi que par leur position dans l'environnement (coordonnées absolues).

10.3. Navigation topologique :

Cette stratégie permet de mémoriser un ensemble de lieux et les possibilités de passer de l'un à l'autre, indépendamment de tout but. Pour rejoindre un but, il faut alors une étape de planification qui permet de rechercher, parmi tous les chemins possibles, le chemin rejoignant le but. Dans cette technique, des points de repères sont utilisés ici comme références pour identifier des lieux dans l'environnement, et un graphe d'adjacence vient représenter les accès possibles à ces lieux.

10.4. Navigation par action associé à un lieu :

Cette méthode requiert une représentation interne de l'environnement qui consiste à définir des lieux comme des zones de l'espace dans lesquelles les perceptions restent similaires, et à associer une action à effectuer à chacun de ces lieux. L'enchaînement des actions associées à chacun des lieux reconnus définit une route qui permet de rejoindre le

but. Ces modèles permettent donc une autonomie plus importante mais sont limités à un but fixé. Une route qui permet de rejoindre un but ne pourra en effet pas être utilisée pour rejoindre un but différent. Changer de but entraînera l'apprentissage d'une nouvelle route.

10.5. Navigation par guidage :

Cette capacité permet d'atteindre un but qui n'est pas un objet matériel directement visible, mais un point de l'espace caractérisé par la configuration spatiale d'un ensemble d'objets remarquables. La stratégie de navigation, souvent une descente de gradient également, consiste alors à se diriger dans la direction qui permet de reproduire cette configuration. Cette stratégie utilise également des actions réflexes et réalise une navigation locale qui requiert que les amers caractérisant le but soient visibles.

10.6. Navigation par approche d'un objet :

Cette capacité de base permet de se diriger vers un objet visible depuis la position courante du robot. Elle est en général réalisée par une remontée de gradient basée sur la perception de l'objet. Cette stratégie utilise des actions réflexes, dans lesquelles chaque perception est directement associée à une action. C'est une stratégie locale, c'est-à-dire fonctionnelle uniquement dans la zone de l'environnement pour laquelle le but est visible.

11. Conclusion

La présentation faite dans cette première partie nous a permis de décrire les éléments constituants d'un robot mobile. A partir de ce descriptif, nous avons constaté que pour réaliser la commande de ce type de robot, nous devions répondre à trois impératifs à savoir la perception, la décision et l'action.

La combinaison d'informations issues de capteur proprioceptifs et de capteurs extéroceptifs permet d'envisager la réalisation de tâches robotiques complexes. Dans le cas de la robotique mobile, le traitement de toutes ces informations nécessite une puissance de calcul importante. Dans le prochain chapitre nous vous présenterons donc, les cartes électroniques qu'utilisées pour gérer et commander le robot mobile.

CHAPITRE II :

LES SYSTEMES EMBARQUES UTILISES

II. CHAPITRE II Les système embarqués utilisés

1. Introduction

Dans ce chapitre, nous découvrirons les deux cartes électroniques que nous avons utilisées dans ce projet. Nous parlerons principalement de la carte Arduino de la carte Raspberry Pi. Raspberry Pi sera absorbé dans le cerveau de notre robot, tandis qu'Arduino aura le rôle de gérer tous les capteurs et les actionneurs du notre robot

Tout d'abord, nous allons décrire en général et tout ce qui concerne la carte Arduino et la carte Arduino UNO que nous avons utilisé spécifiquement.

Deuxièmement, nous allons décrire la structure et le fonctionnement du mini PC Raspberry Pi que nous avons utilisé. De plus, nous présentons le Shield L293D.

2. Arduino

2.1.Définition

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

Un microcontrôleur est un petit ordinateur confiné dans un unique circuit intégré (une puce). Il constitue un excellent moyen pour programmer et pour contrôler des équipements électroniques. Il existe une grande variété de telles cartes à microcontrôleur, certaines des plus utilisées sont la platine Wiring, le PIC, le Basic Stamp et bien sûr Arduino. **[35]**

2.2.Les applications

Arduino peut vous permettre de prototyper et de réaliser de nombreux projets quel que soit votre domaine de prédilection. La gamme d'utilisation d'Arduino est énorme. Pour vous donner quelques exemples, vous pouvez :

*Contrôler des systèmes intelligents (smart home)

* Fabriques des robots et des machines (bras robots, CNC, imprimante 3D)

*Communiquer avec des plateformes différentes (web, serveur)

* Systèmes de détection (gaz, mouvement, visage)

* Gérer des caméras

*Récupérer les informations de consommation via le téléinformatique d'EDF

2.3.Les atouts majeurs d'un système à carte Arduino

Les cartes Arduino sont conçues pour réaliser des prototypes et des maquettes de cartes électroniques pour l'informatique embarquée. Ces cartes permettent un accès simple et peu couteux à l'informatique embarquée. De plus, elles sont entièrement libres de droit, autant sur l'aspect du code source (Open Source) que sur l'aspect matériel (Open Hardware). Ainsi, il est possible de refaire sa propre carte Arduino dans le but de l'améliorer ou d'enlever des fonctionnalités inutiles au projet. Le langage Arduino se distingue des langages utilisés dans l'industrie de l'informatique embarquée de par sa simplicité. En effet, beaucoup de librairies et de fonctionnalités de base occulte certains aspects de la programmation de logiciel embarquée afin de gagner en simplicité. Cela en fait un langage parfait pour réaliser des prototypes ou des petites applications dans le cadre de hobby. Les possibilités des cartes Arduino sont énormes, un grand nombre d'application ont déjà été réalisée et testées par bon nombre d'internautes. On retrouve par exemple diverse forme de robot, des stations météo.**[26]**

2.4.Les avantages d'Arduino

Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, nous allons programmer des systèmes électroniques. Les principaux avantages de l'électronique programmée sont :

- Simplifie grandement les schémas électroniques.
- Diminué le coût de la réalisation.
- La charge de travail à la conception d'une carte électronique.
- Environnement de programmation clair et simple.
- Multiplateforme : tourne sous Windows, Macintosh et Linux.
- Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- Logiciel et matériel open source et extensible.
- Nombreux conseils, tutoriaux et exemples en ligne (forums, site perso, etc.).
- Existence de « Shield » (boucliers en français).

2.5.Arduino Uno

Arduino Uno est une carte microcontrôleur basée sur l'ATmega328P. Il dispose de 14 broches d'entrée / sortie numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, un résonateur céramique 16 MHz (CSTCE16M0V53-R0), une connexion USB, une prise d'alimentation, un connecteur ICSP et un bouton de réinitialisation. Il contient tout le nécessaire pour prendre en charge le microcontrôleur ; connectez-le simplement à un ordinateur avec un câble USB ou alimentez-le avec un adaptateur CA / CC ou une batterie pour commencer**. [38]**



Figure 13 : Carte Arduino UNO

2.6.Présentation

La carte Arduino est présentée dans le dessine si dessous (figure 13) :

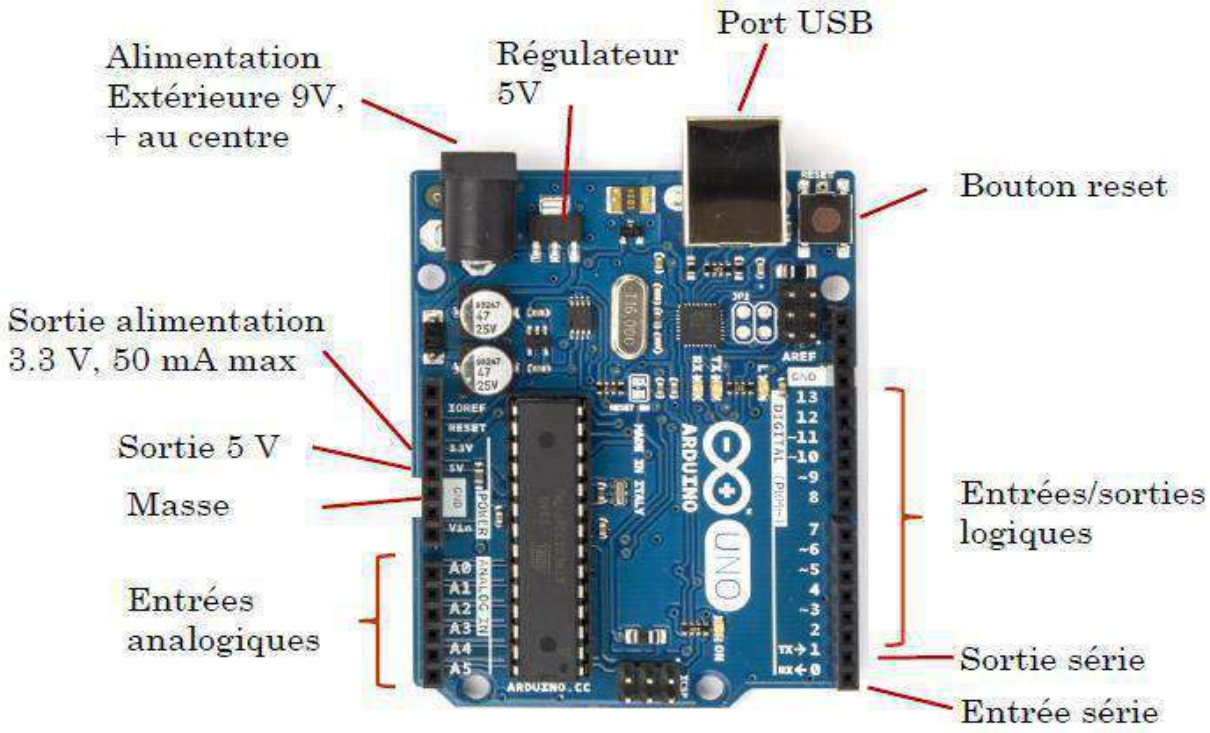


Figure 14 : Les compositions de la carte Arduino UNO

- **Alimentation / Programmation par USB**

La carte Arduino peut être alimentée avec un câble USB relié à votre ordinateur. Tout ce dont vous avez besoin, c'est de connecter votre carte Arduino à votre ordinateur avec le câble USB type A/B.

- **Alimentation via connecteur Jack DC**

La carte Arduino peut être directement alimentée par ce connecteur Jack DC. Ce connecteur est relié au régulateur de tension intégré à la carte. L'alimentation via ce connecteur doit être comprise entre 5 et 12 V.

- **Régulateur de tension**

La fonction du régulateur de tension est de contrôler la tension d'alimentation de l'Arduino pour la stabiliser à la bonne tension du microcontrôleur et de chaque élément de la carte. La tension de stabilisation est de 5 Volts sur les cartes UNO.

- **Oscillateur à quartz**

Un oscillateur à quartz est un élément électronique qui a la particularité de posséder un quartz à l'intérieur qui vibre sous l'effet piézoélectrique. Les propriétés électromécaniques du quartz sont telles qu'on arrive à faire vibrer le quartz à une fréquence très précise. Cet élément aide l'Arduino UNO à calculer les données de temps. Sur le dessus du composant, on peut lire 16.000H9H. Cela signifie que la fréquence est de 16,000,000 Hertz, soit 16 MHz.

- **Arduino Reset**

Vous pouvez redémarrer un Arduino avec un "Reset". Cela aura pour effet de redémarrer votre programme depuis le début. Vous pouvez redémarrer l'Arduino UNO de deux manières : soit en utilisant le bouton "Reset"**,** soit en connectant un bouton externe sur la broche de la carte Arduino mentionnée "RESET".

- **Broches (3.3, 5, GND, Vin)**

-3.3 V Broche d'alimentation de tension 3.3 Volts

-5 V Broche d'alimentation de tension 5 Volts

- La plupart des composants destinés à fonctionner avec Arduino fonctionnent bien en 3.3 Volts ou 5 Volts

- GND (Ground / Masse) – Il y a plusieurs broches de ce type présentes sur la carte Arduino, elles sont toutes communes et peuvent être utilisées comme masse (potentiel 0 Volts) pour vos circuits

-. Vin – Cette broche permet d'alimenter l'Arduino depuis une source de tension extérieure. Elle est reliée au circuit d'alimentation principale de la carte Arduino

- **Broches analogiques**

L'Arduino UNO possède 5 broches d'entrées analogiques numérotée de A0 jusqu'à A5. Ces broches permettent de lire un signal analogique d'un capteur comme un capteur d'humidité ou de température. La carte Arduino utilise un convertisseur analogique/numérique (convertisseur CAN) pour permettre la lecture du signal par le microcontrôleur. Un signal sera converti sur 10 bits. La valeur pourra être lue sur une échelle 1024 points.

- **Microcontrôleur principale**

Chaque carte Arduino possède son propre microcontrôleur. Vous pouvez le considérer comme le cerveau de la carte Arduino. Le microcontrôleur sur l'Arduino est légèrement différent d'une carte à l'autre. Les microcontrôleurs sont généralement de la société ATMEL. Vous devez savoir quel est le microcontrôleur de votre carte avant de charger un nouveau programme depuis l'IDE Arduino. Cette information est disponible directement sur le composant. Pour plus de détails sur la construction et les fonctions du microcontrôleur, vous pouvez vous référer à la fiche technique (*data Sheets*).

- **Connecteur ICSP**

Avant tout, le connecteur **ICSP** (*In-Circuit Serial Programming*) est une connectique AVR comprenant les broches MOSI, MISO, SCK, RESET, VCC et GND. Il s'agit d'un connecteur de programmation. Ce connecteur permet entre autres de programmer directement le microcontrôleur sur les couches les plus basses (bootloader, code ASM…). C'est aussi un port appelé port **SPI** (Serial Peripheral Interface), qui permet de dialoguer avec d'autres composants SPI (écrans, capteurs, etc…). On ne va pas se préoccuper de ce connecteur au début des tutoriels.

- **Indicateur LED d'alimentation**

Ce voyant doit s'allumer lorsque vous branchez votre Arduino sur une source d'alimentation pour indiquer que votre carte est correctement alimentée. Si cette lumière ne s'allume pas, il y a un problème avec votre alimentation, et je ne parle pas de nourriture ici.

- **LED, TX et RX**

Sur votre carte, vous trouverez deux indicateurs : **TX (émission)** et **RX (réception)**. Ils apparaissent à deux endroits sur la carte Arduino UNO. Tout d'abord, sur les broches numériques 0 et 1, pour indiquer les broches responsables de la communication série. Deuxièmement, les LED TX et RX. Le voyant TX clignote à une vitesse variable lors de l'envoi des données série. La vitesse de clignotement dépend de la vitesse de transmission utilisée par la carte. RX clignote pendant le processus de réception. La vitesse de transmission s'exprime en bauds, soit l'équivalent du bits/seconde si le signal est binaire.

- **Entrées/Sorties numériques**

La carte Arduino UNO possède **14 broches d'Entrées / Sorties numériques**, dont 6 peuvent fournir une sortie PWM (Pulse With Modulation). Ces broches peuvent être configurées pour fonctionner comme des broches numériques d'entrée pour lire des valeurs logiques (0 ou 1) ou numériques. Elles peuvent également être utilisées comme des broches de sortie pour piloter différents modules comme des LED, des relais, etc. Les broches étiquetées "~" peuvent être utilisées pour générer des PWM.

- **Broche AREF**

AREF est l'acronyme anglais de "référence analogique". Cette broche est parfois utilisée pour définir une tension de référence externe (entre 0 et 5 Volts) comme limite supérieure pour les broches d'entrée analogiques.

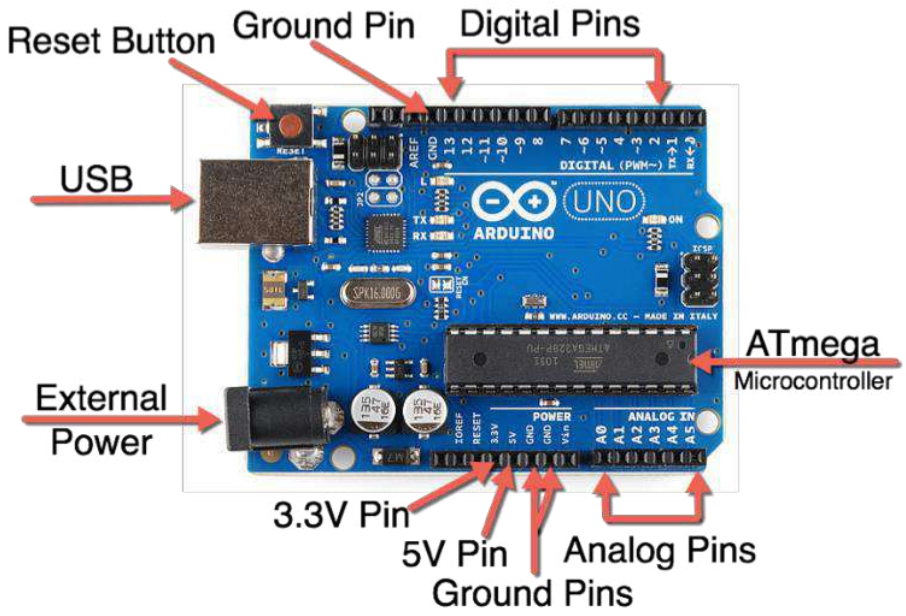


Figure 15 : Structure de la carte ArduinoUno.

Les différents éléments de la carte de commande sont regroupés dans le tableau suivant **[30]**

Courant continu par broche E / S	20 mA
Courant pour broche de 3.3V	50 mA
Mémoire Flash	32Ko (ATmega328P) Dont 0,5 Ko utilisé par boot loader
SRAM	2 Ko (ATmega328P)
EEPROM	1 KB (ATmega328P)
Vitesse de l'horloge	16 MHz
LED_BUILTIN	13
Longueur	68,6 mm
Largeur	53,4 mm
Poids	25 g

Tableau 2 : Les différents éléments de la carte de commande.

