

Sommaire

INTRODUCTION	3
1- besoins et objectifs du projet	4
1.1 Contexte	4
1.2 Motivation	5
2 gestion de projet	5
2.1 L'équipe	5
2.2 La planification de projet et les outils de gestion	5
2.3 Cahier de charge	6
2.4 Organigramme	7
2.5 Diagramme GANTT prévisionnel	9
2.6 le plan de mise en œuvre prévisionnel du projet	10
2.7 Répartition des tâches et les technologies	12
2.7.1 Répartition des tâches	12
2.7.2 Les technologies	12
3. développement technique	Erreur ! Signet non défini.
3.1 La stratégie	16
3.2 Architecture matérielle	17
3.3 Architecture logicielle	17
3.4 Les solutions logicielles	18
4. l'électronique embarquée	19
4.2 Pont en H avec la L293D	19
4.3 Schéma de branchement du servomoteur et principe de fonctionnement	21
4.4 Schéma de branchement du module Bluetooth.	22
4.5 Schéma de câblage final	23
4.6 Test de fonctionnement	24
4.7 Observation lors du fonctionnement de la maquette	24
4.8 Difficultés rencontrées	25
5. Bilan du projet	26
5.1 Apports individuels et collectifs	26
5.2 Liste des composants utilisés pour la réalisation du projet	27
Conclusion	28
BIBLIOGRAPHIE	29

Projet robot détecteur d'obstacle

Liste des figures

Figure 1: diagramme prévisionnel de planification du projet.....	10
Figure 2: capteur ultra son.....	12
Figure 3: servomoteur	13
Figure 4: moteur à courant continu	14
Figure 5: module Bluetooth.....	14
Figure 6: carte Arduino	15
Figure 7: Interface de programmation Arduino.....	15
Figure 8:Schéma de branchement des moteurs	19
Figure 9: schéma de branchement du servomoteur	21
Figure 10:Schéma de branchement du module Bluetooth.....	22
Figure 11: schéma de câblage final de la voiture	23
Figure 12: maquette de la voiture.....	24

INTRODUCTION

La robotique est un domaine vaste qui regroupe un grand nombre de matières telles que l'informatique, l'électronique, la mécanique etc. C'est pour cette raison que nous avons trouvé intéressant de créer un outil plus attirant pour les nouvelles technologies. Nous avons choisi de concevoir un robot mobile capable de mettre en œuvre ces matières. Notre objectif au terme de ce projet est de réaliser un robot éducatif, facile à monter et bon marché. Il devra rassembler des éléments permettant de réaliser des expériences amusantes de découverte de la robotique, l'électronique et l'informatique. Nous voulons donc concevoir un robot capable d'interagir avec son environnement au moyen des capteurs. Il sera commandé à l'aide de différents programmes C Arduino qui aura pour intérêt de mettre en valeur une des matières citée plus haut. Nous diviserons notre travail en deux phases majeures : le développement d'une application androïde de commande d'une part et l'intégration d'une plateforme robotique simple avec électronique de commande et microcontrôleur d'autre part.

1- besoins et objectifs du projet

1.1 Contexte

L'axe Douala – Yaoundé fait partir du circuit routier Douala – Yaoundé – Bafoussam – Douala, appelé au Cameroun « *Triangle de la mort* », à cause de la fréquence des accidents mortels. En 2011, plus de 2600 accidents ayant causé la mort de 1113 personnes ont été enregistrés contre près de 2200 accidents et 721 décès en 2012. Il serait difficile de déterminer le nombre de personnes ayant perdu la vie suite à un accident de la route sur cet axe. Le dernier accident en date a causé une dizaine de morts et une trentaine de blessés dans la petite localité de Mandoumba arrondissement de Matomb région du centre.

Les causes de ces accidents de circulations peuvent être les causes humaines, les causes matérielles, les causes infrastructurelles et les causes environnementales.

- Parmi les causes humaines, on note la responsabilité des conducteurs due aux excès de vitesse, à de mauvais dépassements, aux surcharges de passagers et/ou de marchandises, à la conduite en état d'ébriété, aux mauvais stationnements, au défaut de pré-signalisation de véhicules en détresse, à l'utilisation du téléphone et au déficit de compétences au volant, au chevauchement des lignes continues.
- Quant aux causes matérielles, elles sont directement liées à l'état technique des véhicules majoritairement caractérisé par l'usure des pneus, la défaillance des systèmes de freinage, la complaisance coupable des agents chargés d'assurer la visite technique des véhicules.
- Pour ce qui est des causes infrastructurelles, elles concernent d'une manière générale le mauvais état des routes avec la fréquence des nids de poules, les défaillances de visibilité dues à l'envahissement des accotements par la broussaille et donc, le défaut des abords de chaussées, la rareté voire le manque d'aires de repos, l'absence de matériels de dégagement de véhicules en panne ou abandonnés sur les abords des chaussées, la mauvaise signalisation routière, la disposition de dos d'ânes intempestifs.
- Pour ce qui est des causes environnementales, on retiendra la pluviosité et les temps brumeux qui réduisent la visibilité des conducteurs de manière substantielle.

