

**CONCEPTION ET DEVELOPPEMENT D'UN SYSTEME DE
DETECTION RADAR POUR LA SURVEILLANCE DES
FRONTIERES TERRESTRES**

« CAS DE LA REPUBLIQUE DEMOCRATIQUE DU CONGO »

Présenté par :

TSHIBANGU DESIRE Joël

**Travail de fin de cycle présenté et défendu
en vue de l'obtention du titre de graduat
en Sciences Informatiques**

Directeur : Prof. MUSANGU LUKA
Rapporteur : Assistant ZUIYA
Bernice

EPIGRAPHE

« Toute technologie avancée est magique. »

Arthur Charles Clarke

DEDICACE

Nous dédions ce travail à nos très chers parents **TSHIBANGU André** et **FULAMA Mireille**, eux qui nous ont apporté leur soutien tant émotionnel que matériel tout au long de cette année académique.

Aussi, nous le dédions à nos très chers frère et sœurs, à tous ceux que nous aimons et à tous ceux qui nous aiment

REMERCIEMENTS

Nous remercions premièrement par excellence l’auteur de toutes créations sur terre, notre Dieu, qui nous a donné le courage, la foi ainsi que la patience et qui nous a permis de rester en bonne santé tout au long de cette année académique.

Merci au professeur **MUSANGU LUKA Marcel**, le doyen de la faculté des sciences informatiques d’avoir permis la conception de ce travail, d’y avoir mis son soutien indéniable pour sa réalisation, par la même occasion, nos remerciements s’en vont aussi à notre rapporteur **ZUIYA Bernice** dont l’expertise et la technicité nous ont été d’une grande aide.

Un grand merci à mon oncle Michel NSANZA pour son soutien financier, ce qui a permis la réalisation de ce travail grâce à l’achat de quelques équipements, merci à mes sœurs et frères, Naomi TSHIBANGU, Daniel TSHIBANGU, Gédéon TSHIBANGU, Merveille TSHIBANGU et Zoé TSHIBANGU qui m’ont toujours soutenu spirituellement.

Sans être exhaustif, nous ne saurions oublier nos très chers aînés et ami(e)s, John KAYANGI, Sage OTSHUDI, Bedel KAPUKU, Fils KASONGO, Exaucé BARUTI, Seke DJINGA, Benit TOMBOLA, Nelcy MANDE, Isaac ILLUNGA, Exaucé Ngoyi, Dove KASHOBA, Josué MIYIMI, David KASEREKA ainsi que les membres du groupe « H sur h » pour leur soutien intellectuel, émotionnel et leur encouragement durant notre parcours académique.

LISTE D'ABREVIATION

CW: Continuous wave

DGM : Direction Générale de Migration

DPEF : Division de la police des frontières et des étrangers

DPF: Division de la police des frontières

DSI : Direction des Systèmes d'Information

PWM: Pulse Wit Modulation

RADAR: RADio Detection And Ranging

RCO : Rapport Cyclique d'Ouverture

RDC : République Démocratique du Congo

SNCC : Système Numérique de Contrôle Commande

UML: Langage de Modélisation unifié

TABLE DES MATIERES

INTRODUCTION	9
0. Mise en contexte	9
1. Problématique	9
2. Objectifs de la recherche	9
3. Question de la recherche.....	10
4. Méthodes et techniques de travail.....	10
5. Délimitation du sujet.....	11
6. Difficultés rencontrées.....	11
7. Subdivision du travail	11
CHAPITRE 1 : REVUE DE LA LITTERATURE ET CONTEXTE DE TRAVAIL	12
Section 1. Revue de la littérature	12
1.1.1. Système d'information	12
1.1.2. Système Radar	13
1.1.3. Système numérique de contrôle commande	17
1.1.4. Le réseau Bluetooth	20
1.1.5. Application mobile	23
Section 2 : Contexte de travail.....	24
1.2.1. Direction Générale de Migration	25
CHAPITRE 2 : REVUE DE LA LITTERATURE ET CONTEXTE DE TRAVAIL	26
2.1. Narration.....	26
2.2.Modélisation avec uml	27
2.2.1. Diagramme de cas d'utilisation	27
2.2.2. Diagramme de classe	31
2.2.3. Diagramme de séquence	35
2.2.4. Diagramme d'activité	37
2.2.5. Diagramme de déploiement	38
CHAPITRE 3 : REALISATION ET CONTEXTE DU TRAVAIL	40
Section 1. Partie matérielle.....	40
3.1.1 Principaux composants utilisés.....	40
3.1.2. Schéma de branchement	44
Section 2. Partie logicielle.....	45
3.1.1 IDE utilisés	45
3.1.3. Sous-système 1 : Déplacement du robot par commande vocale.....	45

3.1.4. Sous-système 2 : Détection du radar et visualisation des informations de détections	53
CONCLUSION	57
BIBLIOGRAPHIE	58
NOTE DE COURS	58

LISTE DES FIGURES

Figure I.1 Principe de fonctionnement Radar	13
Figure I.2 Classification Radar.....	14
Figure I.3 Composition du Radar.....	16
Figure I.4 Couche de Spécification	20
Figure I.5 Représentation du pico réseau	22
Figure I.6 Représentation du Bluetooth	23
Figure II.1 Diagramme de cas d'utilisation.....	28
Figure II.2 Diagramme de classe su sous-système 1.....	32
Figure II.3 Diagramme de classe su sous-système 2.....	33
Figure II.4 Diagramme de séquence du cas : Se connecter.....	36
Figure II.5 Diagramme de séquence du cas : Commander le robot	36
Figure II.6 Diagramme de séquence du cas : Détecter et voir les informations	37
Figure II.7 Diagramme d'activité du système	38
Figure II.8 Diagramme de déploiement	39
Figure III.1 Arduino Uno	40
Figure III.2 Servo moteur.....	41
Figure III.3 Ultra son.....	42
Figure III.4 Pont H	43
Figure III.5 Motor Driver	43
Figure III.6 Motor DC	44
Figure III.7 Module Bluetooth	44
Figure III.8 Schéma de branchement 1	45
Figure III.9 Schéma de branchement 2	45
Figure III.10 Lancement de l'ide Arduino	44
Figure III.11 Lancement de l'ide Processing 3	46
Figure III.12 Lancement de l'ide Android studio	47
Figure III.13 Application mobile – Rechercher Bluetooth.....	48
Figure III.14 Application mobile – Connecter Bluetooth	49

Figure III.15 Application mobile – Commande Vocale.....	50
---	----

LISTE DES TABLEAUX

Tableau II.1 Tableau des acteurs et cas d'utilisation	28
Tableau II.2 Description textuelle du cas : Se connecter	29
Tableau II.3 Description textuelle du cas : Commander le robot	29
Tableau II.4 Description textuelle du cas : Se déplacer	29
Tableau II.5 Description textuelle du cas : Détecter un objet	29
Tableau II.6 Description textuelle du cas : Envoi des informations	30
Tableau II.7 Description textuelle du cas : Affichage des informations	30
Tableau II.8 Tableaux explicatif des classes	35

INTRODUCTION GENERALE

0. MISE EN CONTEXTE

Le Radar, de sa signification radio détection and ranging, est l'un des plus sophistiqués technologies pour l'observation de la terre et l'exploration planétaire, ils ont vu le jour après la seconde guerre mondiale, Aujourd'hui, les systèmes radar sont aussi très largement répandus dans des applications civiles telles que la surveillance du trafic aérien et/ou frontalier, la météorologie ou encore la sécurité automobile. Le radar est donc aujourd'hui un système indispensable et très largement utilisé qu'il sert à surveiller un territoire ou pour éviter des collisions.

Dans notre cas nous utiliserons le système radar pour la surveillance des frontières terrestres, celles de la république démocratique du Congo plus précisément; Ainsi la réalisation de notre système est portée sur deux parties principales, la première se compose de la partie matérielle par laquelle les opérations de détections et de déplacement seront effectuées à l'aide des composants disponibles, la deuxième est celle du logiciel, cette partie se focalise sur deux points : La visualisation où les informations sur la détection provenant de la partie matérielle sont affichées; La commande qui contrôlera le déplacement du robot radar à l'aide d'une application mobile, ce contrôle sera fait par commande vocale.

1. PROBLEMATIQUE

« Ensemble des problèmes élaborés par une science donnée et considérés comme délimitant le domaine qui lui est propre » (Aur. –WEIL 1981).

Aux vues des recherches faites sur les frontières terrestres de la République Démocratique du Congo, notre système Radar sera amené à pallier aux problèmes suivant :

- L'immigration illégale créée par la perméabilité de nos frontières
- La contrebande
- Le terrorisme qui mène l'insécurité des villes attachées directement aux frontières

La sécurité et la surveillance aux frontières doivent être exercées 24h/24 et 7jours/7, et aucune interruption ni baisse de vigilance ne doit être tolérées ainsi le radar pourra être d'une aide indispensable pour aussi palier à la fatigue des agents frontaliers

2. OBJECTIF DE LA RECHERCHE

La frontière, cette ligne qui marque la limite d'un territoire, devient de plus en plus source des tensions évoluant souvent vers des conflits armés, parfois vers des problèmes de migrations illégales, il est donc utile de mettre en place un système de surveillance plus efficace et plus sécurisant.

Le numérique est un système indispensable pour un travail plus autonome, plus précis et surtout plus sécurisant c'est pourquoi nous avons pensé à un système de détection radar pour la surveillance des frontières terrestre de la république démocratique du Congo notre pays;

La sécurisation et la surveillance des frontières du pays se heurte à des nombreux obstacles, l'ampleur de zone concernée par les patrouilles et les surveillances en est le principal, c'est dans cette optique que nous avons décidé de placer notre radar sur une sorte robot ambulant contrôlé par commande vocale tout en restant connectés à un écran de visualisation du système radar;

Grace à un capteur ultra sonique, nous pouvons connaitre la distance et L'emplacement de l'objet détecté par le fait que ce capteur envoie des impulsions sonores qui seront ensuite envoyées sous formes d'échos si elles heurtent un objet ainsi ces informations seront envoyées dans notre système de surveillance, Il sera facile à l'agent frontalier de visualiser les informations sur la détection des menaces ou des présences étrangères et de contrôler le déplacement du robot radar par commande vocale.

3. QUESTION DE LA RECHERCHE

Aux termes des problématiques et des objectifs cité(e)s précédemment pour définir l'importance et dans quel cadre notre projet s'oriente, la question de la recherche soulevée est : Comment est-ce que le système radar va amplifier la surveillance des frontières ?

4. METHODES ET TECHNIQUES DE TRAVAIL

Pour une bonne structuration et afin d'acheminer à un bon résultat de notre travail scientifique, nous avons utilisé des méthodes et des techniques de travail permettant de certifier les réponses et les conclusions mises à la disposition de la science.

a. Méthodes

Le mot « méthode » peut avoir plusieurs significations et n'a pu mettre d'accords les différents auteurs qui s'y sont penchés. Mais dans le cadre de ce travail nous allons outrepasser cette polémique tout en nous ralliant à Pierrette RONGERE qui la définit comme étant la procédure particulière appliquée à l'un ou à l'autre de stade de la recherche. C'est dans ce sens que nous avons retenu les méthodes historiques et structuro-fonctionnelles.

- Méthodes historique :

« C'est une méthode employée pour constituer l'histoire, elle sert à déterminer scientifiquement les faits historiques puis à les regrouper en un système scientifique » (*Die Probleme der Geschichtsphilosophie*, 1892.).

Ainsi par cette présente méthode, nous avons pu déterminer les faits historiques sur la localisation des frontières de la RDC et leur délimitation.

- Méthodes structuro-fonctionnelles :

Elle est complétive par rapport à la méthode précédente par le fait qu'elle nous a permis de comprendre le fonctionnement sur les différentes dispositions mises en place pour la protection des frontières par la police des frontières.

b. Techniques

Les techniques utilisées sont les suivantes :

- La technique d'interview :

Elle nous a permis de questionner les responsables du service frontaliers de la DGM, afin de mieux nous orienter et nous éclairer sur le fonctionnement des surveillances frontalières.

- La technique documentaire :

La documentation tant électronique que physique nous ont été d'une grande aide pour la bonne précision de notre enquête sur les frontières de RDC.

5. DELIMITATION DU SUJET

Ce travail concerne les frontières terrestres de la République Démocratique du Congo, leurs sécurisations, leurs délimitations ainsi que les dispositions mises en places pour leurs protections.

6. DIFFICULTÉS RENCONTRÉES

Tout au long de notre travail, nous avons fait face à quelques difficultés, beaucoup plus liées aux manques d'équipements plus sophistiqués suites à leur indisponibilité dans le pays et aux coûts élevés, ces équipements étant nécessaires pour une détection radar de très grande portée.

Malgré ces difficultés, nous avons pu réaliser un travail acceptable pour pallier aux problèmes cités, bien que se trouvant dans le domaine numérique, il est possible de faire plus car le numérique est vaste et évolutif.

7. SUBDIVISION DU TRAVAIL

Mise à part l'introduction et la conclusion, ce présent travail est subdivisé en trois chapitre, notamment :

INTRODUCTION

1 CHAPITRE PREMIER : REVUE DE LA LITTÉRATURE

2 CHAPITRE DEUXIEME : ETUDE ET MODELISATION DU SYSTÈME

3 CHAPITRE TROISIEME : REALISATION ET IMPLEMENTATION DU SYSTEME

CONCLUSION

CHAPITRE 1. REVUE DE LA LITTÉRATURE ET CONTEXTE DE TRAVAIL

En vue de mieux de comprendre les concepts de base liés à notre travail scientifique, nous aborderons les termes sur le développement du Radar et son fonctionnement en y mettant plus d'explications portant sur notre compréhension ainsi que les recherches faites sur les définitions de certains auteurs.

SECTION 1 : REVUE DE LA LITTÉRATURE

1.1.1 SYSTÈME D'INFORMATION

« Un système d'information se définit par l'information qu'il véhicule et par la manière dont il l'exprime sur ses différents supports. C'est un langage de communication d'une organisation, adapté à la nature plus ou moins répétitive de l'information que l'on veut y échanger »¹

Le système d'information (SI) c'est l'ensemble des ressources de l'entreprise qui permettent la gestion de l'information. Le SI est généralement associé aux technologies (matériel, logiciel et communication), aux processus qui les accompagnent, et aux hommes qui les supportent. D'abord simplement appelé informatique, cet ensemble a pris le nom de SI avec l'arrivée des nouvelles technologies qui ont élargi son domaine.

Le système d'information se penche sur trois points de vue :

- Le premier reprend la vision des solutions informatique (matériel, logiciel, communication), et évoque les ressources pour collecter, stocker, traiter et communiquer les informations, c'est le plus communément retenu ;
- Le second est orienté processus, et va au-delà des solutions informatiques pour impliquer la stratégie de l'entreprise et l'alignement des métiers, il prend en compte l'évolution de l'informatique hors de la DSI (Direction des systèmes d'information) ;
- Le troisième, rarement évoqué car récent, se détache de l'approche informatique traditionnelle pour entrer dans celle de la valeur de l'information au travers des modèles économiques emportés par la transformation digitale qui déconstruit les deux premières voix

Ainsi nous pouvons dire que l'importance du système d'information dans notre projet de système Radar consiste à la manipulation de l'information en vue de mieux la gérer et de bien l'utiliser

¹ Peaucelle (1997)

1.1.2 SYSTÈME RADAR

A. DEFINITION DU RADAR

Le terme radar est issu d'acronyme anglais de (Radio Détection And Ranging) désigne un système qui diffuse une onde électromagnétique dans une portion de l'espace, et reçoit les ondes réfléchies par les objets qui s'y trouvent, permettant de détecter leurs existences et déterminer certaines caractéristiques de ces objets tel que la position, l'altitude, la vitesse et parfois la forme de ces objets. Ces données permettent au Radar de renseigner l'utilisateur, mais aussi d'éliminer un grand nombre des objets indésirables pour ne conserver que les « cibles » intéressantes.²

B. PRINCIPES DU RADAR

Le principe de base des systèmes radars peut être résumé en quatre étapes : émission et propagation d'une onde électromagnétique, rétrodiffusion de cette onde par une cible sous forme d'écho de l'onde, réception et analyse du signal reçu par le radar.

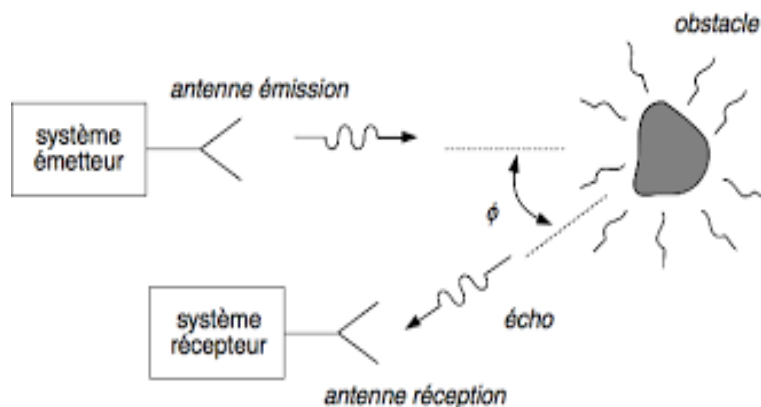


Figure I.1 : Principe d'un système RADAR

• Mesure de distance

L'onde émise par le radar parcourt la distance radar/cible R_0 à la célérité de la lumière c . Elle est rétrodiffusée par la cible dans tout l'espace. L'onde rétrodiffusée parcourt une nouvelle fois la distance R_0 et une partie de l'onde est recueillie par le radar. La détection de l'écho reçu et la mesure du temps de propagation de l'onde nous informe sur la présence et la distance de la cible. L'onde reçue par le radar est une version atténuée et retardée de l'onde émise d'un retard³

• Mesure de direction

² Michel-Henri CARPENTIER, « RADAR », Encyclopaedia Universalis, URL : <http://www.iniversalis.fr/encyclopedie/radar/>

³ Philippe Goy, « Détection d'obstacles et de cibles de collision par un radar FMCW aéroporté », Thèse de doctorat de l'université de Toulouse, France. Mardi 18 décembre 2012.

