

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Projet de Fin d'Études

présenté par

ESSELAMI ABDELHAK

&

ZAIDI LOTFI

pour l'obtention du diplôme Master en Électronique option système de vision et robotique

Thème

Conception et réalisation D'un robot mobil contrôlé par un système de vision embarqué

Proposé par : Mr kased boualam

Année Universitaire 2014-2015

Dédicaces

J'ai le grand plaisir de dédier le fruit de mes années d'études aux êtres qui me sont les plus chers, mes parents.

A ma mère qui s'est donnée tant de mal pour moi et qui m'a offerte amour et soutien depuis mon premier souffle.

A mon cher père pour avoir mis tous les moyens à ma disposition et d'avoir été à mes côtés tout au long de ces années d'études.

A ma chère sœur ainsi que toute ma famille.

A mes chers amis.

Lotfi

Je dédie ce travail aux deux plus chères personnes au monde qui est mes parents, pour tous leurs amours, encouragements, conseils, sacrifices, patience et confiance.

Je le dédie également à mes très chers sœur et frères, même ainsi qu'à toute ma famille.

Je le dédie enfin à tous mes amis et ceux qui me sont chers.

ABDELHAK

Remerciements

*Nous exprimons toute notre reconnaissance à Monsieur **Pr .KAZED BOUALEM** pour avoir assuré l'encadrement de ce travail. Nous le remercions pour son soutien, son orientation et ses consignes. Son expérience et sa connaissance, ont contribué à notre formation scientifique.*

Nous remercions les membres de jury, qui nous fait l'honneur de participer au jugement de ce travail.

*Nous souhaitons remercier également les enseignants du département d'électronique de l'université de **SAAD DAHLAB-BLIDA** et nos camarades amies de promotion 2014-2015, ainsi l'ensemble des personnels pour leurs soutiens.*

Nous remercions tous les membres de notre famille et en particulier les parents, pour leurs encouragements et appuis inconditionnels tout au long de ces longues années d'études.

Nous nous excusons auprès de ceux que nous n'avons pas nommés et nous souhaitons qu'ils comprennent qu'en ces moments cruciaux , il n'est pas aisé d'être entièrement soi-même qu'ils se rassurent , ils sont et seront toujours dans nos cœurs .Merci.

ملخص:

في مشروعنا هذا نقوم بصنع روبوت يحدد موقع الهدف الملون باستعمال الرؤية الصناعية وذلك باحتوائه على كمبيوتر الراسبيري باي الذي يستقبل الصور التي يتم التقاطها عن طريق كاميرة ذاتية في الوقت الحقيقي و يقوم بتحليلها و ارسال احداثيات الهدف الى لوحة الاردوينو التي تعمل على توجيه الروبوت بالتحرك نحو الهدف حيث يقوم هذا الاخير بحمل الهدف الاول عن طريق اليد الالية و التوجه بها الى الهدف الثاني ووضعتها ليعود مرة اخرى للقيام بنفس المهمة.

كلمات المفاتيح: روبوت متحركة, ذاتية الحكم , رؤية صناعية , معالجة الصور , رايسبيري باي , بايثون, اردوينو , مكتبة الرؤية بالحسوب .

Résumé :

Ce projet consiste à concevoir un robot mobile capable de reconnaître un objet de couleur prédéfinie pour ensuite le ramasser et le déplacer vers un autre endroit caractérisé par une autre couleur choisie par l'utilisateur. Pour ce faire nous utiliserons comme plateforme principale une carte de type Raspberry Pi, dotée de ressources suffisantes pour le traitement du flux vidéo issu d'une caméra embarquée sur notre robot. Les mouvements du robot sont contrôlés par une autre carte de type Arduino, laquelle se chargera de délivrer les signaux au module puissance directement lié aux deux moteurs responsables de faire déplacer le robot aux endroits désirés.

Mots clés : Robot mobile, Autonome, Vision artificielle, open cv, traitement d'images, Raspberry Pi, python, arduino.

Abstract :

The aim of this project is to design a mobile robot with the ability to recognize an object according to a specified color; this object will then be fetched and moved to a particular place featured by a user defined color. To achieve this, our robot is equipped with a Raspberry Pi embedded computer connected to its full featured camera; this set of hardware tools provides the necessary requirements for our project. The robot movements are obtained through an Arduino Uno board which is responsible to control the power board that will drive a pair of DC motors which will provide the required movements.

Keywords: Mobile robot, Autonomous, image processing, Raspberry Pi, open cv, python, arduino.

Listes des acronymes et abréviations

HD	high definition
PAL/NTSC	PAL : Phase Alternating Lines, NTSC : National Television Standard Committee
RJ45	Registered Jack
HDMI	High Définition MultiMedia Interface »
USB	Bus série universel « Universal Serial Bus »
MPI CSI-2 15	MPI : Processor Interface CSI-2 : Camera Serial Interface 2
SD :	Secure Digital
GPIO :	General-purpose input/output
LXDE	Lightweight X11 Desktop Environment
PWM	Pulse-width modulation
Tx/Rx	Rx : receiver, Tx : Transmitter
RPI	Raspberry Pi

INTRODUCTION GENERALE

«Pour faire de grandes choses, il ne faut pas être un si grand génie, il ne faut pas être au dessus des hommes, il faut être avec eux »

Montesquieu

Introduction générale

La robotique touche tous les domaines de notre vie, les robots se développent de jour en jour jusqu'à, parfois, devenir plus performants qu'un être humain. Nous sommes, sans nous en rendre compte, à un tournant de l'Histoire. La Robotique fait partie des sciences des objets et des systèmes artificiels. Elle peut être vue comme une science de la perception et du mouvement et de leur intégration en une machine physique, mécanique et informatique.

Un robot est un système matériel possédant des capacités de perception, d'actions, de décision et de communication, parfois capable d'améliorer ses propres performances par apprentissage automatique ou supervisé par des humains.

Le but de notre projet est la conception et la réalisation d'un système mobile, autonome capable de trouver, suivre et déplacer un objet en temps réel. L'objectif principal de notre conception est d'embarquer un nano-ordinateur avec une caméra spéciale pour effectuer le traitement des images captées, afin de reconnaître l'objet à manipuler et de corriger la position de notre robot, ceci est réalisé en superposant le centre de gravité de l'image captée avec le centre de gravité de l'objet détecté sur cette image. L'alignement étant effectué, le système se met en marche dans la direction de l'objet.

Le travail réalisé dans le cadre de ce mémoire s'articule en quatre chapitres. Le premier chapitre présente des notions sur la robotique et de robots mobiles. Le deuxième chapitre est consacré à la vision artificielle. Le troisième chapitre consiste à présenter la technologie utilisée dans notre projet. Et dans le quatrième chapitre, nous évoquons tous les étapes de conception et de réalisation matérielle de notre robot. Nous terminons par une conclusion générale sur notre travail et des perspectives.

CHAPITRE . 1

GENERALITE SUR LA ROBOTIQUE MOBILE

«Une personne qui n'a jamais commis d'erreurs n'a jamais tenté d'innover »

Albert Einstein

Chapitre 1

Généralités sur la robotique mobile

1.1. Introduction :

L'appellation (robot mobile) désigne généralement un véhicule capable de naviguer d'une manière autonome dans un environnement complexe, parfois évolutif, partiellement connu ou inconnu, et d'exécuter les tâches programmées avec ou sans intervention humaine.

1.1.1. Historique :

Le concept de robot mobile autonome est apparu vers la fin des années soixante, de deux sources totalement différentes : tout d'abord des recherches menées au Stanford Research Institute sur les possibilités d'équiper des machines de capacités de déduction et de réaction logique à des événements extérieurs. On a ainsi construit Shakey, machine à roues reliée à un ordinateur et équipée d'une caméra lui permettant d'acquérir des images de son environnement. Elle évoluait dans un univers de cubes de tailles et de couleurs différentes. Shakey avait pour mission de prendre un objet et de le porter ailleurs, quel que soit sa position ; chaque mission dure près de cinquante minutes.

D'autres parts, l'industrie nucléaire a besoin de machines permettant d'agir à distance dans des environnements encombrés et inaccessibles à l'homme. L'entreprise américaine Général Electric développe alors un quadrupède pour essayer de résoudre ce problème, tandis que les projets Luna et Mars Rover s'échafaudent dans le but d'explorer des planètes sans que l'homme ne prenne part au voyage.

Pendant plusieurs années, laboratoires, industriels, informaticiens et mécaniciens vont continuer leurs travaux en parallèle. On accède ainsi cote informatique, on assiste a de grands progrès dans le domaine de l'intelligence artificielle. Ainsi, vers la fin des années soixante-dix, trois pôles géographiques principaux se distinguent (France, Japon, Etats-Unis).

La synthèse de tous les travaux réalisés jusqu' alors donne enfin naissance aux robots mobiles autonomes (du robot domestique au robot militaire).

L'industrie de production, les sociétés d'exploitation minière, les expéditions de recherche sous-marine...Les domaines d'utilisation de robots autonomes sont très variés, allant de la production en chaine dans une usine de voiture à l'exploration d'autre planète Mars.

C'est dans cet environnement d'effectuer des tâches répétitives ou encore impossibles à l'homme (porter des charges lourdes, découpage ultra précis,...), mais aussi de manifester une certaine autonomie de déplacement dans des milieux hostiles à l'homme. On en voit désormais les applications sur des chantiers tels que le désamiantage d'immeubles, la décontamination radioactive, les expériences en milieu dangereux.... Aussi a-t-on besoin de robot mobiles autonomes capables de se déplacer d'un point à un autre sur une simple demande de planification complexe de la trajectoire est fourni par les drones, ces robots volants destinés aussi bien à l'espionnage militaire qu'aux études botaniques nécessitant des prises de vues aériennes. [1]

1.1.2. Définition :

Il existe diverses définitions du terme robot, mais elles tournent en général autour de celle-ci : Un robot est une Machine chargée d'une ou plusieurs tâches de manière autonome. Il peut-être mobile et utiliser différents moyens de locomotion. Il peut être anthropoïde s'il imite la forme humaine. Le mot vient d'une pièce de théâtre tchèque dans laquelle apparaissait un travailleur artificiel employé pour le (servage), désigné en tchèque par (robota).

1.2. Présentation général des 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 générale 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).

L'autonomie du robot mobile est une faculté qui lui permet de s'adapter ou de prendre une décision dans le but de réaliser une tâche malgré un manque d'informations préliminaires ou éventuellement erronées. Dans d'autres cas d'utilisation, comme celui des véhicules d'exploitation de planètes, l'autonomie est un point fondamental puisque la télécommande est alors impossible par le fait de la durée du temps de transmission des informations.

1.3. Classification des robots mobiles :

La classification des robots mobiles se fait suivant plusieurs critères (degré d'autonomie, système de locomotion, énergie utilisée...).

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

- ✓ Robot télécommandé par un opérateur qui lui impose chaque tâche élémentaire à réaliser.
- ✓ Robot télécommandé au sens de la tâche à réaliser. Le robot contrôle automatiquement ses actions.
- ✓ Robot semi-autonome réalisant sans l'aide de l'opérateur des tâches prédéfinies
- ✓ Robot autonome qui réalise des tâches semi-définies. Ce type de robot pose 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. [3]

1.4. Applications :

Le domaine d'application des robots mobiles est vaste, nous présentons quelques applications dans le tableau suivant (figure1.1) et sur figure (1.2).

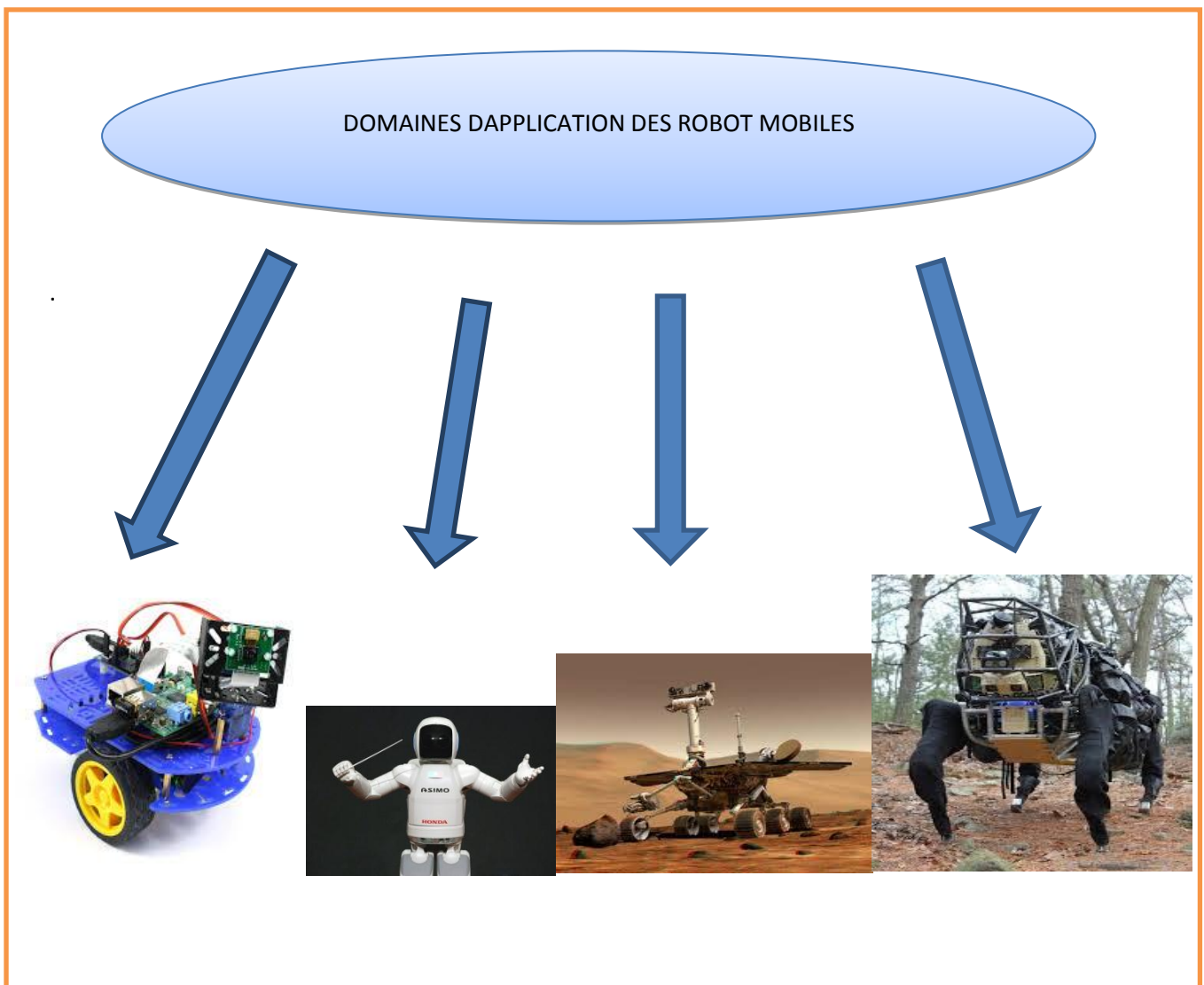


Figure 1.1. Quelques exemples d'application des robots mobiles

Domaines	Applications
Industrie nucléaire	- surveillance de sites - manipulation de matériaux radioactifs - démantèlement de centrales
Sécurité civile	- neutralisation d'activité terroriste - déminage - pose d'explosif - surveillance de munitions
Chimique	- surveillance de site - manipulation de matériaux toxiques
Mine	- assistance d'urgence
Agricole	- cueillette de fruits - traite, moisson, traitement des vignes.
Nettoyage	- coque de navire - nettoyage industriel
Espace	- exploration
industrie	- convoyage - surveillance
Sous-marine	- pose de câbles - recherche de modules - recherche de navires immergés - inspection des fonds marins
Militaire	- surveillance - pose d'explosif - manipulation de munitions

Tableau .1.1 : Domaines d'applications des robots

1.5. Architecture des robots mobiles :

L'architecture des robots mobiles se compose de quatre parties essentielles :

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

1.5.1 La structure mécanique et la motricité :

Il existe quatre types de structures mécaniques assurant la motricité :

- ✓ Les robots à roues
- ✓ Les robots à chenilles
- ✓ Les robots marcheurs
- ✓ Les robots rampant.

a) Les robots mobiles à roues :

La mobilité par roues est la structure mécanique la plus communément appliquée. Cette technique assure l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importantes .le franchissement d'obstacles ou l'escalade de marches d'escaliers est possible. [3]

b) Les robots 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.