2.7.Choix d'Arduino

Les paramètres à prendre en compte pour vous aider à choisir sont :

- **Le prix,** par ce que ça ne sert à rien de payer plus cher ce qu'on n'a pas besoin !
- **La dimension**, par ce que si c'est plus petit c'est plus facile à faire rentrer au chausse pied !
- **Le nombre d'entrée sorties**, par ce que si on a besoins de 50 entrées sorties il vaut mieux prendre celle qui en a suffisamment
- **Le poids**, par ce que si la réduction du poids est la priorité numéro 1 mieux vaut prendre la plus légère
- **La taille mémoire**, par ce que si on a un programme très lourd mieux vaut qu'il y ai suffisamment de place dans la carte
- **Les connecteurs disponibles**, par ce qu'avoir les bons connecteurs et des connecteurs supplémentaires c'est toujours plus pratique !
- **Le besoin de souder les connecteurs**, par ce que si tu n'as pas accès à un fer à souder ce n'est pas pratique

En fonction de votre projet gardez bien en tête ces paramètres pour bien choisir la bonne carte Arduino qui vous conviendra le mieux.

2.8.Le logiciel de gestion de la carte Arduino

Arduino est un espace de développement intégré (EDI) qui vous permet d'écrire, de compiler et d'envoyer du code sur le circuit imprimé du même nom. Pour rappel, la carte Arduino contient un microcontrôleur que l'on peut programmer dans le but d'effectuer des tâches variées, comme la domotique, par exemple**. [4]**

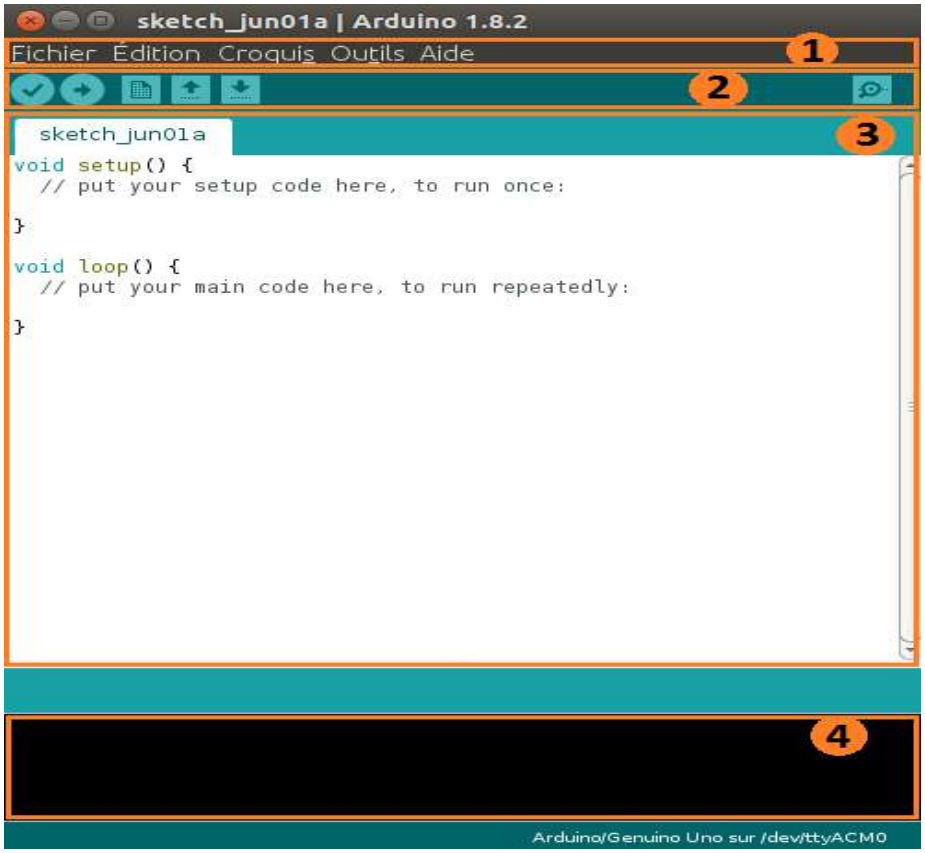


Figure 16 : logiciel Arduino.

- Le cadre numéro 1 : 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.

Les boutons

Voyons à présent à quoi servent les boutons numérotés en rouge.



Figure 17 : Présentation des boutons.

- Bouton 1 (vérifié) : Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
- Bouton 2 (uploade): Compiler et envoyer le programme vers la carte.
- 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).

Moniteur série

Le moniteur série est utilisé pour afficher les informations envoyées de la carte Arduino à l'application (généralement via un câble USB). Il permet également la transmission d'informations à la carte Arduino. Pour ce faire, il vous suffit de taper du texte dans le champ en haut de la fenêtre et de cliquer sur le bouton « Soumettre ». Bien sûr, avant de pouvoir le faire, vous devez encoder un programme approprié, prêt à recevoir ces informations. Il est également possible d'ajuster la vitesse de transmission du dispositif d'affichage série, qui détermine la rapidité avec laquelle la transmission des données est effectuée. Il est également possible de fournir une connexion série entre la carte Arduino et des périphériques ou d'autres logiciels. Nous verrons ce travail plus en détail dans les projets "Bright Text" et "Oscilloscopes".**[30]**

3. La carte Raspberry pi

L'une des grandes forces du Raspberry Pi, c'est qu'il peut être utilisé dans une grande variété de projets. C'est une plate-forme souple, de divertissement, de travail ou d'expérimentation, vous permettant aussi bien de regarder des vidéos et de surfer sur le Web, que de bidouiller, d'apprendre ou de réaliser des montages électroniques.

Voici quelques-unes de ses applications

- UN VÉRITABLE ORDINATEUR

- UN SUPPORT D'APPRENTISSAGE DE LA PROGRAMMATION

- UNE PLATE-FORME POUR INTERAGIR AVEC LE MONDE PHYSIQUE

- UN OUTIL INDISPENSABLE POUR LES MAKERS

- UN LECTEUR MULTIMÉDIA

- UN OUTIL DE PROGRAMMATION BAS NIVEAU **[38]**



Figure 18 : Raspberry pi 3 B

3.1.Pourquoi Pi 3 B ?

- Démarrage plus rapide
- Une meilleure compatibilité avec les derniers programmes et logiciel éducatifs
- Processeur intégré Quad-core ARM Cortex-A53 1.2 GHz (BroadcomBCM2837)

- RAM : 1024 Mo et Lecteur de cartes Micro SD, Wi-Fi b/g/n et Bluetooth4.1

- Distributions GNU/LINUX ARM, telles que Snappy Ubuntu, et Microsoft Windows10.

3.2.La carte mère Raspberry Pi 3 Type B

Le Raspberry Pi 3 Model B est le premier modèle du Raspberry Pi de troisième génération. Il a remplacé le Raspberry Pi 2 Model B en février 2016. Voir aussi le Raspberry Pi 3 Model B +, le dernier produit de la gamme Raspberry Pi 3.

- Processeur intégré Quad-core ARM Cortex-A53 1.2 GHz (BroadcomBCM2837)
- GPU Dual Core Video Core IV Multimedia Co-Processor, RAM: 1024 Mo
- Lecteur de cartes Micro SD
- Ports : HDMI, 4x USB, RJ45, jack 3.5 mm, connecteurs pour APN et écran tactile
- Wi-Fi b/g/n et Bluetooth 4.1
- Support des distributions dédiées basées sur Linux et Windows 10**.[31]**

3.3. Les composantes du Raspberry Pi

Vous serez peut-être tenté de considérer le Raspberry Pi comme une carte à microcontrôleur comme Arduino, ou comme un substitut à votre ordinateur portable. En réalité, la structure du Raspberry Pi ressemble davantage à celle d'un appareil mobile, doté de plusieurs connecteurs permettant d'accéder facilement à ses nombreux ports et fonctions. La figure 18 représentes différents composants de la carte.



Figure 19 : les composants de Raspberry pi

- **Le processeur (Broadcom BCM 2835)**

Le cœur du Raspberry Pi est équipé d'un processeur de même type que celui de l'iPhone 3G. La puce centrale est un System on Chip (SoC) cadencé à 700 MHz, utilisant l'architecture ARM11 conçue par la société anglaise ARM Holdings

- **Le support de carte SD**

(Secure Digital) Comme vous pourrez rapidement le constater, le Raspberry Pi ne possède pas de disque dur, tout est stocké sur une carte SD.

- **La prise USB**

Utiliser pour connecter n'importe quel périphérique de type USB comme clavier, souris, carte réseau sans fil...etc.

- **La prise Ethernet**

Raspberry Pi possède une prise Ethernet standard au format.

- **La prise HDMI**

Le port HDMI fournit les sorties vidéo et audio numériques et il supporte 14 résolutions vidéo différentes.

- **Les LED d'état**

Le Raspberry Pi dispose de cinq LED qui renseignent sur l'activité de la carte

*Une led verte allume d’activité de processeur

*Une led rouge de mise sous tension

*Une led verte signalant le mode full duplex de la connexion réseau

*Une autre led verte signalant l’activité de réseau

*Une led jaune indiquant que le réseau fonctionne à 100 mbit par seconde

- **La sortie vidéo composite**

C’est une prise standard de type RCA qui fournit les signaux vidéo composites. Le format composite a une résolution très basse par rapport à celle du HDMI.

- **Le connecteur d'alimentation**

Le connecteur micro USB sert uniquement à alimenter la carte et il fonctionne comme un interrupteur.

- **Les broches d'entrées-sorties (GPIO)**

Les ports GPIO (General Purpose Input/Output) très utilisés dans le monde des microcontrôleurs et de l'électronique embarquée. Ces broches GPIO offre à la Raspberry Pi la possibilité de communiquer avec d'autres circuits électroniques. Il existe plusieurs bibliothèques facilitent la programmation des connecteurs GPIO par exemple : wiringPi-langage C, Rpi.GPIO-Python, Pi4J-Java

- **Le connecteur Display Serial Interface (DSI)**

Ce connecteur accepte un câble plat à 15 conducteurs qui peut communiquer avec un afficheur LCD ou OLED.

- **Le connecteur Camera Serial Interface (CSI)**

Ce port permet de connecter directement à la carte un module caméra spécifique.

- **La sortie audio analogique**

La carte comporte une prise jack standard de 3,5 mm qui fournit un signal sonore stéréo à haute impédance, destiné à des appareils comme des haut-parleurs amplifiés.

3.4.Programmation

Il existe plusieurs langages de programmation qui peut être compilé sur le Raspberry Pi, telle que le langage C, C++, HTML5, le Java, le JavaScript...etc. Le langage utilisant dans notre projet et le Python.

Le Python est un langage de programmation inventé par Guido van Rossum, la première version de python est sortie en 1991.

Python est un excellent langage de « pilotage » pour les codes scientifiques écrits dans d'autres langues. Cependant, avec des outils de base supplémentaires, Python se transforme en un langage de haut niveau adapté au code scientifique et d'ingénierie qui est souvent assez rapide pour être immédiatement utile mais aussi suffisamment flexible pour être accéléré avec des extensions supplémentaires. **[31]**

3.5.Système d'exploitation

La plupart des systèmes qui fonctionnent sur Raspberry Pi sont des versions du système d'exploitation Linux. Parce que Linux est open source, les développeurs peuvent l'adopter pour des buts spécifiques. Dans le cas du Raspberry Pi, le matériel léger nécessite un système réduit et privé d'applications inutiles.

La distribution est recommandée Raspbian. C'est un système d'exploitation libre basé sur Debian optimisé pour le matériel Raspberry Pi. Cependant Raspbian fournit plus d'un simple système d'exploitation : il est livré avec plus de 35.000 paquets, des logiciels précompilés qui nous facilitent le développement soft **[31]**

4. L’extension (Shield) de commande des moteurs.

Le L293D est un pilote de moteur H-Bridge à double canal capable de piloter une paire de moteurs à courant continu ou un moteur pas à pas unique.

Le Shield contient deux chipsets de pilote de moteur L293D, cela signifie qu'il peut piloter individuellement jusqu'à quatre moteurs CC, ce qui le rend idéal pour la construction de plates-formes de robot à quatre roues.

Il offre un total de 4 ponts en H et chaque pont en H peut fournir jusqu'à 0,6 A au moteur.

il a également un registre à décalage 74HC595 qui étend 4 broches numériques de l'Arduino aux 8 broches de contrôle de direction de deux puces L293D. **[33]**

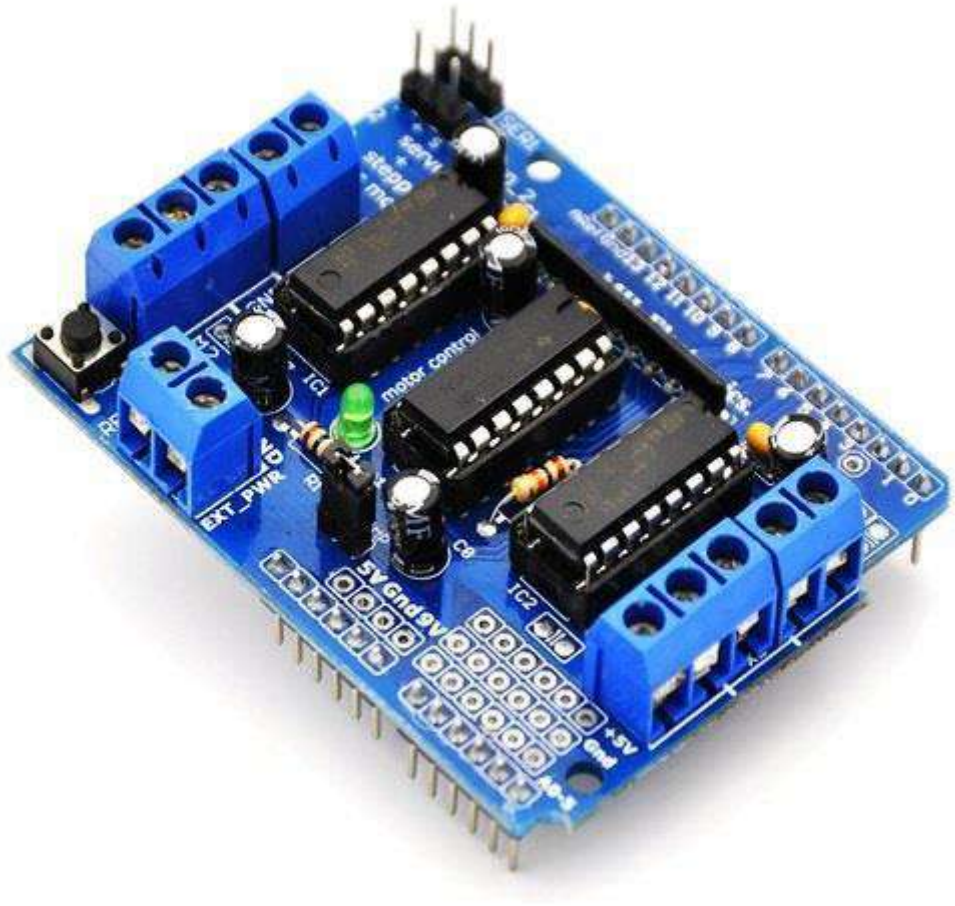


Figure 20 : Adafruit Motor Shield

Application : [34]

- Manipulation des moteurs pas à pas
- Manipulation des moteurs a courant contenu
- Manipulation des servomoteurs

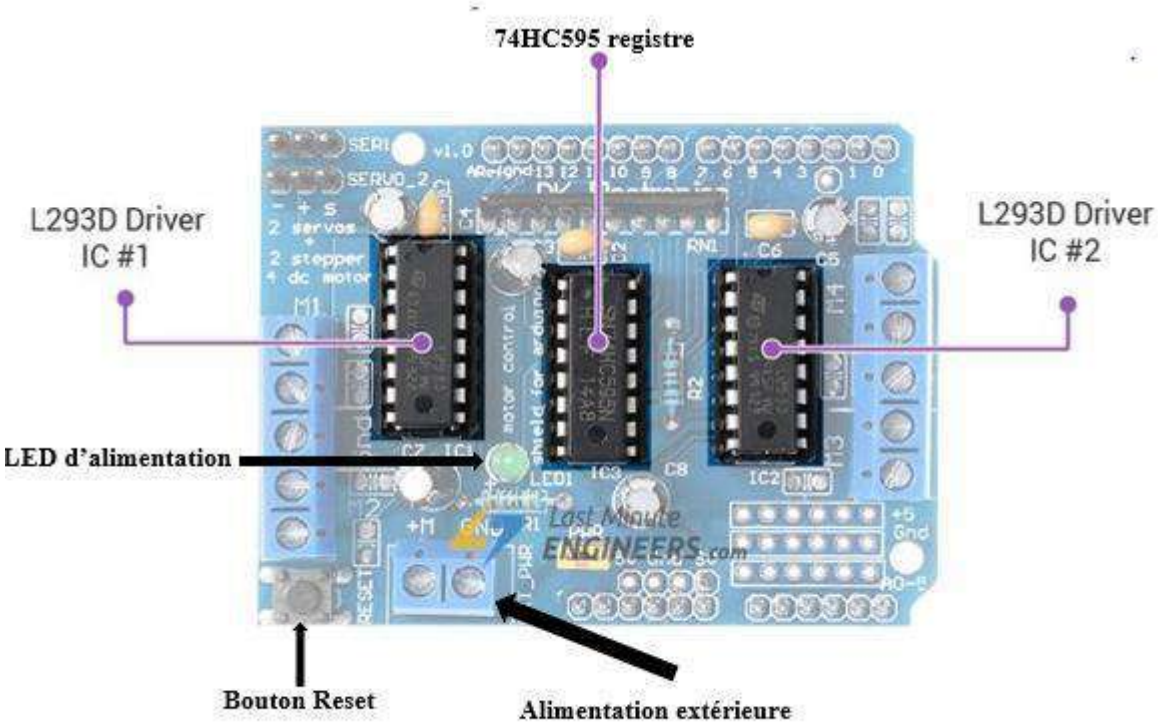


Figure 21. : Description Shield l293D

5. Conclusion

Dans ce chapitre, nous avons fourni une description complète des cartes Raspberry PI 3 B et Arduino, ainsi que la carte l293D que nous avons utilisées pour concevoir l'architecture matérielle nécessaire pour contrôler un robot mobile.