La mesure de la direction est déterminée grâce à l'angle entre la direction du nord et celle de la cible (Azimut), cet angle est obtenu par la directivité de l'antenne. En mesurant cette direction dans laquelle l'antenne est pointée à l'instant où elle reçoit l'écho, on peut déterminer l'azimut et aussi le site de la cible (l'altitude). La précision de la mesure de ces angles dépend de la directivité de l'antenne. Pour une fréquence émise donnée (Ou une longueur d'onde définie), la directivité d'une antenne est en fonction de ses dimensions propres.⁴

• Mesure de la vitesse radiale

La vitesse radiale de la cible peut être mesurée de deux façons : Le décalage dû à l'effet Doppler ou la mesure successive de la distance. Même si cela demande un temps de mesure relativement long pour avoir la précision adéquate, c'est la deuxième méthode qui est préférée.

C. LA CIBLE

Une cible est tout objet qui interfère avec l'onde émise, et réfléchit une partie de l'énergie vers le Radar, elle se comporte comme une antenne de forme complexe. L'énergie émise dans la direction du Radar est fortement fluctuante et dépend énormément de l'orientation de la cible par rapport au Radar.

D. CLASSIFICATION DU RADAR

La classification basée sur la fonction principale du Radar est illustrée dans la figure suivante:

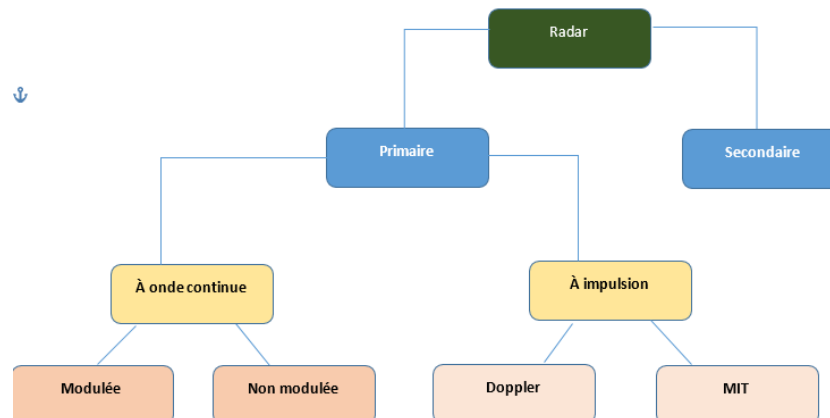


Figure I.2 : Classification des Radars

• Le Radar primaire

Un radar primaire émet des signaux hyperfréquences qui sont réfléchis par les cibles. L'énergie réfléchi ou les échos sont en outre traités pour extraire des informations utiles. Il comprend :

⁴ DJEDOU Nacéra & BOUKERN Dounya, « Étude et réalisation d'un RADAR de détection », mémoire de master 2 en télécommunications, Université Aboubakr Belkaid, Tlemcen. Algérie. 19/06/ 2017.

- **Le Radar à impulsions** : Ce type de Radar transmet des impulsions à hautes fréquence avec une puissance très élevées vers la cible. Ensuite, il attend l'écho du signal transmis pendant un certain temps avant qu'il ne transmet une nouvelle impulsion. Le Radar à impulsions est généralement utilisé lorsque cela est nécessaire pour détecter des cibles au sein d'un certain volume de l'espace et de déterminer la distance et le relèvement et dans certains cas, la vitesse de chaque cible. Le système de radar pulsé nécessite généralement l'émission de grandes puissances et peut-être très complexe et coûteux.
- **Le Radar à ondes continues** : Les Radars CW (Continuous Wave), le signal est émis de façon continu. Les Radars utilisant ce type d'émission sont destinés à mesurer la vitesse en exploitant l'effet doppler.

Radar à ondes continues modulées : ces Radars émettent sans interruption un signal hyperfréquence, l'écho est donc reçu et traité continuellement.

Radar à ondes continues non modulées : Les Radars CW non modulés émet une Seul fréquence à amplitude constante et qui transmet une puissance non modulée, ne peut qu'en utilisant l'effet Doppler.

- **Le Radar secondaire**

Ce type de Radar utilise des impulsions à hautes fréquence, qui se réfléchies sur toute cible, créant ainsi une onde de retour susceptible d'être décelée par un récepteur adapté à ce signal. Chaque impulsion de durée très brève de l'ordre de quelque microseconde se propage dans l'atmosphère à la vitesse de la lumière ($C = 3.10^8$ m/s). Une partie de ce signal est réfléchié par la cible, nous pouvons dire que la cible est illuminée et rayonne une partie de l'énergie émise sous la forme d'une onde de faible amplitude et de caractéristiques temporelles identiques à celle du signal émis. La mesure de la distance se déduit à partir du retard entre l'émission de l'impulsion électromagnétique et sa réception. Les paramètres les plus importants pour déterminer la distance maximale, c'est-à-dire, la plus grande distance mesurable et la résolution du Radar, sont la durée des impulsions et la fréquence de répétition.

Pour des raisons de compatibilité, le type de radar que nous utiliserons dans notre travail est le **Radar à impulsions**

E. COMPOSITION D'UN RADAR À IMPULSIONS

Un Radar à impulsions peut être décrit à l'aide du schéma de principe représenté sur la figure ci-dessous, il se compose d'un émetteur, récepteur, antenne et un duplexeur.

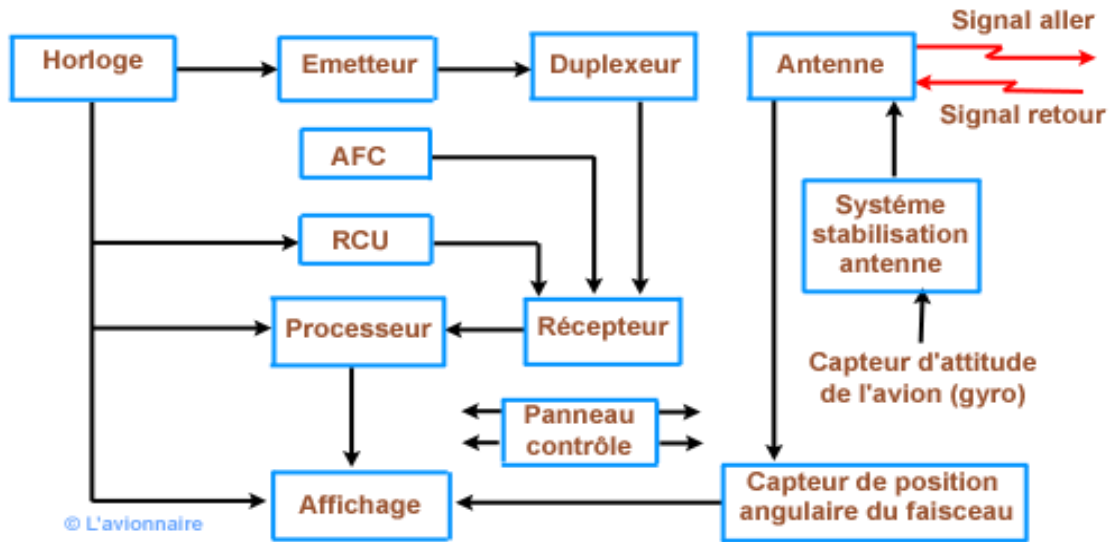


Figure I.3 : Composition du Radar à impulsions

L'émetteur : L'émetteur peut être un oscillateur, tel qu'un magnétron, qui est "pulsé" (Activé et désactivé) par le modulateur pour générer un train répétitif d'impulsions. La forme d'onde générée par l'émetteur se déplace via une ligne de transmission (Guide d'onde) vers l'antenne, où elle est rayonnée dans l'espace.

Une forme commune d'antenne Radar est un réflecteur de forme parabolique, alimenté à partir d'une source ponctuelle à son foyer. Le réflecteur parabolique concentre l'énergie en un faisceau étroit, tout comme un projecteur ou un phare d'automobile.

Le Duplexeur : Le duplexeur est un commutateur électronique qui dirige l'onde vers l'antenne lors de l'émission ou le signal de retour depuis l'antenne vers le récepteur lors de la réception.

Le Récepteur : Le récepteur est généralement du type superhétérodyne. La première étape peut être un amplificateur RF à faible bruit, tel qu'un transistor à faible bruit. Cependant, il n'est pas toujours souhaitable d'utiliser un premier étage à faible bruit dans le Radar. Surtout dans les Radars militaires qui doivent fonctionner dans un environnement bruyant. Le mélangeur et l'oscillateur local (LO) convertissent le signal RF en une fréquence intermédiaire (IF). L'amplificateur FI doit être conçu comme un filtre adapté, c'est-à-dire sa fonction de réponse en fréquence $H(f)$ devrait maximiser le rapport de puissance de signal de crête à moyenne de bruit à la sortie. Cela se produit lorsque l'amplitude de la réponse fréquentielle $|H(f)|$ est égale à l'amplitude de spectre d'écho $|S(f)|$, et la partie négative de phase du spectre d'écho est la même que celle du spectre de filtre adapté. Après avoir maximisé le rapport signal / bruit dans l'amplificateur IF, la modulation d'impulsion est extraite par le second détecteur et amplifiée par l'amplificateur vidéo à un niveau où elle peut être affichée correctement, habituellement sur un tube cathodique (CRT). Des signaux de synchronisation sont également fournis au duplexeur, l'information d'angle est obtenue à partir de la direction de pointage de l'antenne.

1.1.3 SYSTÈME NUMÉRIQUE DE CONTRÔLE-COMMANDE

A. DEFINITION

Un **système numérique de contrôle-commande** (SNCC, ou DCS pour *distributed control system* en anglais) est un système de contrôle d'un procédé industriel doté d'une interface homme-machine pour la supervision et d'un réseau de communication numérique⁵

B. ARCHITECTURE DU SYSTÈME

Les équipements de commande d'un SNCC sont distribués ou géo-répartis. À la différence des systèmes de contrôle centralisés à base d'automates programmables industriels, qui comportent un seul contrôleur central qui gère toutes les fonctions de contrôle-commande du système, les SNCC sont constitués de plusieurs contrôleurs modulaires qui commandent les sous-systèmes ou unités de l'installation globale.

L'attribut clé d'un SNCC est sa fiabilité en raison de la distribution du traitement de contrôle autour des nœuds dans le système. Cela atténue une défaillance d'un seul processeur. Si un processeur tombe en panne, cela affectera seulement une section du processus de l'usine, par opposition à une défaillance d'un ordinateur central qui affecterait l'ensemble du processus. Cette répartition de la puissance de calcul locale sur les baies de connexion de champ d'entrées-sorties (E / S) sur le terrain garantit également des temps de traitement rapides du contrôleur en supprimant les éventuels retards de traitement réseau et central.

Le diagramme ci-joint est un modèle général qui montre les niveaux de fabrication fonctionnels en utilisant un contrôle informatisé :

- Le niveau 0 contient les appareils de terrain tels que les capteurs et actionneurs
- Le niveau 1 contient les modules d'entrées-sorties (E / S) industrialisés et leurs processeurs électroniques distribués associés.
- Le niveau 2 contient les ordinateurs de supervision, qui collectent des informations à partir des nœuds du processeur sur le système et fournissent les écrans de contrôle de l'opérateur.
- Le niveau 3 est le niveau de contrôle de la production, qui ne contrôle pas directement le processus, mais qui concerne le suivi des objectifs de production et de surveillance.
- Le niveau 4 est le niveau d'ordonnancement de la production.

Les niveaux 1 et 2 sont les niveaux fonctionnels d'un SNCC traditionnel, dans lequel tous les équipements font partie d'un système intégré provenant d'un seul fabricant.

Les niveaux 3 et 4 ne sont pas strictement des processus de contrôle dans le sens traditionnel, mais où le contrôle de la production et l'ordonnancement ont lieu.

C. FONCTIONNALITÉS DU SYSTÈME ET SON APPORT DANS LE PROJET

⁵ Jacques le Gallais « Systèmes numériques de contrôle-commande (SNCC) », [Techniques de l'ingénieur. Informatique industrielle](#), vol. S2, n° R7505, 1997, R7505.1-R7505.15 (ISSN 1632-3831).

Un SNCC est composé à la base d'un ensemble d'équipements à base d'un microcontrôleur assurant chacun une tâche spécifique :

- **Acquisition des données** : raccordé généralement et essentiellement avec les équipements du terrain (capteur, actionneur), ils permettent de filtrer et conditionner le signal pour assurer la bonne communication entre le régulateur et ces équipements de terrain. Les données seront donc acquises à travers le **capteur ultra-son** qui permet de calculer la distance
- **régulation et traitement de donnée** : ce sont les équipements qui assurent la fonction de commande de régulation et d'asservissement en utilisant des boucles de régulation PID ou bien à l'aide d'une logique combinatoire. Le traitement des données est effectué par le **microcontrôleur**
- **Un système de communication** : en utilisant des protocoles de communications et des topologies spécifiques pour assurer la communication entre les différents équipements du système. **Le module Bluetooth HC-06** s'occupe de la transmission des données à distance, il va servir à connecter le microcontrôleur avec l'écran de visualisation
- **Supervision** : c'est un **écran de surveillance (ou de visualisation)** qui veille sur le bon fonctionnement de l'ensemble d'équipement du système ainsi que le bon fonctionnement du processus. Cet écran pourra visualiser les informations (la distance et l'emplacement des cibles) sur la détection radar
- **Archivage** : cette fonction assure l'archivage et l'enregistrement des différentes données du système dans la base de données du serveur
- **Commande** : cette fonction permet de commander le robot à distance en assurant ces déplacements, ce sera au travers d'une **application mobile**

D. LE CONDITIONNEMENT DES SIGNAUX

Conversion de signaux

La fonction principale d'un conditionneur de signaux est de capter le signal et de le convertir en un signal électrique plus puissant. La conversion du signal est très utile pour les applications industrielles qui utilisent une large gamme de capteurs pour effectuer des mesures. Lorsque plusieurs capteurs différents sont utilisés, il est nécessaire de convertir les signaux générés pour qu'ils soient utilisables par

les instruments auxquels ils sont connectés. Tous les signaux issus des capteurs peuvent être convertis en n'importe quel signal de processus standard.

Linéarisation

Certains conditionneurs de signaux peuvent effectuer une linéarisation lorsque les signaux produits par un capteur n'ont pas de relation linéaire avec la mesure physique. C'est le processus d'interprétation du signal par le logiciel et c'est assez commun pour les signaux de thermocouple. Cette méthode est utilisée pour atteindre une plus grande précision car chaque capteur n'est pas complètement linéaire. Les paramètres de linéarisation sont évalués lors de l'étalonnage du capteur et mentionnés dans le protocole d'étalonnage du capteur.

Amplification

L'amplification du signal et le processus d'accroissement du signal pour le traitement ou la numérisation. L'amplification du signal peut être effectuée de deux manières différentes : en augmentant la résolution du signal d'entrée ou en augmentant le rapport signal / bruit.

Le conditionnement de signal utilise une gamme de différents amplificateurs à des fins variées, y compris des amplificateurs d'instrumentation qui sont optimisés pour une utilisation avec des signaux CC et se caractérisent par une impédance d'entrée élevée, un taux de réjection de mode commun élevé et un gain élevé. Un autre exemple d'un conditionneur de signal utilisé pour l'amplification serait un amplificateur d'isolement, conçu pour isoler les niveaux de courant continu élevés du dispositif tout en faisant passer un petit signal alternatif ou différentiel.

Filtration

Une autre fonction importante d'un conditionneur de signaux est le filtrage. Le spectre de fréquence du signal est filtré pour inclure uniquement les données valides et bloquer tout bruit. Les filtres peuvent être fabriqués à partir de composants passifs et actifs ou d'un algorithme numérique. Un filtre passif utilise uniquement des condensateurs, des résistances et des inductances avec un gain maximum de un. Un filtre actif utilise des composants passifs en plus des composants actifs tels que les amplificateurs opérationnels et les transistors. Les conditionneurs de signaux de pointe utilisent des filtres numériques, ils sont faciles à régler et aucun matériel n'est requis. Un filtre numérique est un filtre mathématique utilisé pour manipuler un signal, tel que le blocage ou le dépassement d'une plage de fréquence particulière. Ils utilisent des composants logiques tels que les ASIC, les FPGA ou sous la forme d'un programme séquentiel avec un processeur de signal.