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 [3]

c) Les robots à chenilles :

L'utilisation des chenilles présente l'avantage d'une bonne adhérence au sol et d'une faculté de franchissement d'obstacles. L'utilisation est orientée vers l'emploi sur sol accidenté ou de mauvaise qualité au niveau de l'adhérence. [3]

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.

1.5.2. Le système de localisation :

Le système de localisation est l'ensemble constitué par les capteurs et les logiciels de traitement de données utilisé par le véhicule pour estimer de manière autonome son déplacement ou sa situation dans l'espace. Classiquement, on distingue deux types de solution :

- ✓ La localisation relative qui permet au véhicule de naviguer à l'estime (Dead reckoning) en utilisant uniquement les mesures de ses mouvements propres délivrées par les capteurs proprioceptifs.
- ✓ La localisation absolue qui fait appel aux mesures des capteurs extéroceptifs pour estimer la situation du véhicule dans un repère lié à l'environnement.[2]

1.5.3. Les organes de sécurité :

Il est dangereux de laisser le robot mobile complètement libre. Donc il est obligatoire qu'il soit doté d'organes garantissant la sécurité.

Deux types de capteur sont employés :

- ✓ Les capteurs proximétriques assurent la détection avant collision (ultrasons, hyperfréquences, infrarouge...)
- ✓ Les capteurs à contact détectent une collision ou un choc avec l'environnement (contact électrique sur pare-chocs, résistance variable, fibre optique...).

L'organisation de la sécurité d'un robot mobile est représentée sur le schéma suivant :

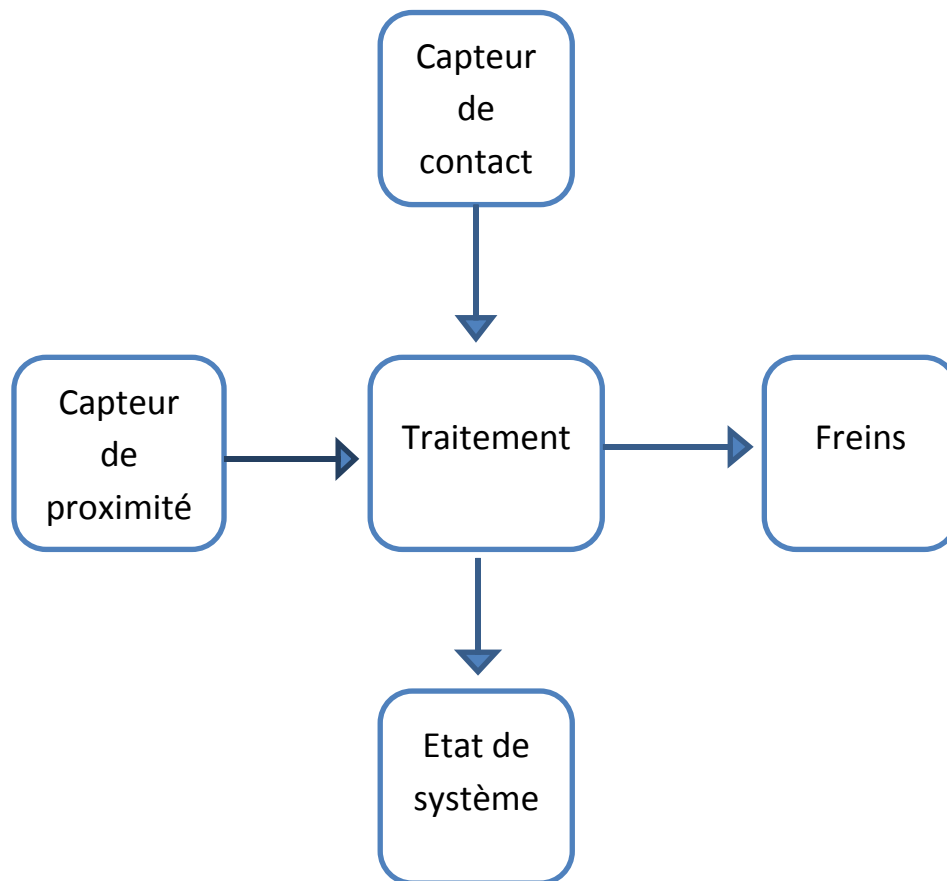


Figure 1.2. Les organes de sécurité

Si on détecte une présence, la stratégie consiste soit à immobiliser le robot en attendant que la personne s'éloigne, soit à ralentir le mouvement si la personne n'est pas très proche, soit à choisir un autre chemin qui l'éloigne de la personne.

1.5.4 .Le système de traitement des informations et gestion de taches :

L'ensemble de traitement des informations et gestion des tâches constitue le module information central qui établit les commandes permettant au mobile de réaliser un déplacement et d'activer les divers organes en accord avec l'objectif. Nous nous limiterons au problème de génération de plan qui consiste à établir la manière dont le robot se déplace par rapport à des connaissances à priori (statiques) ou obtenues en cours d'évolution (dynamiques).

La génération de plus repose sur trois concepts :

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

.1.6. Vision pour la robotique :

La vision par ordinateur met en œuvre des caméras conventionnelles à champ de vue large ou des systèmes permettant d'acquérir des images tridimensionnelles. Elle décrit la géométrie et le traitement de ces images qui rendent possible l'utilisation de la vision dans de nombreuses applications, parmi lesquelles figure l'augmentation de l'autonomie perceptive des robots. [4]

1.7. Conclusion :

Ce premier chapitre fournit une présentation générale sur les robots mobiles. Il constitue également une introduction de l'étude conduite pour résoudre le problème de la localisation d'un robot mobile.

CHAPITRE . 2

NOTION DE VISION ARTIFICIELLE

«En essayant continuellement on finit par réussir, donc : plus ça rate, plus on a de chances de réussir »

Devise shadok

Chapitre 2

Notion de vision artificielle

2.1 Introduction :

La vision artificielle aussi appelée vision par ordinateur permet à un robot de s'adapter à l'environnement qui l'entoure en orientant lui-même ses actions « éviter des obstacles, suivre un objet, se guider sur une trajectoire » lorsqu'on le connecte à une ou plusieurs caméras. La vision par ordinateur se base essentiellement sur le traitement d'images pour effectuer des tâches comme la segmentation d'objets, la reconnaissance des formes, la détection du mouvement, le suivi de cible et autres techniques. Dans ce chapitre on parlera de quelques notions de base de traitement d'image ainsi que les outils qu'on utilise au cours de notre projet « suivre un objet ».

2.2. Notions de base :

2.2.1 Image numérique :

Pour faire un traitement sur une image avec l'outil informatique, il faut qu'elle soit numérisée. L'image numérique est l'image dont la surface est divisée en éléments de taille fixe appelés cellules ou pixels. La numérisation d'une image est la conversion de celle-ci en une image numérique « c'est-à-dire à partir d'une image à distribution continue d'intensité, vers une image numérique » représentée par une matrice bidimensionnelle à valeur numérique $f(x, y)$ où x, y correspondent aux coordonnées cartésiennes « discrètes » d'un point de l'image, et $f(x, y)$ la valeur de ce point [5].

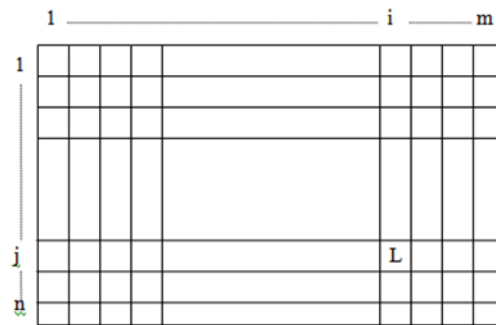


Figure 2.1 représentation matricielle d'une image

L : Le niveau de gris au point (x, y) . $I(i, j) = L$.

n : La hauteur de l'image m : La largeur de l'image.

2.2.2 Les plans de l'image couleur :

a) RGB :

Le modèle RGB propose de coder sur un octet chaque composante de couleur, ce qui correspond à 256 intensités de rouge⁸, 256 intensités de vert et 256 intensités de bleu, soient 16777216 possibilités théoriques de couleurs différentes, c'est-à-dire plus que ne peut en discerner l'œil humain « environ 2 millions ». Toutefois, cette valeur n'est que théorique car elle dépend fortement du matériel d'affichage utilisé[6]

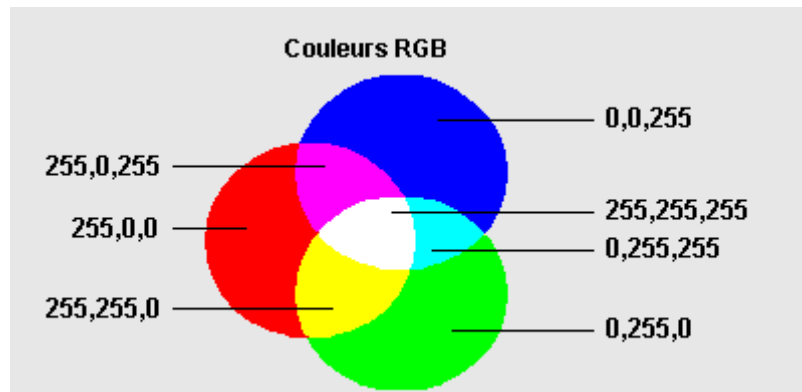


Figure 2.2 : Les plans RGB

b) HSV :

Le système HSV « Hue, Saturation, Value)'' est un mode de coloration des images qui se révèle souvent plus efficace que le système classique RGB (Red, Green, Blue), notamment pour les images fractales. Le système HSV est représenté en coordonnées cylindriques

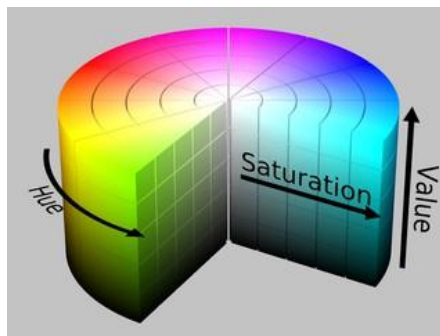


Figure 2.3 : représenté en coordonnées cylindriques

La teinte (Hue) se lit sur un cercle : elle est codée par un angle entre 0 et 360°, la saturation et la luminosité des valeurs entre 0 et 255.

2.3. Vision Assisté par Ordinateur « VAO » :

La VAO est un ensemble d'outils qui permet à l'ordinateur d'imiter la perception humaine afin d'extraire des informations d'une image brute pour pouvoir prendre des décisions. Nous pouvons alors dire que la VAO constitue une chaîne de traitements allant de l'acquisition de l'image brute jusqu'à son interprétation par la machine.[4]

Le schéma suivant « Figure 2.4 » montre les étapes essentielles dans un système de VAO

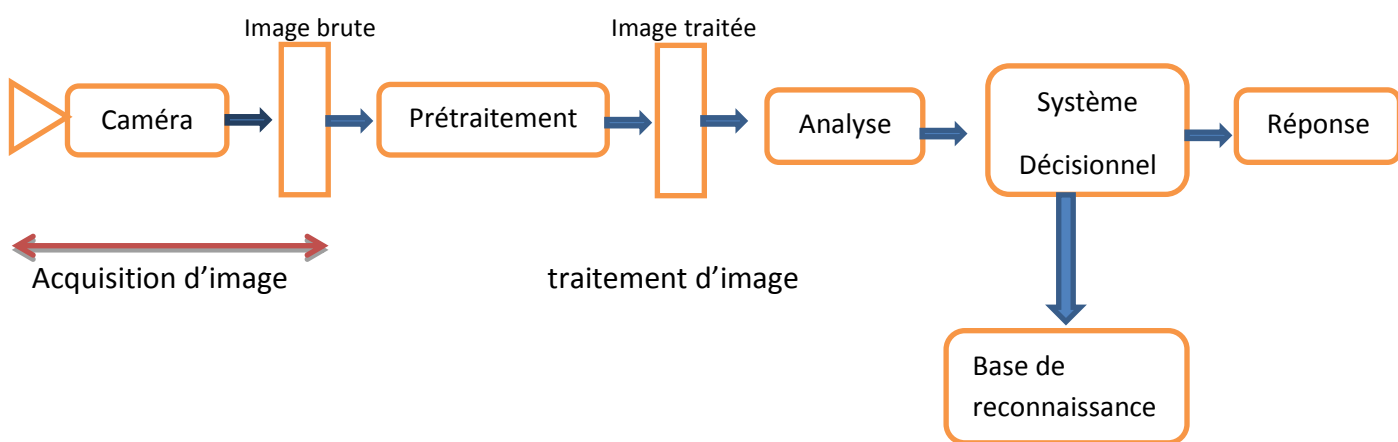


Figure 2.4 La Chain de VAO

Le traitement artificiel d'une image passe essentiellement par les étapes suivantes :

- Acquisition
- Prétraitement.
- Analyse.
- Interprétation.

Les trois premières étapes sont appelées : Traitement de bas niveau et la dernière étape : Traitement de haut niveau.

2.4. Acquisition d'image :

L'acquisition est le mécanisme qui permet l'obtention d'une image numérique «représentée par une matrice » à deux dimensions à partir d'une scène à trois dimensions, en passant par un système optique. Cette opération peut être statique « Appareil Photo, Scanner...etc. » ou dynamique « Caméra vidéo, Webcam », dans ce cas on aura une séquence vidéo. L'image continue $f(x, y)$ est approximée par des échantillons qui sont obtenus par discrétisation des coordonnées (x, y) « ce qu'on appelle l'échantillonnage » et la discrétisation des amplitudes de ces points « c'est la quantification ». A ce niveau on obtient une donnée brute qui nécessitera d'éventuels traitements pour les phases à venir. Un dispositif de visualisation permet l'affichage de l'image. Les reconstituteurs permettent de transformer le signal numérique qui est la matrice image en un signal analogique visible à l'œil humain. Pour cela, on dispose d'une multitude de supports employés à cet effet « moniteur vidéo, impression sur papier... » [5].