1.2 Motivation

Partant des causes humaines, il nous ait venu l'idée de concevoir une voiture capable d'éviter les chocs en fonctionnement automatique et de s'arrêter en fonctionnement commande lorsqu'elle se trouve à une certaine distance d'un l'obstacle afin de réduire les causes humains liés aux accidents de circulation. Nous avons ainsi décidé de réaliser une solution domotique dans le but d'illustré cette idée.

2 gestion de projet

2.1 L'équipe

Notre équipe de projet est composée de personnalités complémentaires provenant de différents cursus :

- NAFACK BAURICE, chef de project, licence en EEA.
- BASSO TSAKENG FRANCK, licence technologique.
- MATZE LAURA, licence en EEA.
- FOKOU TCHINDA ADOLPH HTLER, licence en EEA.
- LOH BORIS-EDMOND BOHTEH, fondateur du projet, licence en EEA.

Nos diverses expériences professionnelles, académique et personnelles au cours de ces années d'études, nous ont permis d'aiguiser notre curiosité et de nous ouvrir à d'autres domaines et technologies. Nos centres d'intérêts, souvent abordés au cours de nos discussions quotidiennes, nous ont rassemblés cette année autour d'un projet domotique, faisant intervenir la programmation des microcontrôleurs et des technologies embarquées.

2.2 La planification de projet et les outils de gestion

Pour accompagner le développement du projet dès le stade d'ébauche, un cours de gestion de projet a rapidement été mis en place. Ce cours introduit la méthodologie à suivre et les outils nécessaires au bon déroulement d'un projet. Étant dans une dimension ingénieur, cette gestion est d'autant plus importante que le respect des délais, des coûts et de la performance sont essentiels dans la conception d'un système complexe. La gestion de projet permet d'autre part de distribuer les travaux à réaliser chaque jours de rencontre mais également de créer une base de référence permettant de surveiller les écarts et l'évolution du projet afin d'assurer sa continuité.

2.3 Cahier de charge

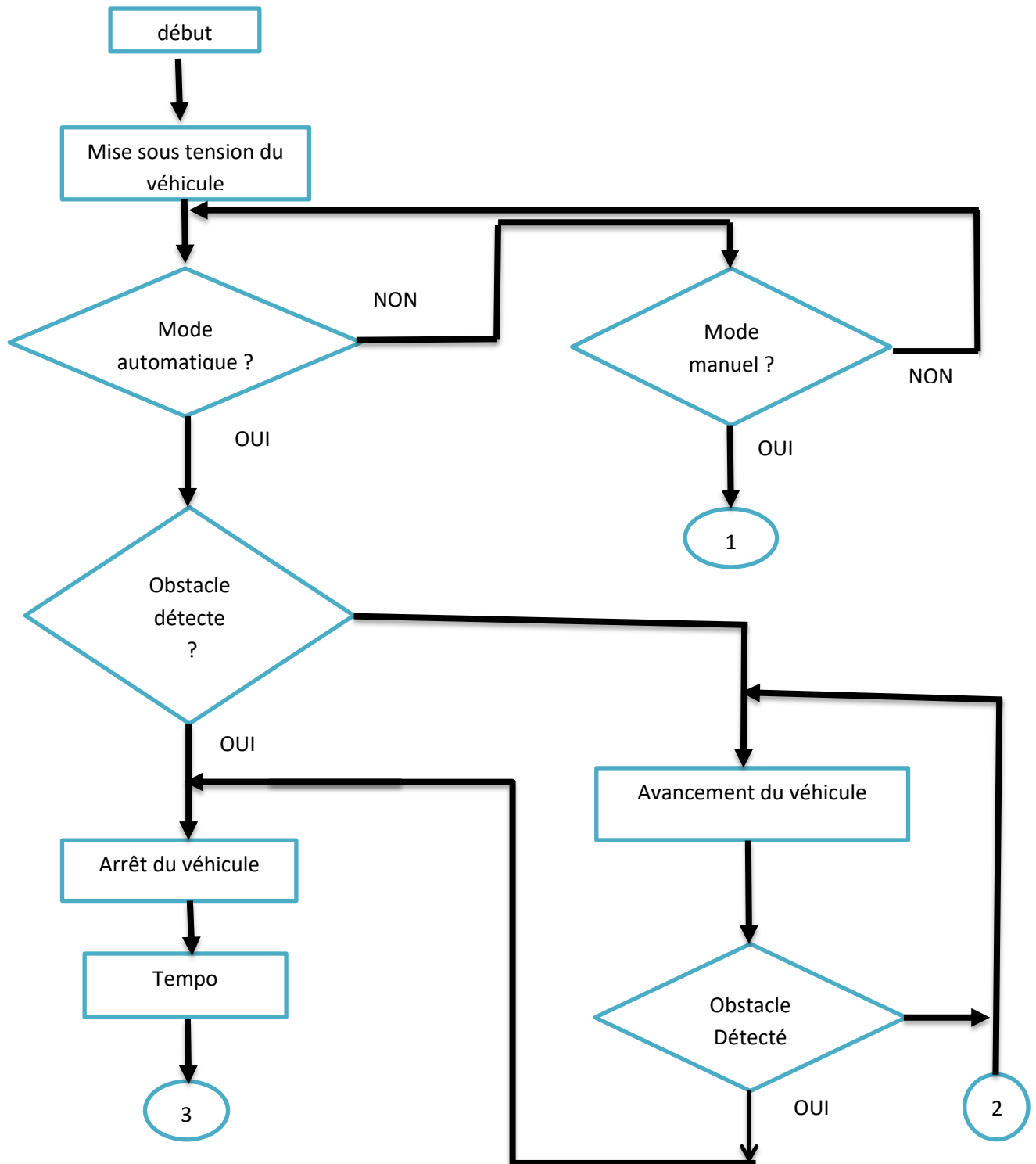
Décrivant l'ensemble des conditions attachées à l'exécution du projet, le cahier de charge nous permet dans un premier temps, de définir le contexte, les enjeux, les objectifs techniques ainsi que les livrables et les axes de développement envisagés.