La maîtrise de ce robot, basée sur l'utilisation de données visuelles, nécessite d'une part une puissance de calcul importante et d'autre part des ordinateurs petits et légers pouvant être intégrés dans la structure mécanique du robot mobile.

CHAPITRE III :

DESCRIPTION DU SYSTEME GPS

III. CHAPPITRE III Description du système GPS

1. Introduction

Le système GPS (Global Positioning System) est un système de radio-positionnement par satellites mis en place par le département de la défense des Etats-Unis d'Amérique à partir de 1978 pour des applications militaires (guidage balistique, positionnement de troupes...) qui s'est ouvert progressivement aux usages civils au cours des années 1990 - 2000. L'objectif est de permettre à un utilisateur au sol de répondre quasi-instantanément, 24h/24 et 7j/7

Ce service est disponible pour un nombre illimité d'utilisateurs à un coût modéré grâce au choix d'un système passif où ce sont les satellites qui émettent les informations utiles aux utilisateurs qui ne font que les interpréter **[10]**

2. Comment fonction la géolocalisation :

Le principe de la triangulation est considéré comme la méthode la plus utilisée pour la détermination d'une position d'un point à la surface du globe.

La triangulation permet de localiser un objet en connaissant la distance qui le sépare de 3 points de référence (Figure21).

En 3D il est nécessaire de rajouter une quatrième distance reliant l'objet à un autre point de référence pour connaître l'altitude,

On parle alors de multilatéralisons. Extrapolée à la géolocalisation d'un récepteur à la surface du globe,

Il est possible de définir les coordonnées X, Y et Z d'un point s'il on connaît les distances qui le séparent de quatre satellites.

Les coordonnées du point d'intersection des sphères représentant les signaux émis par les différents satellites seront d'autant plus précises que le nombre de satellites captés sera important.



Figure 22 : Principe de la triangulation pour géolocaliser un récepteur a la surface du globe

3. Description du système

G.P.S est un système de radiolocalisation par satellite qui permet de positionnement des objets sur la surface terrestre

Le système est composé de trois secteurs **[11]** :

-le secteur spatial qui est constitué d'un ensemble des satellites ;

-le secteur de contrôle qui permet de piloter le système ;

-le secteur utilisateur qui est l'ensemble des utilisateurs civils et militaires du système ;

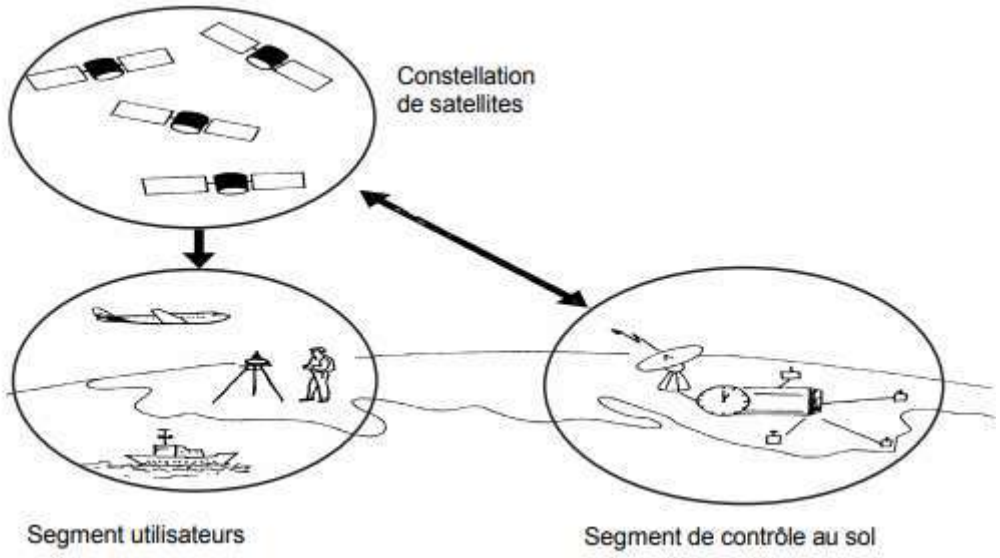


Figure 23 : Les trois segments du GPS

3.1.Le secteur spatial

Le segment spatial est constitué d'une constellation nominale de 24 satellites opérationnels qui transmettent des signaux unidirectionnels, les quels donnent le positionnement et l'heure du réseau GPS **[12]**

Les satellites évoluent sur six plans orbitaux ayant une inclinaison d'environ 55° sur l'équateur**[13],** Ils suivent une orbite quasi-circulaire de rayon 26 600 km environ (soit une altitude de 20 200 km) qu'ils parcourent en 11 h 58 min 2 s, soit un demi-jour sidéral. Ainsi, les satellites, vus du sol, reprennent la même position dans le ciel au bout d'un jour sidéral

Chaque satellite est muni d'un émetteur-récepteur, d'une horloge atomique, d'un ordinateur et d'équipements auxiliaires destinés au fonctionnement du système.

L'énergie pour l'équipement est fournie par 7m2 de capteurs solaires tandis qu'un système de fusées d'appoint permet de réajuster la position du satellite sur son orbite et de contrôler sa stabilité **[14].**

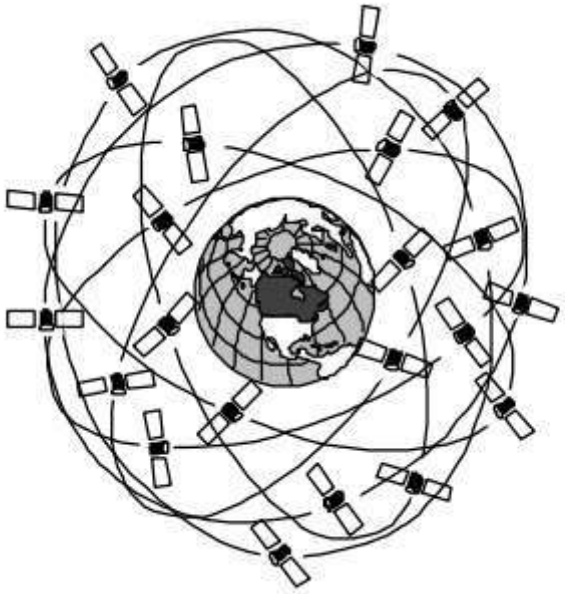


Figure 24 : Constellation des satellites du GPS

3.1.1. Les différents types de satellites

En 1974, le Rockwell Il a remporté un contrat pour la construction de huit satellites Bloc I. En 1978, le contrat a été prolongé pour construire trois autres. En commençant par

le premier satellite Navstar en 1978, 10 satellites Block I ont été lancés avec succès. Un satellite, le Navstar 7, a été perdu en raison d'un lancement raté 18 Décembre 1981. **[15]**

Les satellites ont été lancés le bloc I de la base aérienne Vandenberg en utilisant la fusée Atlas. Les satellites ont été construites dans la même usine où les deuxièmes étages des fusées ont déjà été construits Saturn V

L'objectif principal des satellites Block I était la validation du système et la conception des satellites. L'expérience acquise lors de la construction de 11 satellites a été utilisé dans la construction de la série Bloc II.

Un double panneau solaire a été en mesure de générer 400W, charger les batteries NiCd à fonctionner même à l'ombre. Pour le contrôle de la télémétrie, il a été utilisé un canal S-Band, alors qu'un canal UHF a été utilisé pour les communications entre satellites. Pour la correction orbitale était présente une hélice à l'hydrazine. La charge principale est composée de deux émetteurs dans la bande L des signaux de navigation GPS à des fréquences de 1575,42 MHz (L1) et 1227,60 MHz (L2).

Le dernier lancement de satellites Block I était de 9 Octobre, 1985, mais le dernier des satellites n'a pas été mis hors service jusqu'au 18 Novembre 1995, bien au-delà de sa durée de vie prévue de 5 ans.

SATELLITES BLOC II

Bloc Série II Accueil

La série de satellites Bloc II ont été les premiers à obtenir un fonctionnement complet de la constellation GPS, avec une conception qui a permis au système de fonctionner même en l'absence de contact avec les stations de contrôle au sol jusqu'à 14 jours. Le principal fabricant a été le Rockwell, qui a construit 12 satellites pour se qualifier comme un ajout aux satellites du contrat Bloc I. En 1993, la société a remporté un nouveau contrat pour construire 28 satellites Bloc II / II

Les satellites Bloc II sont stabilisé 3-axes, avec pointant vers le sol sur la base de roues de réaction. Les deux panneaux solaires étaient capables de livrer bien 710W. Le canal en bande S a été utilisé pour contrôler et télémétrie tandis que les perles ont été utilisé canal UHF communications entre satellites. Pour la correction orbitale a été utilisé une hélice à l'hydrazine. La charge principale est composée de deux émetteurs dans la bande L des signaux de navigation GPS à des fréquences de 1575,42 MHz (L1) et 1227,60 MHz (L2).

Chaque satellite est équipé de 4 horloges atomiques (deux césium et rubidium 2) et un système de détection des explosions nucléaires avec un total de 1660 kg.

Les neuf premiers satellites ont été lancés à partir du 14 Février 1988. Le dernier a été lancé le 1er Octobre de 1990. Le dernier satellite de la série a été retiré du service de 15 Mars 2007, bien au-delà de sa vie pour laquelle il a été construit pour 7,5 ans

Bloc série II

La série IIA Block était une version légèrement améliorée de la série bloc II, construit pour permettre au système de fonctionner jusqu'à 180 jours sans contact avec les stations au sol. La masse des satellites a augmenté, pour atteindre 1816 kg.

Satellites ont été lancés IIA 19 série du 26 Novembre 1990 jusqu'au 6 Novembre 1997. Le 17 Janvier 2009 Les six des satellites ont été retirés du service actif.

Deux satellites de cette série, les numéros 35 et 36, sont équipées de laser catadioptres qui permettent leur suivi indépendamment de leurs signaux radio, en fournissant une séparation non ambiguë entre les erreurs d'éphéméride et de leurs horloges internes.

Bloc IIR Série

La série Bloc IIR a été développé par Lockheed Martin. Chaque satellite a une masse de 2.030 kg et 1.080 kg au lancement, une fois en orbite. Le premier lancement tentative 17 Janvier, 1997 ne sont pas en raison de l'explosion de la fusée Delta II après 12 secondes de vol. Le premier lancement a été achevée avec succès le 23 Juillet 1997. Il a été lancé avec succès 12 satellites du bloc série IIR.

Série bloc IIR-M

Représentation artistique d'un satellite Bloquer IIR-M en orbite

La série bloc IIR-M comprend un nouveau signal militaire et un signal civil plus robuste connu sous le nom L2C. Il y a 8 satellites de bloc série IIR-M, fabriqués par Lockheed Martin Le premier satellite de la série a été lancée 26 Septembre 2005 et 17 derniers Août 2009.

Bloc série IIF

Représentation artistique d'un satellite en orbite bloc IIF

Les satellites bloc IIF de la série ont été développés par Boeing. Le premier satellite a été rempli 09 Septembre 2007. Boeing a signé un contrat pour la construction de 12 satellites de la série. a été lancé le premier satellite en mai 2010, une fusée Delta V. Le deuxième satellite de la série a été lancé en Juillet 2011. Les satellites ont une masse de 1630 kg et un cycle de service de 12 ans.

L'innovation du bloc série IIF sont :

- Nouvelle utilisation pour le signal de démonstration pour une utilisation safefy de la vie dans la bande L5
- Nouveau signal militaire M-Code
- Précision attendue Doubler
- Meilleure résistance Jamming
- Processeur reprogrammable qui peut recevoir des mises à niveau logicielles
- Premiers satellites ne sont pas le matériel Disponibilité sélective pour la dégradation du signal installé.

SATELLITES BLOC IIIA

Les satellites de la série Bloquer IIIA sont la nouvelle génération de satellites GPS, avec de nouveaux signaux et des niveaux plus élevés de puissance de transmission. Ils seront jusqu'à 32 satellites avec une durée de vie prévue de 15 ans. Le premier contrat comprend l'option 4 autres satellites de test pour 8 autres satellites opérationnels a été attribué à Lockheed Martin. Le lancement du premier satellite de la série est prévu en 2014.



Figure 25 : satellite bloc IIIA en orbite



Figure 26 : satellite Navstar-2F

3.1.2. Identification des satellites

Plusieurs systèmes de numérotation des satellites coexistent :numérotation séquentielle dans l'ordre de lancement, numéro lie au plan d'orbite et la position sur l'orbite, numérotation PRN donne pour codes, numéro dans le catalogue de la NASA et numérotation internationale.

*Le numéro PRN est celui qui est généralement utilise car il fournit aux utilisateurs dans le message de navigation. **[11]**

3.1.3. Le signal satellite

Actuellement, chaque satellite GPS émet en continu en utilisant deux radiofréquences dans la bande "L" appelées Link 1 (L1) et Link 2 (L2). La bande "L" couvre les fréquences comprises entre 1 GHz et 2 GHz et est un sous-ensemble de la bande ultra-haute fréquence (UHF). Les fréquences centrales de L1 et L2 **[16]**

Chaque fréquence est un multiple de la fréquence de référence 10.23 MHz telle que :

$$L1 = 1575.42\ MHz = 154 \times 10.23\ MHz \quad L2 = 1227.60\ MHz = 120 \times 10.23\ MHz$$

Sur ces deux fréquences porteuses sont modulés **[17]** :

*Le message de navigation : est envoyé sur les deux porteurs L1 et L2, la durée d'un message complet est de 750 sec. (12.5 min.). Il contient les informations sur la position des satellites, l'état des satellites, les corrections sur leurs horloges ainsi que des données atmosphériques

*les codes PRN :

Ici encore il y a eu de deux types :

« C/A code » (pour coarse acquisition code) est destiné au service SPS (« Standard Positioning Service ») il est transmis sur un seul porteur (L1).

« P-code » (pour precise positioning code) destiné au service PPS (« Precise Positioning Service ») est transmis sur les deux porteurs L1 et L2. Lorsque ce dernier était réservé exclusivement aux militaires il était encrypté en un nouveau code, dit « Y-code ».

Les figures (26 et 27) résument la structure des signaux, transmis sur les deux fréquences, par chaque satellite GPS.

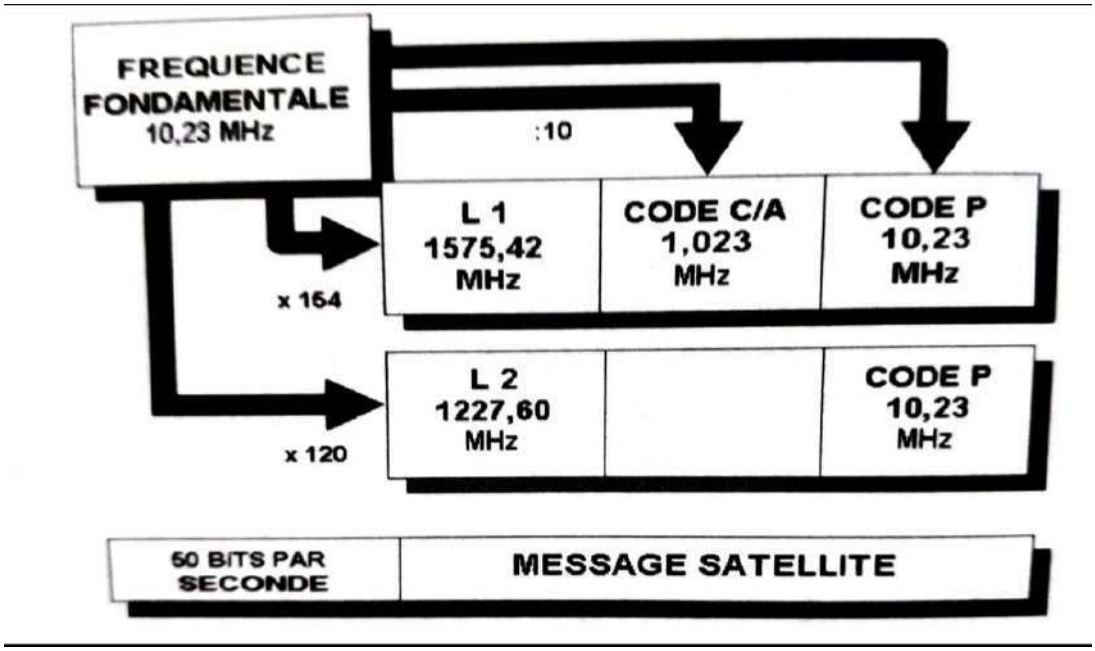


Figure 27 : Structure du signal GPS

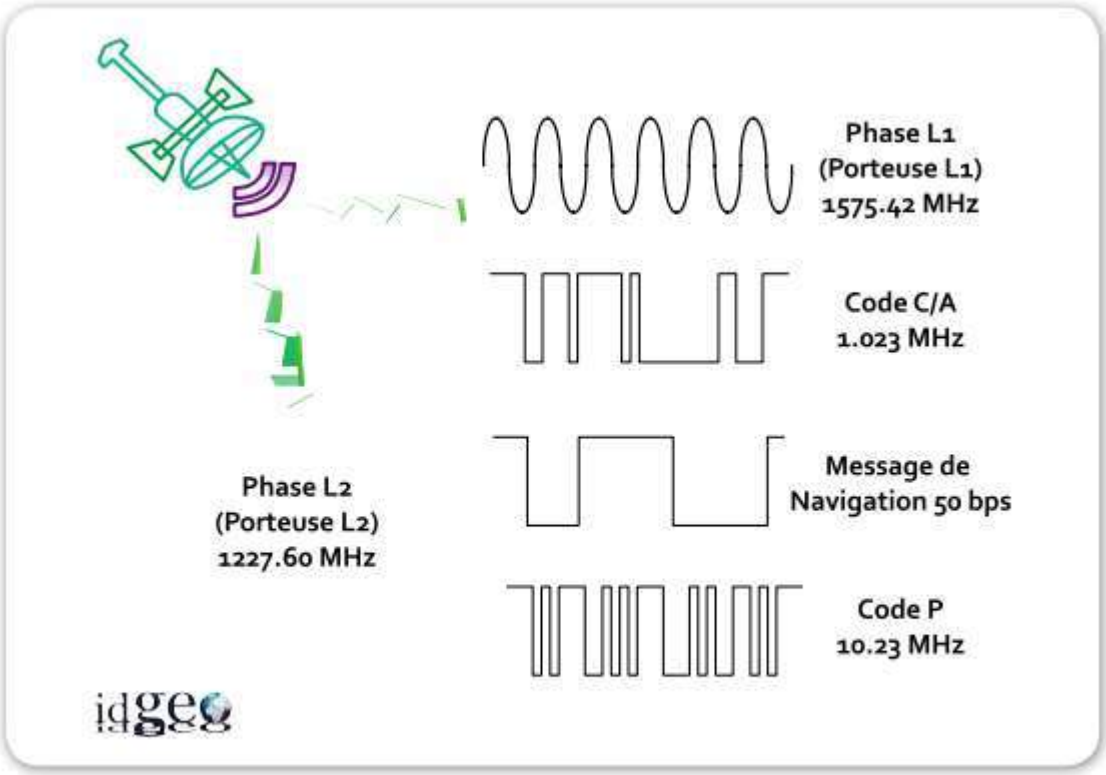


Figure 28 : Structure du signal GPSD Deux frequence porteuse L1 et L2 modulé sur des codes C/A des codes P et des codes de navigation