2.5 Prétraitement :

Le prétraitement regroupe toutes les opérations de manipulation de l'image dans le but d'améliorer les caractéristiques d'une image afin de faciliter l'analyse d'images. On cite quelques techniques on citer :

- Le filtrage : il s'agit de supprimer le bruit, ou les petites variations, présent dans une image. L'intensité d'un pixel est transformée en fonction des intensités sur un petit voisinage du pixel.
- L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image (contraste, ...) pour faciliter son interprétation par l'œil humain.
- La restauration d'images a pour but de supprimer les dégradations subies par une image à l'aide de connaissance a priori sur ces dégradations.

2.5.1. Filtrage :

Le filtrage est une opération fondamentale dans le traitement d'images, l'objectif est de réduire le bruit afin de compenser certains défauts liés au capteur et améliorer la qualité visuelle de l'image. On peut citer quelques types de filtrage :

a) Filtres linéaires :

Le filtrage linéaire est une transformation mathématique appelée convolution. Le principe est de modifier la valeur des pixels d'une image, généralement dans le but d'améliorer son aspect. En pratique, il s'agit de créer une nouvelle image en se servant des valeurs des pixels de l'image d'origine. La définition de la convolution est donnée par l'équation suivante.

$$(I * h)[x, y] = \sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} h[i, j] \cdot I[x - i, y - j]$$

b) Filtres non linéaires :

Ce type de filtrage s'oppose au précédent, le résultat n'est pas une combinaison linéaire des pixels du voisinage mais plutôt une fonction non linéaire tel que « min, max, médiane ». Exemple : du filtre médian sur un voisinage à huit, le nouveau niveau de gris du pixel centre est choisi comme étant la valeur médiane de tous les pixels de la fenêtre d'analyse centrée sur ce dernier

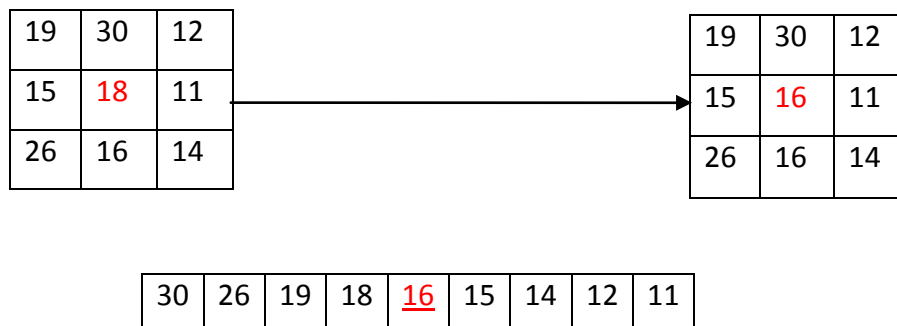


Figure 2.5 : Principe du filtre médian

2.5.2 Egalisation de l'histogramme :

Elle consiste à redistribuer les valeurs de l'histogramme afin d'équilibrer la distribution des niveaux de gris c'est à dire s'approcher le plus possible de la situation où chaque niveau de gris est représenté par le même nombre de pixels [4].

La figure 6 montre une image peu contrastée et la même image après égalisation.



Figure 2.6 Egalisation d'histogramme

2.6. L'analyse :

L'analyse est une suite d'opérations permettant l'extraction de l'information essentielle contenue dans une image. Elle est essentiellement composée par une phase de segmentation qui fournit une partition de l'image. A partir de cette segmentation, des techniques de description / modélisation permettent d'obtenir la description de l'image. Celle-ci peut prendre la forme d'une image ou de toute structure de données permettant une description des entités contenues dans l'image elle-même. Par opposition avec la phase d'interprétation, l'analyse tente dans la mesure du possible, de ne pas prendre en compte le contexte [4]. La segmentation d'images ainsi définie est un domaine vaste où l'on trouve essentiellement deux approches :

2.6.1. Segmentation par extraction de contour :

Le but de l'analyse d'images est d'obtenir une description synthétique des divers éléments qui la constituent à partir de la masse énorme d'information qu'elle contient à l'état brut. L'un des processus fondamentaux dans la chaîne de reconnaissance visuelle, consiste à diminuer cette quantité d'information en ne gardant que les points essentiels de l'image. Les points de contours constituent en ce lieu une approche rationnelle. Le processus d'extraction de contours peut être décomposé en plusieurs étapes distinctes [4] :

- La mise en évidence des contours : Elle s'obtient par une différenciation de l'image en utilisant des opérateurs telles que:

Opérateur de Roberts : il est basé sur la notion du gradient et utilise deux masques :

$$H_x = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

$$H_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$



a) image initial



opérateur Roberts

Figure 2.7 Opérateur de Roberts

Opérateur de Sobel : utilise comme masque :

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$



opérateur Sobel

Figure 2.8 Opérateur de Sobel sur la précédente image

Laplacien : il consiste à calculer la dérivée seconde de l'image.

Il utilise le masque suivant :

$$H = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



laplacien

Figure 2.9 : Laplacien sur la même image

Cet opérateur est particulièrement sensible au bruit.

- La réduction des contours : elle agit de manière que les contours mis en évidence ne fassent qu'un seul pixel d'épaisseur.
- La binarisation des contours : Suivant les méthodes, ce processus et le précédent peuvent être exécutés suivant un ordre différent, c'est à dire binarisation-réduction ou réduction-binarisation.
- La description des contours : Elle permet d'organiser les points du contour en structures simples telles que les segments de droites, arcs,...etc., et de détecter les contours fermés afin d'isoler ceux-ci.

2.6.2. Segmentation en régions homogènes :

Contrairement à l'extraction de contours qui s'intéresse aux bords d'une région, la segmentation en régions homogènes vise à segmenter l'image en se basant sur les propriétés intrinsèques de la région. Le choix des propriétés détermine ce qu'on appelle le critère de segmentation. Les principaux critères utilisés sont le niveau de gris ou la couleur, la texture et le mouvement [4].

2.6.3. Opérations morphologiques :

Les opérations morphologiques telles que la dilatation, l'érosion, la fermeture et l'ouverture sont utilisées pour l'amélioration des images binaires en éliminant les faux pixels considérés comme des bruits.

a) Dilatation :

La dilatation consiste à éliminer les points noirs isolés : on dilate les parties blanches ce qui élimine les points noirs. L'opération de dilatation est définie par:

$$I \oplus S = \{x | [(S)_x \cap I] \neq \emptyset\}$$

Où I est l'image qui sera dilatée, S est un ensemble de points de coordonnées connues. L'effet de base de l'opérateur de dilatation sur une image binaire est d'agrandir progressivement

les limites des régions noires de l'image. Donc les régions blanches de l'image grandissent en dimensions.

b) Erosion :

Il s'agit ici d'éliminer les points blancs isolés. La méthode est assez similaire à celle de la dilatation. L'opération d'érosion est définie par:

$$I \ominus S = \{x | [(S)_x \subseteq I]\}$$

- L'ouverture est une opération d'érosion suivie d'une dilatation, elle est définie par

$$B \circ I = (B \ominus I) \oplus I$$
- La fermeture est une opération de dilatation suivie d'une érosion, elle est donnée par

$$A \bullet I = (A \oplus I) \ominus I$$

2.7. L'interprétation :

L'interprétation a pour but le passage de la description structurelle à la description sémantique en regard à certains objectifs. Ces objectifs peuvent être très simples « mesure de certains paramètres sur des formes » ou beaucoup plus complexes « description du contenu de la scène en termes de concepts non mathématiques »[4].

2.8. Conclusion :

Après avoir présenté quelques notions sur la vision par ordinateur, ainsi que quelques techniques de traitement d'image, Le prochain chapitre consiste à définir la nature et les caractéristiques de l'ordinateur utilisé pour faire ce type de traitement

CHAPITRE . 3

RASPBERRY PI ET LA ROBOTIQUE

«Il faut faire de la vie un rêve et faire d'un rêve une réalité »

Pierre Curie

Chapitre 3

Raspberry Pi et la robotique

3.1 Introduction :

Le module Raspberry pi est l'un des nombreux exemples d'ordinateurs embarqués existants sur le marché. Ce genre d'outils ouvre une grande porte pour le développement de la robotique mobile, la miniaturisation et la connectivité de cette carte sont deux clés pour le développement de la robotique. Ce chapitre décrit ce module et les outils utilisés pour son exploitation dans le cadre de notre application.

3.2 Le Raspberry pi :

3.2.1 Définition :

Le Raspberry pi est un nano ordinateur mono carte sous le système d'exploitation Linux. Il est destiné à des applications d'informatique embarquée. Le cœur de l'ordinateur est un FPGA intégrant un processeur ARM et de nombreux périphériques. Le connecteur d'extension supporte les entrées/sorties parallèles ainsi que la plupart des bus de communication. C'est un support particulièrement économique et puissant qui peut être facilement mis en œuvre dans de petits systèmes nécessitant un accès au monde physique par des capteurs ou des actionneurs disposants d'interfaces numériques [7].

3.2.2 les connexions :

Tout autour de la partie centrale du Raspberry pi, on retrouve différentes connectiques permettant de brancher des périphériques afin d'interagir avec l'ordinateur. Voici donc la connectique exhaustive offerte par le Raspberry Pi :

- Sortie vidéo HD 1080p
- Sortie vidéo composite (PAL/NTSC)
- Sortie audio stéréo
- Prise Ethernet RJ45 10/100 Base-T

- Prise audio et vidéo HDMI 1.3 et 1.4
- Prise jack de sortie vidéo audio composite de 3,5 mm à 4 pôles
- 4 prises USB 2.0
- Connecteur MPI CSI-2 15 contacts pour caméra vidéo HD Raspberry Pi (775-7731)
- Port pour carte micro SD
- 40 broches pour GPIO et bus de communication
- Alimentation : +5 V via le port micro-USB

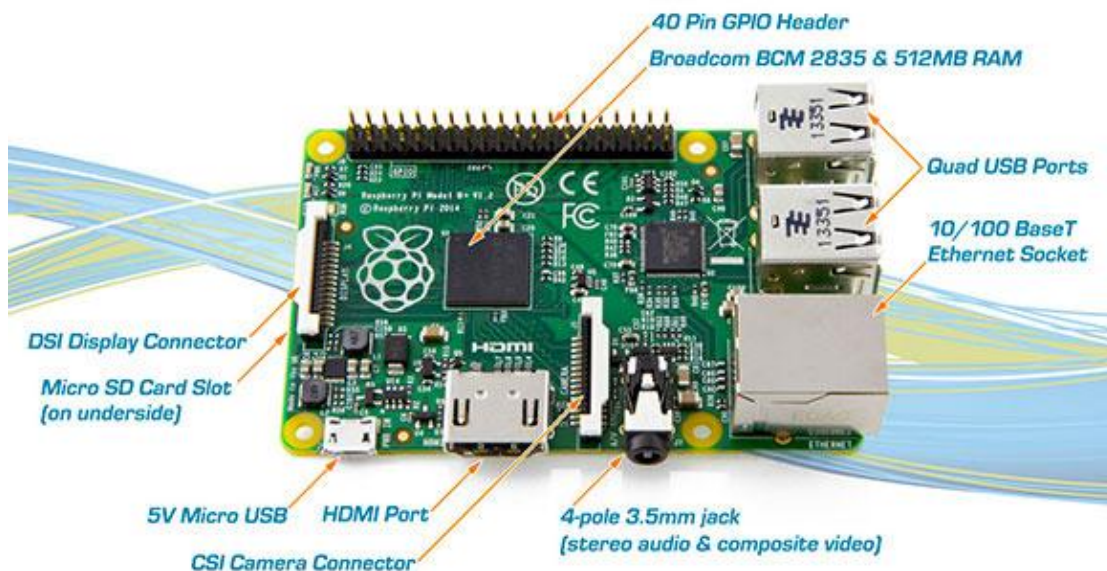


Figure3.1 : connections de Raspberry

3.2.3 GPIO :

Les ports GPIO (General Purpose Input/Output) soit entrées/sorties à usage général, sont des ports qui sont très utilisés dans le monde des microcontrôleurs, en particulier dans le domaine de l'électronique embarquée. Les périphériques GPIO comportent un ensemble de ports d'entrée/sortie qui peuvent être configurés pour jouer soit le rôle d'une entrée, soit le rôle d'une sortie[11]. Dans le Raspberry pi le connecteur GPIO « Figure 3.2 » est composé

de 40 broches dont 27 sont dédiées au GPIO (permet des changements d'état on/off). Les autres broches ont le rôle d'alimentation (3.3v, 5v) et de la masse (ground). Le connecteur dispose ainsi de différents types de connexion :

- Broches pour une interface I2C. (permet de se connecter sur du matériel avec seulement 2 broches.)
- Broches utilisables en entrée ou sortie numérique. Également dite « tout ou rien ».
- Broches Rx et Tx. (Communication avec des périphériques séries.)
- Broches pouvant être utilisé en PWM « Pulse Width Modulation » ou PPM (« Pulse Position Modulation »).
- Une interface SPI.

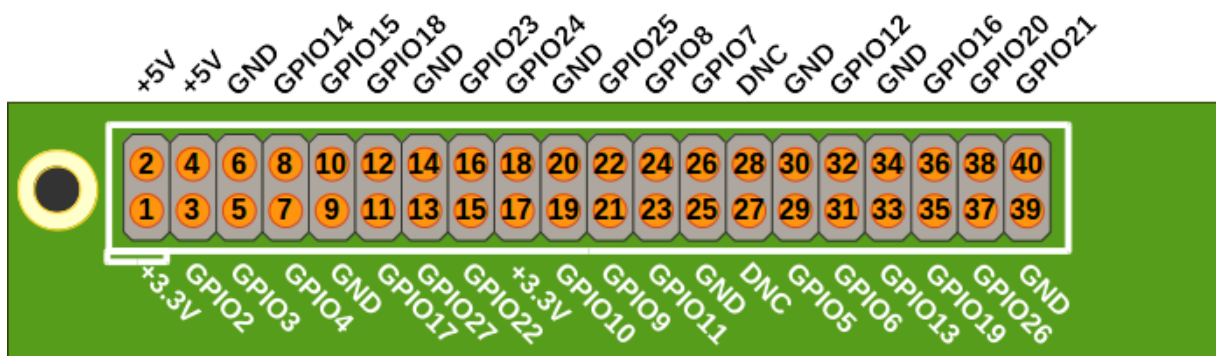


Figure 3.2: les 40 broche de raspberry pi modale B+

3.2.4 Equipement recommandé :

En fonction des besoins de l'application des accessoires sont nécessaires pour utiliser et configurer la carte Raspberry pi.

- Une carte micro SD
- Une Alimentation micro USB 2A
- Un Boitier transparent
- Un Câble HDMI
- Un écran HDMI
- Un clavier et une souris USB

3.3 : Installation de système d'exploitation Raspbian :

L'installation du système d'exploitation se fait sur une carte mémoire « micro SD » car le Raspberry ne possède pas de mémoire morte. Il existe plusieurs systèmes d'exploitation pour le Raspberry dont telles que la distribution Raspbian, Raspbian est un système d'exploitation libre basé sur Debian optimisé pour le matériel Raspberry Pi. Un système d'exploitation est l'ensemble des programmes et des services de base qui font fonctionner le Raspberry Pi. Cependant, Raspbian fournit plus qu'un simple système d'exploitation : il est livré avec plus de 35.000 paquets, des logiciels précompilés livrés dans un format pour une installation facile du Raspberry Pi [10]. Sur le site Internet dédié à ce module nous pouvons trouver les étapes d'installation et configuration du Raspbian sur la carte SD [8]. Il suffit d'insérer la carte SD avec le système Raspbian et brancher les équipements.