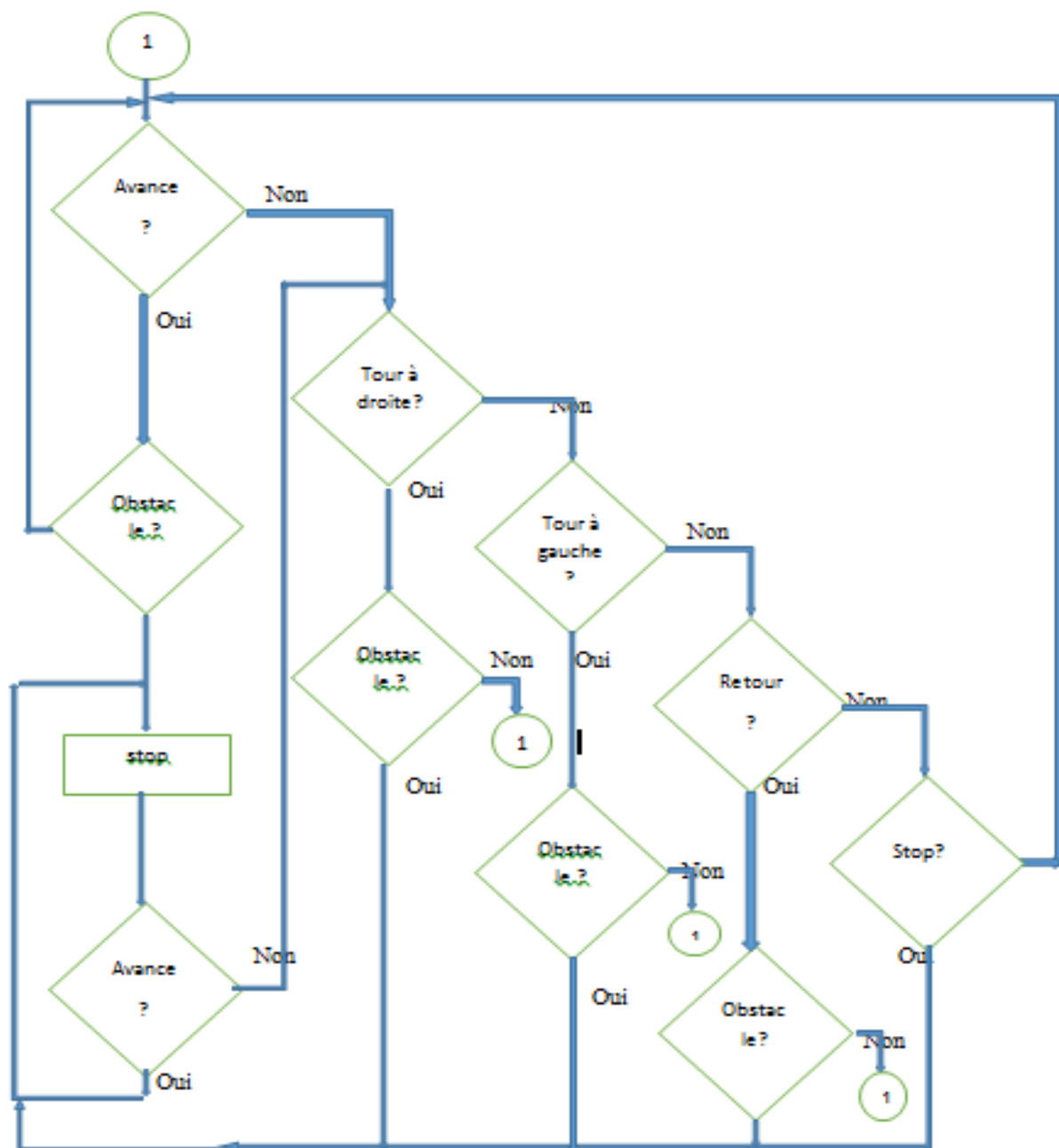
Deux types de fonctionnement seront adoptés dans la réalisation de notre projet :