3.1.4. L’échelle de temps GPS

L'heure GPS est alignée sur UTC, sauf que l'heure GPS est continue et n'inclut pas les secondes intercalaires, ce qui signifie que l'heure GPS diffère de l'heure UTC par un nombre entier de secondes, 16 au début de 2013. L'heure du système GPS est donnée par son horloge embarquée. Cette horloge comprend toutes les stations de surveillance GPS opérationnelles et les normes de fréquence des satellites. L'heure du système GPS, à son tour, est référencée à l'horloge principale (MC) de l'Observatoire naval des États-Unis (USNO) et orientée vers UTC tel que maintenu par l'USNO, appelé UTC (USNO), à partir de laquelle l'heure GPS ne s'écartera pas de L'origine de l'heure GPS la plus récente, lorsque l'heure GPS a été initialisée à zéro, était l'heure à 23 :59 :47 UTC le 21 août 1999 (00 :00 :19 TAI le 22 août 1999). 1024 semaines, soit environ 19,6 Ans **[18]**

3.2.Le secteur de contrôle

Le segment de contrôle est représenté par cinq stations au sol (la principale est basée Au Colorado Spring, États-Unis) répartis le long de l'équateur.

Le Segment de Contrôle est responsable de la maintenance des satellites et de leur bon fonctionnement. Cela comprend le maintien des satellites dans leurs positions orbitales appropriées et la surveillance de la santé et de l'état du sous-système satellite.

Le segment de contrôle surveille également les panneaux solaires des satellites, les niveaux de puissance de la batterie et l’état des propulseurs utilisés pour les manœuvres.

De plus, le segment de contrôle active des satellites de rechange (si disponibles) pour maintenir la disponibilité du système.

Le segment de contrôle met à jour l’horloge, les éphémérides, l’almanach et les autres indicateurs de chaque satellite dans le message de navigation au moins une fois par jour.

Les mises à jour sont plus fréquemment programmées lorsque des précisions de navigation améliorées sont nécessaires.

Les paramètres des éphémérides correspondent précisément aux orbites des satellites GPS et ne sont valables que pour un intervalle de temps de 4 heures avec le programme de téléchargement normal une fois par jour. Selon le bloc satellite, les données des messages de navigation peuvent être stockées pendant un minimum de 14 jours à un maximum de 210 jours par intervalles de 4 heures ou 6 heures pour les téléchargements aussi peu fréquents

qu'une fois toutes les deux semaines et des intervalles de plus de 6 heures dans le cas où un téléchargement ne peut pas être fourni pendant plus de 2 semaines.

L'almanach est un sous-ensemble à précision réduite des paramètres des éphémérides. L'almanach se compose de 7 à 15 paramètres orbitaux des éphémérides. Les données d'almanach sont utilisées pour prédire la position approximative du satellite et l'aide à l'acquisition du signal satellite.

En outre, le segment de contrôle résout les anomalies des satellites, contrôle SA et AS, et collecte des mesures de pseudo-portée et de phase de porteuse à distance.
Surveiller les stations pour déterminer les corrections d'horloge satellite, l'almanach et les éphémérides.

Pour accomplir ces fonctions, le segment de commande est composé de trois composants physiques différents : le poste de commande principal (MCS), les stations de surveillance et les antennes de terrain. **[19]**

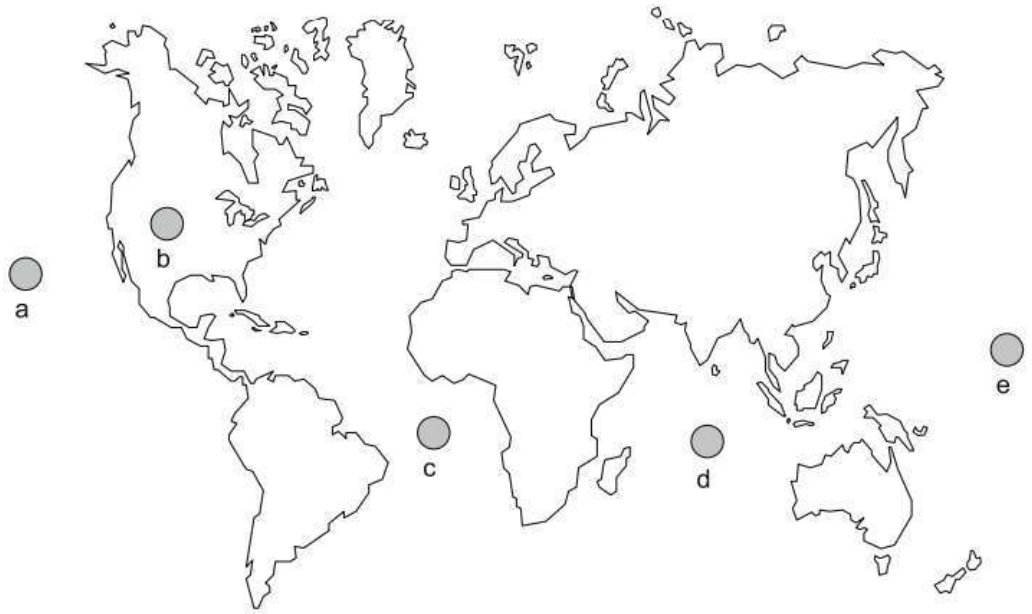


Figure 29 : Dislocation géographique mondiale de la station contrôle

3.3.Le secteur utilisateur

L'équipement de réception d'utilisateur comprend le segment d'utilisateur. Chaque ensemble d'équipement est généralement appelé récepteur GPS, qui traite les signaux en bande L transmis par les satellites pour déterminer le PVT (position, vitesse, temps) de l'utilisateur.

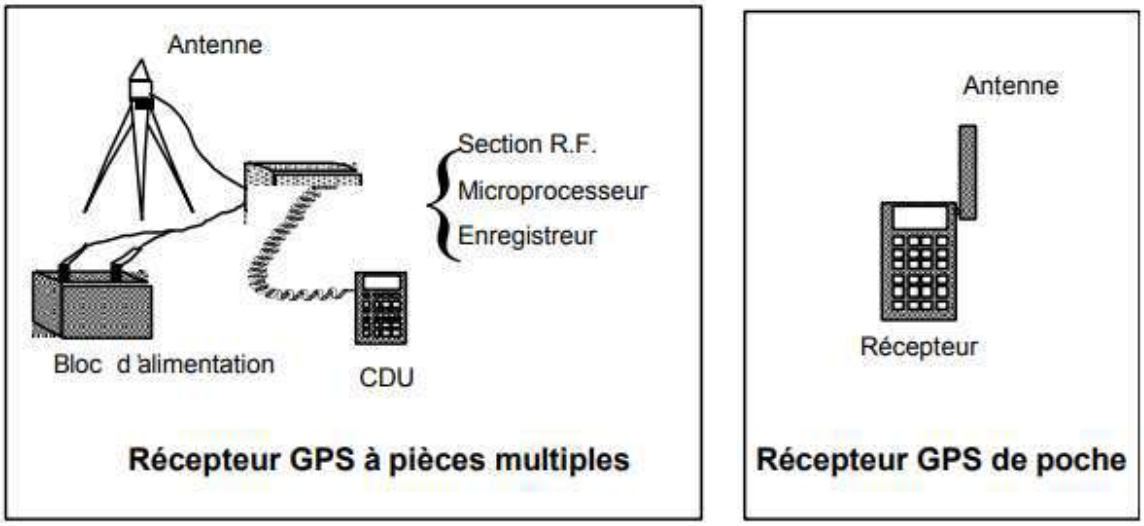


Figure 30 : Équipement GPS

3.3.1. Mode de fonctionnement GPS

Le système GPS, comme indiqué ci-dessus, peut être utilisé avec succès pour les levés géodésiques-topographiques et pour la navigation maritime, aérienne ou terrestre.
Selon l'application, les données GPS doivent être acquises et traitées de différentes manières. Les stratégies d'enquête possibles sont :

En positionnement absolu, I 'utilisateur muni d'un récepteur mesure la distance entre son récepteur et plusieurs satellites à l'aide des codes C/A ou P (ou Y). Les éphémérides du satellite transmises par le message permettent de calculer la position du satellite dans un repère terrestre à l'instant de l'observation. Chaque mesure distance entre le récepteur et un satellite définit une sphère dont le centre est le satellite et sur laquelle se trouve le récepteur. Trois satellites sont donc nécessaires pour obtenir un point intersection des trois sphères. En réalité quatre satellites son nécessaires pour tenir compte de la non-synchronisation des horloges du récepteur et des satellites. Ce type de positionnement est peu précis : de quelques mètres à 100 mètres suivant le code, la qualité des éphémérides et les dégradations

volontaires. L'intérêt de cette méthode est la possibilité de temps réel et l'obtention d'une position instantanée. Son utilisation principale est la navigation terrestre ou maritime

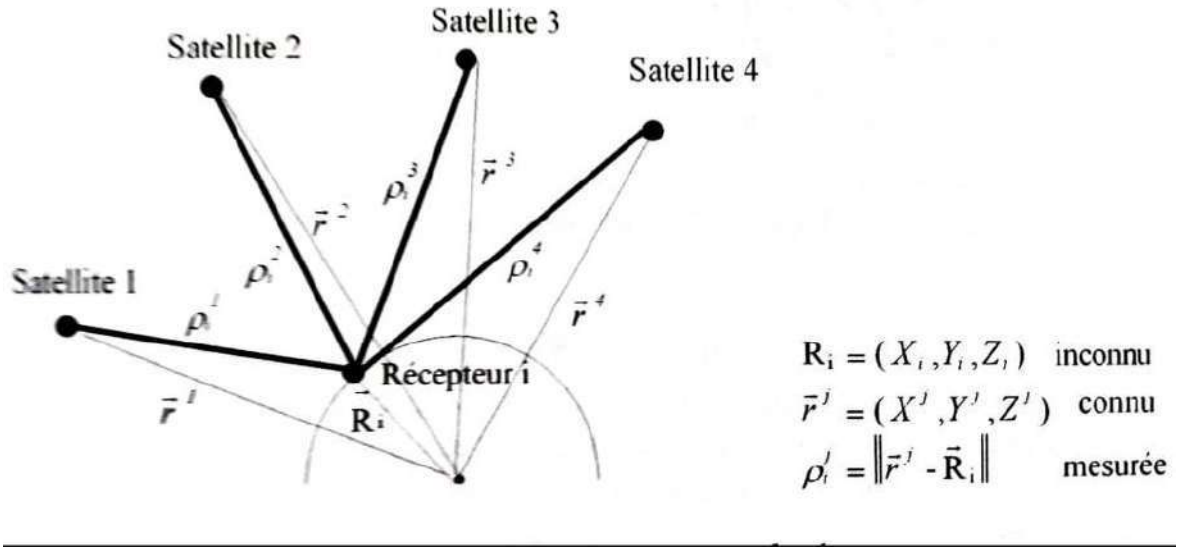


Figure 31 : positionnement absolu

En positionnement relatif la distance aux satellites est calculée soit à partir des codes P ou C/A (pseudou-distance), soit à partir de mesures de phase sur les ondes porteuses L1 et L2. Simultanément avec deux récepteurs sur les mêmes satellites et le résultat est le vecteur entre les deux récepteurs (positionnement relatif). Ce positionnement permet de diminuer l'influence des erreurs d'orbites, de propagation, d'horloges satellites et récepteurs, des dégradations volontaires et est plus précis. La gamme de précision varie selon le type de mesure : métrique en pseudo-distance (DGPS : GPS différentiel), à millimétrique (mode géodésique statique). **[11]**

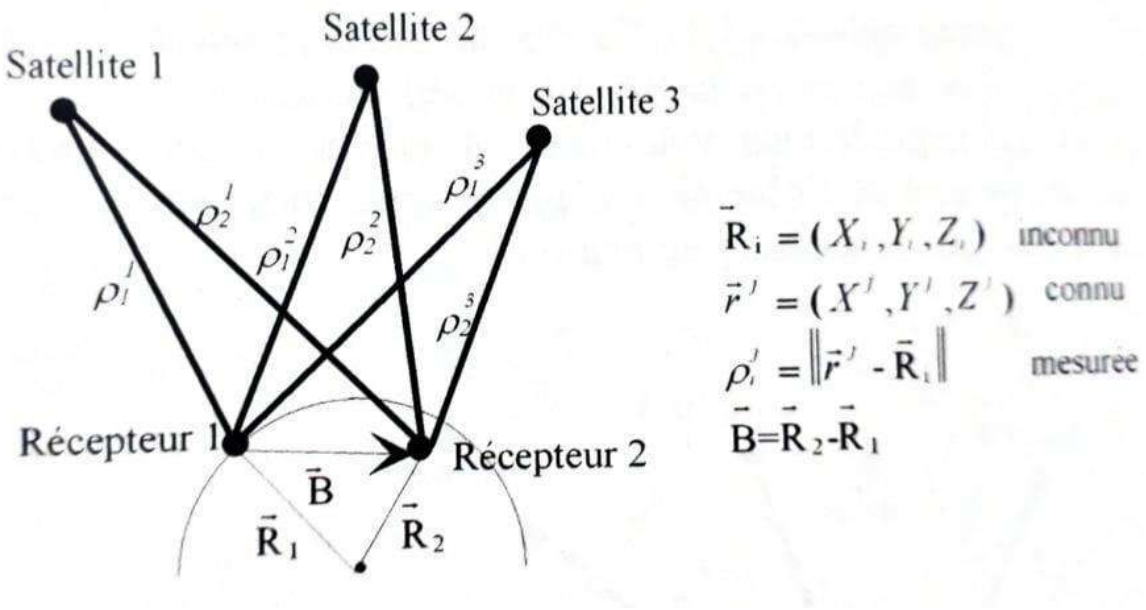


Figure 32 : positionnement relatif

4. Les mesures GPS

La mesure de base effectuée par la technologie GPS est la distance entre le satellite et le récepteur. Ceci est dérivé indirectement par des mesures auxiliaires relatives à la fois à la transmission du signal des satellites et à la dynamique d'enregistrement du récepteur.

Une copie du signal reçu est obtenue à l'intérieur du récepteur, via l'oscillateur. Les signaux reçus et générés sont ensuite comparés pour calculer leur déplacement lié à la période du signal du satellite au récepteur. **[20]**

Ensuite, le signal GPS peut être traité selon deux stratégies différentes :

• mesure de pseudo-distance ou de code, permettant le positionnement en temps réel ;

• mesure de phase ou de déplacement, portée sur les fréquences L1 et L2 qui nécessite un post-traitement des données.

4.1.La pseudo-distance

La mesure de la pseudo-distance nécessite la connaissance des codes (C/A et (ou) P (ou Y)). Ces derniers arrivent au récepteur avec un retard Δt correspondant au temps fourni lors du parcours du signal entre le satellite et le récepteur.

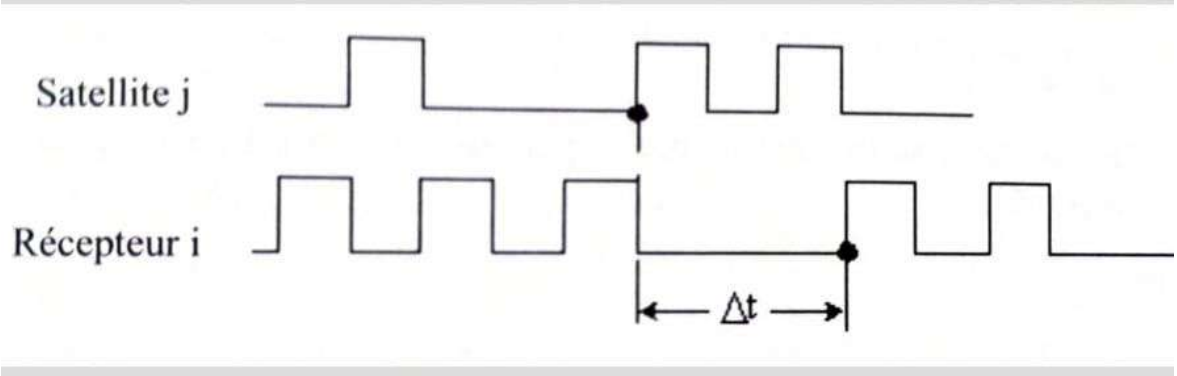


Figure 33 : Comparaison entre le signal reçu et le signal propre du récepteur sur les codes

Il est à savoir que La réplique du code satellite généré par le récepteur est synchronise avec le code satellite donc pour mesurer le décalage entre les deux signaux en temps Δt, il suffit de faire une comparaison entre eux. Cette mesure se fait par l'alignement des deux codes (en retardant le code récepteur afin qu'il soit aligné sur celui du satellite)

L'équation de pseudo distance est donnée par **[21] :**

R=c. ΔT= (T^R-T^S)=c. (ΔT^{GPS}+δT_s- δT_R- δT_a)= R_G+c. (δT_s- δT_R- δT_a)

R : Pseudo distance mesurée entre le satellite et le récepteur

R_G: Distance géométrique entre la position du satellite S à l'instant T_s (x_s, y_s, z_s) ; et la position du récepteur R à l'instantT_R (x_R, y_R, z_R) ;

C : Vitesse de propagation de la lumière dans le vide ;

T^R : Instant de réception du signal (dans l'échelle du temps du récepteur R) ;

T^S : Instant d'émission du signal (dans l'échelle du temps du satellite S) ;

ΔT^{GPS} : Temps de propagation exprimé dans l'échelle de temps GPS ;

δT_s : Décalage de l'horloge récepteur par rapport à l'échelle du temps GPS;

δT_R : Décalage de l'horloge satellite par rapport à l'échelle du temps GPS;

δT_a: Retard atmosphérique.