3.4 : l'interface graphique de Raspbian :

L'environnement graphique du bureau LXDE propose un dialogue simplifié avec l'utilisateur auquel sont habitués ceux qui utilisent Windows ou Mac OS X.



Figure 3.3 : le bureau de Raspbian

- 1 —————→ **La barre de menu**
- 2 —————→ **Gestionnaire de fichiers**
- 3 —————→ **Les icônes**
- 4 —————→ **Le Pi Store**
- 5 —————→ **Bouton Power**

1 .La barre de menu :

Cette barre de menu correspond à peu près au menu démarrer de Windows. Cliquez dans le coin en bas à gauche pour la faire apparaître, tous les logiciels que vous téléchargez que ce soit en ligne de commande, sur le web ou sur le Pi Store apparaîtront ici. Nous pouvons aussi gérer les réglages et les préférences à partir de cet endroit [9].

2. Gestionnaire de fichiers :

Un gestionnaire de fichiers classique comme sur n'importe quel ordinateur, nous pouvons faire des copier-coller, créer des dossiers, organiser nos fichiers[9].

3. Les icônes :

Quelques logiciels préinstallés ont leur icône sur le bureau, nous pouvons personnaliser le bureau à notre guise en déplaçant les icônes ou en rajoutant d'autres. Un clic droit sur une icône permet de la paramétrer, pour ouvrir un logiciel il faut un double clic sur l'icône correspondante[9].

4. Le Pi Store :

Quand le Pi Store est ouvert, cette petite icône apparait dans la barre des tâches. Nous pouvons accéder au Pi Store en cliquant dessus. Le Pi Store est une boutique de logiciels pour le Raspberry pi, nous devons créer un compte pour pouvoir commencer à télécharger des logiciels[9].

5. Bouton Power :

Le bouton Power permet d'éteindre ou de redémarrer le Raspberry pi, il est préférable d'éteindre le Raspberry pi avec le bouton Power qu'en le débranchant brusquement[9].

3.5 Les commandes Linux les plus utiles pour le Raspberry Pi :

Toutes les actions utilisées dans l'interface graphique Raspbian « changer de répertoire, créer un fichier, lancer un programme... » Sont réalisables en ligne de commandes c'est à dire en saisissant des commandes dans un terminal. Pour ouvrir le terminal il suffit de cliquer sur l'icône LXTerminal sur le bureau ou le retrouver dans le menu des programmes, dans la catégorie « Accessoires ». Une fois ouvert, nous devrions nous retrouver face à une fenêtre noire comme celle-ci Figure 3.4

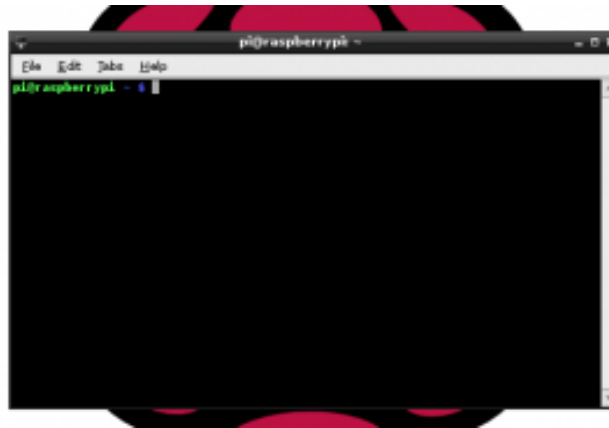


Figure 3.4 : fenêtre de ligne de commande

Il est indispensable, pour configurer et utiliser le Raspberry pi de connaître quelques commandes Linux. Une aide exhaustive pour chaque commande est disponible sous Linux par la commande « man nom-de-la-commande ». Le tableau 3.1 présente les commandes les plus utiles ainsi que leur description :

shutdown –r now	redémarrer
shutdown –h now	éteindre
ls	voir le contenu du dossier
ls –l	voir le contenu du dossier et les droits sur les fichiers
pwd	afficher le nom du dossier courant
cd dossier	descendre dans l’arborescence
cd..	remonter dans l’arborescence
nano nom_fichier	éditer un fichier
exit	se déconnecter e la machine
cp source destination	copier un fichier
mkdir nom	créer un dossier
mv source destination	déplacer un fichier/dossier
mv nom1 nom2 (dans le même dossier)	renommer un fichier
rm fichier et rm –r dossier	supprime un fichier ou un dossier
chmod +x nom_du_fichier	Rendre un fichier exécutable
./nom_du_fichier	Exécuter un fichier

Tableau 3.1. Les commandes le plus utilise

3.6 Les langages de programmation que l'on peut utiliser avec un Raspberry pi :

Le langage de programmation officielle qui a été pris en charge comme langue d'apprentissage éducatif sur le Raspberry pi est le Python. Comme c'est le langage officiel du Raspberry pi, il est installé par défaut dans Raspbian. Ce langage est puissant et pas très difficile à apprendre. D'autres langages de programmation peuvent être pratiqués avec un Raspberry pi comme le HTML5, le Java, le JavaScript, le C et le C++ ou encore le Perl. Cependant, n'importe quel langage de programmation qui compilera sur un processeur ARMv6 est utilisable comme un moyen de développement sur le Raspberry pi[9].

3.7 Utilisation comme un système embarqué

Le Raspberry est un mini-PC aux caractéristiques physiques impressionnantes. L'idée est donc de pouvoir préserver ce potentiel de légèreté et de simplicité sans avoir besoin d'ajouter de nombreux périphériques pour contrôler le système, ceci est primordiale pour la robotique mobile autonome. L'idéal serait donc de n'utiliser que l'indispensable à savoir une alimentation 5V régulée et une connexion réseau sans fil par wifi pour pouvoir contrôler le Raspberry Pi à distance. Au final, on va obtenir le système fonctionnel suivant (Figure 3.5), accessible sans fil et par wifi, programmable à distance, ce qui ouvre pas mal de possibilités pour la robotique autonome.

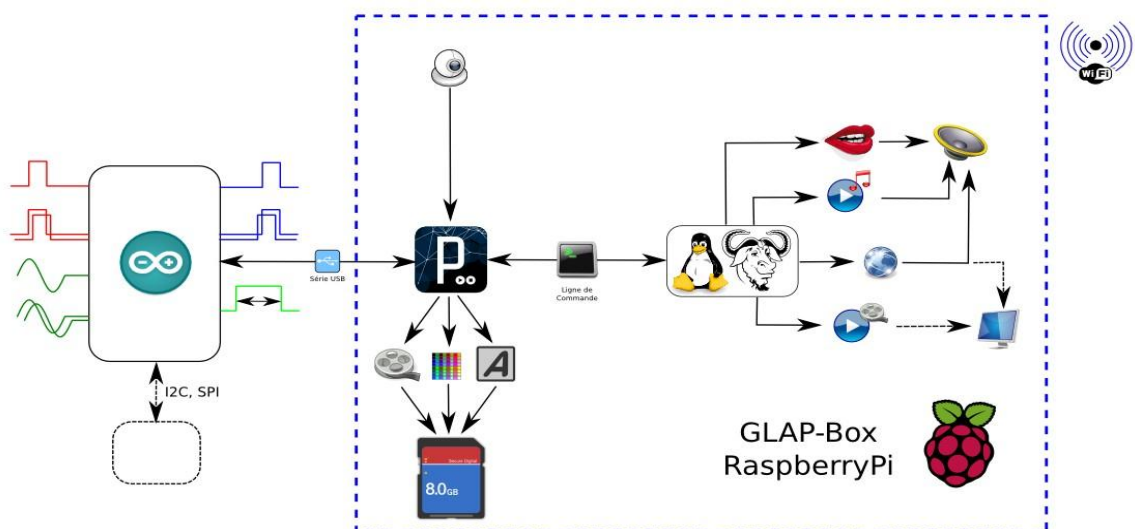


Figure 3.5 : système fonctionnel de Raspberry pi

3.7.1 Rendre le système accessible à distance :

L'utilité de pouvoir contrôler le Raspberry pi à distance rend son utilisation très attractive pour la robotique mobile autonome. Grâce à un ordinateur connecté au réseau et au protocole SSH, il est possible de piloter le Raspberry pi à distance. Le SSH « Secure Shell » est un protocole de communication sécurisé qui permet de se connecter à distance à un ordinateur dans notre cas « le Raspberry pi ».

3.7.2 Composants nécessaires on mode embarqué :

a) une mini-alimentation :

L'alimentation est un point essentiel pour utiliser le Raspberry pi en mode embarqué. Le Raspberry Pi s'alimente via un connecteur micro USB, en 5V/400mA. On pourra aussi alimenter le Raspberry Pi en 5V régulé directement sur le connecteur droit GPIO, sur les broches 0 et 5V.

b) Module Caméra pour Raspberry Pi :

Cette caméra est un accessoire spécifiquement réalisé pour la Raspberry Pi. La caméra se connecte à la Raspberry Pi par l'une des deux sockets CSI qui se trouve sur la Raspberry Pi. L'interface CSI a été spécifiquement conçue pour transférer des données caméra et propose donc une bande passante élevée. La carte caméra a une taille de 25mm x 20mm x 9mm et un poids de 3g, faisant de cette caméra une solution particulièrement adaptée aux solutions embarquées telle que la robotique. La résolution de la caméra est de 2592 x 1944 pixels pour une image statique et supporte également les résolutions vidéos suivantes : 1080p30, 720p60 et 640x480p60/90.

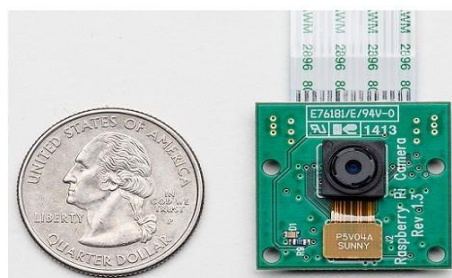


Figure 3.6 : Module camera pi

c) Arduino :

Pour commencer pour ceux qui ne connaissent pas un Arduino est un microcontrôleur open source qui se programme via une logicielle aussi open source bien sûr, le tout dans une forme de C++, son avantage sur le Raspberry Pi est qu'il dispose de plus de ports d'entrée/sortie, des entrées analogiques, une dizaine de sorties PWM et quelques sorties Tx/Rx.

Et importante il dispose de port I2C, car le but dans mon cas est d'utiliser l'Arduino comme esclave du Raspberry Pi pour simplement éviter d'utiliser les GPIO du RPi. L'autre avantage de l'Arduino sur le Raspberry Pi qui est due notamment à son ancienneté, c'est la quantité de bibliothèques disponibles pour tout et n'importe quoi.



Figure 3.7 : Arduino Uno

3.8 Conclusion :

Dans ce chapitre nous avons donné les caractéristiques essentielles du module Raspberry Pi afin de justifier son choix pour réaliser notre projet. Nous avons entre autres présenté les outils de développements nécessaires à l'exploitation de ce PC embarqué.

CHAPITRE . 4

MISE ŒUVRE ET RESULTATS

«La théorie, c'est quand on sait tout et que rien ne fonctionne .La pratique, c'est quand tout fonctionne que personne ne sait pourquoi .Ici, nous avons réuni théorie et pratique : rien ne fonctionne ...et personne ne sait pourquoi ! »

Albert Einstein

4.1 Introduction :

Dans ce chapitre nous allons présenter tous les aspects matériels et logiciels pour la réalisation de notre projet, à l'issue de ce dernier nous donnerons d'une manière graphique, les résultats obtenus à travers un échantillon des différents tests expérimentaux que nous avons effectué.

4.2 La conception mécanique :

L'étape de la conception mécanique est primordiale pour la réalisation d'un robot mobile. Notre robot est un robot à 4 roues ; Deux roues motrices disposées au centre du robot qui permettront d'avancer/reculer/tourner, les deux autres roues libres sont disposées à l'avant et à l'arrière du robot, celles-ci assurent la stabilité de la plateforme. Ce robot est équipé d'une pince en aluminium destinée à manipuler l'objet à suivre. .

4.2.1:Moteur et roues :

a) Moteur

Des nombreux types de moteurs sont utilisés pour la robotique mobile, notre choix est le moteur à courant continu EMG30. L'EMG30 est un moteur 12v entièrement équipé avec des encodeurs et réducteurs. De plus à proximité de l'axe du moteur on trouve deux filets permettant de les fixer à un support pour assurer un montage facile sur le robot.



Figure 4.1 : moteur à courant continu EMG30 avec support

b) les roues

Pour ce robot mobile, on a choisi deux types de roues, à savoir, deux roues motrices et deux autres libres pour la stabilité du robot.



Figure 4.2 : Roue libre



Figure 4.3 : roue motrice

4.2.2. Pince et servomoteur :

Pour faire bouger/déplacer des objets, on a fabriqué une pince en aluminium. Cette pince s'ouvre et se ferme à l'aide d'un servomoteur. Ces moteurs sont intéressants car ils intègrent la mécanique et l'électronique dans un même boîtier et offrent un couple suffisant pour notre application.

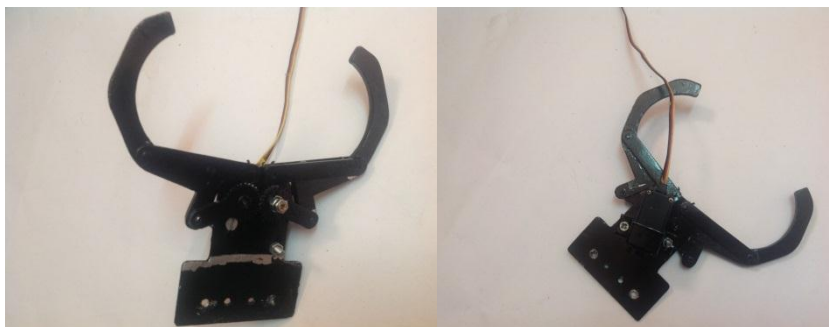


Figure 4.4 : pince d'aluminium

4.2.3 Châssis :

Le châssis est la base même de notre robot, la stabilité et la solidité de celui-ci va jouer un rôle important dans la qualité de la réalisation. Le châssis doit être imperturbable, il ne doit pas plier et ne former qu'une seule pièce de manière à ce qu'il n'y ait pas de jeu entre les différents montants. L'assemblage des différentes pièces mécaniques est fait avec des vis, afin de faciliter l'accessibilité à l'enceinte du robot, pour d'éventuelles interventions.