Évaluation et fonctions intelligentes

Les conditionneurs de signaux modernes ont des fonctions supplémentaires pour l'évaluation du signal et le prétraitement des données de mesure. Ces fonctions "intelligentes" permettent de surveiller et évaluer les alertes et les alarmes directement via une sortie de commutation électrique rapidement. D'autres fonctions "intelligentes" comme la voie interne calculée peuvent gérer des fonctions mathématiques, telles que l'ajout de signaux de capteur ou encore un contrôleur PID. Ces fonctions permettent d'obtenir un système plus réactif et plus rapide, elles réduisent également la charge de contrôle de la machine.

Interfaces

Les convertisseurs de signaux doivent transmettre les signaux des capteurs via des interfaces standard et des protocoles au contrôle de la machine. Ces interfaces peuvent être analogiques ou numériques. Les interfaces analogiques communes sont des signaux de tension (+/- 10V) ou de courant (+/- 20mA) faciles à manipuler, mais chaque signal nécessite un câblage séparé. Les interfaces numériques modernes sont conçues comme des interfaces bus Ethernet (Profinet, Ethercat, Ethernet / IP) et permettent de connecter plusieurs composants avec un seul câble. Le câblage est ainsi réduit. De plus, les interfaces numériques permettent également de transmettre des informations

supplémentaires, telles que les informations de diagnostic des composants, ce qui est très important pour réduire les temps d'arrêt et accélérer la maintenance.

1.1.4 LE RESEAU BLUETOOTH

Bluetooth est une spécification de l'industrie des télécommunications. Elle utilise une technique radio courte distance destinée à simplifier les connexions entre les appareils électroniques. Elle a été conçue dans le but de remplacer les câbles entre les ordinateurs et les imprimantes, les scanners, les claviers, les souris, les manettes de jeu vidéo, les téléphones portables, les PDA, les systèmes et kits mains libres, les autoradios, les appareils photo numériques, les lecteurs de code-barres, les bornes publicitaires interactives. Les premiers appareils utilisant la version 3.0 de cette technologie sont apparus début 2010

A. ORIGINE DU NOM

Le nom *Bluetooth* est directement inspiré du roi danois Harald I^{er} surnommé Harald Blåtand (« homme à la dent bleue »), connu pour avoir réussi à unifier les États du Danemark, de Norvège et de Suède. Le logo de *Bluetooth*, est d'ailleurs inspiré des initiales en alphabet runique de *Harald Blåtand*

B. SPECIFICATIONS

Le SIG travaille sur la spécification de la [norme](#), qui a évolué des versions 1.0, 1.1, 1.2, 2.0, 2.0 + EDR (*Enhanced Data Rate*), 2.1 + EDR et 3.0. Le site [Internet](#) bluetooth.org regroupe les travaux du SIG, et toutes les versions de la norme *Bluetooth*.

C. PILE DE PROTOCOLES

Afin d'assurer une compatibilité entre tous les périphériques *Bluetooth*, la majeure partie de la pile de protocoles est définie dans la spécification.

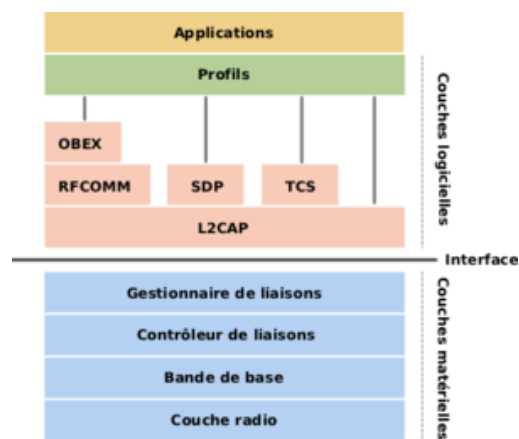


Figure I.4 : Couche de spécifications Bluetooth

D. NORMES BLUETOOTH

Le standard *Bluetooth* se décompose en différentes normes :

- IEEE 802.15.1 définit le standard *Bluetooth* 1.x permettant d'obtenir un débit de 1 Mbit/s ;
- IEEE 802.15.2 propose des recommandations pour l'utilisation de la bande de fréquence 2,4 GHz (fréquence utilisée également par le Wi-Fi). Ce standard n'est toutefois pas encore validé ;
- IEEE 802.15.3 est un standard en cours de développement visant à proposer du haut débit (20 Mbit/s) avec la technique *Bluetooth* ;
- IEEE 802.15.4 est un standard en cours de développement pour des applications sans fils à bas débit et à faibles coûts. Il est actuellement utilisé par Zigbee pour ses couches basses.

Les éléments fondamentaux d'un produit *Bluetooth* sont définis dans les deux premières couches protocolaires, la couche *radio* et la couche *bande de base*. Ces couches prennent en charge les tâches matérielles comme le contrôle du saut de fréquence et la synchronisation des horloges.

E. COUCHES RADIO

La couche radio (la couche la plus basse) est gérée au niveau *matériel*. C'est elle qui s'occupe de l'émission et de la réception des ondes radio.

Elle définit les caractéristiques telles que la bande de fréquence et l'arrangement des canaux, les caractéristiques du transmetteur, de la modulation, du récepteur, etc.

Le système *Bluetooth* opère dans les bandes de fréquences ISM* (*Industrial, Scientific and Medical*) 2,4 GHz dont l'exploitation ne nécessite pas de licence. Cette bande de fréquences est comprise entre 2 400 et 2 483,5 MHz. Un *transceiver* à saut de fréquences est utilisé pour limiter les interférences et l'atténuation.

Deux modulations sont définies : une modulation obligatoire utilise une modulation de fréquence binaire pour minimiser la complexité de l'émetteur ; une modulation optionnelle utilise une modulation de phase (PSK à 4 et 8 symboles).

La rapidité de modulation est de 1 Mbaud pour toutes les modulations. La transmission duplex utilise une division temporelle. Les 79 canaux RF sont numérotés de 0 à 78 et séparés par 1 MHz en commençant par 2 402 MHz. Le codage de l'information se fait par sauts de fréquence. La période est de 625 µs, ce qui permet 1 600 sauts par seconde.

Il existe trois classes de modules radio *Bluetooth* sur le marché ayant des puissances différentes et donc des portées différentes :

Classe	Puissance	Portée
1	100 mW (20 dBm)	100 mètres
2	2,5 mW (4 dBm)	10 à 20 mètres
3	1 mW (0 dBm)	Quelques mètres

Tableau I.1 : Classe de modules *Bluetooth*

F. BANDE PASSANTE

La bande de base (ou *base band* en anglais) est également gérée au niveau matériel.

C'est au niveau de la bande de base que sont définies les adresses matérielles des périphériques (équivalentes à l'adresse MAC d'une carte réseau). Cette adresse est nommée *BD_ADDR* (*Bluetooth Device Address*) et est codée sur 48 bits. Ces adresses sont gérées par la *IEEE Registration Authority*.

C'est également la bande de base qui gère les différents types de communication entre les appareils. Les connexions établies entre deux appareils *Bluetooth* peuvent être synchrones ou asynchrones.

La bande de base peut donc gérer deux types de paquets :

- les paquets SCO (*Synchronous Connection-Oriented*) ;
- les paquets ACL (*Asynchronous Connection-Less*).

Picoréseau

Un picoréseau (on emploie également l'anglicisme *piconet*) est un mini-réseau qui se crée de manière instantanée et automatique quand plusieurs périphériques *Bluetooth* sont dans un même rayon. Un picoréseau est organisé selon une topologie en étoile : il y a un « maître » et plusieurs « esclaves ».

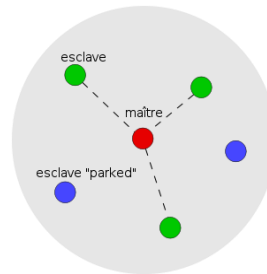


Figure I.5 : représentation du picoreseau

Un périphérique « maître » peut administrer jusqu'à :

- 7 esclaves « actifs » ;
- 255 esclaves en mode « *parked* ».

La communication est directe entre le « maître » et un « esclave ». Les « esclaves » ne peuvent pas communiquer entre eux.

Tous les « esclaves » du picoréseau sont synchronisés sur l'horloge du « maître ». C'est le « maître » qui détermine la fréquence de saut pour tout le picoréseau.

Inter-réseau Bluetooth (scatternet)

Les périphériques « esclaves » peuvent avoir plusieurs « maîtres » : les différents *piconets* peuvent donc être reliés entre eux. Le réseau ainsi formé est appelé un *scatternet* (littéralement *réseau dispersé*)

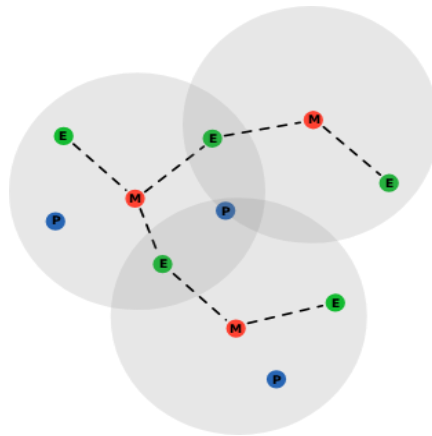


Figure I.6 : représentation du Bluetooth scatternet

1.1.5. APPLICATION MOBILE

Une application mobile est un logiciel applicatif développé pour un appareil électronique mobile, tel qu'un téléphone portable ou un smartphone, fonctionnant avec le système d'exploitation Windows Phone, Chrome OS ou Android.

a. Application native

Une application native est une application mobile qui est développée spécifiquement pour un des systèmes d'exploitation utilisé par les smartphones et tablettes (IOS, Android, etc.). Les applications natives pour iPhones sont par exemple développées avec les langages de développement Swift et Objective-C. Le développement natif permet d'utiliser les fonctionnalités liées au système d'exploitation visé (GPS, accéléromètre, appareil photo, etc.) et proposent des applications plus performantes que celles développées en HTML et CSS

b. Application hybride

Les applications hybrides sont des applications mobiles dites cross-platform, elles permettent de concevoir des applications Android et IOS à la fois. Ces applications ne dépendent pas du Système d'exploitation. Les applications hybrides sont les plus développées en utilisant les langages connus du web ou en passant par des Framework tels que React Native, Ionic ou Cordova.

SECTION 2 : CONTEXTE DE TRAVAIL

2.1 DIRECTION GENERALE DE MIGRATION / RDC

A. historique

La Direction Générale de Migration (DGM) a vu le jour le 27 mai 1997 par l'ordonnance loi constitutionnel n°002/2003 qui stipule la création et organisation de la Direction General de Migration, Depuis son emplacement, la DGM/BC a fonctionné avec quatre (4) division dont celle de la police des frontières et des étrangers en sigle "DPEF".

En effet, avec la nouvelle structure organique contenue dans la note n° 06/DP/00/DGM/BC/044/2003 du 24 janvier 2003, la DPEF est éclatée en deux nouvelles divisions à savoir :

- La Division de Police des Frontières (DPF) qui est notre domaine de travail précis ;
- La Division de Police des Étrangers (DPE)

B. Décret : de la création et de la mission

Il est créé un service doté de l'autonomie administrative et financière dénommé Direction Générale de Migration, en sigle « DGM ». Le président de la république détermine, par décret le ministre sous l'autorité duquel la Direction Générale de Migration exerce ses activités sous réserve d'autres missions qui lui sont conférées;

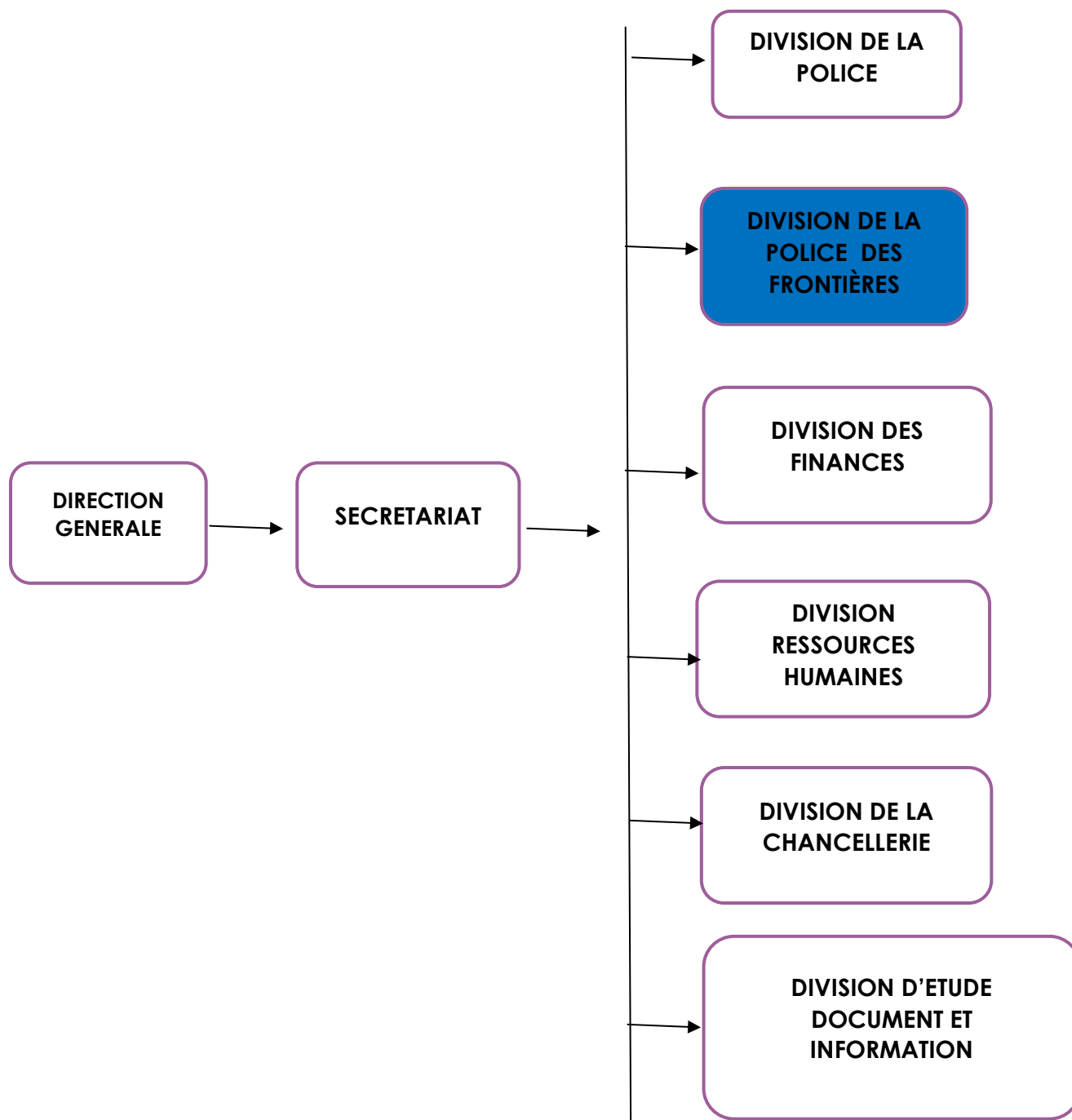
La Direction Générale de Migration est chargée questions ayant trait à :

- l'exécution de la police du gouvernement en matière d'immigration
- l'exécution sur le sol congolais, des loi et règlements sur l'immigration et l'émigration
- La police des frontières
- la délivrance du passeport ordinaire aux nationaux
- la collaboration dans la recherche des criminels et malfaiteurs signalées par l'organisation internationale de la police criminelle (INTERPOL)

La DGM exerce ses activités sur l'ensemble du territoire national et dans toutes missions diplomatiques de la république démocratique du Congo

Notre système sera implémenté dans la La Division de Police des Frontières (DPF) pour la surveillance des frontières terrestres

C. Organigramme



CHAPITRE 2. MODELISATION DU SYSTÈME

2.1. NARRATION

Notre système de détection Radar sera représenté en deux sous-systèmes, le premier sous-système s'occupe de la commande à distance fait sur une application mobile et le second s'occupe de la visualisation sur un logiciel, Nous aurons donc un robot radar qui est commandé à distance et va permettre à son tour d'envoyer les informations de détection qui seront affichées sur un écran afin de faciliter le contrôle (la visualisation).

L'objectif du système est de surveiller et de détecter les activités sur les frontières, sur ce fait, le déroulement sera présenté comme suite :

➤ **Sous-système 1 : Commande à distance**

Étape 1 : Connexion au robot

L'agent frontalier se connecte au robot radar à partir de l'application mobile, cette connexion est possible grâce au signal émis du module Bluetooth se trouvant sur le robot, Dans la liste des Bluetooth qui apparaissent dans l'application, l'agent choisit le nom du Bluetooth **HC-O6**, ce qui permet l'appariage entre l'application et le robot

Gestion d'exceptions (les erreurs):

- Si l'agent se connecte à un autre émetteur Bluetooth, l'application lui renvoie un message pour lui dire de se connecter au bon émetteur Bluetooth.

Étape 2 : Commander le robot

Après avoir connecté l'application et le robot, l'agent frontalier peut maintenant introduire les commandes qui se feront de manière vocale, il doit appuyer sur le bouton ayant l'icône d'un microphone, ensuite il peut prononcer les mots suivants :

- **Avancer** : pour le déplacement du robot vers l'avant
- **Arrière** : pour le déplacement du robot vers l'arrière
- **À Gauche** : pour le déplacement du robot vers la gauche
- **À Droite** : pour le déplacement du robot vers la droite

C'est donc avec ces commandes vocales que l'on pourra déplacer le robot.