4.2.Mesure de phase :

La mesure de phase se fait sur les ondes porteuse L1 et L2 de longueurs d'ondes 19,0 cm et 24,4 cm

Cette mesure consiste donc à la comparaison de la phase du signal reçu par le récepteur avec la phase du signal généré par le récepteur et qui est une réplique du signal satellite.

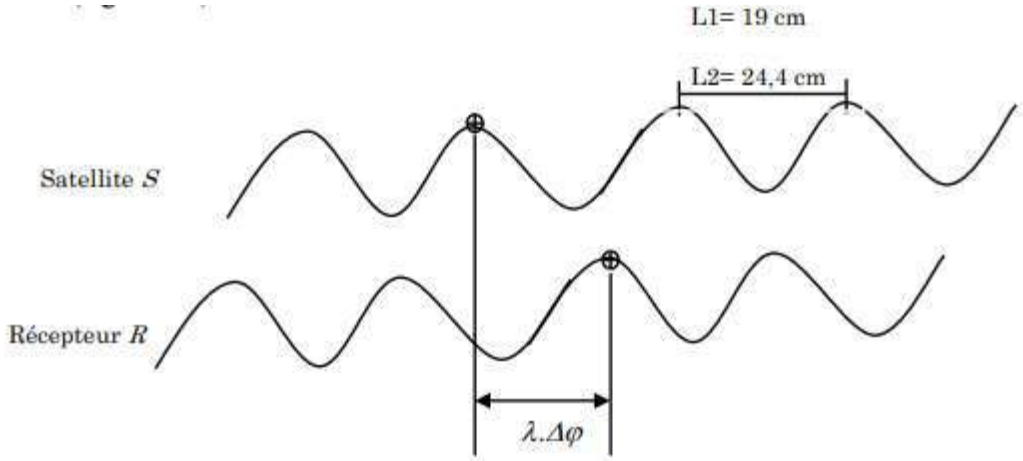


Figure 34 : Comparaison entre le signal reçu et le signal du récepteur sur la porteuse

Pour un satellite S et un récepteur R, la phase de battement $\Phi_{s,r}$ s'exprime par **[21]**

$\Phi_{s,R}(T^R)= \Phi_s(T^s)-\Phi_R(T^R)$

Avec : $\Phi_s(T^s)= \Phi_R(T^R-\Delta T) =\Phi_S(T^R)-f_S\Delta T$ (développement au 1er ordre)

$\Phi_s(T^s)$: phase du signal émis à l'instant T^s (phase de S comme observée par R)

$\Phi_R(T^R)$: phase du signal généré par le récepteur R à l'instant T^R

5. Positionnement statique et positionnement cinématique

Le positionnement GPS peut être également dit statique ou cinématique. En positionnement statique, un récepteur GPS est stationnaire alors qu'en positionnement cinématique, le récepteur recueille les données GPS tout en se déplaçant. La figure 34 illustre les notions de positionnements statique et cinématique aussi bien en positionnement autonome que relatif. Noter qu'en positionnement cinématique relatif, le récepteur, dit de contrôle, est stationnaire en un point connu alors qu'un second récepteur, dit mobile, se déplace suivant la trajectoire à positionner **[22].**

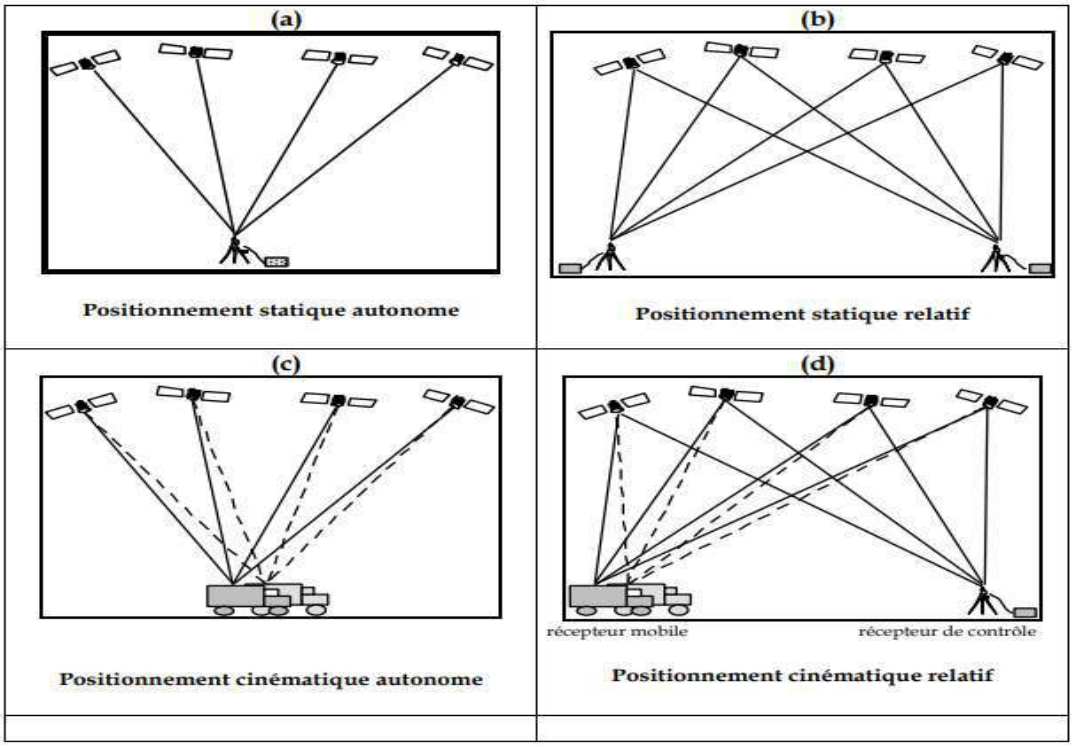


Figure 35 : Positionnements statique et cinématique

6. Sources d'erreur

La précision de la géolocalisation d'un récepteur est en théorie métrique si on utilise les pseudo-distances (code C/A) et millimétrique si la phase utilisée (porteuse L1 ou L2). En pratique les précisions observées sont généralement comprises entre 3 et 50 mètres et quelques centimètres, respectivement.

De nombreuses erreurs peuvent venir entacher le calcul de la géolocalisation (Figure35). Certaines de ces erreurs pourront être corrigées par des modèles et algorithmes et d'autres pourront être évitées par des bonnes pratiques **[23]**

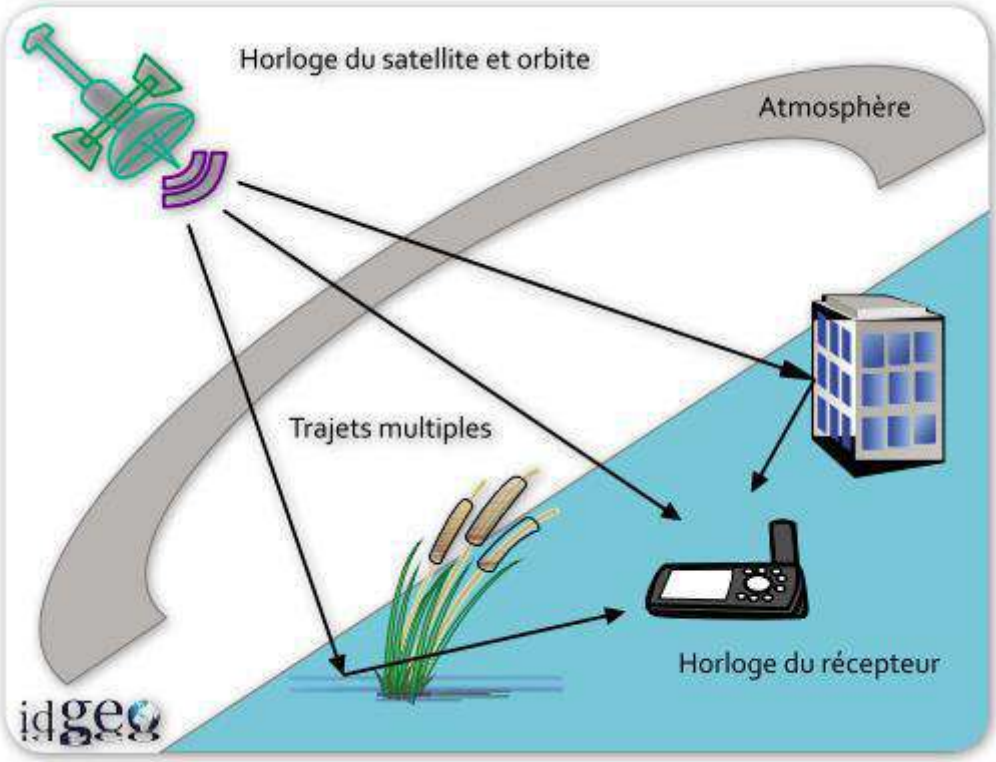


Figure 36 : Principales sources d'erreur impactant la précision du calcul de la géolocalisation

Synchronisation des horloges Bien que les horloges embarquées à bord des satellites soient extrêmement précises et que les stations de contrôle surveillent leur synchronisation, elles peuvent être sujettes à de légères dérives et entraîner des erreurs affectant la précision de la position (1 ns = 30 cm). L'horloge du récepteur a rarement la même précision que celle des satellites. Seules les différences de temps entre satellites sont donc précises. Ce problème est résolu s'il y a suffisamment d'émetteurs (quatre satellites au minimum)

Troposphère et ionosphère Les ondes ont des vitesses variables selon le milieu qu'elles traversent. Émises des satellites les ondes auront à traverser des couches de l'atmosphère plus ou moins humides, plus ou moins ionisées. La présence d'humidité et les modifications de pression de la troposphère modifient la vitesse et la direction de propagation des ondes radios. Certains récepteurs intègrent des modèles de correction. Une autre couche traversée par les signaux : l'ionosphère. Cette couche ionisée par le rayonnement solaire va modifier la vitesse de propagation du signal. La plupart des récepteurs intègrent là aussi des modèles de correction, mais en période de forte activité solaire cette correction n'est pas assez précise. Pour corriger plus finement cet effet, certains récepteurs bi-fréquences utilisent les deux fréquences L1 et L2 qui ne sont pas affectées de la même manière pour recalculer plus précisément la position.

Géométrie de la distribution des satellites dans l'espace : dilution de la précision (DOP) Si les satellites visibles sont très proches dans l'espace, la précision sera moins bonne que s'ils sont répartis régulièrement sur une large étendue au-dessus de l'utilisateur. Les distances utilisées pour le calcul doivent en quelques sortes être le plus hétérogènes possible afin d'échantillonner au mieux l'espace au-dessus du récepteur. Les effets de la géométrie du système peuvent être appréciés par un paramètre : le DOP (dilution of précision). La précision attendue est d'autant plus importante que le DOP est petit. Certains dispositifs permettent de filtrer des positions calculées en dessous d'un certain DOP. Ce coefficient DOP peut se décliner horizontalement (HDOP) et verticalement (VDOP)

Cryptage Le GPS étant conçu par et pour les militaires américains, un certain nombre de moyens ont été mis en place pour empêcher l'utilisation du système par un hostile. Ce but est atteint par deux brouillages : la dégradation volontaire de la précision du positionnement et le cryptage du code P. L'accès sélectif ou « selective availability » (SA) est une dégradation de la précision des horloges embarquées à bord des satellites. Cette dégradation volontaire a fonctionné de manière permanente jusqu'en mai 2000. L'anti-leurrage ou « anti spoofing » (AS) est le cryptage du code P (précis) transformé en code dit « Y ». Ce cryptage rend impossible la mesure directe des pseudo-distances basée sur le code P.

Réflexion et perte des signaux : trajets multiples et effets « canyons » La topographie du milieu et sa structuration peuvent influer sur la qualité de la géolocalisation. Le relief ou des bâtiments peuvent occulter des satellites et influer directement sur la précision du positionnement. Ce phénomène de « canyon » se rencontre particulièrement en zone urbaine

et en zone montagneuse. Les signaux de satellites peuvent également être réfléchis par des surfaces d'eau voire même des murs : c'est le problème des multi-trajets. Dans ce cas le géonavigateur ne réceptionne que des échos des signaux et la géolocalisation calculée tarde à se stabiliser.

Erreur	Ordre de grandeur	
Horloge du satellite	de 10 à 30m	(incluant les effets de l'accès sélectif et en supposant que les corrections diffusées ont été apportées)
Orbite	de 5 m à 25 m	(éphémérides diffusées)
Ionosphère	de 1 à 100 m	(après application du modèle diffusé)
	1 m	(observations bifréquences)
Troposphère	0,5 m	(après application d'un modèle)
Multitrajets		
Obs. du code C/A	de 50 cm à 20 m	(selon l'équipement GPS et le lieu)
Obs. de phase	jusqu'à quelques cm	(selon l'équipement GPS et le lieu)
Bruit du récepteur		
Obs. du code C/A	de 10 cm à 3 m	(selon l'équipement GPS)
Obs. de phase	de 0,5 à 5 mm	(selon l'équipement GPS)

Tableau 3 : les déférentes erreurs GPS

7. Conclusion

Dans ce chapitre, nous avons présenté les concepts clés du système GPS, tout en mettant l'accent sur les éléments essentiels et nécessaires afin de mieux comprendre son fonctionnement. Nous avons également décrit le principe du positionnement par GPS, source d'erreurs et les systèmes de référence, le signal GPS et ses composants ainsi que les différents types de mesures utilisées.

Dans le prochain chapitre, nous détaillerons l'architecture de plate-forme dotée du nouveau capteur GPS. Nos choix quant à l'utilisation des cartes électroniques embarquées et des algorithmes de commandes pour la navigation du robot mobile seront testes au travers d'application.

CHAPITRE IV :

Description de la plateforme robot mobile

IV. CHAPPITRE IV DESCRIPTION DE LA PLATEFORME ROBOT MOBILE

1. Introduction

Comme il est mentionné dans les chapitres précédents, nous avons constaté que la réalisation de la commande du robot nécessite une stratégie de commande spécifique, donc afin de Controller le déplacement du robot nous devons utiliser un asservissement par navigation en utilisant un capteur GPS. Pour atteindre cet objectif nous n'avons qu'utilisé les coordonné GPS.

L'objectif que nous avons devons atteindre sont :

- L'exploiter de la plate-forme robotique composé de (Raspberry، l'ensemble Arduino /Shield) et intégrer un nouveau capteur GPS pour améliore son autonomie.
- L'asservissement de notre robot à l'aide d'un algorithme qui lui permet d'attendre point cible en utilisant que les données de géolocalisation

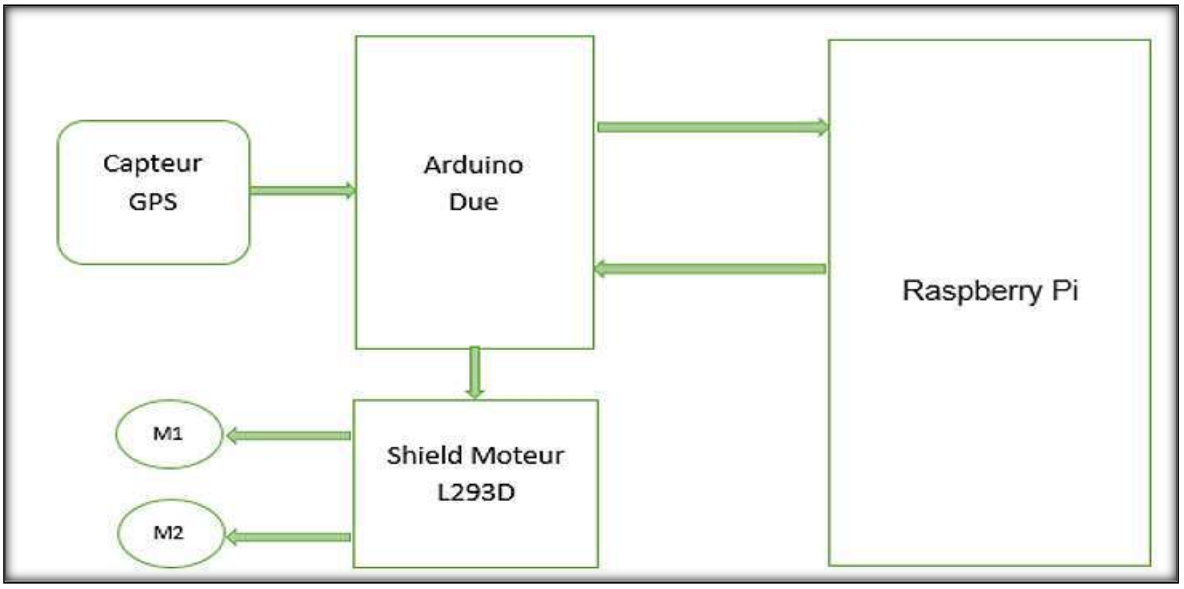


Figure 37 : Schéma bloc du projet.