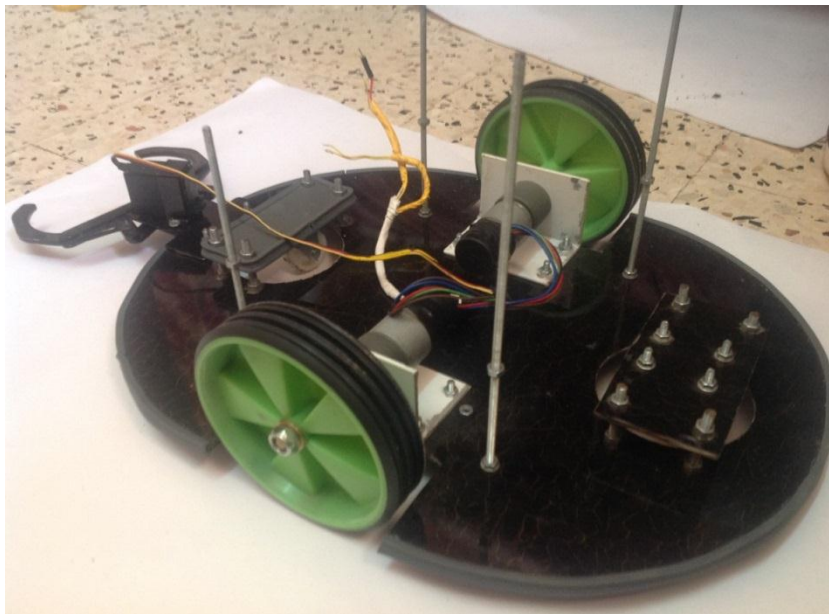


Figure 4.5 : Base mécanique du robot

La solution présentée dans cette section est le résultat de plusieurs possibilités envisagées à l'entame de ce projet. Le résultat obtenu est très satisfaisant puisque la solution est simple, tant au niveau mécanique qu'au niveau contrôle, et elle est surtout très efficace.

Pour que notre robot soit fonctionnel il nous faut concevoir des interfaces électroniques qui nous permettront de gérer correctement l'ensemble des systèmes mécaniques embarqués et principalement sa mobilité, ce que nous verrons dans la partie suivante.



Figure 4.6 : Vues isométriques du robot

4.3 Conception électronique :

La conception électronique doit débiter avec le choix de l'architecture. Le choix des parties électroniques constitue en fait l'identification de toutes les composantes qui formeront le système. La Figure 4.7 présente la structure du système électronique de notre robot. Dans cette partie nous décrivons chacun des éléments de cette structure, ainsi que les entrées/sorties entre chaque élément.

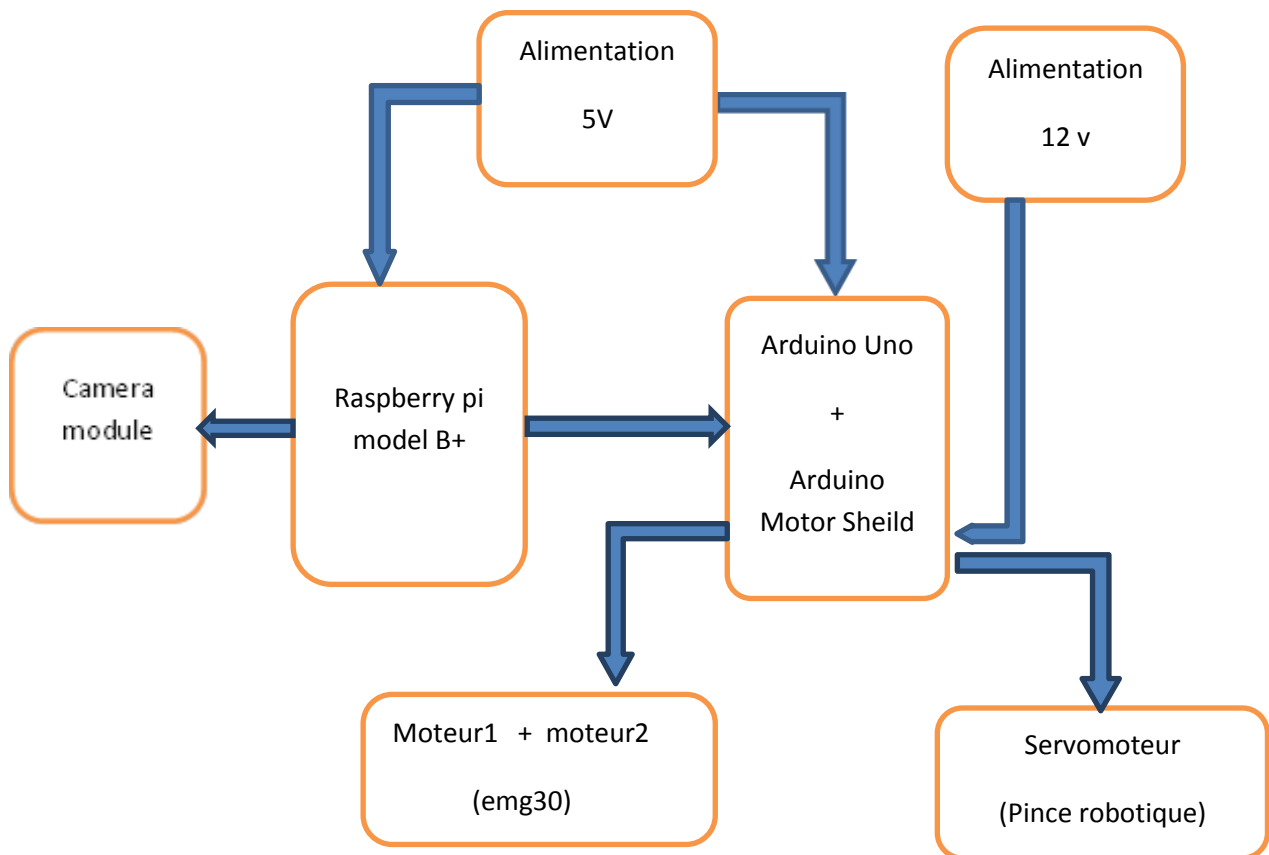


Figure 4.7: Structure électronique du robot

4.3.1 LE RASPBERRY PI :

Comme on l'a décrit dans le chapitre précédent. Le Raspberry Pi est un ordinateur particulièrement petit possédant des entrées GPIO, il est souvent utilisé dans le cadre de projets liés à l'électronique, en tant que contrôleur central d'un système informatique. Il est utilisé dans notre projet comme unité de traitement informatique qui permet d'implanter les différents algorithmes désirés. Grâce à cet ordinateur, nous pouvons facilement utiliser notre système d'exploitation préféré et les langages de programmation choisis.

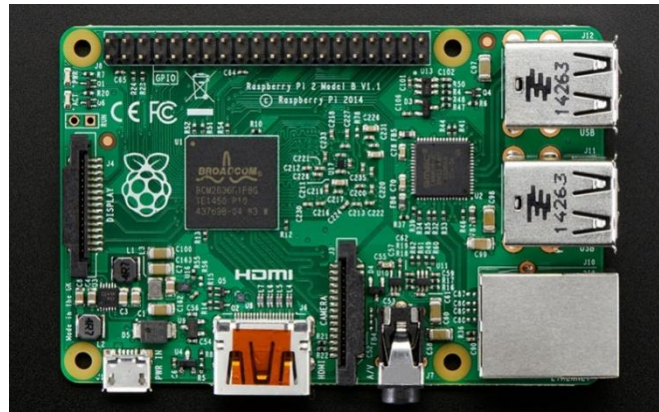


Figure 4.8: l'ordinateur embarqué Raspberry pi

4.3.2 Arduino Uno :

Il existe plusieurs moyens qui nous permettent de commander un robot ; parmi elles : les microprocesseurs, les microcontrôleurs, les automates programmables, etc. Dans notre cas nous avons choisi l'Arduino Uno. Le cœur de cette carte est un microcontrôleur ATMEL de référence ATmega328 dont la programmation peut être réalisée en langage C. L'intérêt principal d'utiliser Arduino comme carte de commande est de partager le calcul avec le Raspberry pi.

4.3.3 CONECTION RASPBERRY-ARDUINO :

Pour lier les deux modules, nous allons passer par le protocole I2C. Le protocole I2C, créé par Philips avec une orientation domotique, permet à plusieurs composants de dialoguer entre eux de manière bidirectionnelle.

Ce protocole se base sur un principe de maître et esclave. Ici nous allons donc utiliser le Raspberry Pi en tant que maître et l'Arduino en tant qu'esclave. Le branchement Pour le Raspberry se fait sur les GPIO 2 et 3 et sur les broches A4 et A5 pour l'Arduino UNO, Il faudra ajouter une connexion pour les masses des deux cartes.

Avant de connecter les deux modules, il faut s'assurer qu'ils vont pouvoir communiquer. Ceci doit passer par une affectation d'une adresse particulière à l'esclave tout en tenant compte de cette valeur lors des transmissions émises par le circuit maître.

Nous pouvons maintenant relier l'Arduino au Raspberry et allumer voir la Figure 4.9

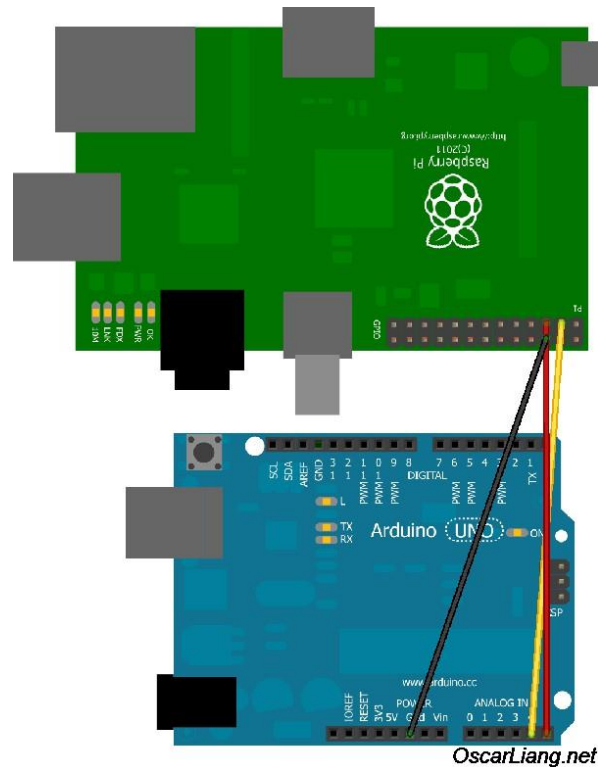


Figure 4.9: connexion Arduino Raspberry via I2C

Pour récupérer l'adresse de communication on ouvre le terminal de Raspberry et on tape la commande `i2cdetect`. La Figure 4.10 montre un exemple d'une adresse récupéré «0x12».

```

root@raspberrypi:~# sudo i2cdetect -y 1
00:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: 40 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: 70 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
root@raspberrypi:~#

```

Figure 4.10: adresse récupéré sur terminal de Raspberry

4.3.4 Shield moteur v1.0 Arduino :

On a utilisé comme carte de puissance le Shield moteur v1.0 Arduino pour piloter les deux moteurs à courant continu et un servomoteur.

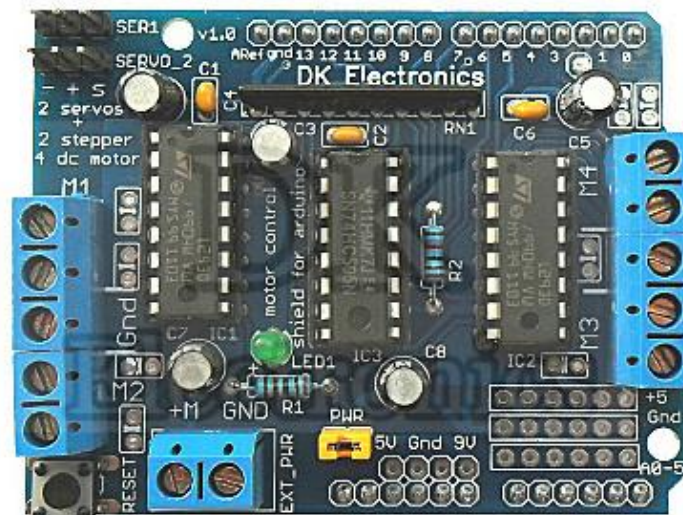


Figure 4.11: Shield moteur v1.0

Sur le Shield moteur v1.0 Arduino existe quatre ponts en H basé sur composant L293D qui peut fournir jusqu'à 0.6A par pont (1.2A crête) avec une protection d'arrêt thermique et diodes de roue libre internes. Il peut fonctionner sur les moteurs 4.5VCC jusqu'à 25VDC et deux connexions pour servomoteur 5V connectés. Dans notre réalisation on va utiliser que deux ponts pour piloter les deux moteurs assurant la motricité du robot et une connexion servomoteur pour actionner la pince.

a) Principe de pont de H

Un pont en H est un circuit destiné au pilotage d'un moteur ayant deux rôles principaux: Fournir le courant nécessaire au fonctionnement du moteur et donner la possibilité d'inverser le sens du courant « donc le sens de rotation du moteur ». Les caractéristiques du pont en H sont:

- Tension Max: 46V.

- Courant Max: 2A.
- Contrôle d'un moteur à courants continu.
- Utilisation: 3 entrées par pont IN1, IN2 et ENABLE.

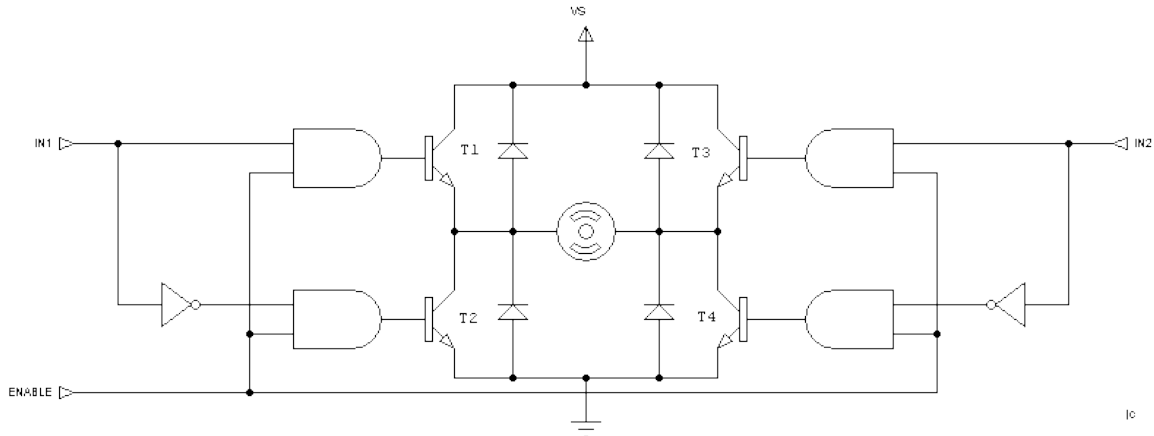


Figure 4.12: schéma électrique d'un pont H

Le codage de ses entrées (Voir la table de vérité: Tableau 4.1)

- si ENABLE=0 : Le moteur est libre.
- si IN1=1 et IN2=0 : Il tourne dans un sens.
- si IN1=0 et IN2=1 : Il tourne dans l'autre sens.
- si IN1=IN2 : Frein moteur.

IN1	IN2	moteur
0	0	Arrêt
0	1	Sens 1
1	0	Sens 2
1	1	Arrêt

Tableau 4.1 : Table de vérité des signaux de **commande d'un moteur**

b) Commande du moteur DC par les signaux PWM :

Il existe plusieurs méthodes pour commander un moteur DC : Commander son sens de rotation par un relai monté en tout ou rien ou bien un transistor. Commander sa vitesse par variation de la tension à ses bornes ou par génération de signaux PWM. La variation de

vitesse par commande des signaux PWM consiste à fournir au moteur une tension constante soit la tension maximale. Cette tension ne sera appliquée que par très courtes périodes de temps. En ajustant la longueur de ces périodes de temps, on arrive à augmenter la vitesse des moteurs. La vitesse des moteurs devient proportionnelle à la longueur des périodes de temps. Cette méthode est très utilisée en contrôle des systèmes et en électronique.