Gestion d'exceptions (les erreurs):

- L'agent ne pourra prononcer que les mots cités ci-hauts pour faire déplacer le robot sans quoi l'application lui enverra un message lui disant de ne pas prononcer d'autres mots

- Avant que l'agent puisse prononcer un autre mot, il doit appuyer sur le bouton pour activer la commande vocale

➤ **Sous-système 2 : Visualisation des informations**

Étape 1 : Détection de la donnée (l'information)

Robot Radar doté d'un module ultra-son, lors de la détection d'un objet, émet des signaux ultrasoniques, ces signaux rencontrent un obstacle et sont retournés sous formes d'échos, ainsi on aura deux temps, le temps d'émission et le celui de réponse, le tout fonctionnant à la vitesse du son, ce qui nous permet de localiser l'objet en trouvant la distance entre ce dernier et le robot.

Nous avons notre information, la **distance**

Étape 2 : Envoie de l'information vers l'écran de contrôle

Grâce au module Bluetooth nous pouvons envoyer nos informations vers l'écran de visualisation

Étape 3 : Affichage de l'information sur l'écran de contrôle

Sur l'écran de visualisation, nous avons une interface graphique dans laquelle nous verrons la représentation dynamique du radar appelée **Affichage radar**, nous avons opté pour la **vue panoramique à angle d'élévation constant** plus connu sous son nom anglophone de **Plan Position Indicator ou PPI**, il s'agit d'une coupe horizontale à une altitude donnée des données recueillies par le balayage de plusieurs PPI à des angles d'élévation différents. Ce balayage est parabolique et se fait de 0 à 180 degrés suivant la rotation d'un servomoteur sur lequel est mis notre module ultra-son (rappelons que ce module envoie les informations sur cet écran). Ainsi si un objet est détecté le balayage change de couleur et se met en rouge, ensuite un message nous donne sa localisation selon un angle donné.

2.2. MODELISATION AVEC UML

La **modélisation UML** permet de vulgariser les aspects liés à la conception et à l'architecture, propres au logiciel, au client. Aussi, elle apporte une compréhension rapide du programme à d'autres développeurs externes en cas de reprise du logiciel et facilite sa maintenance, à d'autres utilisateurs professionnels et tous ceux qui souhaitent comprendre le fonctionnement du système.

2.2.1. DIAGRAMMES DE CAS D'UTILISATIONS

En UML les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Elles décrivent les fonctions générales et les portées d'un système

Dans ce type de diagramme, nous distinguons deux concepts UML fondamentaux pour la spécification des exigences qui sont : Les cas d'utilisations et les acteurs.

- Un cas d'utilisation est un ensemble d'actions réalisées par le système et c'est un acteur qui les déclenche ;
- Un acteur est un rôle joué par une personne ou un processus qui interagit avec le système.

Selon la narration faite précédemment, le tableau d'acteurs et de cas d'utilisations se présente comme suit :

Acteurs	Cas d'utilisations
Agents frontaliers	<ul style="list-style-type: none"> Se connecter au robot via Bluetooth au travers d'une application mobile Commander le robot
Robot radar	<ul style="list-style-type: none"> Se déplacer Détecter un objet Renvoyer l'information concernant l'objet
Écran de surveillance	Afficher les informations de détections

Tableau II.1 : tableau des acteurs et cas d'utilisations

Le diagramme de cas d'utilisation global sera alors représenté comme suit :

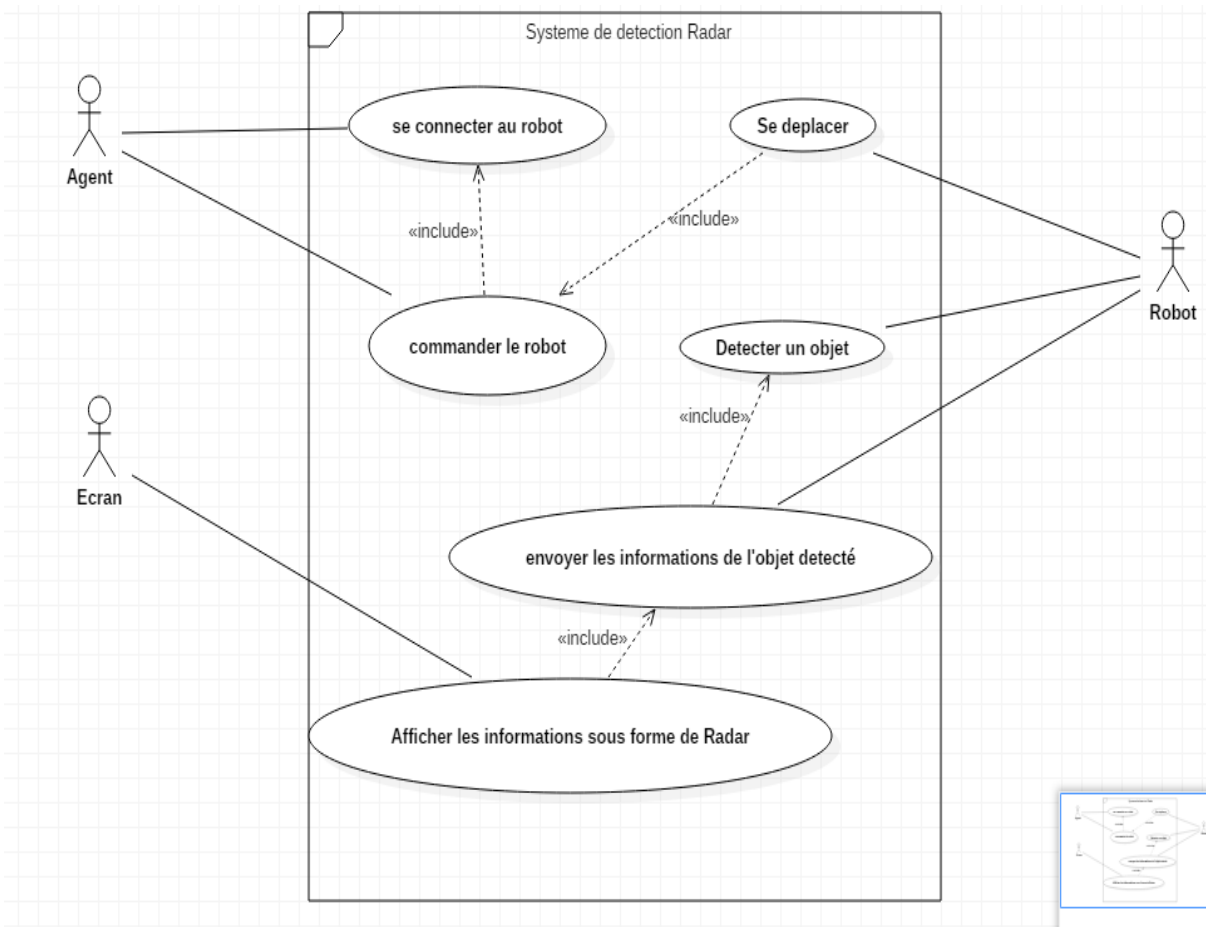


Figure II.1 diagramme de cas d'utilisation

Description textuelle des cas d'utilisation

Cas d'utilisation : Se connecter au robot
Objectif : permet de connecter l'application mobile au robot par Bluetooth
Précondition : activer la connexion Bluetooth du téléphone
Acteur concerné : L'agent
Scenario nominal L'agent active le Bluetooth du téléphone, appuie sur le bouton rechercher dans l'application mobile L'application affiche la liste des Bluetooth émis L'agent choisi le Bluetooth HC-O5
Scenario alternatif L'application renvoie un message d'erreur de connexion

Tableau II.2 : description textuelle du cas : Se connecter au robot

Cas d'utilisation : Commander le robot
Objectif : Commander le robot à distance
Précondition : activer la connexion internet du téléphone
Acteur concerné : L'agent
Scenario nominal L'agent appuie sur le bouton et fait une note vocale pour déplacer le robot

Tableau II.3 : description textuelle du cas : Commander robot

Cas d'utilisation : Se déplacer
Objectif : le robot doit se déplacer pour améliorer la détection
Précondition : recevoir l'ordre(la commande) de l'agent
Acteur concerné : Le robot
Scenario nominal Après avoir reçu l'ordre de faire un déplacement, le robot l'exécute

Tableau II.4 : description textuelle du cas : Se déplacer

Cas d'utilisation : Détecter un objet
--

Objectif : détection des objets(obstacle)
Précondition : Aucune
Acteur concerné : Le robot
Scenario nominal Le robot émet des signaux ultrasoniques s'il y a obstacle, ces signaux reviennent sous forme d'échos, ce qui permet de savoir si un objet est détecté Le système calcule la distance de cet objet détecté.

Tableau II.5 : description textuelle du cas : Détecter un objet

Cas d'utilisation : Envoyer l'information concernant l'objet détecté
Objectif : permettre l'affiche des informations de détections sur un écran
Précondition : bonne connexion Bluetooth
Acteur concerné : Le robot
Scenario nominal Le robot envoie les informations de détections vers un écran de visualisation par le module Bluetooth qui est bidirectionnel

Tableau II.6 : description textuelle du cas : Envoi des informations

Cas d'utilisation : Afficher les informations de détections radar
Objectif : permettre de voir sur un affichage-radar, les informations de détections
Précondition : Activer la communication série
Acteur concerné : L'Écran
Scenario nominal L'écran nous montre un affichage-radar faisant un balayage elliptique de 0 à 180 degrés et vice-versa, si un objet est détecté une concentration d'énergie se forme selon un angle donné

Tableau II.7 : description textuelle du cas : Affichage des informations

2.2.2. DIAGRAMMES DE CLASSES

L'intérêt du diagramme de classe est de modéliser les entités du système d'information. Le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées, c'est-à-dire qu'elles ont regroupées dans des classes. Le diagramme met en évidence d'éventuelles relations entre ces classes.

Les diagrammes de classes seront faits pour les deux sous-systèmes :

1. Diagramme de classes pour le sous-système 1 : Commande à distance

Pour rappel, l'objectif de ce sous-système est de commander à distance le déplacement du robot radar, pour se faire, nous avons les classes suivantes :

- ConnexionBluetooth
- Contrôleur
- Initialiseur
- Préférences

Relation Entre les classes

- La classe Contrôleur établis une relation de **contrôle** sur la classe ConnexionBluetooth, cela veut dire qu'une connexion Bluetooth est contrôlée par une et une seule commande venant de la classe Contrôleur;
- La classe préférences **sauvegarde** les états d'affichage que la classe initialiseur a à offrir, parmi les états d'affichages de la classe initialiseur, nous avons : mode portrait et mode paysage;
- La classe initialiseur affiche l'état de la classe contrôleur soit en mode portrait soit en mode paysage.

Ci-dessous, les différentes classes mises en relation ,leurs attributs et leurs opérations concernant le sous-système 1 :

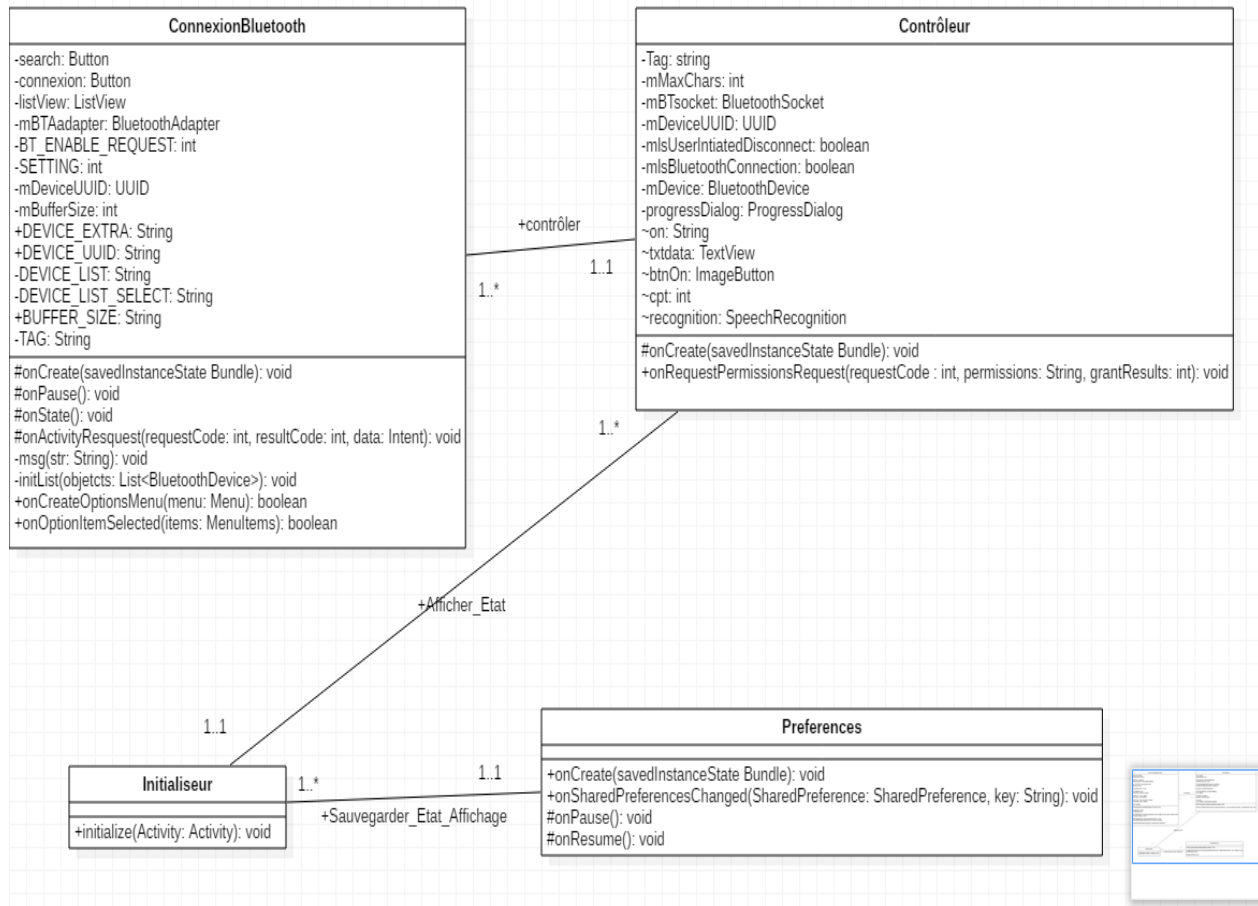


Figure II.2 : diagramme de classe du sous-système 1

2. Diagrammes de classes pour le sous-système2 : Visualisation des informations

Dans ce système nous avons les classes suivantes :

- DetecterRadar
- VoirRadar

Relation entre les classes

- La classe DetecterRadar relaye un ou plusieurs informations de détections vers la classe VoirRadar

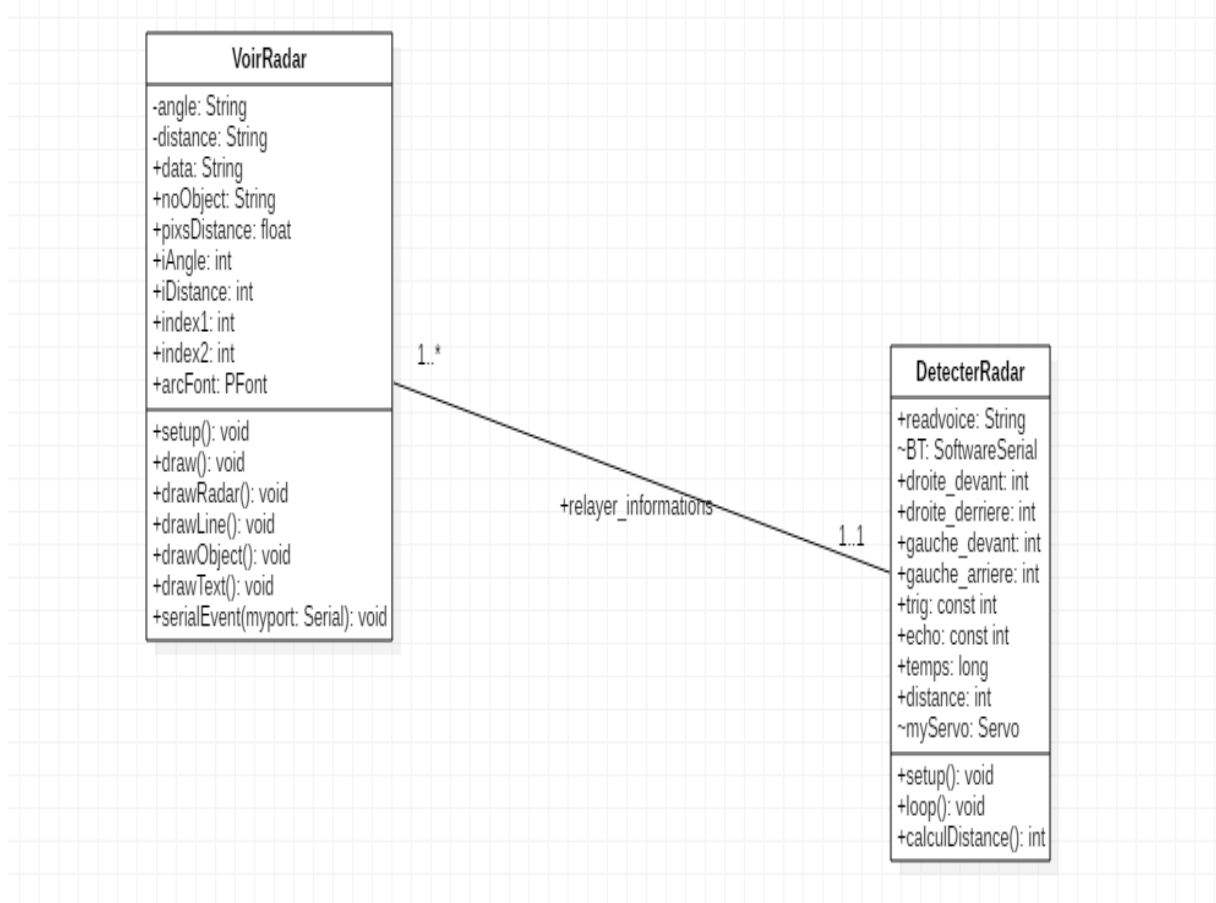


Figure II.3 : diagramme de classe du sous-système 2

Classe	Attributs	Operations
ConnexionBluetooth	Search Connexion listView mBTAadapter BT_ENABLE_REQUEST SETTINGS mDeviceUUID mBufferSize DEVICE_EXTRA DEVICE_UUID	onCreate() onPause() onResume() onActivityResult() msg() initlist() onCreateOptionsMenu() onOptionsItemSelected()