Notre schéma blocs est composé de trois blocs :

- Les deux premiers blocs représentent l'association du robot mobile (avec DC moteurs) et de la carte Arduino qui permet de gérer le capteur GPS et commander les actionneurs (les moteurs du robot).

- Concernant le dernier bloc (la Carte Raspberry)، il sert à calculer les données reçues et les transférer sous forme de commande permettant au robot d'arriver au point cible.

Dans le premier chapitre nous avons proposé quelques définitions sur les différents types de robots mobiles utilisés dans les différents projets, le robot utilisé dans notre projet, est un robot de type tricycle, composé de deux roues motrices arrière et une roue de guidage avant, la raison pour laquelle nous avons choisi ce type de robot qui est techniquement un Arduino robot est la simplicité de son utilisation , nous avons pu développer une plateforme qui permet de guider un robot mobile par GPS

2. Robot mobile Arduino

2.1. Description

L'architecture d'un robot mobile de type tricycle est donnée sur la figure 37, il utilise une roue motrice avant et deux roues passives arrières (ou vice versa). Le mouvement est conféré au robot par deux actions : la vitesse longitudinale et l'orientation de la roue orientable, ce type de robots est très connu dans les applications des AGV pour leur simplicité inhérente. **[24]**

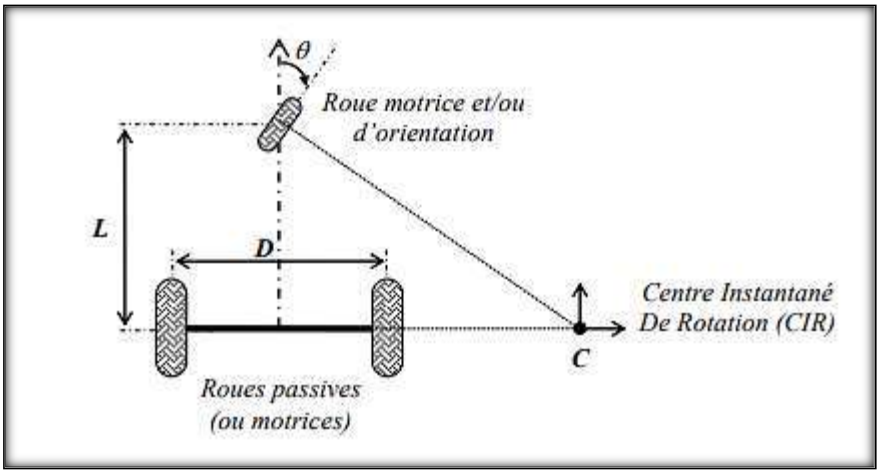


Figure 38 : Robot mobile de type tricycle

2.2. Modélisation cinématique du robot mobile

Pour la modélisation cinématique du robot mobile, nous avons besoin de déterminer les équations de quelques paramètres du robot. La figure 38 représente le paramétrage nécessaire pour la modélisation de notre robot mobile **[24]**

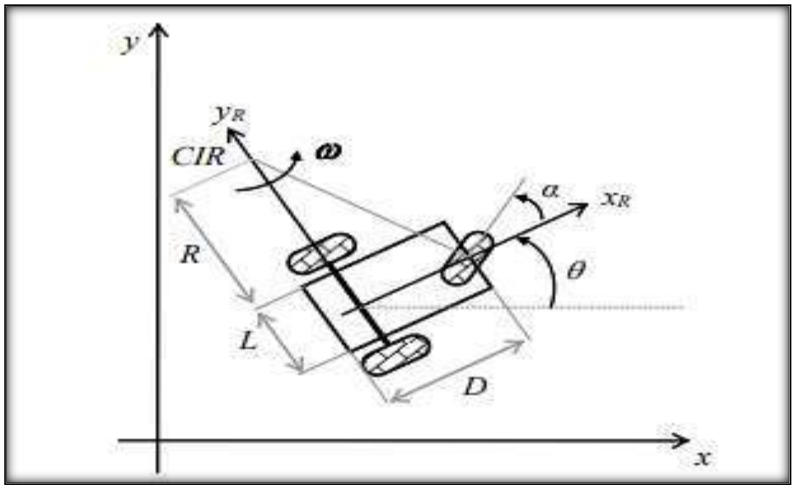


Figure 39 : Paramétrage du robot mobile

L'angle de braquage est donné par :

$\alpha = \text{arctg}\left(\frac{D}{R}\right)$

$R = \frac{L(V_d+V_g)}{2(V_d-V_g}$

$\alpha = \text{arctg}\left(\frac{2D(V_d-V_g)}{L(V_d-V_g)}\right)$

Avec :

α : L'angle de braquage ;

θ : L'orientation du robot ;

ω : La vitesse de rotation du robot autour de son centre instantané de rotation (CIR) ;

R : Le rayon de courbure ;

L : L'entre axe ;

D : La longueur du châssis ;

V_d : La vitesse linéaire de la roue droite;

V_g : La vitesse linéaire de la roue gauche;

V : La vitesse linéaire du robot.

Le modèle cinématique d'un robot mobile peut est donné par :

$\dot{x}$ =V cos θ

$\dot{y}$= V sin θ

$\dot{\theta}=\frac{V}{D}$tngα

L'intégration des équations cinématiques pour des déplacements élémentaires entre deux instants t et (t + Δt) donne :

− Si $\dot{\alpha} \neq 0$, ($V_g \neq V_d$) l'orientation du robot par rapport à l'axe horizontal à l'instant (t + Δt) est donnée par : $\theta(t+\Delta t) = \theta(t)+\frac{V}{D}\int_t^{t+\Delta t} tg\alpha dt$

$$= \theta(t)+\frac{V}{D\dot{\alpha}}(-\ln|\cos\alpha(t+\Delta t) + \ln|\cos\alpha(t)|$$

- Si $\dot{\alpha} = 0$, ($V_g = V_d$) ,l'orientation du robot à l'instant (t + Δt) est donné par:

$\theta(t+\Delta t) = \theta(t)+\frac{V}{D}tg\alpha\ \Delta t$

La position du robot à l'instant (t + Δt) peut être calculée comme suit.

-si $\dot{\theta} \neq 0$,$\begin{cases} x(t+\Delta t) = x(t) + \frac{V}{\dot{\theta}}\left(sin\theta(t+\Delta t) - \sin\theta(t)\right), \\ y(t+\Delta t) = y(t) + \frac{V}{\dot{\theta}}\left(cos\theta(t+\Delta t) - \cos\theta(t)\right) \end{cases}$

- si $\dot{\theta} = 0$,$\begin{cases} x(t+\Delta t) = x(t) + V\cos\theta\Delta t\ , \\ y(t+\Delta t) = y(t) + V\ sin\ \theta\Delta t\ , \end{cases}$

3. L'architecture matérielle de gestion et de commande de la plateforme robot mobile

3.1. L'extension (Shield) de commande des moteurs

Cette extension (Adafruit Motor Shield) permet de commander 2 moteurs DC avec notre carte Arduino, en contrôlant la vitesse et la direction de chacun d'eux indépendamment.

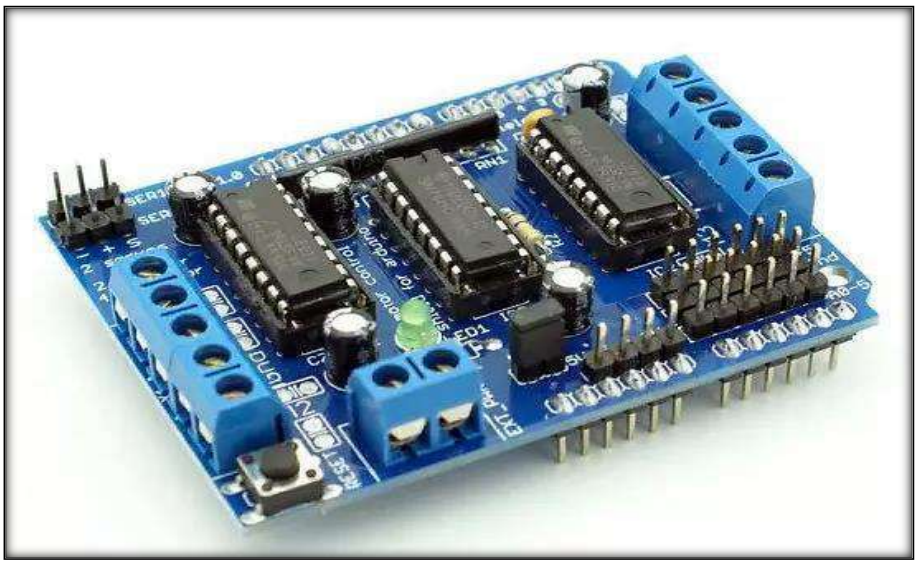


Figure 40 : Shield contrôle de moteurs L293D

A- Caractéristiques : **[25]**

- 2 interfaces pour servomoteurs 5V, connectés au timer haute résolution de l'Arduino.
- Peut piloter 4 moteurs à courant continu DC, ou 2 moteurs pas à pas, ou 2 servos 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).
- Fournit 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 désactivent 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 Méga, Diecimila & UNO

B- Les moteurs et le registre à décalage

Le blindage du moteur peut contrôler jusqu'à quatre moteurs à courant continu à l'aide de deux circuits intégrés de moteur à moteur à double pont L293D. Vous pouvez lire la fiche technique pour plus de détails. Comme le montre la Figure 40, chaque CI possède quatre entrées numériques qui commandent bi directionnellement deux moteurs à courant continu. Les entrées sont par paires, et chaque paire contrôle un moteur. Par exemple, si la broche 2 est réglée en haut et que la broche 7 est bas, le moteur 1 est allumé. Si les entrées sont inversées, le moteur est à nouveau activé, mais son sens de rotation est inversé, et si les deux sont bas, le moteur est éteint. Il existe d'autres configurations possibles, mais celle-ci est supportée par le bouclier du moteur **[26]**

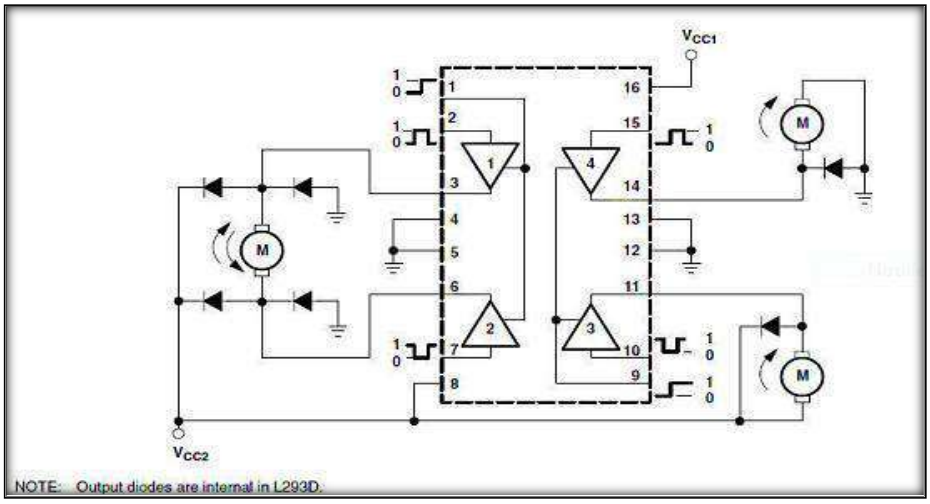


Figure 41 : fonctionnement de L293D.

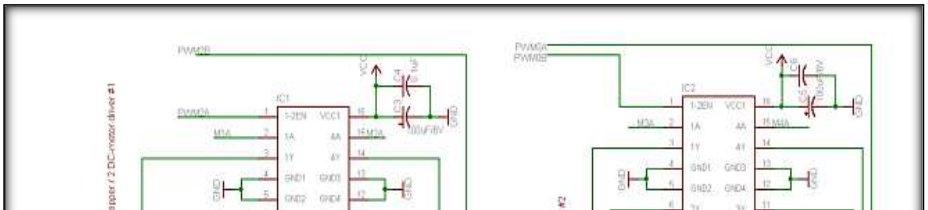


Figure 42 : montage L293D avec Shield.

Les entrées des moteurs sont effectivement câblées sur les sorties d'un registre à décalage 74HCT595N à 8 bits Figure 42 Cet IC vous permet d'alimenter en bits en série, mais il les présente ensuite comme des sorties parallèles. Il est contrôlé par quatre entrées numériques, de sorte qu'il vous permet de contrôler huit entrées aux moteurs (donc quatre moteurs) en utilisant quatre broches de l'Arduino.

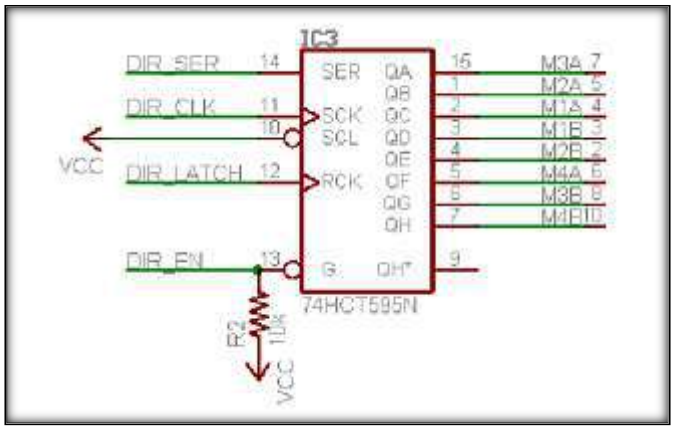


Figure 43 : IC 74HCT595N

Les broches étiquetées QA à QH sont les broches de sortie parallèles, et vous pouvez les considérer comme des bits 0 à 7. Les chiffres immédiatement à droite du CI sont des chiffres de broches plutôt que des nombres de bits, bien qu'ils coïncident surtout. QA est un bit zéro, broche 15. La fiche technique les appelle Q0 à Q7, de sorte que je ne sais pas pourquoi le schéma utilise une convention différente. Comme vous pouvez le voir, le moteur 1 est contrôlé par les bits 2 et 3, le moteur 2 par les bits 1 et 4, etc. Les broches illustrées à gauche sur Tableau4 et Figure 42 "74HC595" sont les quatre broches d'entrée plus la réinitialisation et la validation de sortie. La convention de dénomination d'Adafruit est en

contradiction avec celle fournie dans la fiche technique 74HCT595N à 8 bits de registre à décalage Philips. Voici un tableau de référence croisée.

ARDUINO BROCHES	IC 74HCT595N BROCHES
8	SERIE DATA DS
4	CLK SH_CP
12	LATCH ST_CP
7	SELECTION EN

Tableau 4 : brochage d'IC 74HCT595N

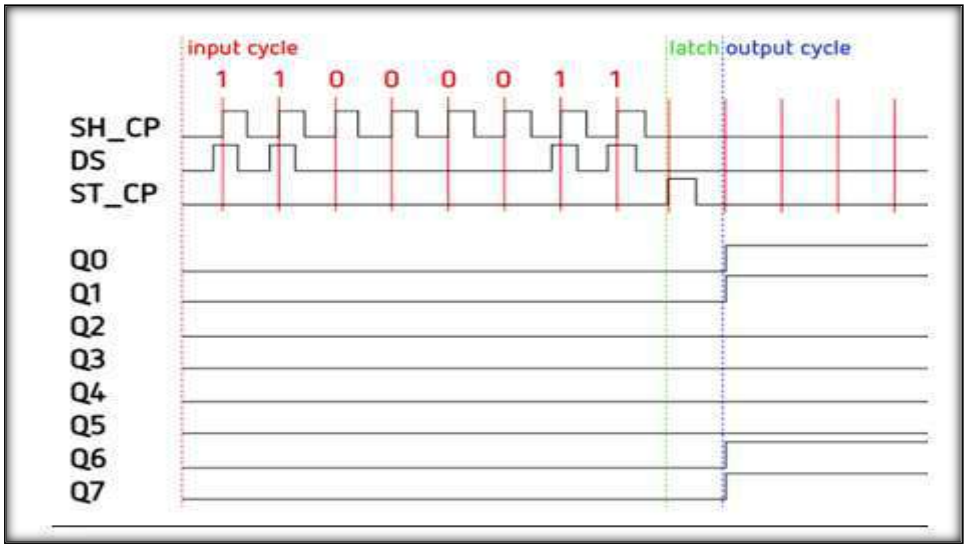


Figure 44 : fonctionnement de IC 74HCT595N

3.2. Le capteur GPS

Le récepteur GPS est l'unité de réception, de traitement interne, de navigation, et si nécessaire de sauvegarde des mesures GPS en vue du post-traitement.