Un signal PWM est un signal dont la période est fixe, mais le rapport cyclique varie. En d'autres termes, t_1 et t_2 varient tout en conservant la période $T = \text{constante}$. (Figure 4.13) On peut définir: t_1 : temps d'enclenchement

t_2 : temps de déclenchement

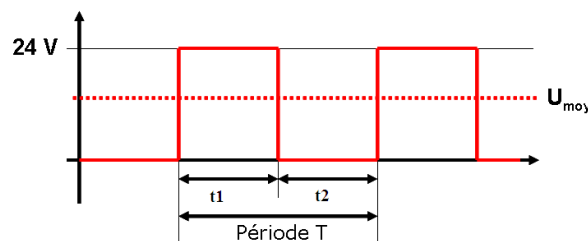


Figure 4.13 : Signal PWM

La tension moyenne vue par la charge est donc dépendante du rapport entre le temps d'enclenchement et le temps de déclenchement, nous parlons de taux PWM, donné habituellement en pourcentage (%). Pour l'exemple montré avec 24 Volts, la PWM nous permet de balayer toute la plage de 0V (taux de 0%) à 24V (taux de 100%) et ainsi avoir une source variable de 0 à 24V.

4.3.5 Camera module :

Cette caméra permet de faire l'acquisition d'un signal vidéo analogique provenant d'une caméra vidéo afin d'effectuer la vision artificielle nécessaire au contrôle du robot. Elle a une définition maximale de 5 millions de pixels et est capable de filmer en 1080p à 30 images/s ou en 720p à 60 images/s. Elle est montée sur une petite carte qui se connecte au Raspberry Pi via une nappe « un bus de type CSI-2 », ce qui laisse l'avantage de rendre disponible les ports USB de la carte peu nombreux.

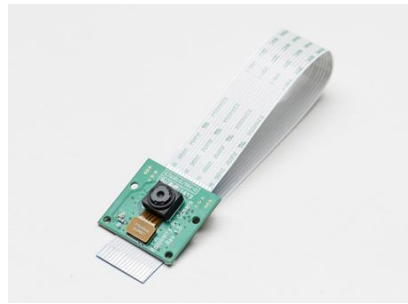


Figure 4.14 : Camera module

4.4 Conception informatique

L'ordinateur embarqué performant dont nous disposons nous permet dans un premier temps d'éliminer plusieurs alternatives telles que la programmation en assembleur ou autre langage de bas niveau. Il est évident que l'utilisation d'un système d'exploitation évolué et de langages de programmation de haut niveau permettent l'implémentation d'un logiciel de contrôle complexe et modulaire. Les éléments qui doivent donc être déterminés avant de pouvoir débiter l'implémentation du logiciel de contrôle sont le système d'exploitation ainsi que le/les langage(s) de programmation qui seront utilisés. Nous avons choisi le Raspbian comme système d'exploitation et python comme langage de programmation.

4.4.1 LANGAGES DE PROGRAMMATION :

Pour l'implémentation et le développement de de robot on a utilisé le langage Python.

Python est un langage de programmation orienté objet de très haut niveau. Il a l'avantage d'être un langage interprété et non compilé. Ceci signifie qu'il n'y a pas besoin de compilateur pour commencer à programmer. Python est le principal langage utilisé par la communauté internet pour le développement d'applications sur Raspberry Pi

4.4.2 L'outil OpenCV :

La bibliothèque open CV est orientée vers des applications en temps réel. Un des objectifs buts d'OpenCV est d'aider les utilisateurs à construire rapidement des applications de vision

à l'aide de simples opérations de vision par ordinateur. Elle propose plus de 2500 algorithmes de vision par ordinateur plusieurs interfaces ont été développées pour Python.

4.4.3 L'algorithme de vision :

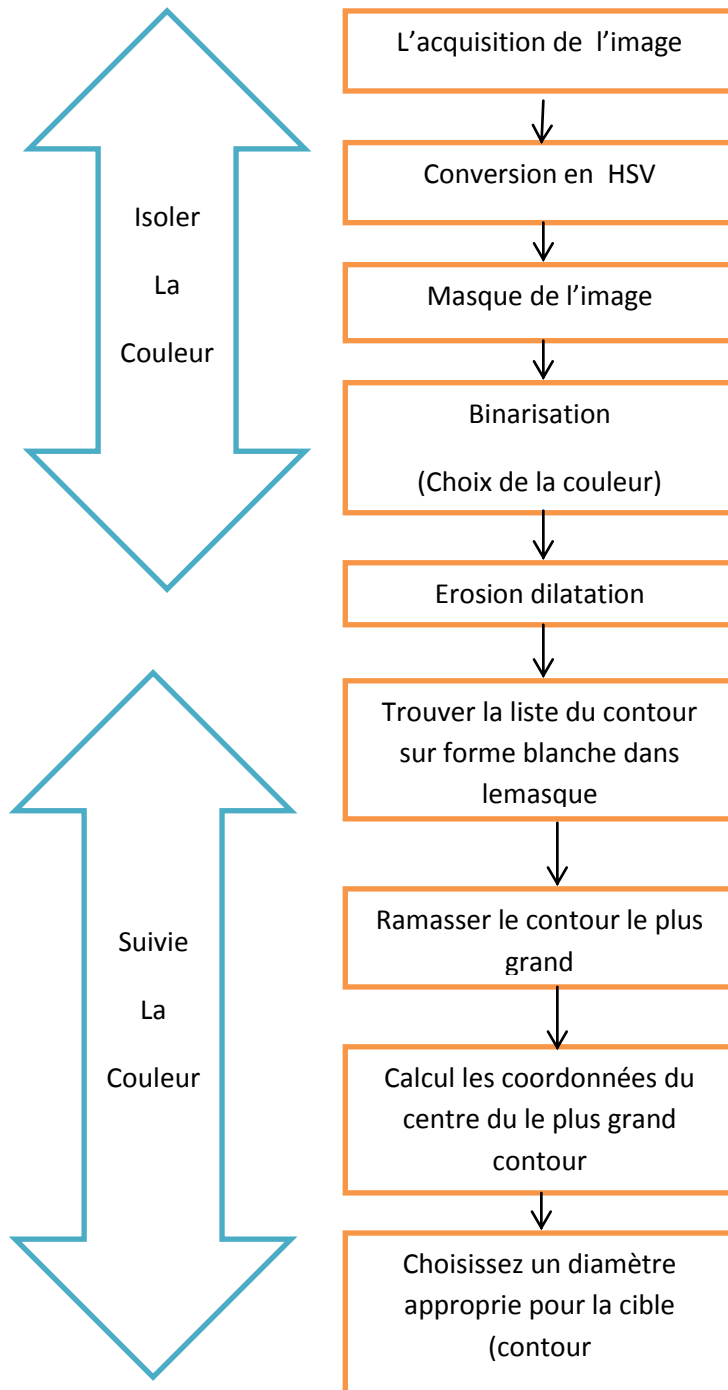


Figure 4.15 : organigramme de suivi de couleur

Dans la présente partie l'algorithme présenté dans la figure 4.15 sera détaillé étape par étape.

a) Conversion de l'image en HSV

Il faut tout d'abord convertir l'image qui est en RGB en HSV. Il est préférable de passer l'image en HSV car ainsi il est possible de se baser sur la teinte et la saturation de la couleur en laissant plus libre la brillance (V- valeur) de cette dernière. Ce qui permet, en partie seulement, d'éloigner les problèmes liés à l'éclairage.

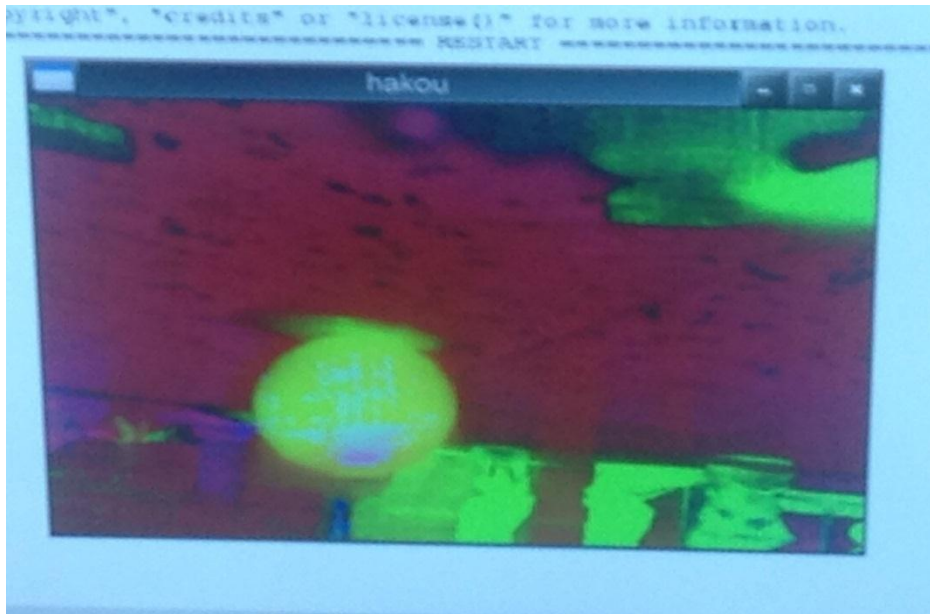


Figure 4.16 : conversion BGR- HSV

b) Création du masque :

La création de masque est nécessaire pour permettre de représenter l'image binarisée, il doit donc être lui aussi de la même taille que notre image originale.

c) Binarisation Choix de la couleur :

La binarisation d'une image consiste à transformer une image de niveau de gris ou couleur en une image binaire. On choisit un seuil, soit fixe soit aléatoire, et on compare tous les

pixels de l'image à ce seuil. Si la valeur de l'intensité du pixel est plus grande que la limite définie on mettra un 1 pour le pixel de l'image en deux classes distinctes.

Notre première classe correspondra à notre couleur à isoler, la seconde à toutes les autres couleurs. Ainsi, si la couleur sélectionnée est le rouge, nous verrons sur notre image binarisée tous les éléments rouges apparaître en blanc sur fond noir. Afin de remplir le masque il faut faire le choix des paramètres HSV de la couleur traquée.

Il est possible de donner les valeurs initiaux HSV de couleur directement dans le code ou le faire manuellement en choisissant la couleur désirée grâce à la souris dans ce projet le choix se fera directement dans le code.

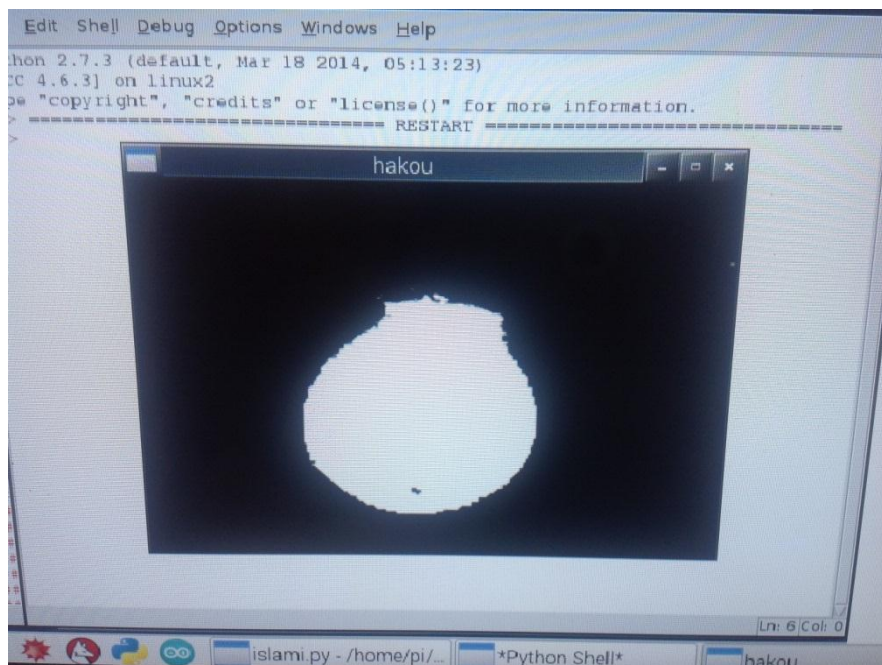


Figure 4.17 : binarisation (choix de couleur)

d) Erosion et dilatation :

Après le choix de la couleur, et la création du masque binarisée vient l'étape de l'érosion et la dilatation. Afin d'isoler l'objet voulu dans le masque ces deux étapes sont obligatoires, a part si on choisit la tâche la plus présente dans le masque. Une érosion permet de supprimer les pixels (isolés) qui ne correspondent pas à notre objet traqué. Ensuite vient l'étape de la

dilatation qui nous permet de renforcer les groupes denses de pixels notamment l'objet suivi.

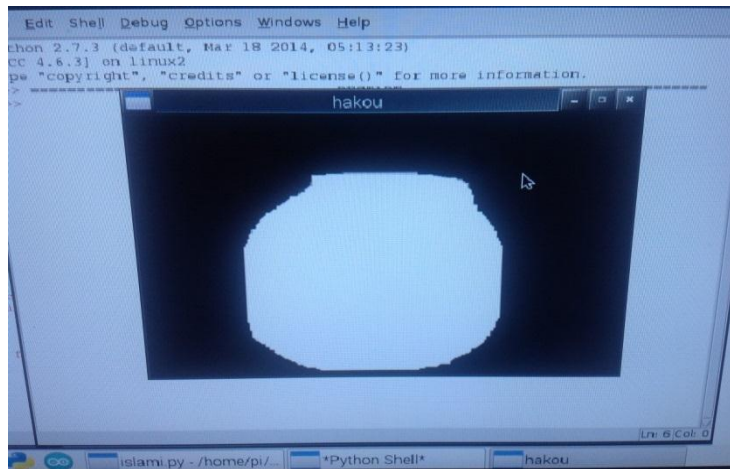


Figure 4.18 : érosion et dilatation

e) Trouver la liste des contours :

Nous pouvons utiliser une fonction pour trouver les « contours » qui sont les contours d'une région d'une seule couleur. Les contours sont eux-mêmes des listes de points (donnés par les coordonnées x et y) qui entourent une région de couleur dans un style joint-le-points. Une deuxième fonction dessine alors les contours.

f) Calcul des coordonnées de centre du plus grand contour :

Il nous faut récupérer les coordonnées x et y de notre objet afin de calculer son barycentre. Pour cela nous parcourons le masque et nous additionnons les coordonnées (x et y respectivement) de tous les pixels blancs dans notre cas. Nous comptons au passage le nombre de point afin de pouvoir diviser la somme.

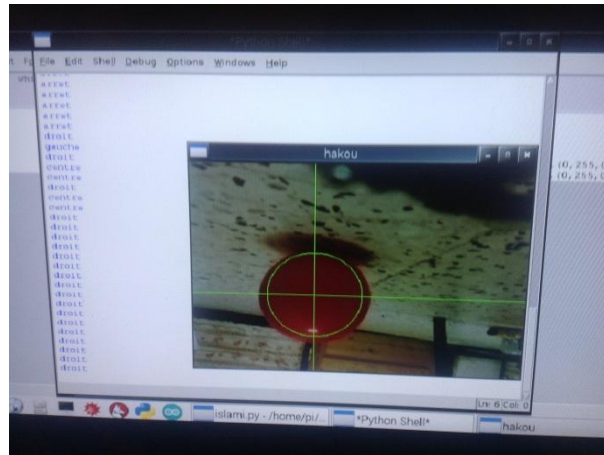


Figure 4.19 : calcul du centre de l'objet

4.4.4 Localisation de l'objet et déplacement du système :

Après la récupération des coordonnées du centre de gravité $G (j_G, i_G)$, on cherche à superposer ce centre avec l'axe central vertical de l'image, en comparant $x = j_G$ et $x = w/2$ tel que w est la largeur de l'image.