	DEVICE_LIST DEVICE_LIST_SELECT BUFFER_SIZE TAG	
Contrôleur	Tag nMaxChars nBTsocket nDeviceUUID nlsBluetoothConnexion nlsUserInitiateDisconnect nDevice progressDialog on textdata btnOn ept recogniton	onCreate() onRequestPermissionRequest()
Initialiseur	_____	initialize()
Préférences	_____	onCreate() onPause() onResume
VoirRadar	Angle distance Data noObject pixsDistance Angle Distance	Setup() Draw() drawRadar() drawLine() drawObject() drawText() serieEvent()

	index1 index2 arcFont	
DetecterRadar	Readvoice BT Droite_devant Droite_derriere Gauche_devant Gauche_arrire Trig Écho Temps Distance myServo	Setup() Loop() CalculDistance()

Tableau II.8 : Tableau explicatif des classes

2.2.3 DIAGRAMMES DE SEQUENCES

Les diagrammes de séquence possèdent intrinsèquement une dimension temporelle mais ne représente pas explicitement les liens entre les objets. Ils privilégient ainsi la représentation temporelle à la représentation spatiale et sont plus aptes à modéliser les aspects dynamiques du système. En revanche, ils ne rendent pas compte du contexte des objets de manière explicite, comme les diagrammes de collaboration. Le diagramme de séquence permet de visualiser les messages par une lecture de haut en bas. L'axe vertical représente le temps, l'axe horizontal les objets qui collaborent. Une ligne verticale en pointillé est attachée à chaque objet et représente sa durée de vie.

Le sous-système 1 : Commande à distance

- Cas d'utilisation : Se connecter au robot

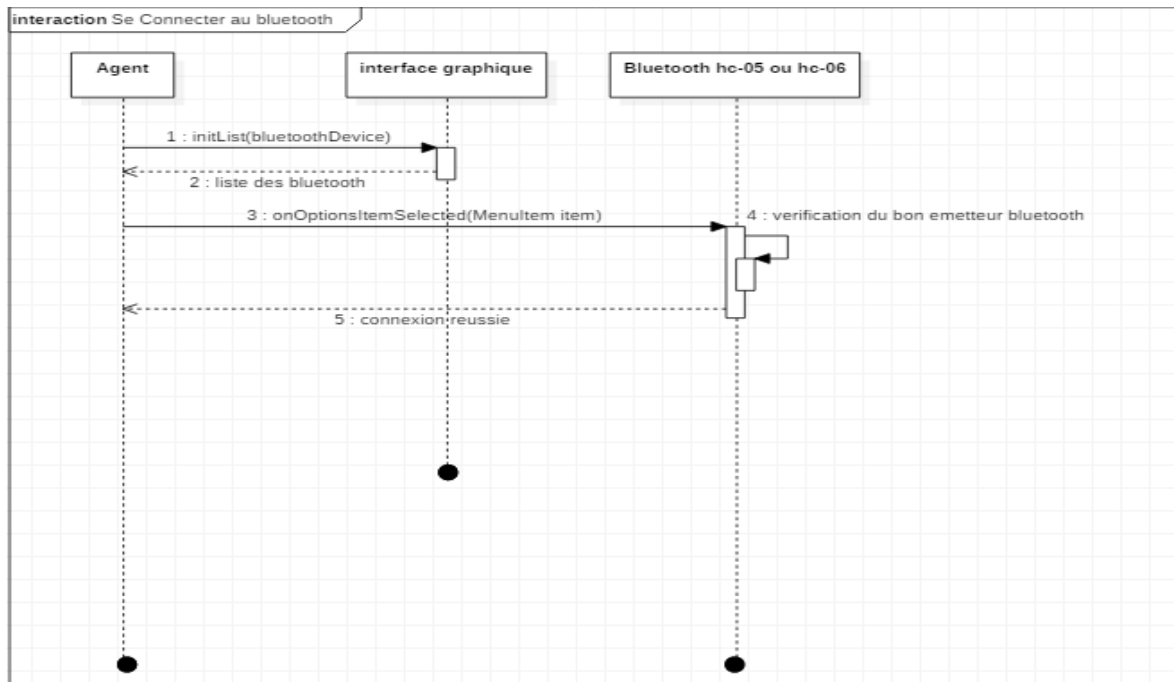


Figure II.4 : diagramme de séquence pour le cas : Se connecter au robot

- Cas d'utilisation : Commander le robot qui inclut le déplacement de celui-ci

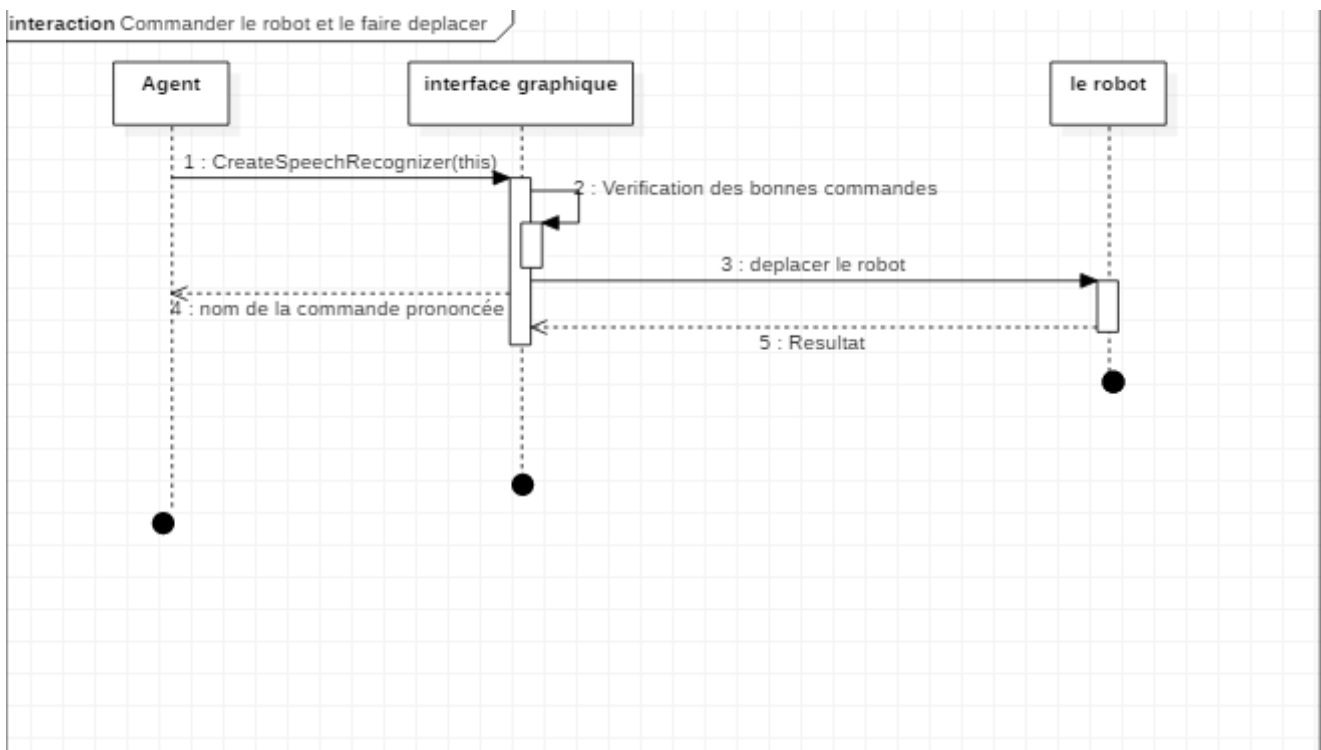


Figure II.5 : diagramme de séquence pour le cas : Commander le robo

Le sous-système 2 : Visualisation d'informations

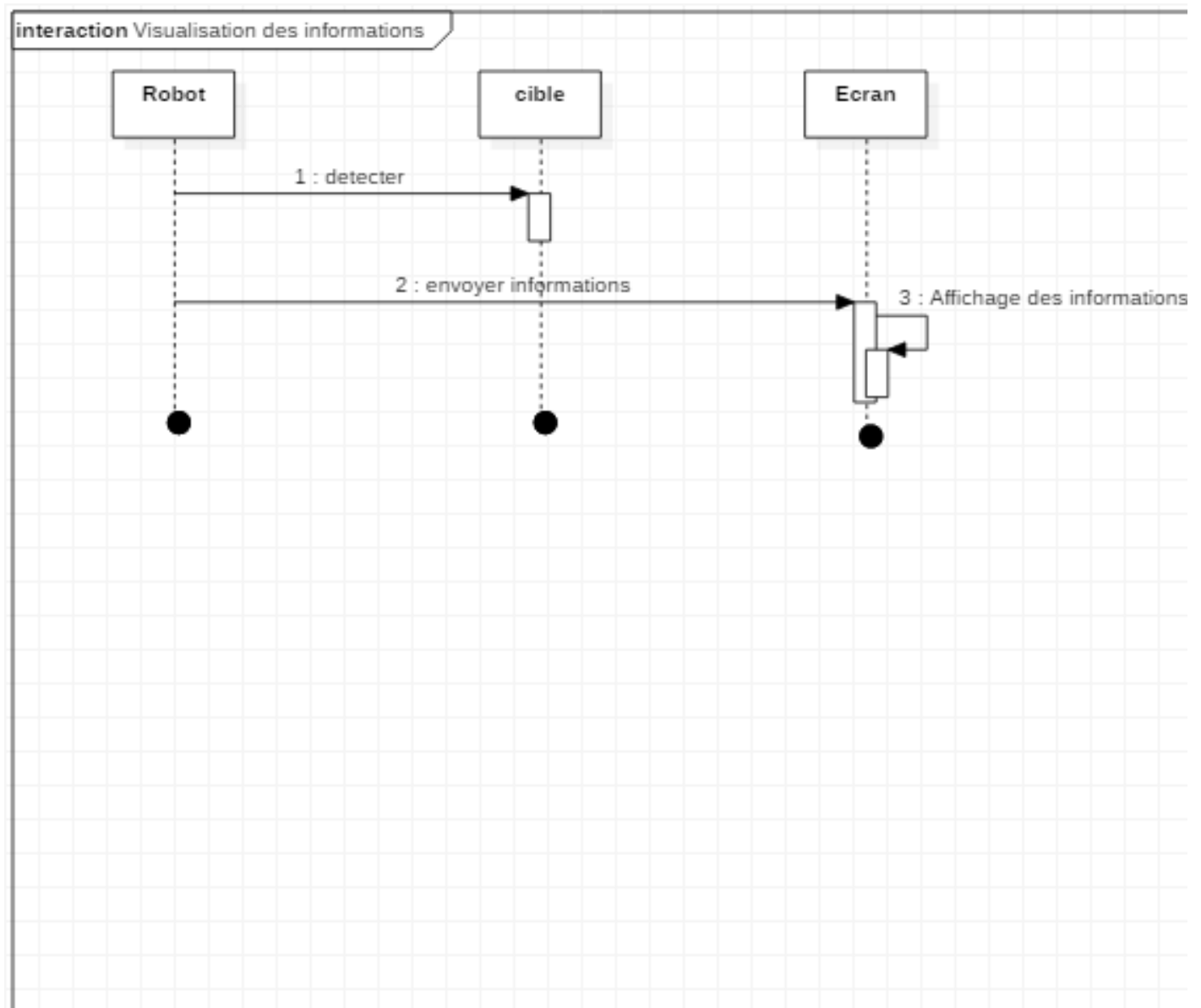


Figure II.6 : diagramme de séquence pour le cas : Détecter et voir les informations de détections

2.2.4. DIAGRAMMES D'ACTIVITÉS

Le diagramme d'activité est attaché à une catégorie de classe et décrit le déroulement des activités de cette catégorie. Le déroulement s'appelle "flot de contrôle". Il indique la part prise par chaque objet dans l'exécution d'un travail. Il sera enrichi par les conditions de séquencement. Il pourra comporter des synchronisations pour représenter les déroulements parallèles. La notion de couloir d'activité va décrire les responsabilités en répartissant les activités entre les différents acteurs opérationnels.

Notre diagramme d'activité se présente comme suit :

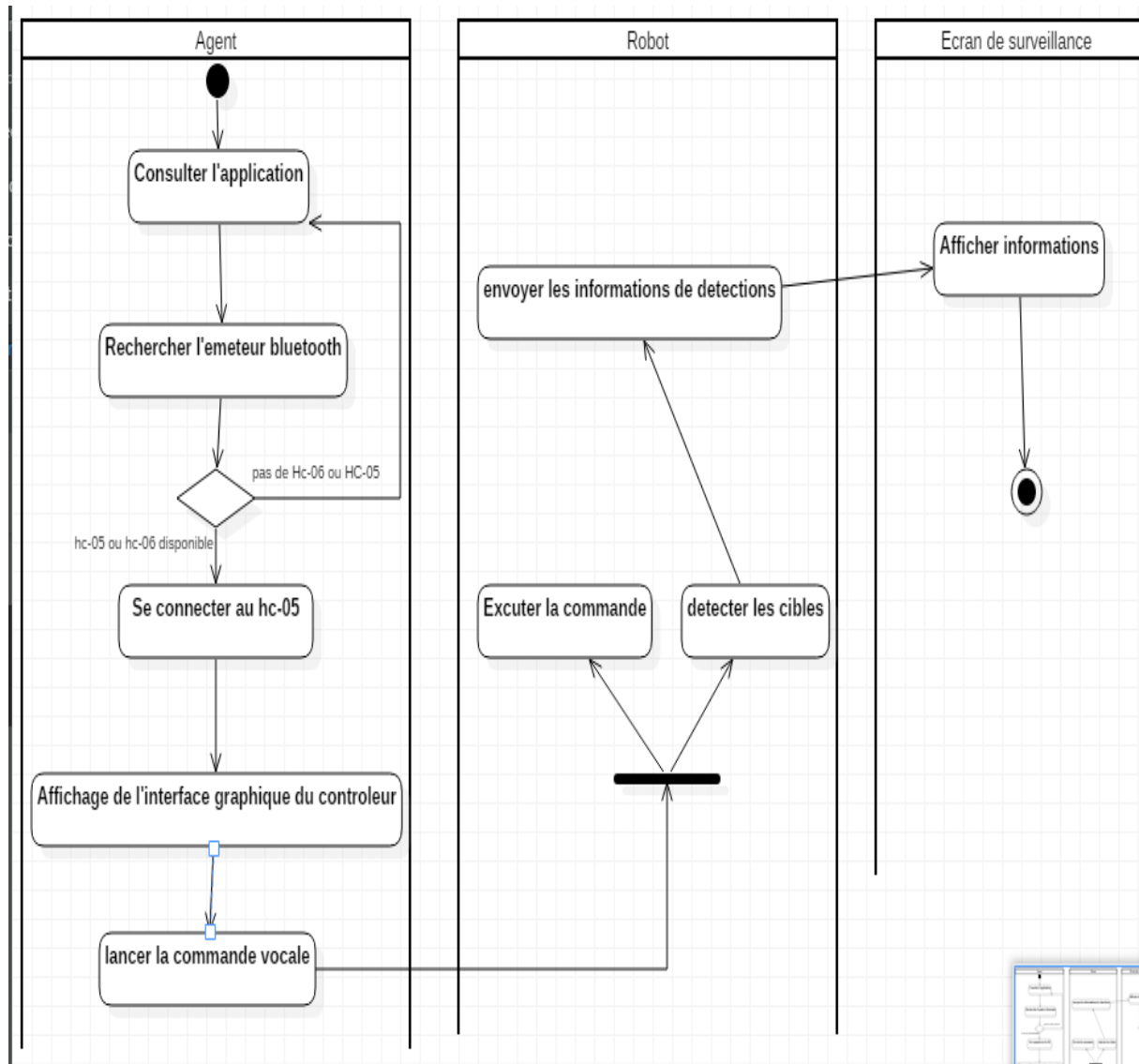


Figure II.7 : diagramme d'activité du système

2.2.5. DIAGRAMMES DE DEPLOIEMENT

Les diagrammes de déploiement montrent la disposition physique des différents matériels (les nœuds) qui entrent dans la composition d'un système et la répartition des instances de composants, processus et objets qui « vivent » sur ces matériels.