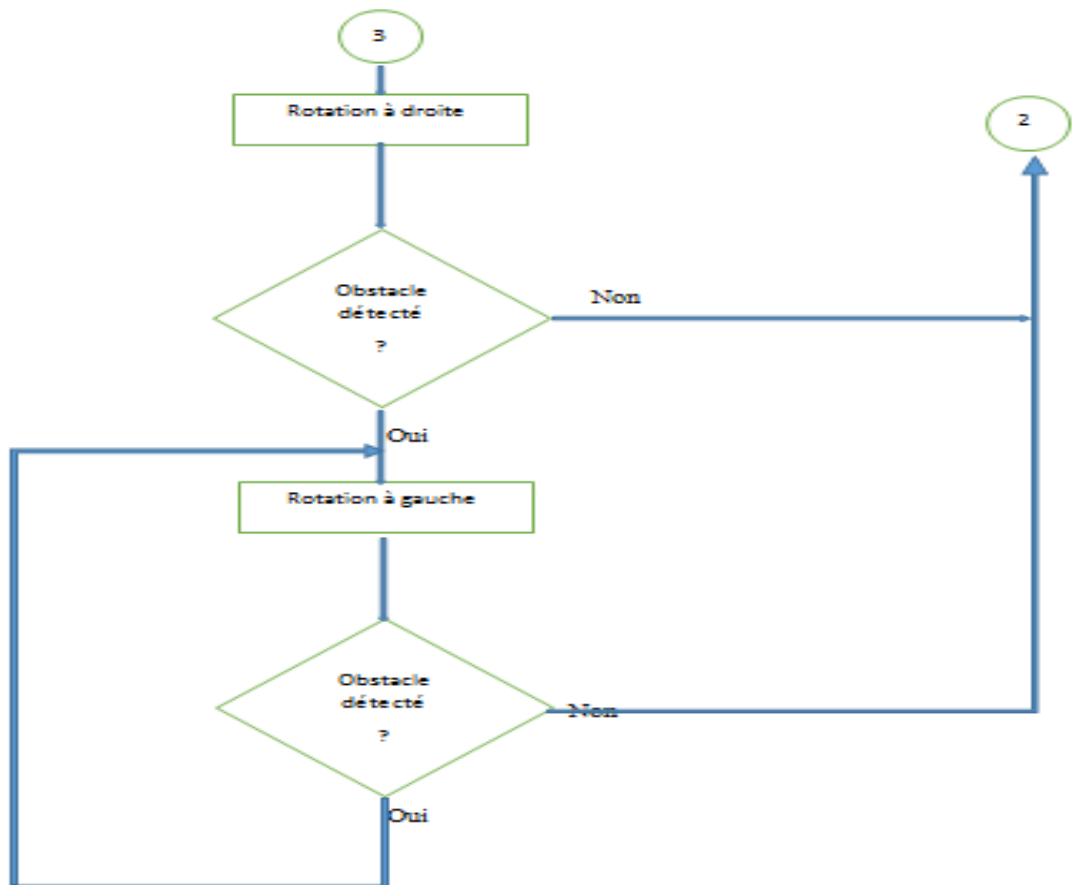
- Fonctionnement en mode automatique : ici le fonctionnement du prototype est autonome c'est à dire une fois ce mode activé via un interrupteur, il suit une trajectoire précise jusqu'à détection d'obstacle il s'arrête pivote à droite et avance s'il y'a pas d'obstacle sinon, il revient à la position d'obstacle initiale, puis pivote vers la gauche et avance s'il y'a pas d'obstacle sinon retour à la position initiale d'obstacle et rebrousse chemin.
- Fonctionnement en mode commande : ici le mode de commande utilisé sera le mode texte. C'est à dire, une fois ce mode activé, il attend de recevoir une information texte afin d'exécuter l'instruction reçue envoyer depuis un téléphone Androïde à travers le module Bluetooth, lors de la détection d'un obstacle, il s'arrête jusqu'à nouvelle instruction différente d'avancer tant qu'il y a présence d'obstacle.

En organisant nos idées, nous avons ainsi pu vérifier la concordance et la faisabilité de notre projet.

2.4 Organigramme







2.5 Diagramme GANTT prévisionnel

Le diagramme de GANTT est un outil efficace exploitant des données brutes tel que les dates de début et de fin y compris les durées de chacune des tâches afin de générer une visualisation de l'avancement du projet. Il permet de donner une vue globale des tâches à réaliser, des responsabilités et des ressources associées, de l'idée jusqu'à la mise en service en passant par l'analyse des exigences, l'étude de faisabilité, la conception fonctionnelle, les spécialisations, la réalisation et enfin, les tests. Aussi, il est possible d'organiser une gestion des ressources, leurs disponibilités, leurs coûts, etc.