- Si $j_G > \frac{w}{2}$: il faut tourner le robot à gauche
- Si $j_G < \frac{w}{2}$: il faut tourner le robot à droite
- Si $j_G = \frac{w}{2}$: il faut que le robot avance

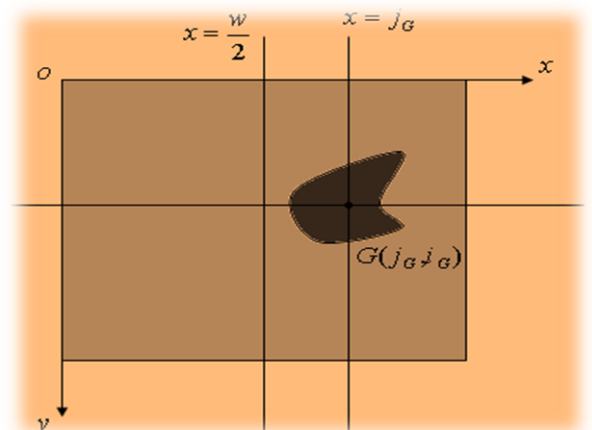


Figure 4.20 : localisation de l'objet
Dans le cas idéal

On sait qu'on ne peut jamais arriver à $j_G = \frac{w}{2}$, car le système va osciller autour de l'axe

$x = \frac{w}{2}$. Pour éliminer ces oscillations on va utiliser une bande morte B au voisinage de cet axe.

S_p : La surface pour laquelle le robot va fermer la pince

- Si $j_G > \frac{w}{2} + \frac{B}{2}$: il faut tourner le robot à gauche
- Si $j_G < \frac{w}{2} + \frac{B}{2}$: il faut tourner le robot à droite
- Si $j_G = \frac{w}{2} + \frac{B}{2}$: le robot avance

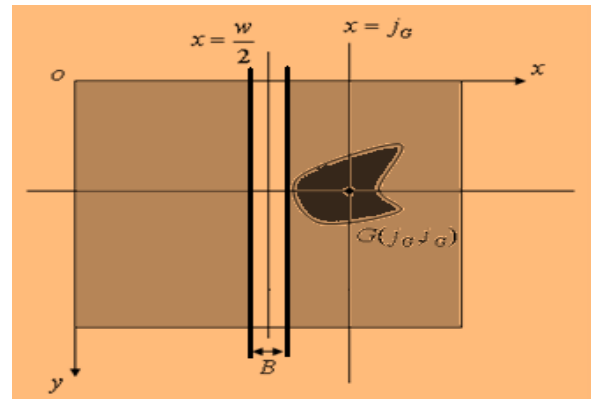


Figure 4.21 : localisation de l'objet
Dans le cas réel

- ✓ si $SG \leq S_p$: il faut que le système avance.
- ✓ si $SG > S_p$: il faut que le système arrête et ferme la pince.

Les étapes de localisation du système sont illustrées sous forme d'organigramme de la figure suivante :

Si « val > 5 » : le système
Cherche le deuxième objet

si « val ≤ 5 » : le système
cherche le premier objet

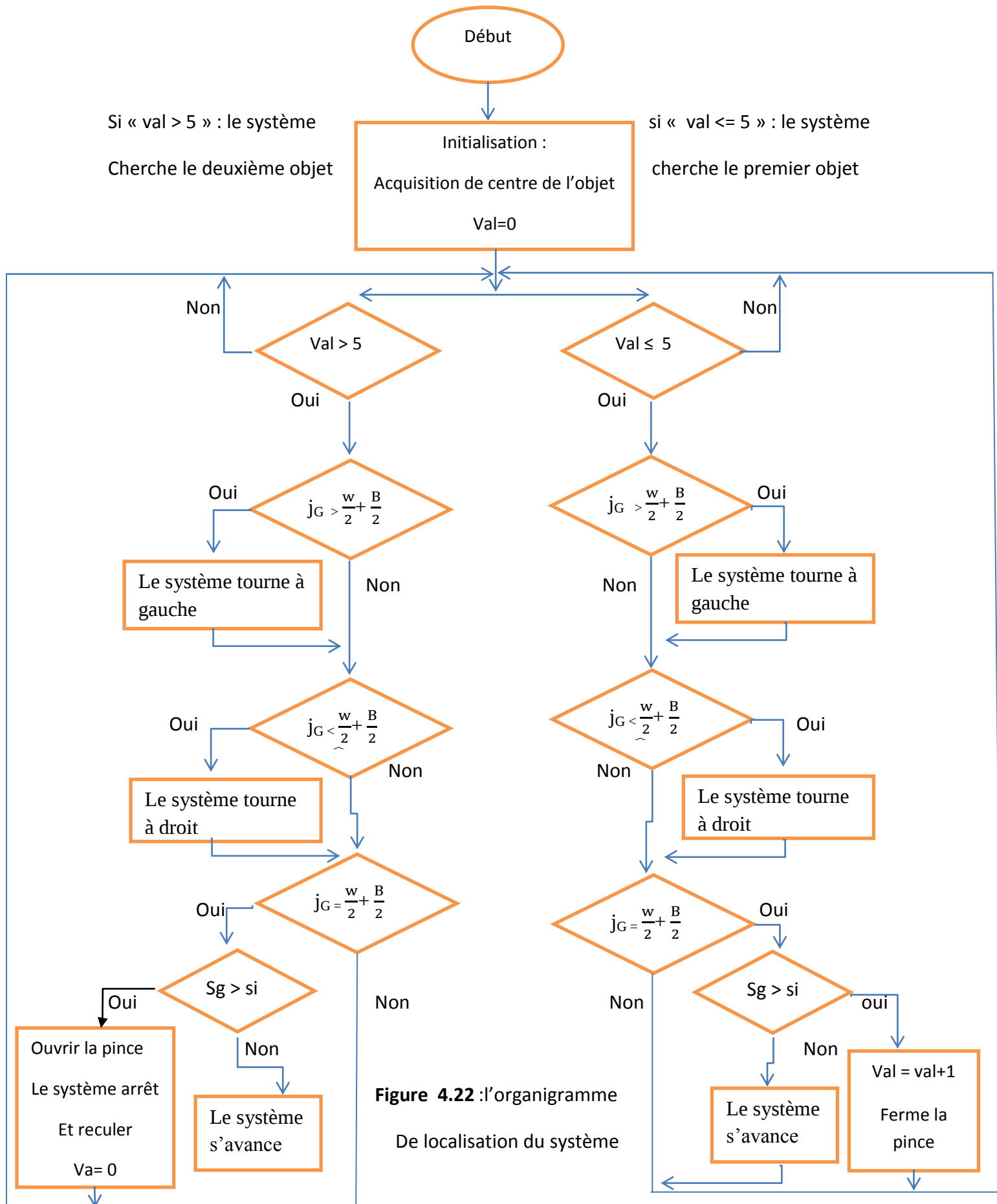


Figure 4.22 : l'organigramme
De localisation du système

Afin d'illustrer le fonctionnement de notre robot tel que décrit dans ce chapitre nous présentons ci-dessous une séquences d'images prises à des instants réguliers. Ces photos donnent les différentes positions du robot pour accomplir sa mission, à savoir :

- 1- se déplacer vers l'objet,
- 2- ramasser l'objet
- 3- se diriger vers l'endroit de dépôt de l'objet
- 4- se préparer pour une nouvelle mission



Le robot prend la direction de l'objet (balle rouge)



Le robot ramasse l'objet



Le robot se dirige vers l'endroit (couleur bleue) de dépôt de l'objet



Objet déposé à l'endroit prévu



Le robot se retire pour une nouvelle tâche

Figure 4.23 : Exemple de comportement du robot pour accomplir les tâches prévues

4.6 Conclusion :

Ce chapitre a été consacré à la description des éléments matériels et logiciels entrant dans la réalisation de ce projet. Nous avons donné l'essentiel des détails pour faire fonctionner notre robot et avons présenté un échantillon des résultats obtenus ; à savoir les mouvements effectués par ce robot tel que prévu dans le cahier des charges proposé.

CONCLUSION GENERALE ET PERSPECTIVES

« La théorie, c'est quand on sait tout et que rien ne fonctionne. La pratique, c'est quand tout fonctionne et que personne ne sait pourquoi. Ici, nous avons réuni théorie et pratique : rien ne fonctionne... et personne ne sait pourquoi ! »

Albert Einstein

Liste des tableaux

Tableau.1.1 :	Domaines d'applications des robots.....	06
Tableau.3.1 :	Les commandes le plus utilise.....	28
Tableau.4.1 :	Table de vérité des signaux de commande d'un moteur.....	40

Annexe 1: motoréducteur emg30

Alimentation: 12 Vcc

Consommation: 530 mA (150 mA à vide)

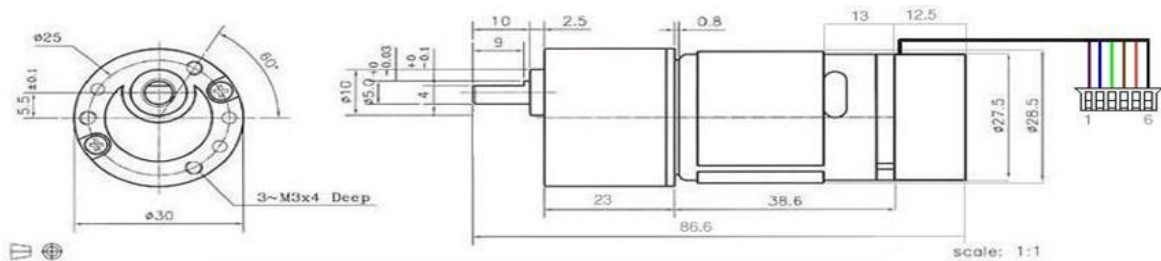
Courant de blocage: 2,5 A

Vitesse de rotation: 170 t/min (216 t/min à vide)

Couple: 1,5 kg.cm

Encodeur: 360 impulsions/tour

Dimensions: $\varnothing 28,5 \times 86,6$ mm (axe de 10 mm inclus)



Connector:

The EMG30 is supplied with a 6 way JST connector (part no PHR-6) at the end of approx 90mm of cable as Standard.

The connections are:

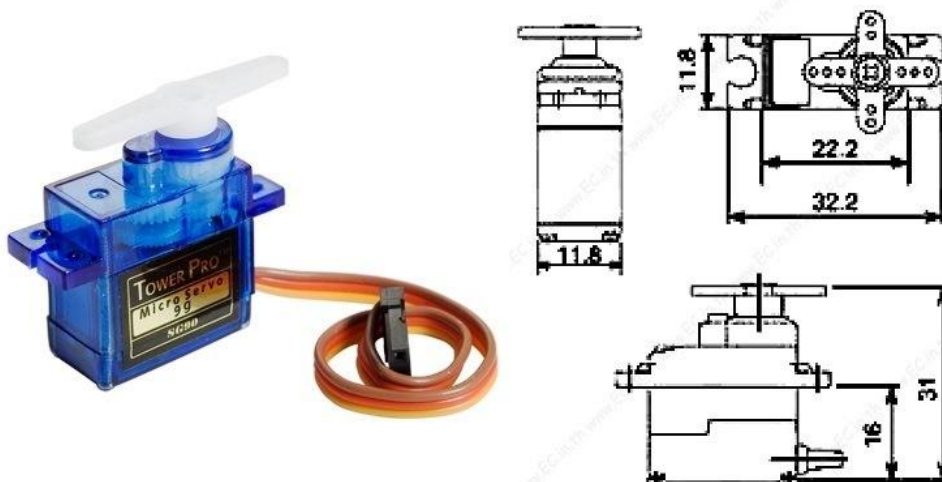
Wire colour	Connection
Purple (1)	Hall Sensor B Vout
Blue (2)	Hall sensor A Vout
Green (3)	Hall sensor ground
Brown (4)	Hall sensor Vcc
Red (5)	+ Motor
Black (6)	- Motor

Spécification

Rated voltage	12v
Rated torque	1.5kg/cm

Rated speed	170rpm
Rated current	530mA
No load speed	216
No load current	150mA
Stall Current	2.5A
Rated output	4.22W
Encoder counts per output shaft turn	360

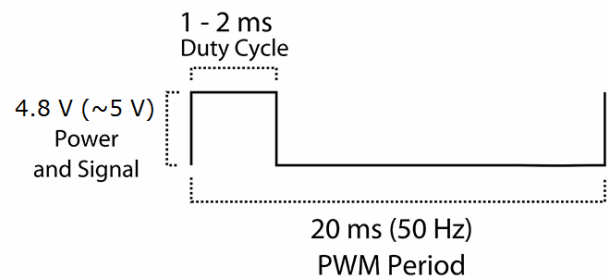
Annexe 2 : SG90 9 g Micro Servo



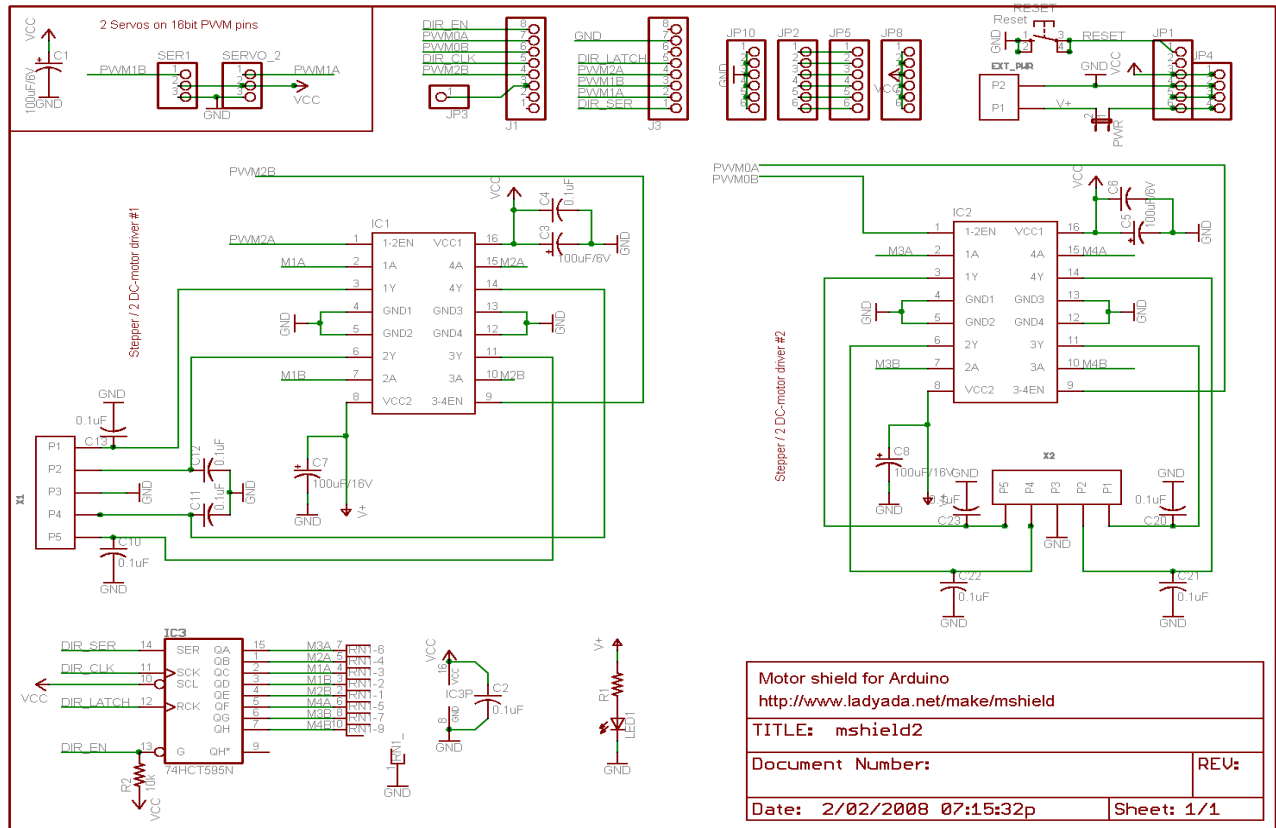
Spécifications :