3.2.1. Les différents types de GPS

Il existe différents types de récepteurs GPS :

GPS de poche, montre, traceur, GPS fixe, PND (Personale Navigation Device), GPS souris, GPS Bluetooth, etc., chacun ayant des fonctions et une utilisation particulière, les plus faciles à exploiter pour enregistrer des traces sont les GPS portables, parmi lesquels nous distinguons les GPS de poche, les PND, les PDA (Personale Digital Assistant ou ordinateur

de poche), les smartphones et téléphones mobiles AGPS, les GPS de type montre que l'on porte au poignet et ceux de type traceur **[37]**

3.2.2. Principes de fonctionnement et d'architecture de récepteur GPS

Les récepteurs implémentent quatre fonctions principales : la réception des signaux, le traitement des signaux, l'oscillateur local et le calcul de position. Leur schéma de fonctionnement est illustré à la figure suivant **[36].**

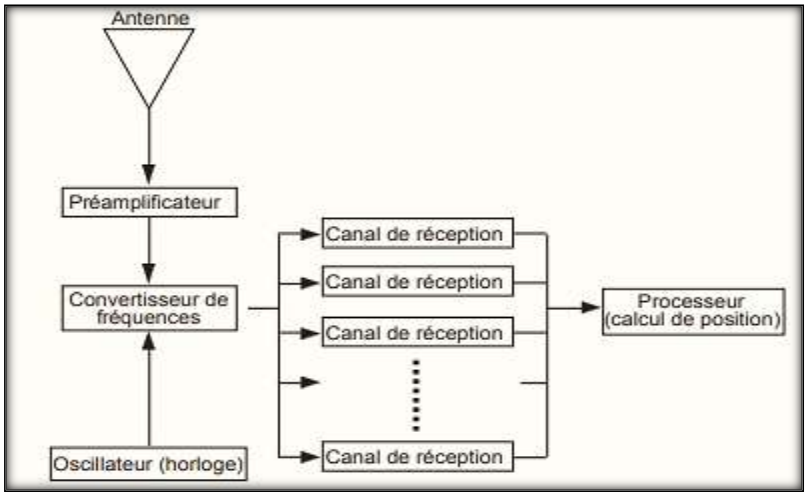


Figure 45 : Schéma de fonctionnement des récepteurs GPS

Le récepteur GPS reçoit la fréquence LI à un niveau de signal de -130 dBm et la largeur de sa bande est de 2,046 MHz. Un premier étage constitué d'un amplificateur faible bruit LNA (Low Noise Amplifier) amplifie le signal. À la suite de cet amplificateur, un filtre passe-bande isole les fréquences utiles du signal (fréquences radio du satellite) et un amplificateur RF ajoute un gain supplémentaire, à la sortie du mélangeur, le signal est de nouveau filtré à l'aide d'un filtre bande de base.

Enfin, un amplificateur bande de base, qui permet de passer de la fréquence RF en bande de base, amplifie le signal contenu dans la bande de fréquence désirée. Le DSP génère le code pseudo-aléatoire et le compare au signal reçu. De plus, il analyse l'almanach et apporte les corrections appropriées. Basé sur ce principe, le récepteur conçu respecte les normes de la réalité, mais l'analyse des almanachs n'est pas considérée pour ce projet. Cependant, cet aspect pourra être ajouté au projet afin d'améliorer les performances du modèle. **[27].**

3.2.3. MODULE GPS utilisé (SIM 28) :

Le module GPS SEN10752P du fabricant SEEED est un Shield équipé d'un capteur GPS "E-1612-UB" et d'un module GPS SIM28 compatible avec le protocole NMEA et le protocole propriétaire U-blox 6. La communication avec module est réalisée au moyen d'une liaison série asynchrone (UART).

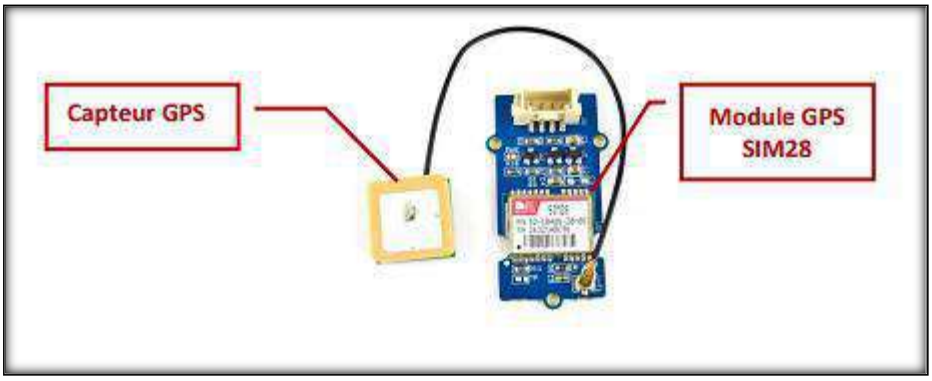


Figure 46 : Capteur GPS SIM 28

3.2.4. Caractéristiques

Les caractéristiques du capteur GPS sont expliquées dans le tableau suivant :

[28]

Caractéristiques	Valeurs
Tension d'alimentation	3,3 V ou 5 V
Consommation	40 mA sous 3,3 V
Sensibilité	-160 dBm
Vitesse de transmission	4800 bits/s (version SIM28)
Dimensions shield	40 x 20 x 13 mm
Dimensions capteur GPS	15 x 15 x 7 mm
Température de fonctionnement	– 40 °C à + 85 °C
Protocoles	NMEA, U-blox 6

Tableau 5 : Les caractéristiques du capteur GPS SIM28

3.2.5. Schéma de câblage

Le Shield GPS est alimenté sous 5 V. La liaison série entre la carte Arduino et le Shield GPS sera réalisée par une liaison série logicielle sur les broches numériques D2 (Rx) et D3 (Tx) de la carte Arduino. La carte Arduino ne va que recevoir des données issues du Shield GPS.

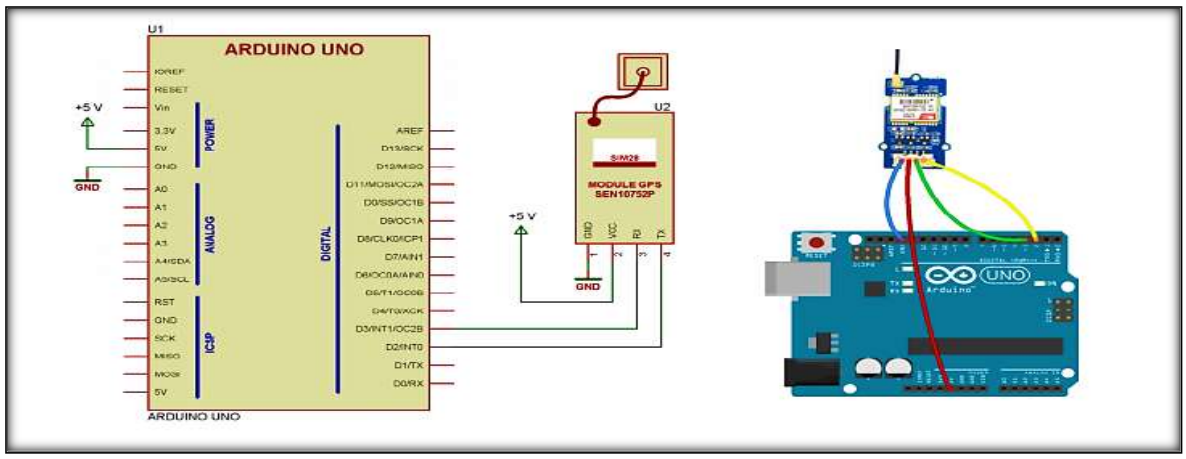


Figure 47 : Schéma de câblage

3.2.6. Résultats moniteur série

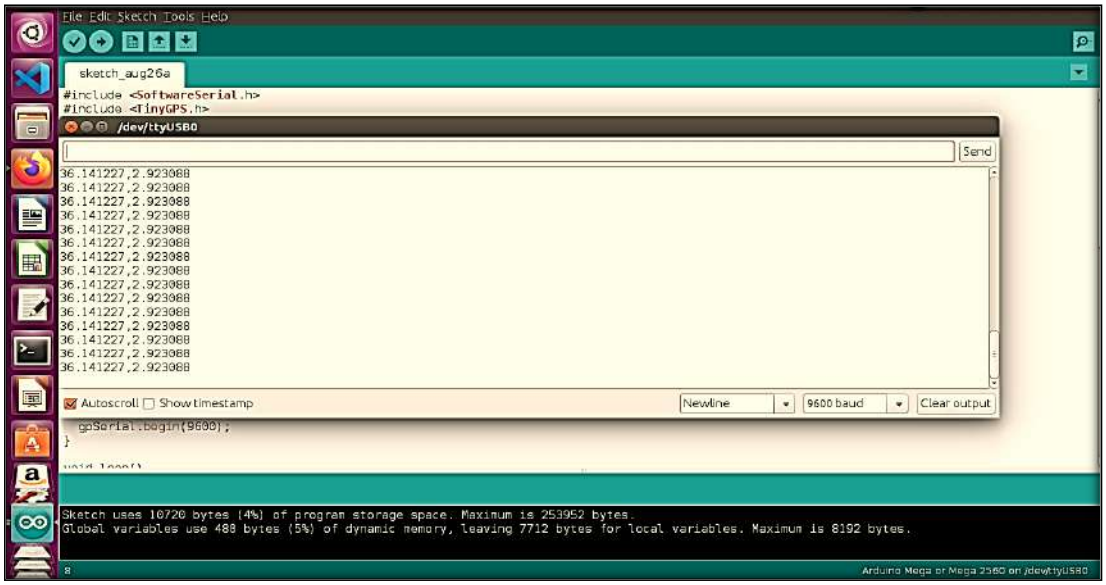


Figure 48 : Résultats moniteur série

3.2.7. Vérification du résultat sur « Google Maps »

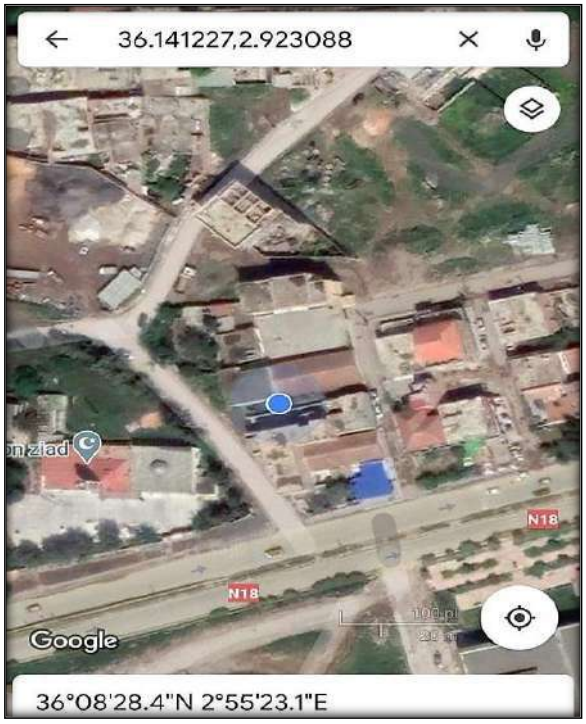


Figure 49 : Résultats sur « Google Maps »

4. La carte Raspberry Pi : le cerveau

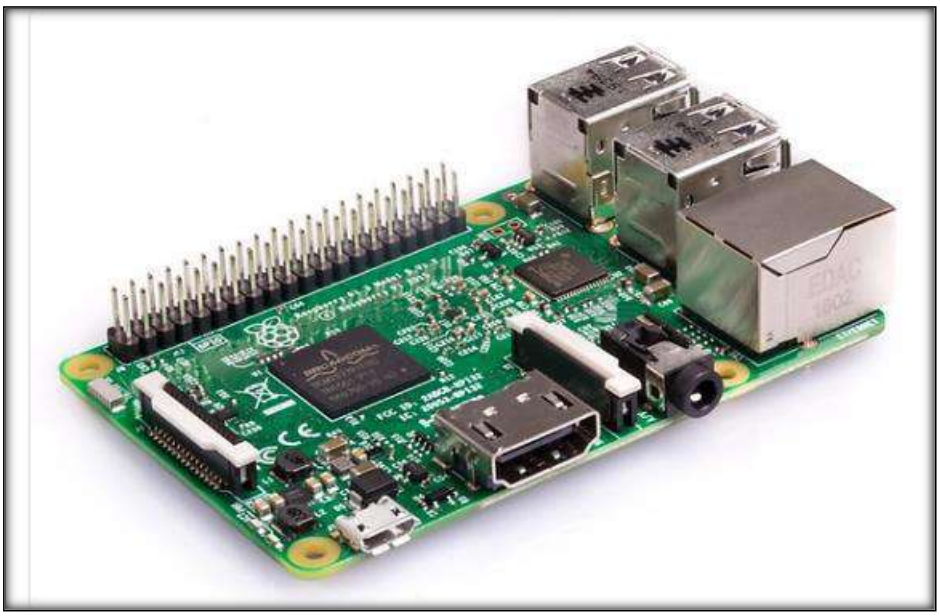


Figure 50 : la carte Raspberry Pi utilisé.

Raspberry Pi est un système d'exploitation libre et gratuit, cette carte est le cerveau de notre robot par ce qu'elle est responsable de l'acquisition et des calculs de lois de commande.

Elle se compose d'une RAM et un CPU puissant qui fonction sous le système d'exploitation LINUX, le choix de ce modelé (Raspberry pi 3 Model B) était basé sur les caractéristiques que nous avons cités dans Chapitre 2, le système linux nous a facilité la programmation de l'ensemble des algorithmes de commandes en langage python ainsi que l'exploitation des sources ouvertes (open source),

Dans le reste de ce chapitre nous allons décrire d'une façon brève les différentes techniques que nous avons utilisées pour contrôler notre robot mobile ainsi :

- Le guidage par navigation basé sur l'exploitation des données fournie par GPS ;
- Des programmes en langage python qui ont des procédures de transmissions des consignes via le port USB vers l'Arduino/Shield qui va les convertir en mouvement des moteurs de robot.

5. L'organigramme du contrôleur de robot mobile

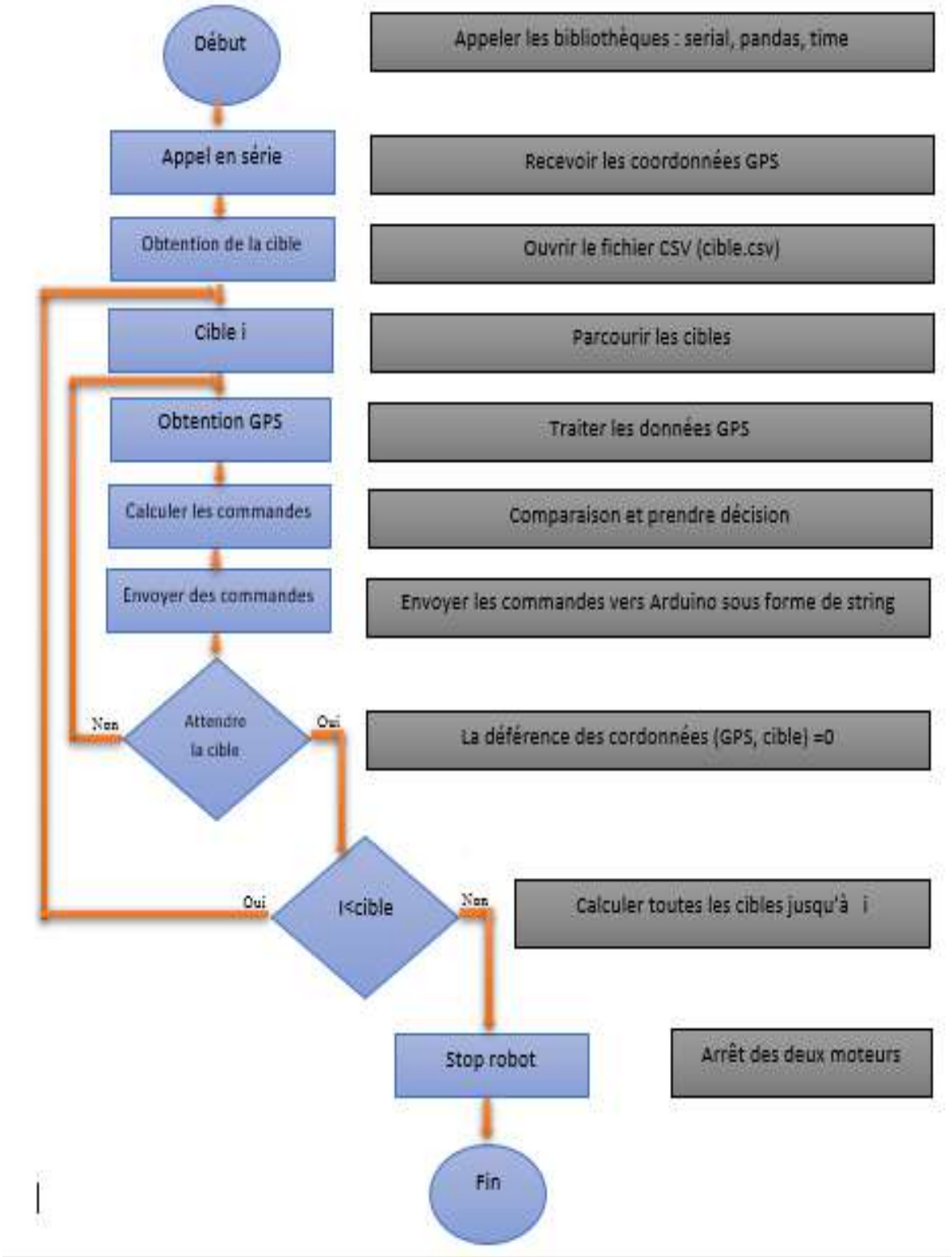


Figure 51 : L'organigramme du contrôleur de robot mobile

6. Description du programme d'asservissement par navigation :

L'organigramme ci-dessus présente d'une façon générale le fonctionnement du programme utilisé dans la réalisation de l'asservissement du robot mobile. Au début de programme nous avons eu recours aux bibliothèques suivantes :

-serial: est une bibliothèque du langage de Python permet d'accès au port série.

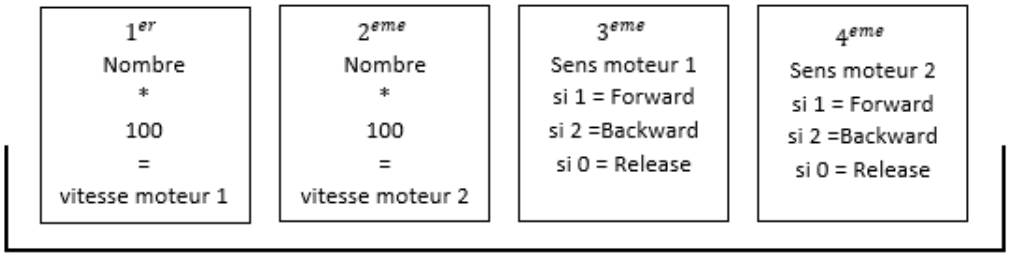
-pandas: est bibliothèque permettant la manipulation et l'analyse des données

-time: dans cette bibliothèque il y a plusieurs fonctions concerné par le temps

Par la suite, nous nous sommes servis des fonctions pour récupérer les coordonnées GPS depuis l'Arduino. Ensuite nous avons créé un fichier dans lequel nous avons classé les cibles selon le cheminement du robot que nous avons choisi.