Projet robot détecteur d'obstacle

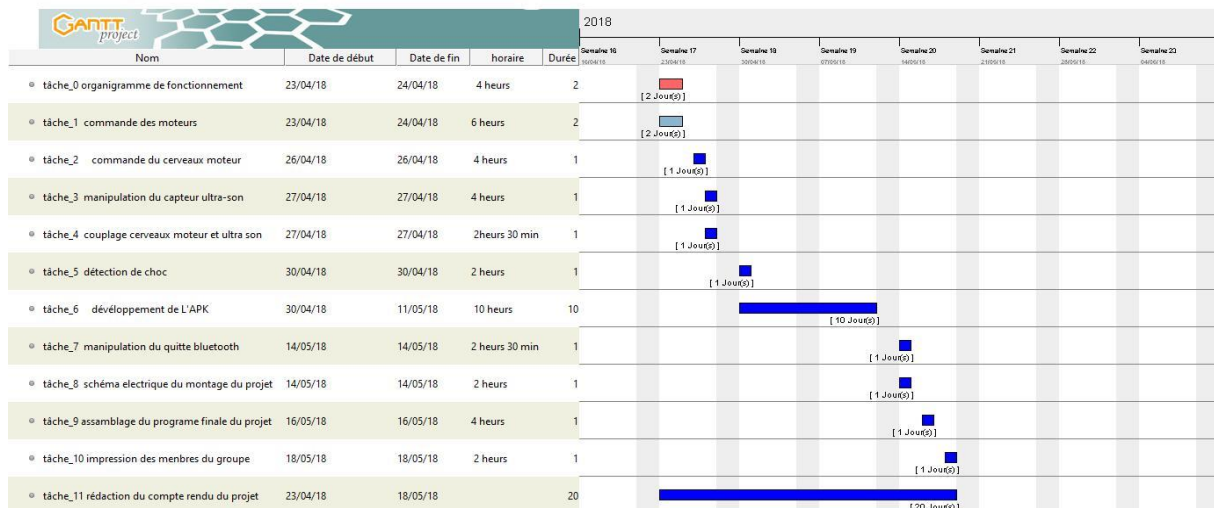


Figure 1: diagramme prévisionnel de planification du projet

2.6 le plan de mise en œuvre prévisionnel du projet

Le plan de mise en œuvre est un plan qui permet d'avoir un aperçu des différentes tâches à effectuer les jours des rencontres. Ainsi le plan de mise en œuvre de notre projet avant le début de réalisation de tâches est le suivant :

Période	Activités	Tâches	Observations
23/04/2018 au 24/04/2018	Mettre en place le comportement complet de la voiture face aux différentes situations	Etablir l'organigramme de fonctionnement de la voiture	Faire au préalable le cahier de charge Et lire le cours sur la réalisation d'un organigramme.
	Alimenter les moteurs à partir du driver et établir le schéma électrique de branchement dans Fritzing	Commande des moteurs des roues de la voiture	Installer au préalable Fritzing et Arduino pour programmer la carte. Prévoir un générateur de tension continue pour alimenter les moteurs indépendamment d'Arduino
26/04/2018	Brancher un servomoteur à la carte Arduino	Commande du servomoteur	Etablir le schéma électrique de branchement et écrire le programme Arduino permettant de commande le servomoteur
27/04/2018	Brancher un capteur ultra-son à la carte Arduino	Mesure d'une distance à l'aide d'un capteur ultra-son	Etablir le schéma électrique du montage et après branchement récupérer la distance mesurée par le capteur à partir d'une ligne de code afin de définir la distance seuil d'arrêt de la

Projet robot détecteur d'obstacle

	Brancher un capteur ultra-son à un servomoteur	Coupler un servomoteur et un capteur ultra-son	voiture après détection du choc Faire pivoter le capteur ultra-son dans de différente direction afin d'analyse la présence d'un obstacle.
30/04/2018	Relier le couplage servomoteur-capteur à la chassie de la voiture	Détecter le choc	Définir l'état des LED pour différentes situations et écrire le programme de commande des LED et de mesure sous forme de fonction
30/04/2018 Au 11/05/2018	Se connecter sur internet et chercher comment développer un apk pour Arduino	Développement de l'application Androïde pour commander l'échange des données avec le kit Bluetooth	Chercher le site de création sans programmation java afin d'organiser simplement l'interface de l'apk
14/05/2018	Brancher le kit Bluetooth à la carte Arduino	Manipuler le kit Bluetooth	Ecrire une fonction en C Arduino permettant d'interprété les informations reçu par la carte depuis ces broches analogique
	Dessiner le schéma complet de fonctionnement électrique de la voiture dans Fritzing	Schéma électrique de fonctionnement de la voiture	Télécharger les bibliothèques des composants Arduino afin d'incorporer dans Fritzing. Imprimer le schéma final en pdf ou l'enregistrer en fichier .JPGE
16/05/2018	Organiser Le programme de commande complet de la voiture	Assembler du programme final du projet	Incorporer toutes les sous fonctions écrites dans un fichier principal, faire appel à chaque sous fonction à l'emplacement correspondant dans le programme
18/05/2018	Donner leur avis concernant un travail en groupe	Impression des membres du groupe	Chaque membre du groupe doit écrire un fichier texte permettant d'avoir leurs avis concernant la réalisant du projet et du travail en équipe. Dire s'il a aimé travail avec chaque membre du groupe
23/05/2018	Imprimé le rapport du projet.	Finalisation du compte rendu du projet	Prévoir les frais d'impression

2.7 Répartition des tâches et les technologies

2.7.1 Répartition des tâches

Le diagramme ci-dessous illustre la répartition du temps attribué à chacune des grandes étapes du projet. C'est une conclusion graphique résultant de notre gestion de projet, nous permettant d'identifier au premier coup d'œil les travaux chronophages et l'homogénéité entre les tâches.