Weight: 9 g
 Dimension: 22.2 x 11.8 x 31 mm approx.
 Stall torque: 1.8 kgf · cm
 Operating speed: 0.1 s/60 degree
 Operating voltage: 4.8 V (~5V)
 Dead band width: 10 μ s
 Temperature range: 0 °C – 55 °C

PWM=Orange (⏏)
 Vcc = Red (+)
 Ground=Brown (–)



Annexe 3 : Shield moteur v1.0 Arduino



- **2 connections for 5V 'hobby' servos** connected to the Arduino's high-resolution dedicated timer - no jitter!
- **Up to 4 bi-directional DC motors** with individual 8-bit speed selection (so, about 0.5% resolution)
- **Up to 2 stepper motors** (unipolar or bipolar) with single coil, double coil, interleaved or micro-stepping.
- 4 H-Bridges: L293D chipset provides 0.6A per bridge (1.2A peak) with thermal shutdown protection, 4.5V to 25V
- Pull down resistors keep motors disabled during power-up
- Big terminal block connectors to easily hook up wires (10-22AWG) and power
- Arduino reset button brought up top
- 2-pin terminal block to connect external power, for separate logic/motor supplies

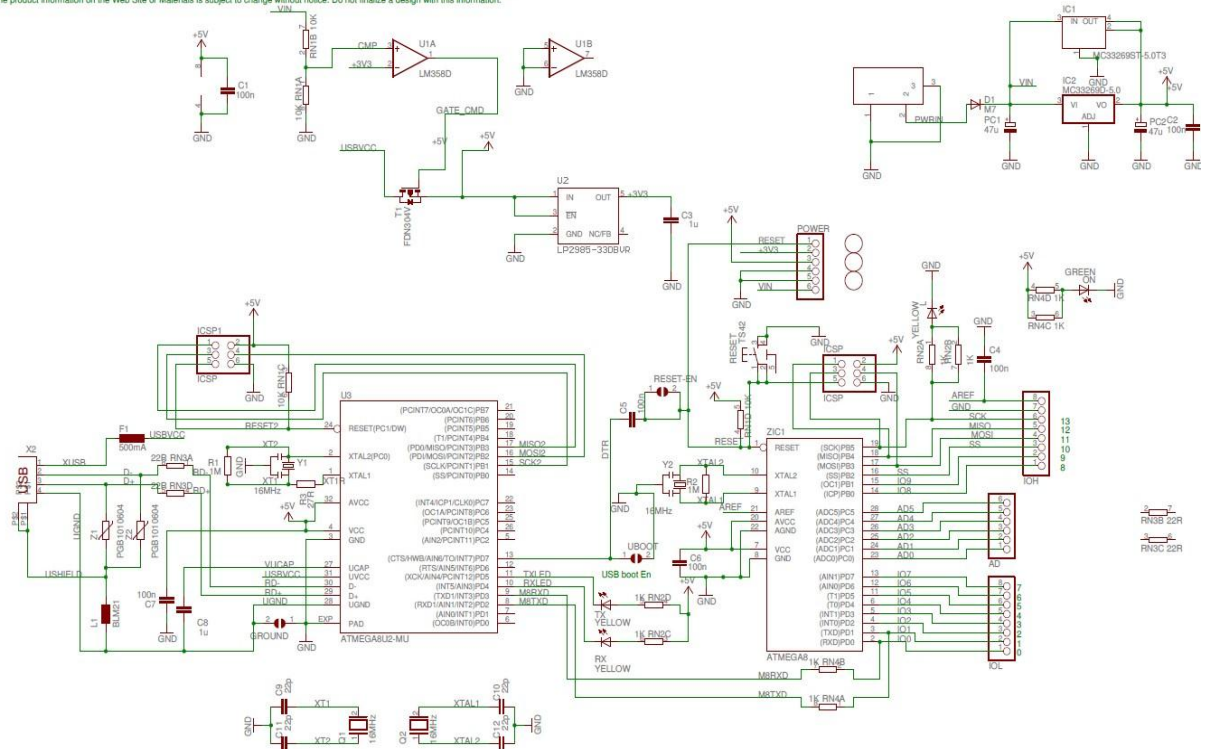
- Tested compatible with Mega, Diecimila, & Duemilanov

Annexe 3 : arduino Uno

Arduino™ UNO Reference Design

Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



ATmega328 Pin Mapping

Arduino function

reset
digital pin 0 (RX)
digital pin 1 (TX)
digital pin 2
digital pin 3 (PWM)
digital pin 4
VCC
GND
crystal
crystal
digital pin 5 (PWM)
digital pin 6 (PWM)
digital pin 7
digital pin 8

(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

Arduino function

analog input 5
analog input 4
analog input 3
analog input 2
analog input 1
analog input 0
GND
analog reference
VCC
digital pin 13
digital pin 12
digital pin 11 (PWM)
digital pin 10 (PWM)
digital pin 9 (PWM)

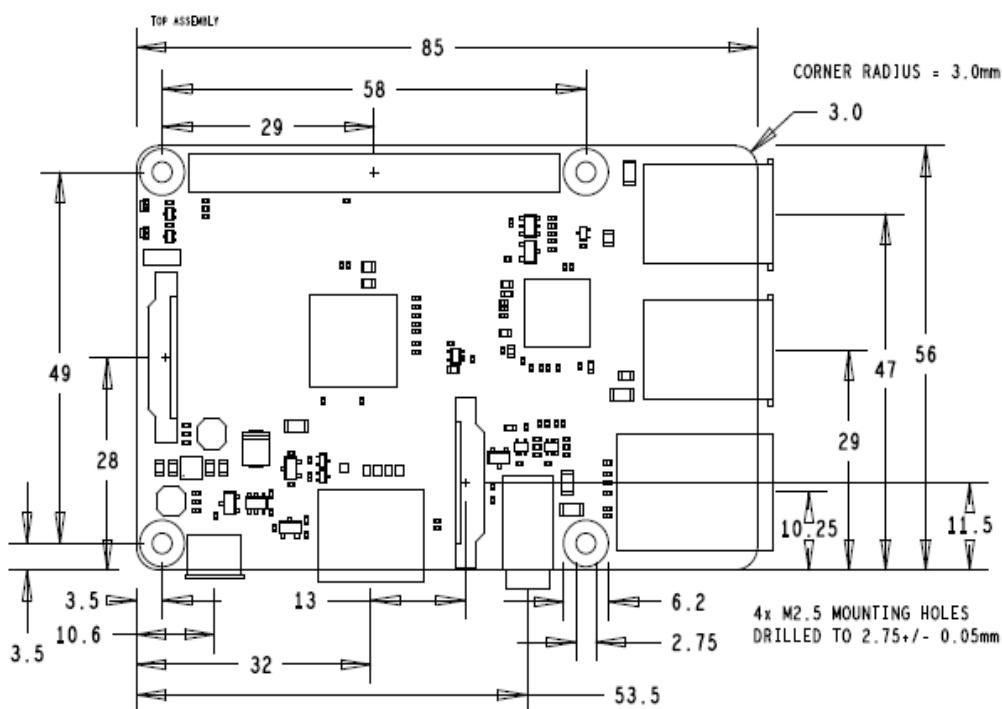
Digital Pins 11, 12 & 13 are used by the ICSP header for MISO, MOSI, SCK connections (Atmega 168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

annex 3 : Raspberry Pi model B+

Spécifications :

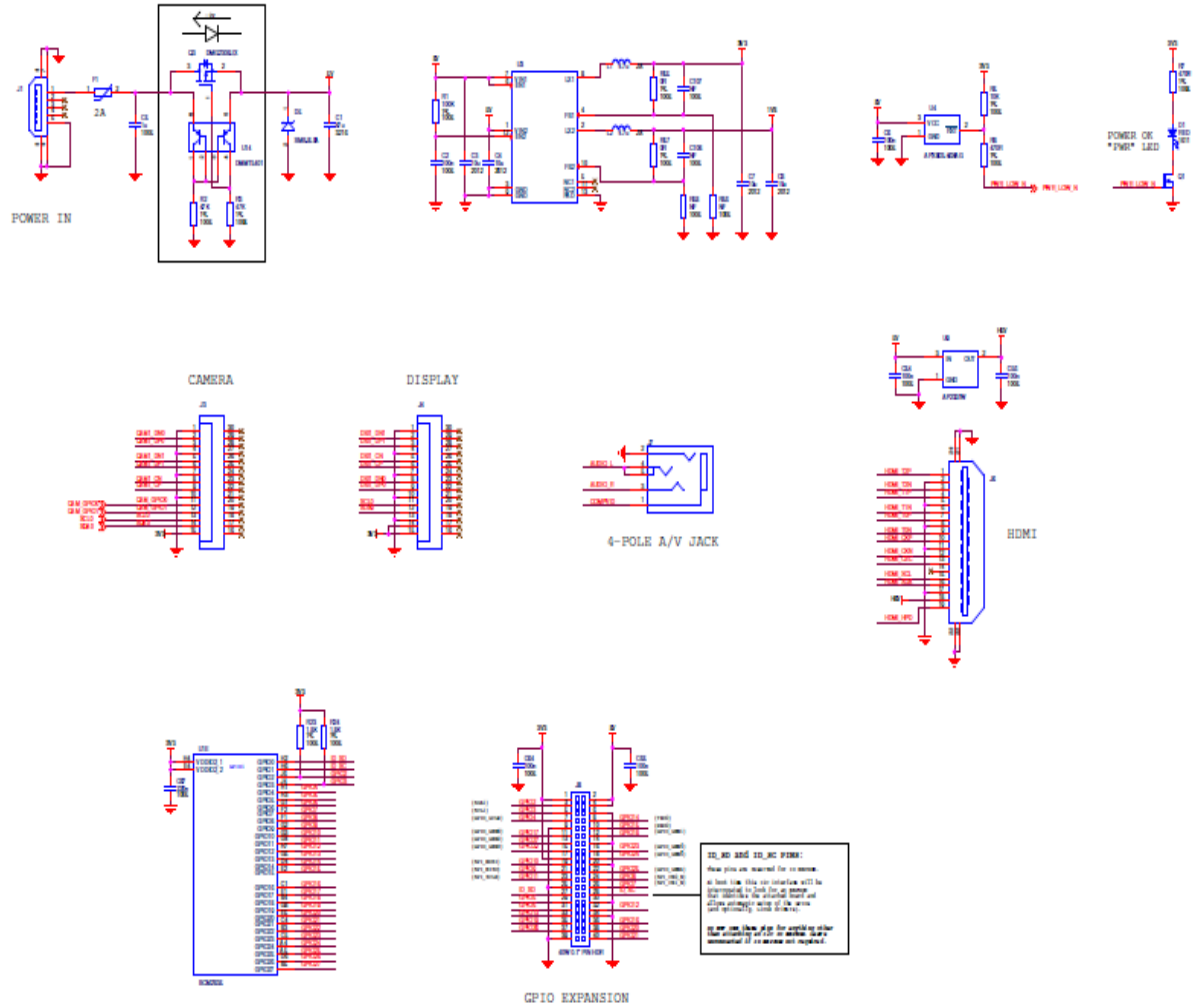
- Broadcom BCM2835 700MHz processor
- 512 MB SDRAM
- HD 1080p video output
- Composite video output (PAL / NTSC)
- Stereo audio output
- Ethernet port 10/100 BaseT RJ45
- Video / audio HDMI 1.3 and 1.4
- 4-pin Audio/Composite -Video Out jack, 3.5 mm
- 4 USB 2.0 ports
- 15-pin MPI CSI-2 connector for Raspberry Pi HD Video Camera
- 15-pin serial display interface connector
- MicroSD card slot
- Power supply: +5 V @ 2 A via microUSB jack
- Dimensions: 85 x 56 x 17 mm
- 40-pin connector for GPIO and serial buses (compatible with 26-pin male connector for Raspberry Pi Model A / B)

Dimensions



 Raspberry Pi www.raspberrypi.org © Raspberry Pi 2014			
TITLE	RASPBERRY PI MODEL B+		
DATE	07/03/2014	REF	RP1-BPLUS-V1_2
DRAWN	James Adams	APVD	James Adams

- **Raspberry Pi B+ IO Connectors Schematic**



Conclusion générale :

Durant ce projet nous avons eu à réaliser un robot mobile autonome dont la tâche consistait à reconnaître un objet de couleur prédéfinie pour ensuite aller le ramasser et le transporter à une zone caractérisée aussi par une autre couleur prédéfinie. La diversité de ces tâches nous a donc conduits à, non seulement étudier l'aspect intelligent du comportement du robot, mais aussi à la construction même de ce robot. A travers les différents chapitres de ce mémoire nous avons donné les éléments essentiels menant à la concrétisation de ce projet.

Parmi les nouveautés que nous pouvons citer dans ce projet, l'utilisation d'un système de traitement d'images totalement embarqué, occupe la place prépondérante, ceci étant donné que ce genre de traitements demande des ressources très importantes par rapport à la majorité des processeurs utilisés dans les systèmes de contrôle pour les robots mobiles. L'utilisation du module Raspberry Pi avec une caméra spécifique nous a permis de réaliser ses opérations de traitements grâce à l'exploitation de la bibliothèque de traitements d'images open cv (computer vision).

Bien que pour la réalisation de ce projet nous avons dû affronter différents types de problèmes, ces derniers nous ont permis d'approfondir notre expérience dans le domaine pratique et donc de mieux compléter notre formation pédagogique. Il faut toutefois signaler qu'en termes d'acquisition et de traitements vidéo en temps réel, le problème lié à l'éclairage de la scène reste le premier souci pour lequel il est très difficile de trouver une solution fiable.

Tel qu'illustré dans le dernier chapitre, les résultats obtenus sont très satisfaisants et ceci a contribué à nous donner plus de confiance dans la réalisation de ce genre de projets. Afin d'améliorer le déplacement du robot et éviter que ce dernier subisse des oscillations lors de ses mouvements, nous préconisons d'ajouter un asservissement de vitesse à même d'accélérer les différentes opérations effectuées par le robot.