Puis, nous avons développé un algorithme qui nous sert à faire la comparaison des coordonnées GPS de robot avec celles de la cible ; cette comparaison se fait entre latitude et longitude tout au long du passage du robot vers la cible (comparaison entre les nouvelles coordonnées GPS de robot avec cordonnée cible), cette opération est faite à la base des conditions qui se trouve dans le code source (code python).

Après avoir terminé la comparaison, le programme python envoie un code (sous forme String) vers l'Arduino qui va à son tour le convertir en commandes moteur, le dessin ci-dessous explique la signification de chaque chiffre du code envoyé :



Une fois les commandes sont reçues le robot commence à marche jusqu' a qu'il atteint la cible.

7. Résultat

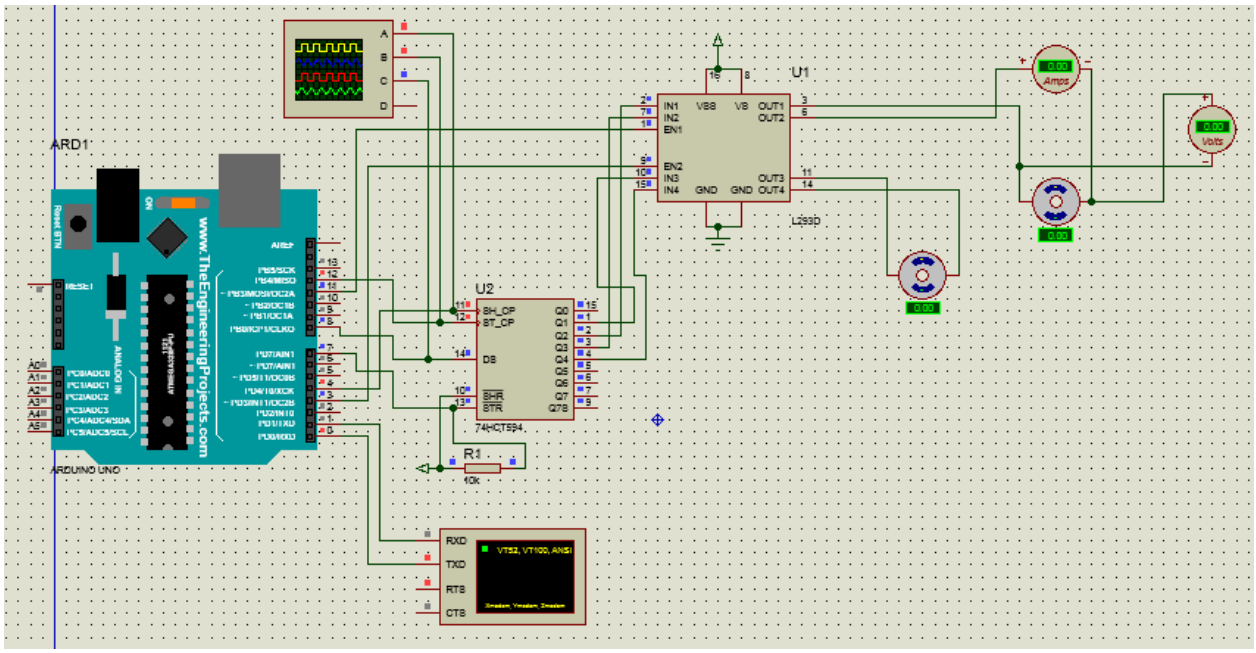


Figure 50 : simulation DC moteur Shield

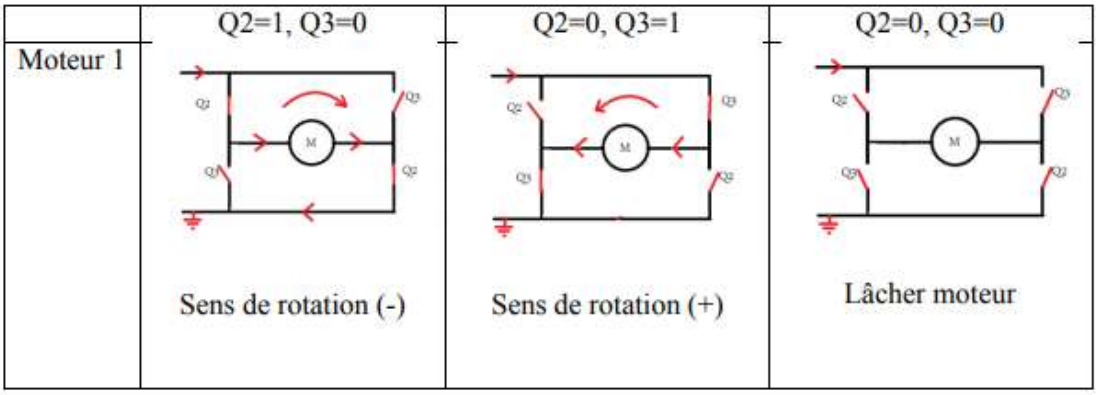


Figure 51 : Synoptique du système d'alimentation en pont des moteurs du robot.

Afin de faire la simulation DC moteur/Shield et Arduino à l'aide du programme Proteus, nous avons dû connaitre les entrées Qi pour la gestion des sens de rotations des moteurs

Depuis le schéma (Figure 51), nous avons pu connaitre le sens de rotation des moteurs et pour bien expliquer nous avons utilisé le tableau suivant :

Moteur 1	Q2=1	Q3=0	En avant
Moteur 1	Q2=0	Q3=1	En marche arrière
Moteur 1	Q2=0	Q3=0	Lâcher moteur
Moteur 2	Q1=1	Q4=0	En avant
Moteur 2	Q1=0	Q4=1	En marche arrière
Moteur 2	Q1=0	Q4=0	Lâcher moteur

Tableau 6 : Le sens de rotation des moteurs Shield l293d

Nous avons fait notre simulation de circuit à l'aide du programme Proteus comme il est montré dans la **(Figure 50)** et par l'utilisation de l'oscilloscope, ampèremètre, voltmètre nous avons eu quelque résultat concernant (PWM, courant, tension).

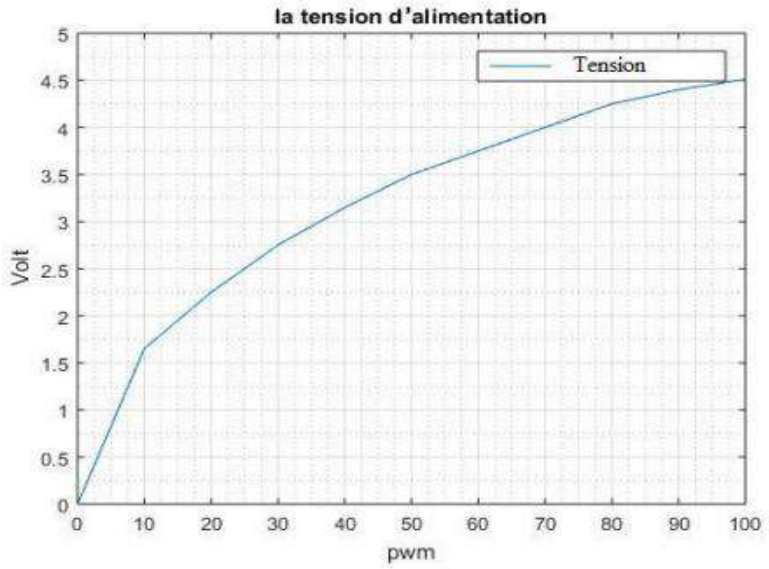


Figure 52 : graphe de la tension d'alimentation du moteur en fonction de PWM.

Le graphe illustré dans la **(Figure 52)** représente l'évolution de la tension d'alimentation (5V) En fonction de PWM

Nous avons remarqué que son évolution n'est pas linéaire à cause de lois des commandes utilisées.

Le graphe ci-dessous nous montre la variation de de courant d'excitation en fonction PWM

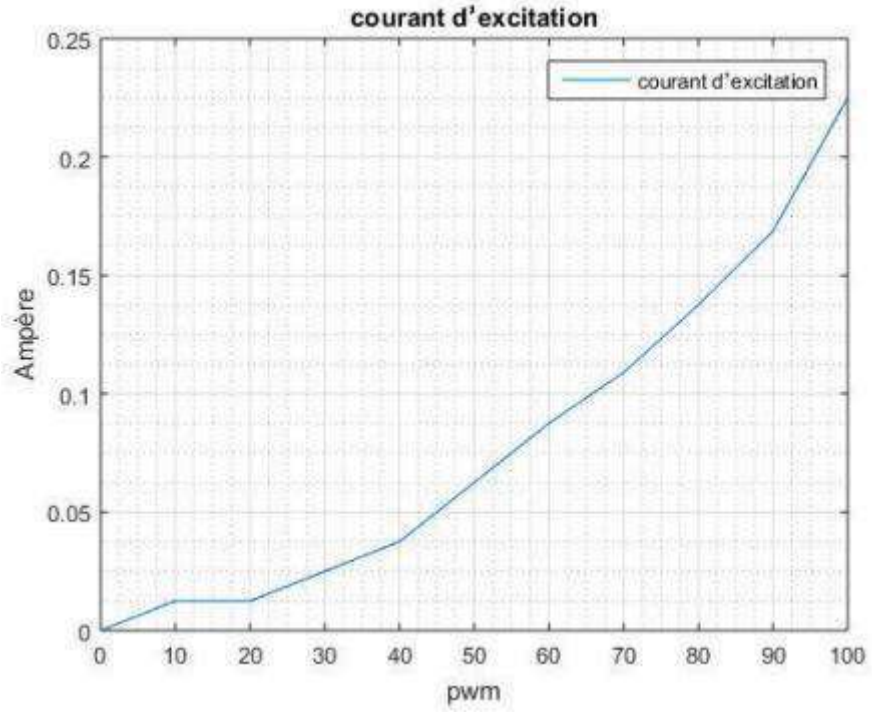


Figure 53 : : graphe du courant d'excitation du moteur en fonction de PWM

Après avoir étudié les graphes, nous avons remarqué que la tentions et le courant d'excitation influencent sur les moteurs.

8. Conclusion

Dans ce chapitre, nous avons décrit avec détail le fonctionnement ainsi que les modifications faites sur la plateforme robot mobile pour réaliser le suivi d'une cible en utilisant un capteur GPS.

Au début, nous avons mis en évidence le schéma de blocs, le système embarqué utilisé qui est composé de capteur GPS sim28, carte Raspberry Pi3 et de la carte Arduino / Shield

Ensuite, pour faire la transmission des données et des commandes nous avons utilisé la communication série via le port USB.

Par la suite, nous avons expliqué les rôles des étages de notre organigramme avec détail et nous avons constaté que le rôle le plus important est l'acquisition, le traitement des données GPS

Un programme écrit en langage python regroupe les opérations citées précédemment et la loi de commande et les exécute sur la carte Raspberry pour contrôler le robot mobile.

Enfin, l'exécution de notre expérience sur terrain et les résultats obtenus nous ont permis de justifier nos choix en ce qui concerne les systèmes embarqués et les algorithmes de commande utilisés

Pendant la réalisation de notre expérience nous avons rencontré quelques problèmes :

- Durant sa trajectoire le robot s'arrête plusieurs fois (arrêt de quelques secondes pour le traitement des données et la prise de décision concernant son orientation).
- Le robot n'a pas atteint la cible mais il s'est arrêté près de la cible (5m)

Ces erreurs permettent d'ouvrir d'autres pistes de recherche à l'avenir, il serait préférable dans les recherches à venir d'utiliser :

- Un capteur GPS puissant plus chère que le nôtre ;
- Ou bien ajouter un deuxième capteur GPS pour un positionnement relatif.

Pour avoir de meilleurs résultats

CONCLUSION GENERALE

La poursuite d'une trajectoire est une tâche importante que doit exécuter un robot mobile avec le minimum d'erreurs. Elle constitue la base de toute mission du robot, pour les robots à commande différentielle, le maintien du robot sur son chemin revient à régler les vitesses des roues motrices de façon à donner au robot l'orientation désirée.

En robotique mobile la poursuite de trajectoire est une tâche élémentaire, qui représente la base de toute autre tâche du robot. Au terme de ce travail, nous avons apporté des modifications majeures à la plate-forme robot mobile pour réaliser le suivi d'une cible.

En effet, les différentes applications que nous avons réalisées nous ont montrées que nous avons intégré avec succès le capteur GPS à la plate-forme robotique existante. Nous avons pu réaliser la commande de ce robot mobile en n'utilisant que les données de géolocalisation. Cependant, nous sommes limités par la précision de notre capteur. Ce dernier n'est précis qu'à cinq mètres près.

D'où les perspectives futures que nous envisageons pour compléter ce travail :

- L'utilisation de capteurs GPS plus performant ou mettre en œuvre un système de GPS différentiel pour réduire l'erreur de positionnement à quelques dizaines de centimètres.
- Remplacer la carte Raspberry par une autre carte plus performante (carte FPGA par exemple).

.

Bibliographie

[1] Mazen Al Homsi, “ main humaine, main robotique, qu’elle est la différence ? ”, UTBM, 2016
http://http://mrmhutbm.e-monsite.com/pages/apercu-historique.html

[2] Alain COURCELLE, “Localisation d’un robot mobile : Application à l’aide à la mobilité
des personnes handicapées moteur”, Thèse de doctorat, l’Université de Metz, France, janvier
2000.

[3] Généralités sur la robotique mobile par ISTIA, Université Angers

[4] ROBOTIQUE, ISTIA, Université Angers Jean-Louis Boimond

[5] Damien LHOMME-DESAGES, “Commande d’un robot mobile rapide à roues non Directionnelles sur sol naturel ”, Thèse de doctorat, l’Université Paris 6, 12 Apr 2013

[6] Alain COURCELLE, “Localisation d’un robot mobile : Application à l’aide à la mobilité
Des personnes handicapées moteur”, Thèse de doctorat, l’Université de Metz, France, janvier
2000.

[7] Etude et conception d'un robot marchant par HEDDOUCHE Kamel Université Mohamed Khider Biskra, année 2014

[8] SYSTEME DE LOCALISATION POUR ROBOTS MOBILES par Dr. SLIMANE Noureddine UNIVERSITE DE BATNA en 2005

[9] Robotique Mobile – David FILLIAT, École Nationale Supérieure de Techniques Avancées ParisTech

[10] Positionnement GPS précis et en temps-réel dans le contexte de réseaux de capteurs sans ?l type Geocube Thèse de doctorat Lionel Benoît

[11] GPS localisation et navigation par satellites Francoise duquenne

[12] https://www.gps.gov/systems/gps/french.php

[13] Paul Correia, Guide pratique du GPS, Éditions Eyrolles, décembre 2012, 6e éd., 245 p

[14] Introduction aux méthodes de télédétection des signaux géophysiques

[15] https://boowiki.info/art/reseaux-satellites/satellite-gps.html

[16] Global Positioning System Signals, Measurements, and Performance by Pratap Misra, Per Enge

[17] Introduction to GPS - The global positioning system, Ahmed El-Rabbany, 2002 Artech House INC., 685 Canton Street, Norwood MA 02062.

[18] John W. Betz - Engineering Satellite-Based Navigation and Timing_ Global Navigation Satellite Systems, Signals, and Receivers-Wiley-IEEE Press (2015)

[19] Principles and Applications Second Edition Elliott D. Kaplan Christopher J. Hegarty

[20] Gomarasca, M. A. (2009). Satellite Positioning Systems. Basics of Geomatics, 341–368. doi:10.1007/978-1-4020-9014-1_7

[21] Positionnement par GPS basé sur le filtre de Kalman : application à l'antenne de poursuite d'ALSAT-1 Diplôme de Magister BELAIDOUNI Kharofa

[22] Guide pour le positionnement GPS (Juillet 1993) Troisième impression, avec corrections, avril 1998

[23] Les principes du positionnement par satellite : GNSS Renaud Lahaye, Sylvie Ladet

[24] Benmakhlouf Abdeslam diplôme de : Magister En Robotique Thème Contrôleur Flou Pour La Navigation D'un Robot Mobile D'intérieur

[25] http://tiptopboards.com/89-controleur-de-moteurs-l293d-pour-arduino.html

[26] Asservissement visuel d'un robot mobile avec évitement d'obstacle Master en Génie Electriques univ medea ZAIBAK Sérhané

[27] ÉTUDE ET ANALYSE TEMPS RÉEL D'UN CANAL DE RÉCEPTION GPS par BERNARD DIONNE

[28] https://wiki.seeedstudio.com/Grove-GPS/

[29] « Arduino : Premier s pas en informatique embarquée » : https://zestedesavoir.com

[30] Arduino, "Arduino Board Uno", https://www.arduino.cc/en/main/arduinoBoardUno

[31] Raspberry pi, "Raspberry pi 3 model B, Raspberry pi foundation UK registred Charity
https://www.raspberrypi.org/products/raspberry-pi-3-model-b/
[32]http://www.monclubelec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.DebuterPresentationLogiciel

[33] https://lastminuteengineers.com/l293d-motor-driver-shield-arduino-tutorial/

[34] L293x Quadrpule Half-H Drivers datasheet (Rev.D)

[35] Arduino pour les Nuls poche, 2e édition

[36] Jean-Marc Pieplu - GPS et Galileo _ Systemes de navigation par satellites (2006, Eyrolles) - libgen.lc

[37] Mon GPS en action Créer et enrichir ses cartes avec Google Earth, Google Maps, OpenStreetMap by Correia, Paul

[38] https://store.arduino.cc/arduino-uno-rev3

[39] Shawn Wallace Matt Richardson - À la découverte du Raspberry Pi (2014, Eyrolles