2.7.2 Les technologies

- capteur ultra-son : plusieurs technologies permettent la mesure de la distance entre le capteur et un obstacle. Trois technologies ont été prises en considération, soit les capteurs ultrasoniques, les capteurs à infrarouge et les capteurs laser. La technologie qui a été choisie est la technologie à ultra-son, car elle est abordable, précise et elle n'est pas affectée par la lumière du soleil comme le sont les capteurs à infrarouge. Les capteurs lasers auraient été de meilleurs candidats, mais ils sont hors prix. Pour effectuer les mesures, le capteur ultrasonique émet des pulsations ultrasoniques et mesure le temps entre l'émission du signal et la réception de son écho. Pour utiliser le capteur srf05, on envoie une impulsion à la patte d'entrée/sortie. Ensuite, le capteur envoie un signal d'une durée proportionnelle à la distance mesurée. Le capteur srf05 est illustré à la figure suivante :

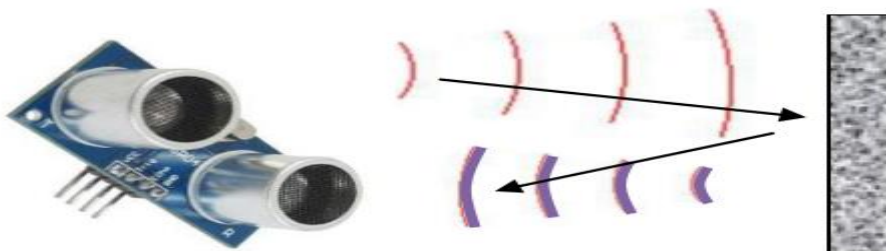


Figure 2: capteur ultra son

- ❖ Le fonctionnement: Il faut envoyer une impulsion niveau haut (à +5V) pendant au moins 10µs sur la broche 'TrigInput' ; cela déclenche la mesure. En retour la sortie 'Output' ou 'Echo', va fournir une impulsion +5V dont la durée est proportionnelle à la distance si le module détecte un objet. Afin de pouvoir calculer la distance on utilisera la formule suivante :

$$distance = \frac{\text{durée d'impulsion en } \mu s}{58}$$

- ❖ Les caractéristiques techniques du module sont les suivantes:
 - Tension de fonctionnement : 5Vdc
 - Courant de fonctionnement : moins de 2mA
 - Signal de sortie en tension (0-5Vdc)
 - Angle max de capteur : 15°

- Plage de détection : 2cm - 450cm
- Haute précision : 0,3cm
- Fonctionnement à l'impulsion sur la pin Trigger
- Largeur de pulse de trigger : 10µs
- Sortie sur la pin Echo, durée de niveau haut fonction de la distance mesurée

➤ Les servomoteurs

Les servomoteurs sont des actionneurs. Très utilisés en modélisme et dans l'industrie, ils ont comme caractéristique principale leur « couple », c'est-à-dire la force de rotation qu'ils peuvent exercer. Plus un servomoteur aura de couple et plus il pourra actionner des « membres » lourds comme déplacer un bras qui porte une charge. Pour la robotique de loisirs, les servomoteurs ont en général peu de couple et sont de taille réduite, bien adaptée à un encombrement minimal et à une énergie disponible limitée. Les servomoteurs sont pilotés par un fil de commande et alimentés par deux autres fils. Habituellement, ces 3 fils sont rassemblés dans une prise au format standard.



Figure 3: servomoteur

Un fil rouge est relié à l'alimentation positive (+5 ou +6 V selon le servomoteur), le fil noir est relié à la masse (GND) et le fil jaune est utilisé pour la commande. Il y aurait beaucoup à dire sur le fonctionnement d'un servomoteur : ses composants, son moteur et le petit potentiomètre qui permet de connaître sa position mais nous nous limiteront à son utilisation avec l'Arduino.

➤ moteur

Réf : Mini motoréducteur 6V (FIT0016)



Figure 4: moteur à courant continu

Ce moteur + engrenages, par définition, motoréducteur possède une tension de commande de 6V et une vitesse de rotation de 180 tours/min.

❖ **Spécifications:**

- Rapport de réduction : 1:120
- Vitesse hors-charge (3V) : 100 tours/min
- Vitesse hors-charge (6V) : 200 tours/min
- Courant hors-charge (3V) : 60mA
- Courant hors-charge (6V) : 71mA
- Courant d'arrêt (3V) : 260mA
- Courant d'arrêt (6V) : 470mA
- Dimension : 55mm x 48.3mm x 23mm
- Poids : 45g

Nous n'avons cependant aucune information sur la consommation en charge.