Les diagrammes de déploiement sont donc très utiles pour modéliser l'architecture physique d'un système.

Ainsi par rapport aux dispositions physique de notre système, nous avons comme diagramme de déploiement, la figure suivante :

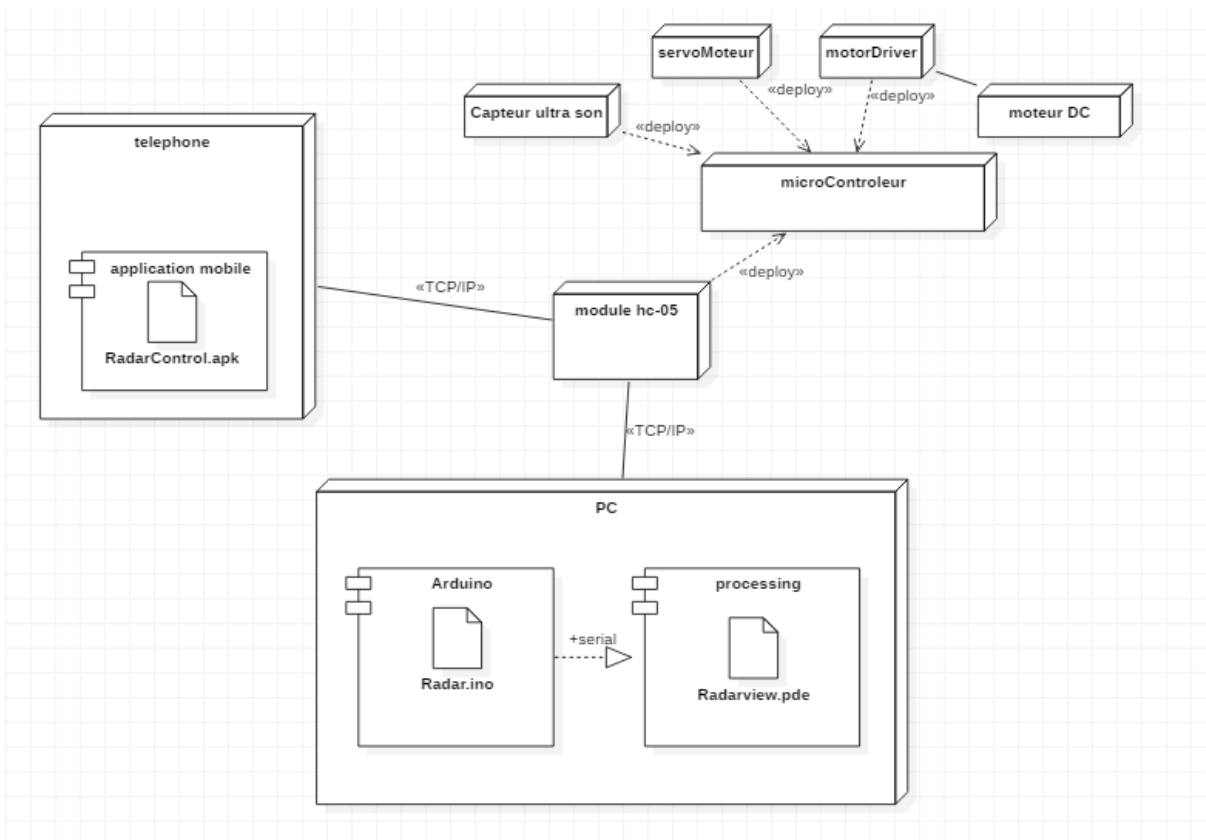


Figure II.8 : diagramme de déploiement du système

Dans ce chapitre, nous avons commencé par faire une narration de notre système qui est divisé en deux sous-systèmes, puis par rapport à cette narration nous avons choisi la méthode des processus unifiés pour modéliser nos sous-systèmes utilisant le langage UML.

Sue ce nous passons au dernier chapitre pour l'explication du fonctionnement de notre système ainsi que son implementation.

CHAPITRE 3. REALISATION ET IMPLEMENTATION DU SYSTÈME

L'objectif de ce chapitre est de présenter les composants matériels et logiciels utilisés pour la conception et la mise en œuvre du robot radar de détection, l'explication de son principe de fonctionnement et la représentations de son organigramme. Ainsi, nous avons 2 sections, la première section est la partie matérielle qui concerne les composants utilisés et leur connections entre elles, la deuxième section est celle du logiciel qui va quant à elle expliquer les environnements de développement utilisés pour le bon déroulement du système.

SECTION 1. PARTIE MATERIELLE

3.1.1. PRINCIPAUX COMPOSANTS UTILISÉS

a. La carte Arduino

La carte Arduino Uno fondée sur un microcontrôleur ATmega328 de 8bits cadencé à 16 MHz qui peut être programmée pour analyser et produire des signaux électriques. Aux niveaux de programmation on peut contrôler l'Arduino en langage C ou C++. Et contient un boot loader qui permet de modifier le programme sans passer par un programmeur.

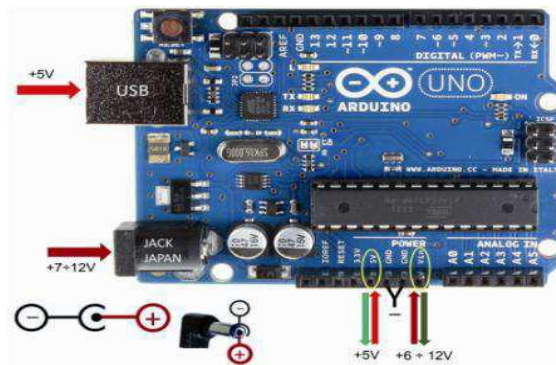


Figure III.1 : Arduino Uno

La carte UNO dispose de (06) six entrées analogiques (Numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de (10 bits) (C'est à dire sur 1024 niveaux soit de 0 à 1023) à l'aide de la fonction analogRead () du langage Arduino.

Puce d'interface USB : En haut à gauche de la carte à côté de la prise USB est

L'interface USB puce. Celle-ci convertit les niveaux du signal utilisé par le standard USB à des niveaux qui peuvent être utilisés directement par l'Arduino

Oscillateur à quartz

Interrupteur de réinitialisation

b. Le servomoteur

Un servomoteur (souvent abrégé en « servo », provenant du latin servus qui signifie « esclave ») est un moteur capable de maintenir une opposition à un effort statique et dont la position est vérifiée en continu et corrigée en fonction de la mesure. C'est donc un système asservi. Le servomoteur intègre dans un même boîtier, la mécanique (moteur et engrenage), et l'électronique, pour la commande et l'asservissement du moteur. La position est définie avec une limite de débattement d'angle de 180 degrés, mais également disponible en rotation continue.

Un servomoteur est composé :

- D'un moteur à courant continu.
- D'un axe de rotation et potentiomètre.
- D'une carte électronique pour le contrôle de la position de l'axe et le pilotage du moteur.

Le principe du moteur est assez simple, il suffit d'envoyer une impulsion, le temps que durera cette impulsion déterminera l'angle du servomoteur. Ce temps d'impulsion est de quelques millisecondes et doit être répété à un intervalle régulier.

Le servomoteur est commandé par l'intermédiaire d'un câble électrique à 3 fils qui Permettent d'alimenter le moteur et de lui transmettre des ordres de positions sous forme d'un signal codé en largeur d'impulsion plus communément appelés PWM (Pulse Wit Modulation ou Modulation de Largeur d'Impulsion) ou RCO (Rapport Cyclique d'ouverture). Cela signifie que c'est la durée des impulsions qui détermine l'angle absolu de l'axe de sortie et donc la position du bras de commande du servomoteur. Le signal est répété périodiquement, en général toutes les 20 millisecondes, ce qui permet de contrôler et de Corriger continuellement la position angulaire de l'axe de sortie, cette dernière étant mesurée par le potentiomètre.



Figure III.2 : servo moteur

c. Capteur Ultrason HC-SR04.

Un capteur est une interface entre un processus physique et une information manipulable. Il ne mesure rien, mais fournit une information en fonction de la sollicitation à laquelle il est soumis.

Le **HC-SR04** est un capteur de distance qui utilise les ultrasons, Il offre une excellente plage de détection avec des mesures stables. Son fonctionnement n'est pas influencé par la lumière du soleil ou des matériaux sombres, bien que des matériaux comme les vêtements puissent être difficiles à détecter. Ce capteur fonctionne avec une tension d'alimentation de 5 volts, dispose d'un angle de mesure de 15° environ et permet de faire des mesures de distance entre 2 centimètres et 3 mètres avec une précision de 3mm.

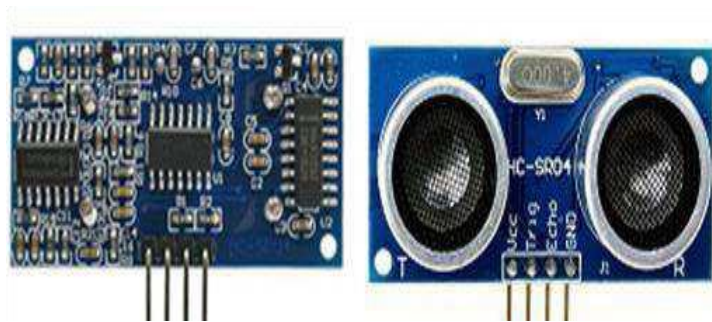


Figure III.3 : Module ultra-son

d. Motor driver L298N

Ce module intègre une entrée de tension. La plage de tensions bascule entre les 3 V et les 35 V, selon la configuration du cavalier régulateur de tension; et supporte une intensité de 2 A. Avec le cavalier régulateur on peut contrôler des moteurs de 5 à 12 V et on pourra utiliser la broche de 5 V du module pour ajouter d'autres *shields*. Sans ce cavalier régulateur, on peut contrôler des moteurs de 12 à 35 V, et dans ce cas la troisième broche agira comme entrée de 5 V pour alimenter la partie logique du driver.

Les sorties des moteurs 1 et 2 nous fourniront l'énergie nécessaire pour démarrer les moteurs. Assurez-vous que **la polarité** des moteurs est la même sur les deux entrées. Dans le cas contraire, vous devrez peut-être les échanger lorsque vous réglerez les deux moteurs vers l'avant et vers l'arrière!

Les pins IN 1 e IN 2 nous servent à contrôler le sens de rotation du moteur 1, et les pins IN 3 et IN 4, celui du moteur 2. Ces pins fonctionnent:

- IN 1 -> HIGH; IN 2 -> LOW: le moteur tourne dans un sens.
- IN 1 -> LOW; IN 2 -> HIGH: le moteur tourne dans le sens inverse.

Et c'est pareil pour les pins IN 3 et IN 4 du moteur 2.

Pour contrôler la vitesse de rotation des moteurs on doit retirer les cavaliers des pins ENA et ENB. On les connecte à deux sorties PWM de la plaque Arduino de façon qu'on puisse envoyer une valeur entre 0 et 255 pour gérer cette vitesse des moteurs. Avec les cavaliers installés, les moteurs tourneront toujours à la même vitesse.

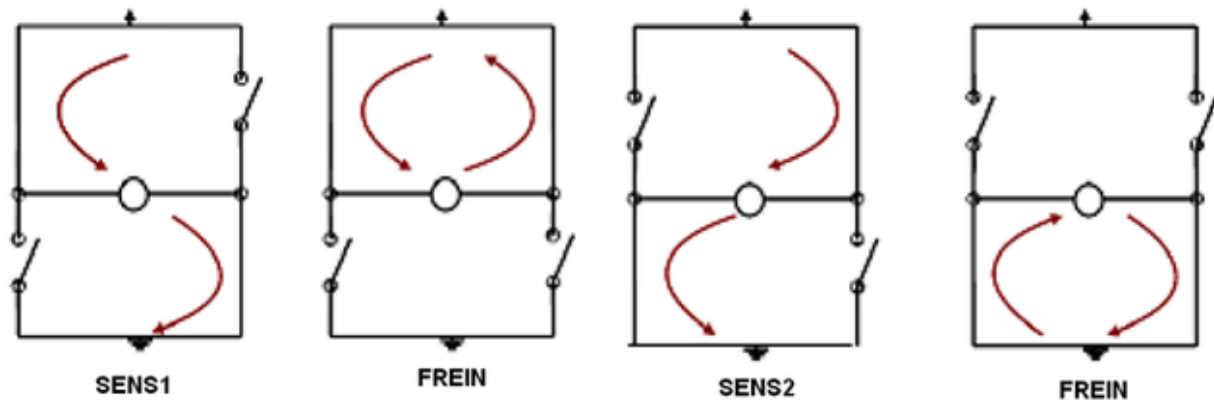


Figure III.4 : pont H

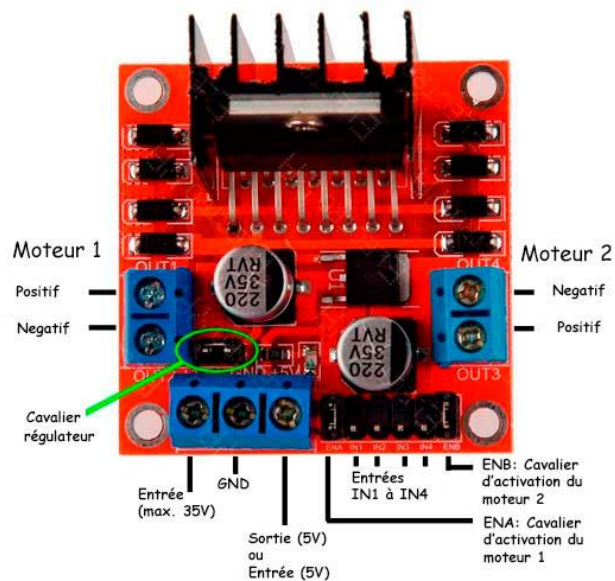


Figure III.5 : Motor driver

Ce montage consiste à réaliser un **pont H** pour **variateur de vitesse** pour un **moteur à courant continu** très utile pour la commande ou l'**asservissement** en **vitesse** et en **position** pour les **robots**

Le **pont en H** est une structure électronique servant à contrôler la polarité aux bornes d'un dipôle. Il est composé de quatre éléments de commutation généralement disposés schématiquement en une

forme de H d'où le nom. Les commutateurs peuvent être des relais, des transistors, ou autres éléments de commutation en fonction de l'application visée. Cette structure se retrouve dans plusieurs applications de l'électronique de puissance incluant le contrôle des **moteurs**, les convertisseurs et **hacheurs**, ainsi que les onduleurs.

e. Motor DC

Un **moteur à courant continu** fait partie d'une classe de moteurs électriques rotatifs qui convertit l'énergie électrique à courant continu en énergie mécanique. Les types les plus courants reposent sur les forces produites par les champs magnétiques. Presque tous les types de moteurs à courant continu ont un mécanisme interne, électromécanique ou électronique, pour changer périodiquement le sens du courant dans une partie du moteur.



109 Photo by DootyPook

Figure III.6 : Motor Dc

f. Module Bluetooth HC-05

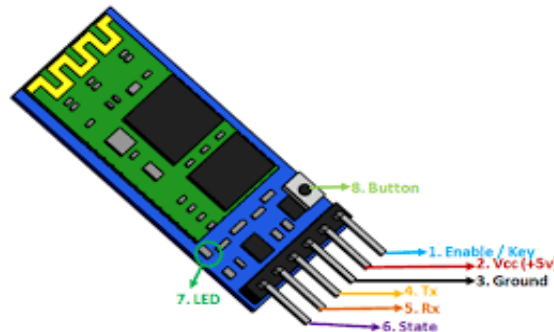


Figure III.7 : module bluetooth

3.1.2. SCHEMA DE BRANCHEMENT

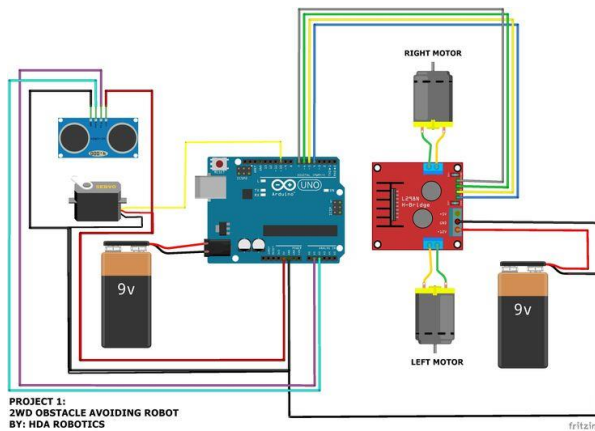


Figure III.8 : schéma 1

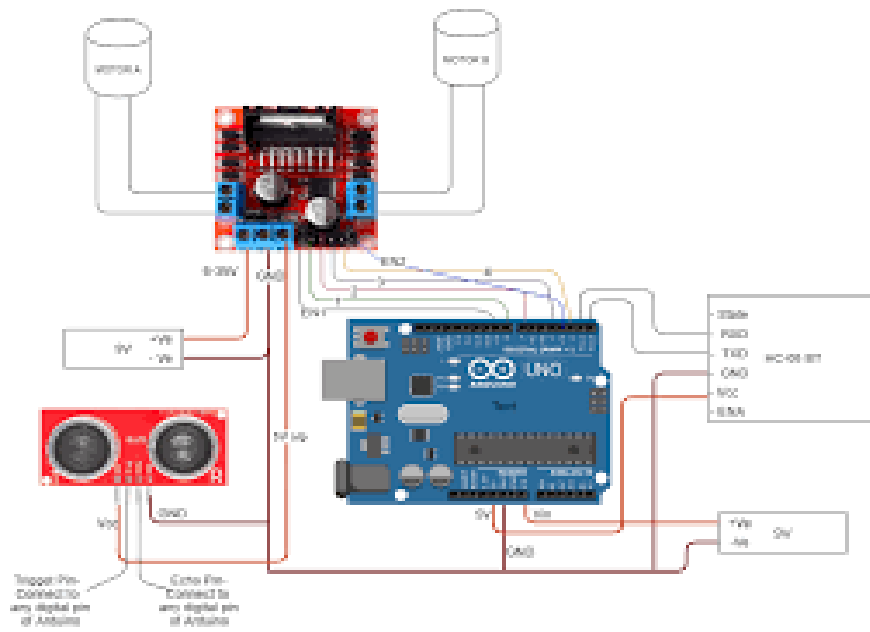


Figure III.9 : schema2

SECTION 2. PARTIE LOGICIELLE

3.2.1. IDE UTILISÉS

- ARDUINO

Arduino 1.8.12 est environnement de développement où sera effectué le code qui permettra de contrôler notre microcontrôleur et aussi de communiquer avec cette dernière à travers l'interface du communicateur série se trouvant dans l'environnement, cette communication sera faite par des port

série (à l'aide d'un câble USB connectant microcontrôleur à l'ordinateur), il est important de spécifier le type de port utilisé par le microcontrôleur.