➤ **Module Bluetooth**



Figure 5: module Bluetooth

Le Bluetooth utilise un protocole “maitre-esclave”.

➤ **carte Arduino**

Projet robot détecteur d'obstacle

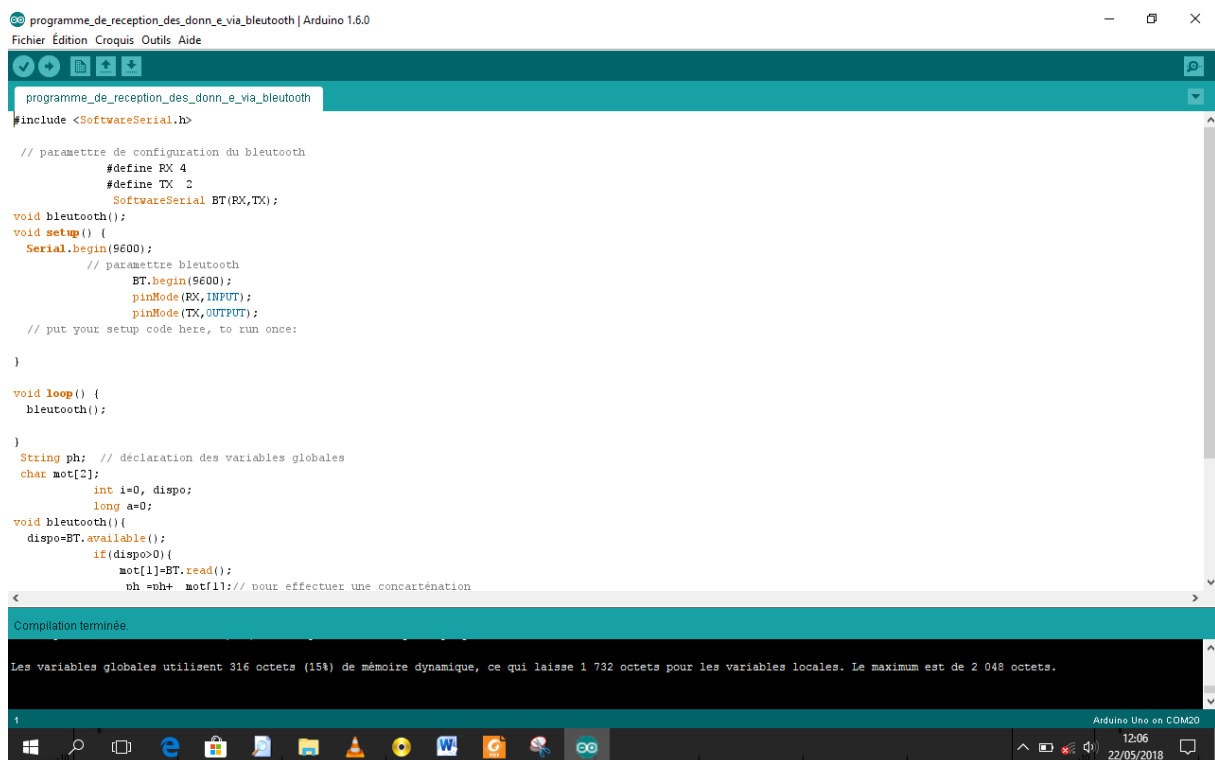


Figure 6: carte Arduino

Pour pouvoir programmer notre carte, nous avons besoin de trois choses : un ordinateur, une carte Arduino et connaître le langage Arduino. La Plateforme de programmation Arduino comprend les parties suivantes:

- Partie programme : Une telle carte d'acquisition qui se base sur sa construction sur un microcontrôleur doit être dotée d'une interface de programmation comme est le cas de notre carte. L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux).
- L'environnement de la programmation : Le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce programme appelé IDE Arduino.

Structure générale du programme (IDE Arduino): Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation Arduino basé sur la programmation en C.



```
programme_de_reception_des_donn_e_via_bleutooth | Arduino 1.6.0
Fichier Édition Croquis Outils Aide

programme_de_reception_des_donn_e_via_bleutooth
#include <SoftwareSerial.h>

// parametre de configuration du bluetooth
#define RX 4
#define TX 2
SoftwareSerial BT(RX,TX);

void bluetooth();
void setup() {
  Serial.begin(9600);
  // parametre bluetooth
  BT.begin(9600);
  pinMode(RX, INPUT);
  pinMode(TX, OUTPUT);
  // put your setup code here, to run once:
}

void loop() {
  bluetooth();
}

String ph; // déclaration des variables globales
char mot[2];
int i=0, dispo;
long a=0;
void bluetooth(){
  dispo=BT.available();
  if(dispo>0){
    mot[i]=BT.read();
    ph=ph+ mot[i]; // pour effectuer une concaténation
  }
}
```

Compilation terminée.

Les variables globales utilisent 316 octets (15%) de mémoire dynamique, ce qui laisse 1 732 octets pour les variables locales. Le maximum est de 2 048 octets.