Langage utilisé : C et C++

Ci-dessous le lancement de l'IDE Arduino1.8.12

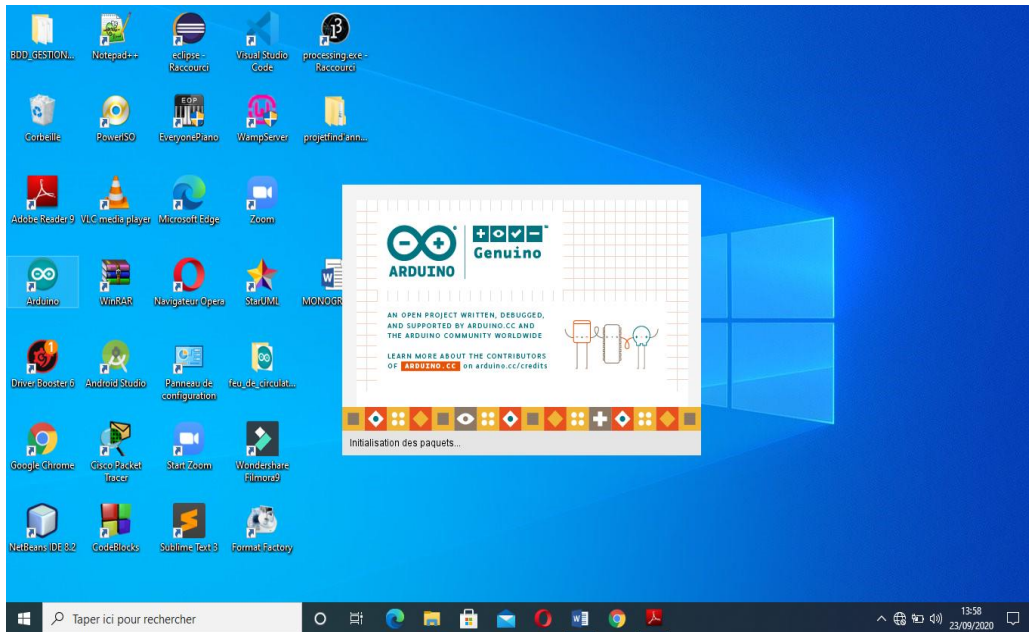


Figure III.10 : lancement de l'IDE arduino

- PROCESSING 3

Pour l'affichage du radar nous utiliserons l'environnement processing 3.5.4 qui sera connecter au premier IDE Arduino 1.8.12 grâce au port série qui sera spécifié dans le code

Langage utilisé : Java

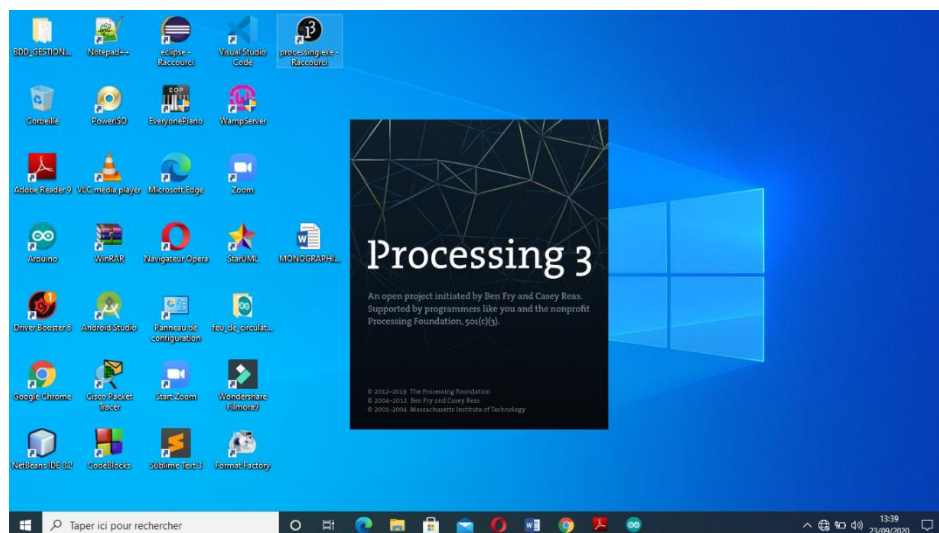


Figure III.11 : lancement de l'IDE processing

- ANDROID STUDIO

Pour créer notre application mobile nous avons eu à utiliser l'environnement Android Studio

Langage utilisé : Java pour le traitement logique et Xml pour la structuration (design) de l'Application



Figure III.12: lancement de l'IDE Android studio

3.2.2. SOUS SYSTEME 1 : DEPLACEMENT DU ROBOT PAR COMMANDE VOCALE

Ici nous utiliserons l'application mobile pour commander le déplacement du robot, Premièrement nous devons établir une connexion entre l'application mobile et le robot cette connexion se fait grâce à l'émetteur bluetooth qui se trouve dans le robot, les étapes à faire sont les suivantes :

Première étape : Dans l'interface graphique qui s'affiche, appuyer sur le bouton **CHERCHER** pour faire apparaître la liste des Bluetooth disponibles, bien sûr avoir la connexion bluetooth du téléphone activé

Deuxième étape : Dans la liste qui s'affiche, appuyer le bluetooth **HC-05**, puis appuyer sur bouton **CONNECTER**

Troisième étape : Dans la nouvelle interface graphique qui s'affiche, commencer d'abord par activer les données mobiles, puis après appuyer sur bouton ayant l'icône d'un microphone, puis effectuer la commande vocale spécifique (AVANCER, ARRIÈRE, À GAUCHE, À DROITE, ET STOP)

PREMIERE ETAPE : RECHERCHER LES BLUETOOTHS DISPONIBLES

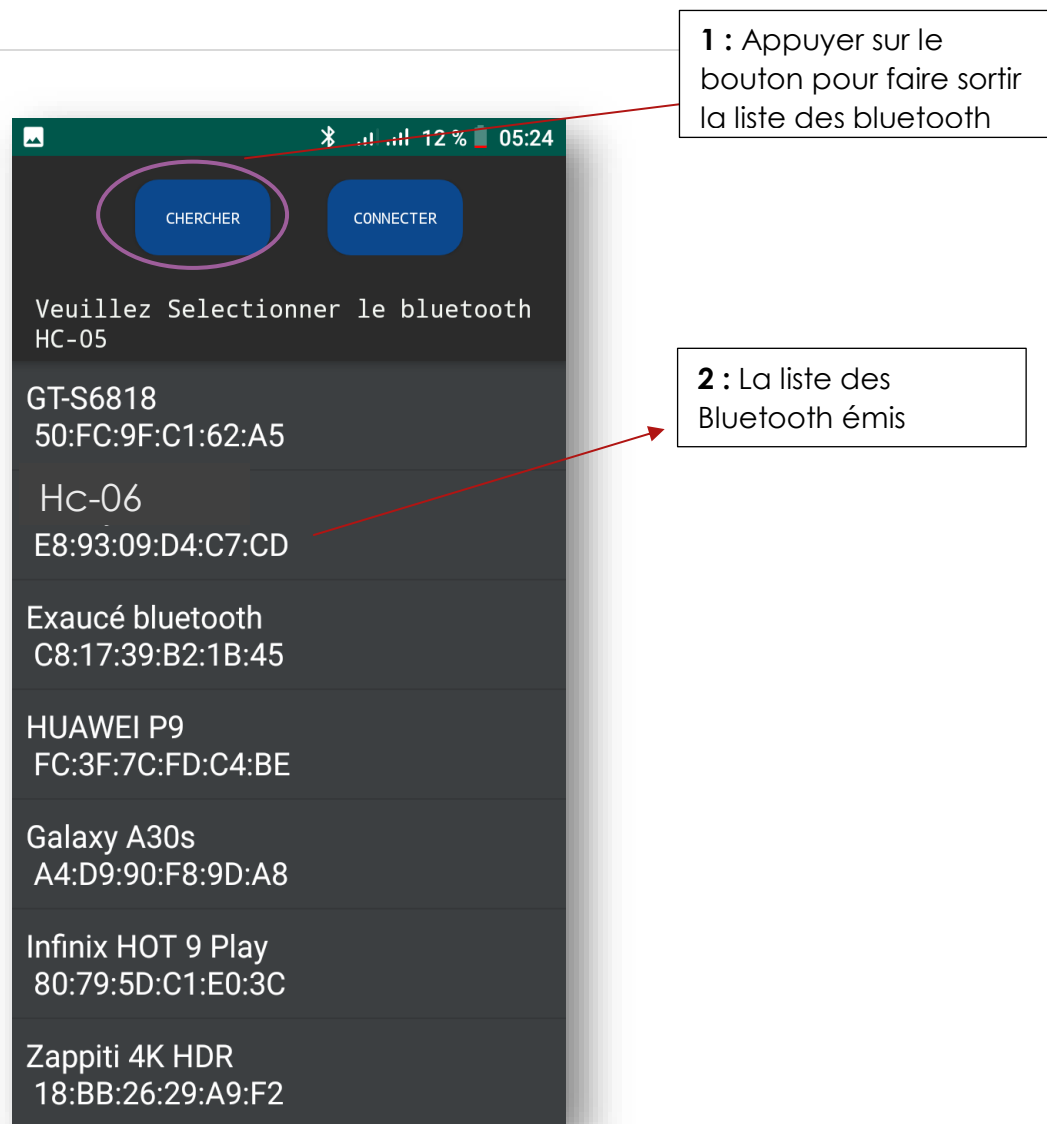


Figure III.13 : application mobile-rechercher bluetooth

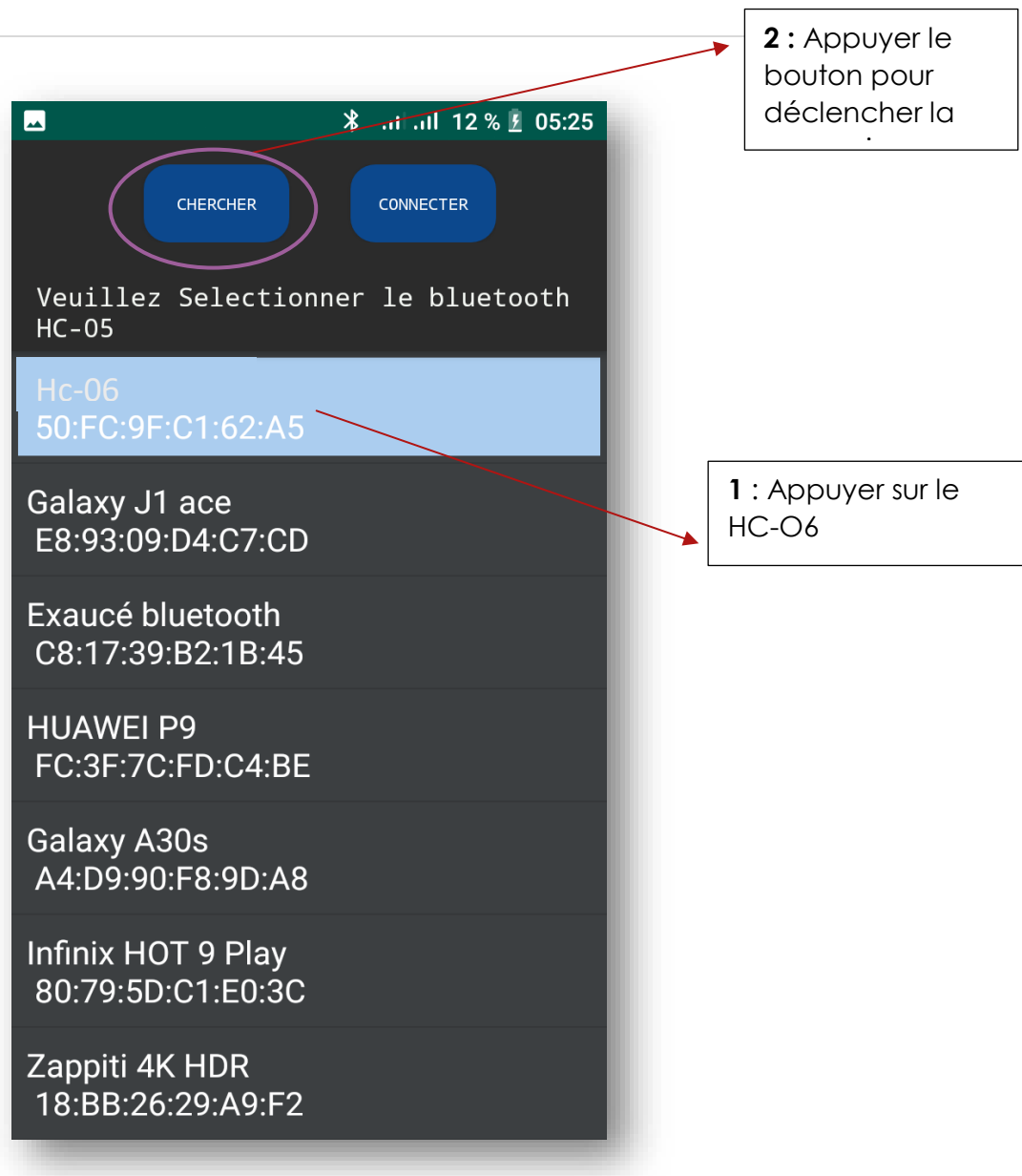


Figure III.14 : application mobile-connecter bluetooth

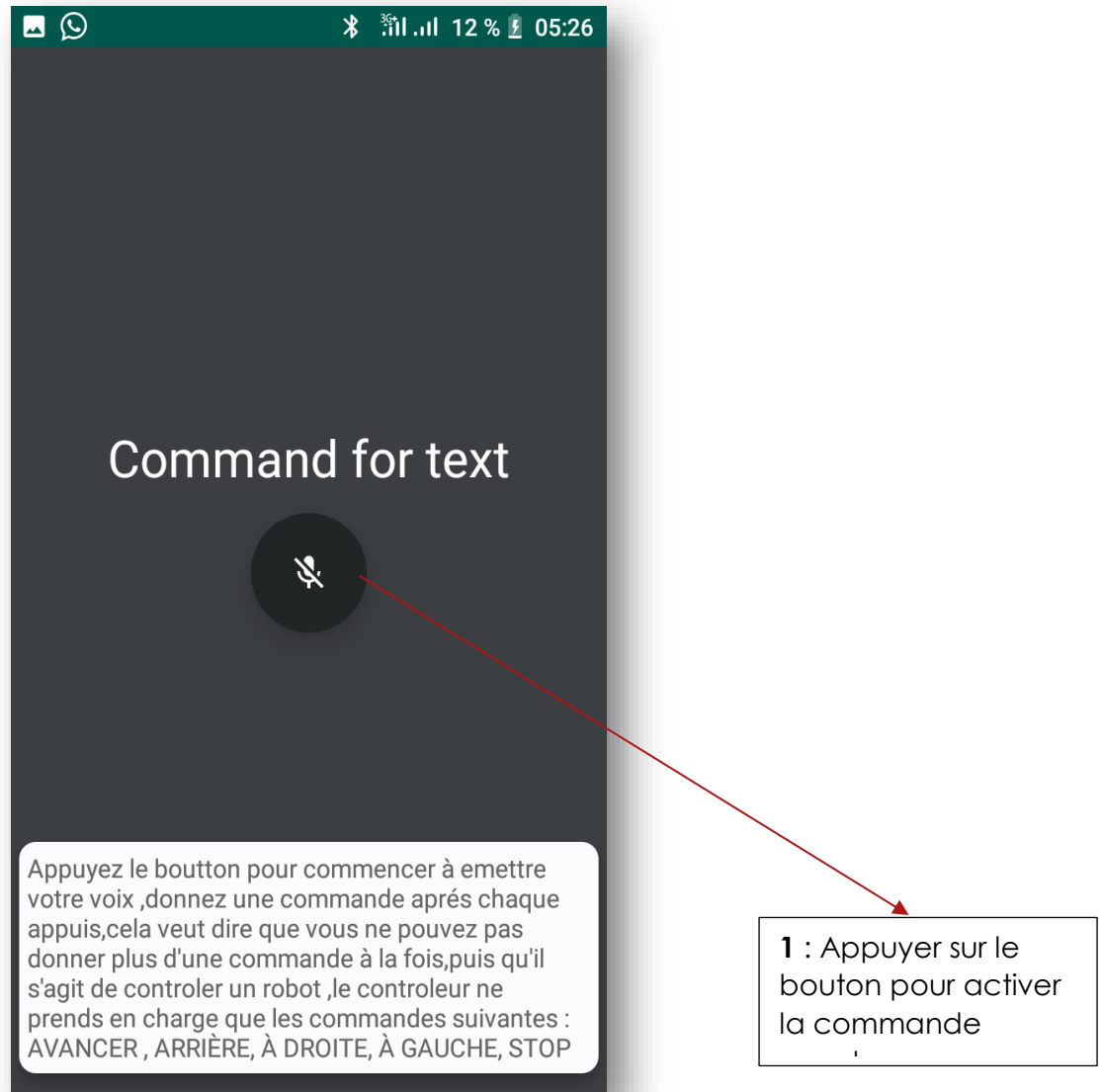


Figure III.15 : l'application mobile-Commande Vocale

Comment les données sont transmises vers le robot? Et comment le déplacement se fait – il ?

Quand une commande vocale est prononcée, une méthode **SpeechRecognition** est employée pour lire, reconnaître et transcrire la voix humaine en chaîne de caractère lettre par lettre et les stocker dans une variable, ainsi cette variable est transmise en réseau vers le module bluetooth, ces données passent par des ondes électromagnétiques pour que le module bluetooth puissent les recevoir.

À noter qu'il y a 4 moteurs DC reliés à 4 roues sur le robot et ils sont contrôlés par le module bluetooth ;

La commande **AVANCER** : lorsque le module bluetooth reçoit cette commande, il envoie une impulsion de 5 volt aux 4 moteurs, les 4 moteurs sont en états de fonctionnement et du coup le robot avance.

La commande **ARRIÈRE** : lorsque le module reçoit cette commande, il bloque les deux roues avant, et envoie une impulsion de 5 volts ainsi le robot recule.

La commande **À GAUCHE** : Lorsque le module reçoit cette commande, il actionne la roue avant se trouvant du côté droit en envoyant une impulsion de 5 volts, et bloque les autres roues, ce qui permet que le robot aille vers la gauche.

La commande **À DROITE** : Lorsque le module reçoit cette commande, il actionne la roue avant se trouvant du côté gauche en envoyant une impulsion de 5 volts, et bloque les autres roues, ce qui permet que le robot aille vers la droite.

La commande **STOP** : le module bloque toutes roues et le robot s'arrête

Extrait de code source pour montrer comment la commande vocale est lue, reconnu, transcrite sous forme de string et ensuite transmise vers le module bluetooth

```
@Override
public void onResults(Bundle bundle) {

    ArrayList<String> data = bundle.getStringArrayList(recognizer.RESULTS_RECOGNITION);

    String getData = data.get(0);

    //algorithme pour prendre les mot et les envoyer au bluetooth
    if(getData.equalsIgnoreCase( anotherString: "avancer"))||
    getData.equalsIgnoreCase( anotherString: "arrière"))||
    getData.equalsIgnoreCase( anotherString: "à gauche"))||
    getData.equalsIgnoreCase( anotherString: "à droite"))||
    getData.equalsIgnoreCase( anotherString: "stop")
    )
    {
        txtdata.setText(getData);
        on = ""+txtdata.getText();

        try {
            mBISocket.getOutputStream().write(on.getBytes());
        } catch (IOException e) {
```

Extrait de code pour montrer comment le module lit les données qui lui sont transmises

```
while (BT.available()){  
  delay(10);  
  char c = BT.read();  
  readvoice += c;  
}
```

Extrait de code pour montrer comment la commande « Avancer » s'exécute

```
if(readvoice == "avancer")  
{  
  digitalWrite(droite_devant, HIGH);  
  digitalWrite (gauche_devant, HIGH);  
  digitalWrite(droite_derriere, LOW);  
  digitalWrite (gauche_arriere, LOW);  
  delay(100); }
```

Extrait de code pour montrer comment la commande « Arrière » s'exécute

```
else if(readvoice == "arrière")  
{ digitalWrite(droite_devant, LOW);|  
  digitalWrite(gauche_devant, LOW);  
  digitalWrite(droite_derriere, HIGH);  
  digitalWrite(gauche_arriere, HIGH);  
  delay(100); }
```

Extrait de code pour montrer comment la commande « à gauche » s'exécute

```
else if (readvoice == "à gauche")  
{ digitalWrite (droite_devant, HIGH);  
  digitalWrite (gauche_devant, LOW);  
  digitalWrite (droite_derriere, LOW);  
  digitalWrite (gauche_arriere, LOW);  
  delay (100);}
```

Extrait de code pour montrer comment la commande « à droite » s'exécute

```
else if ( readvoice == "à droite")
{digitalWrite (droite_devant, LOW);
 digitalWrite (gauche_devant, HIGH);
 digitalWrite (droite_derriere, LOW);
 digitalWrite (gauche_arriere, LOW);
 delay (100);}
```

Extrait de code pour montrer comment la commande « Stop » s'exécute

```
else if (readvoice == "stop")
{digitalWrite (droite_devant, LOW);
 digitalWrite (gauche_devant, LOW);
 digitalWrite (droite_derriere, LOW);
 digitalWrite (gauche_arriere, LOW);
 delay (100);}
```

3.2.3. SOUS SYSTEME 2 : DETECTION RADAR ET VISUALISATION DES INFORMATIONS DE DETECTION

A. DETECTION D'UNE CIBLE AVEC L'IDE ARDUINO

Pour utiliser le module ultrason afin de calculer la distance, nous avons créé deux constantes du type Int (entier) trig et echo qui recevront respectivement les valeurs 10 et 11 (en correspondance avec le trig et l'écho du module ultrason qui sont connectés sur la carte Arduino), un entier temps pour calculer le temps de l'émission,

Pour calculer la distance, nous avons créé une fonction **calculdistance ()**,

Dans cette fonction, on enverra une impulsion de 5v pendant 10millisecondes dans le trig pour pouvoir émettre un ultrason, s'il y'a un obstacle devant le module, le son émis retournera sous forme d'écho

D'où la durée du son émis sera reçu dans la variable temps en microsecondes, on connaît la vitesse du son dans l'air qui est de 340 m/s (qui sera converti en cm/microsecondes 0.034), on peut maintenant calculer la distance par la formule (temps*0.034) /2, la variable distance recevra le résultat et la fonction retournera la distance

Pour le servomoteur, nous avons commencé par inclure la librairie **Servo.h** afin de pouvoir utiliser ses classes et de les instancier à un objet que nous avons appelé myservo, notre objet va contrôler le servomoteur à partir du pin 12,

Dans une boucle qui va de 0 à 180, notre objet myservo recevra une impulsion à chaque itération afin d'effectuer une rotation de 0 à 180 degrés et appellera la fonction **calculDistance ()**, on fera de même dans une boucle qui effectue la rotation inverse en allant de 180 à 0
Ainsi pour chaque angle on aura une distance en cm.

Extrait de code pour montrer comment la détection s'effectue

```
void loop() {

    // rotation du servoMoteur de 0 à 180 degree

    for(int i=0;i<=180;i++){
        myServo.write(i);

        delay(30);
        distance = calculDistance();// appel de la fonction calculDistance pour calculer la distance pour chaque degre
        Serial.print(i); // envoie le nombre de degre pour chaque itteration de i
        Serial.print(",");
        Serial.print(distance); // envoie le nombre de distance calculee pour chaque itteration de degre
        Serial.print(".");

    }

    // // rotation du servoMoteur de 180 à 0 degree
    for(int i=180;i>0;i--){
        myServo.write(i);
        delay(30);
        distance = calculDistance();
        Serial.print(i);
        Serial.print(",");
        Serial.print(distance);
        Serial.print(".");

    }

}

//fonction pour calculer la distance dont la formule est (temps*vitesse)/2
int calculDistance(){
    digitalWrite(trig, LOW);
    delayMicroseconds(2);
    digitalWrite(trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig, LOW);
    temps = pulseIn(echo, HIGH); // lecture de l'echo qui retourne une valeur recue par l'emission du trigger en microsecondes
    distance= temps*0.034/2;

    return distance;
}
```

B. VISUALISATION DES DETECTIONS RADAR AVEC L'IDE PROCESSING

Nous avons la fonction **drawRadar ()** pour construire la ligne radar et les arcs en calculant le cosinus par rapport à la taille de l'écran,

La fonction **drawLine ()** pour construire les lignes qui désignerons les angles,

La fonction **drawText ()** pour écrire des textes dans l'écran,

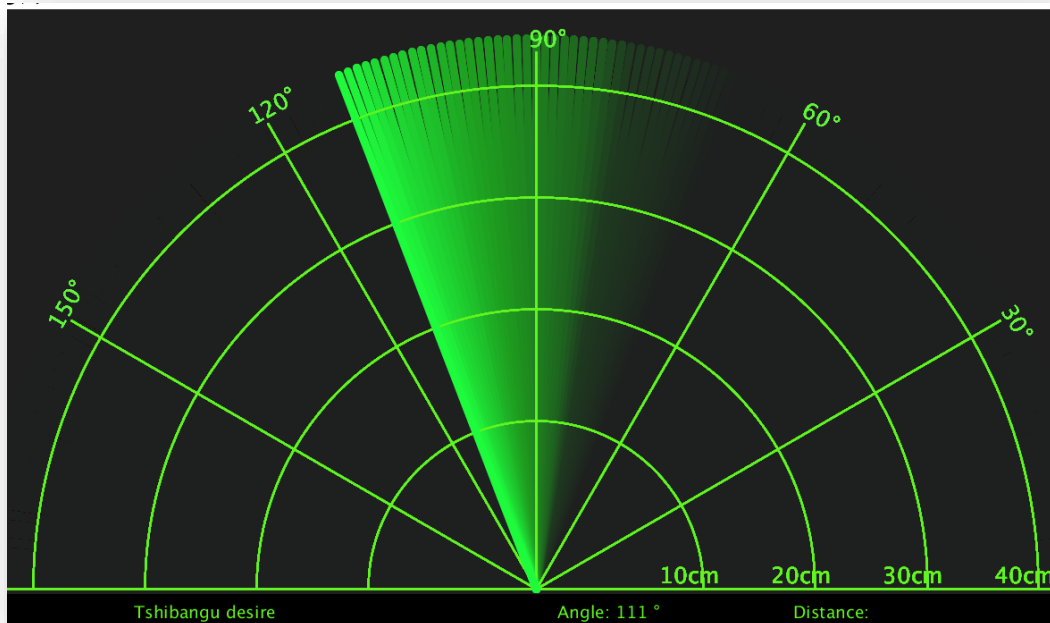
La fonction **drawObjet ()** pour spécifier la détection d'un Object dans un angle donné en le colorant en rouge

La fonction **draw ()** qui appellera toutes les fonctions précédentes,

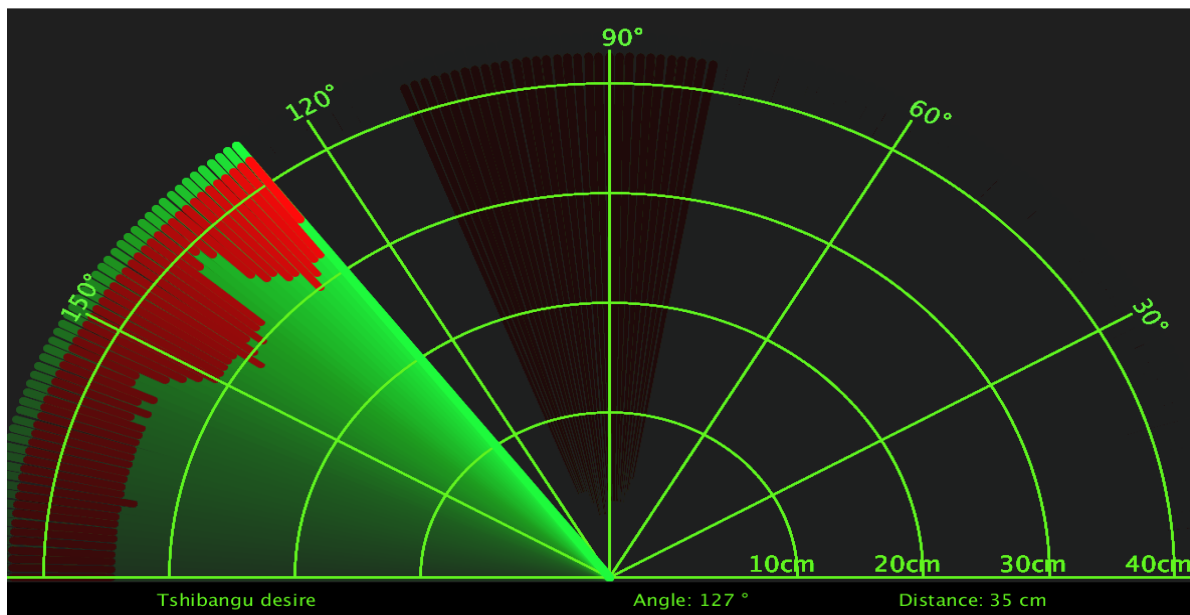
L'importation de la librairie **processing.Serial** qui va permettre de communiquer avec le port série de L'IDE arduino,

La fonction **serialEvent ()** qui va permettre de lire les données dans le port série à savoir l'angle et la distance

Là, nous avons l'écran de visualisation, la partie verte qui fait un balayage, nous indique qu'aucun objet n'a été détecté par rapport à un angle donné (ici 111°)



Sur l'image ci-dessous, nous avons l'écran de visualisation, la partie rouge, nous indique qu'un objet a été détecté par rapport à un angle donné



Dans ce chapitre, nous expliquons la mise en place de notre système, du point de vue matériel, nous avons utilisé les composants disponible et nécessaire à la réalisation de notre système, du point de vue logiciel, nous avons montré les différents algorithmes qui ont été élaborés pour le bon fonctionnement du dit système.

CONCLUSION GENERALE

Pour résumer, Le Radar est un système de détection de cibles qui utilise des ondes électromagnétiques pour déterminer la portée, l'altitude, la direction et la vitesse des cibles. L'antenne Radar transmet des impulsions d'ondes qui rebondissent sur la cible qui se trouve sur leur chemin. La cible renvoie une partie de l'onde reçue par le récepteur qui est en ligne de vue avec l'émetteur.

Notre travail consiste donc à la réalisation d'un robot de détection RADAR pour la surveillance des frontières terrestres, de ce fait, notre système est en mesure de résoudre les problèmes liés aux frontières terrestres de la RDC, pour pouvoir le faire, nous l'avons divisé en deux sous-systèmes, le premier sous-système traite le déplacement du robot par commande vocale à partir d'une application mobile, le deuxième sous-système traite la détection et la visualisation des informations de détections sur un écran de surveillance.

En perspective, l'intégration des technologies plus avancées permettra d'améliorer la sécurité des frontières terrestres dans notre pays :

- Un module ultra-son plus performant ayant une portée de plus de 500km.
- Une caméra de surveillance
- Un module de détection des bombes nucléaires.
- L'intégration d'une intelligence artificielle pour la reconnaissance faciale des criminels recherchés par la police essayant de passer la frontière.

Nous sommes d'avis que ce travail reste limité par cause du manque d'équipements beaucoup plus sophistiqués dans notre pays néanmoins il permet de résoudre les problèmes pour lesquels il a été implémenté. Se heurtant à l'erreur humaine, aux limites d'un chercheur et avec une probité intellectuelle, nous donnons une orientation pour que les brillants chercheurs qui nous succèdent améliorent le fruit de ce travail. Nous nous mettons à la disposition de toute critique et de toute remarque pour l'amélioration et le recadrage de ce travail.

BIBLIOGRAPHIE

Michel-Henri CARPENTIER, «RADAR», Encyclopaedia Universalis, URL : <http://www.iniversalis.fr/encyclopedie/radar/>

M. Skolnik , « Introduction to radar systems». Mc-Graw-Hill, New-York, 1981

Philippe Goy, «Détection d'obstacles et de cibles de collision par un radar FMCW aéroporté». Thèse de doctorat de l'université de Toulouse, France. mardi 18 décembre 2012.

B.Cottenceau, « Carte ARDUINO Uno Microcontrôleur ATmega328 ». Microcontrôleurs EI3 Option AGI.

Capteur à Ultrasons HC-SR04 .www.robot-maker.com.

LAKRID DJAHID, « Emulation de Système SCADA pour le Contrôle et Distribution d'eau », mémoire de master 2 en automatique industriel, Université Badji Mokhtar Annaba. Algérie. Promotion 2017.

NOTE DES COURS

- ZUIYA B. (2021). Introduction à la robotique, Kinshasa
- NKONGOLO KABWA-KA-NTANDA M. (2021) Note de cours de télématique, Kinshasa
- MUSANGU LUKA M. (2021). Langage de modélisation unifié (UML), Kinshasa