Arduino Uno en COM20 12:06 22/05/2018

Figure 7: Interface de programmation Arduino

3. Développement technique

3.1 La stratégie

La stratégie prévisionnelle : Ce projet étant personnel, notre première mission fut de définir nous-mêmes une stratégie prévisionnelle ainsi que les objectifs à atteindre. Bien entendu, cette stratégie a évolué au cours du temps afin de satisfaire nos exigences, mais aussi les contraintes auxquelles nous avons fait face. Nous avons ainsi tissé notre réflexion, avec pour fil conducteur la relation entre l'utilisateur et la voiture. Tout d'abord l'utilisateur : même si celui-ci ne fait pas partie à proprement parler du développement technique, il est primordial d'identifier le public concerné. Quel est le profil de l'utilisateur? Voilà le fondement de notre réflexion. Nous avons finalement privilégié une clientèle novice en développant une solution facile d'utilisation, la plus intuitive possible.

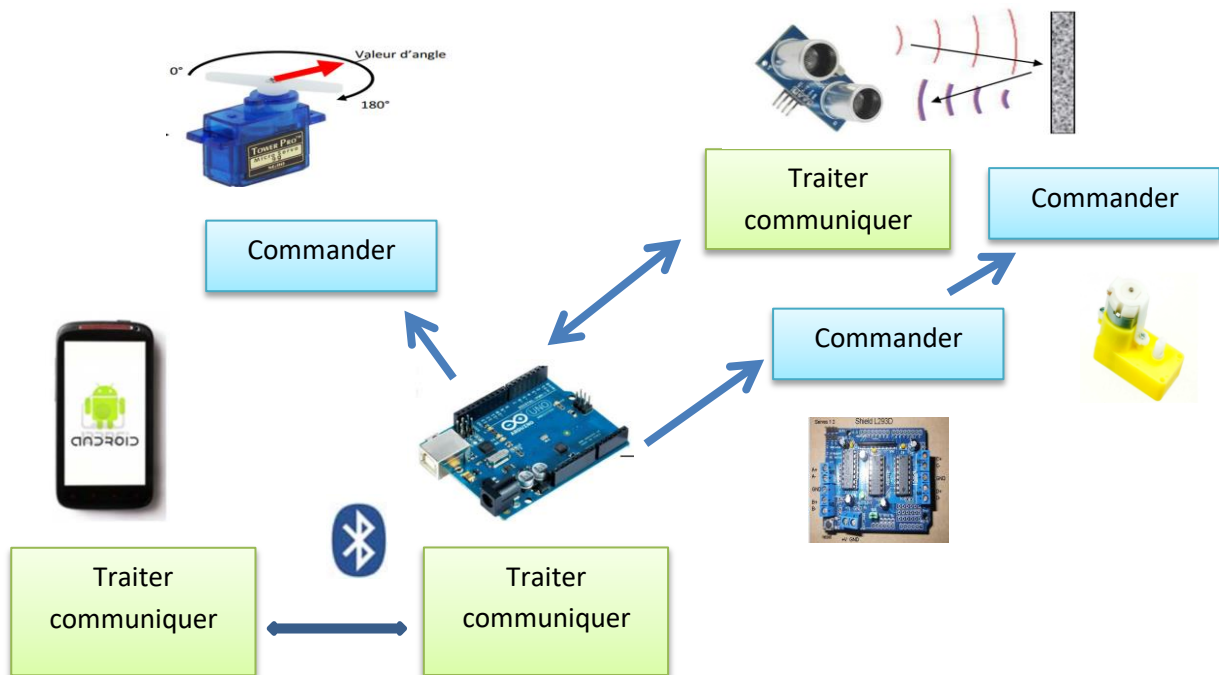
- À la manière d'une interface homme-machine, l'application Androïde permet d'interagir avec le système, sans nécessiter de connaissances pointues de la part de l'utilisateur pour être configuré et utilisé. Cette application inclut une vue d'ensemble du système, les boutons de commande de la voiture, et une zone de l'état à laquelle se trouve la voiture. Cette partie logicielle est installée dans un téléphone Androïde.
- Le microcontrôleur Arduino est le cœur du projet : il joue le rôle d'intermédiaire entre l'utilisateur et la partie mécanique. Grâce à une communication Bluetooth maître-esclave, la carte commande le fonctionnement de chaque dispositif embarqué à partir de la requête de l'utilisateur. Relié par une simple liaison série, le rôle de l'Arduino esclave est de commander le fonctionnement de chaque technologie.
- Enfin, nous avons agrémenté notre architecture domotique d'un capteur ultra son et d'un servomoteur qui nous permettent d'effectuer une rotation (servomoteur) selon l'angle imposé dans le programme et de mesurer la distance entre le capteur et un obstacle(capteur ultra son) pour une prise de décision.

Conclusion : L'utilisateur peut agir de deux manières différentes sur le système: en utilisant des commandes manuelles (considéré dans notre cas comme l'échange Bluetooth -Arduino) ou des commandes automatisées. Dans sa version finale, notre solution fournit un ensemble de fonctionnement contrôlables par l'utilisateur grâce à une application androïde et un autre en automatique.

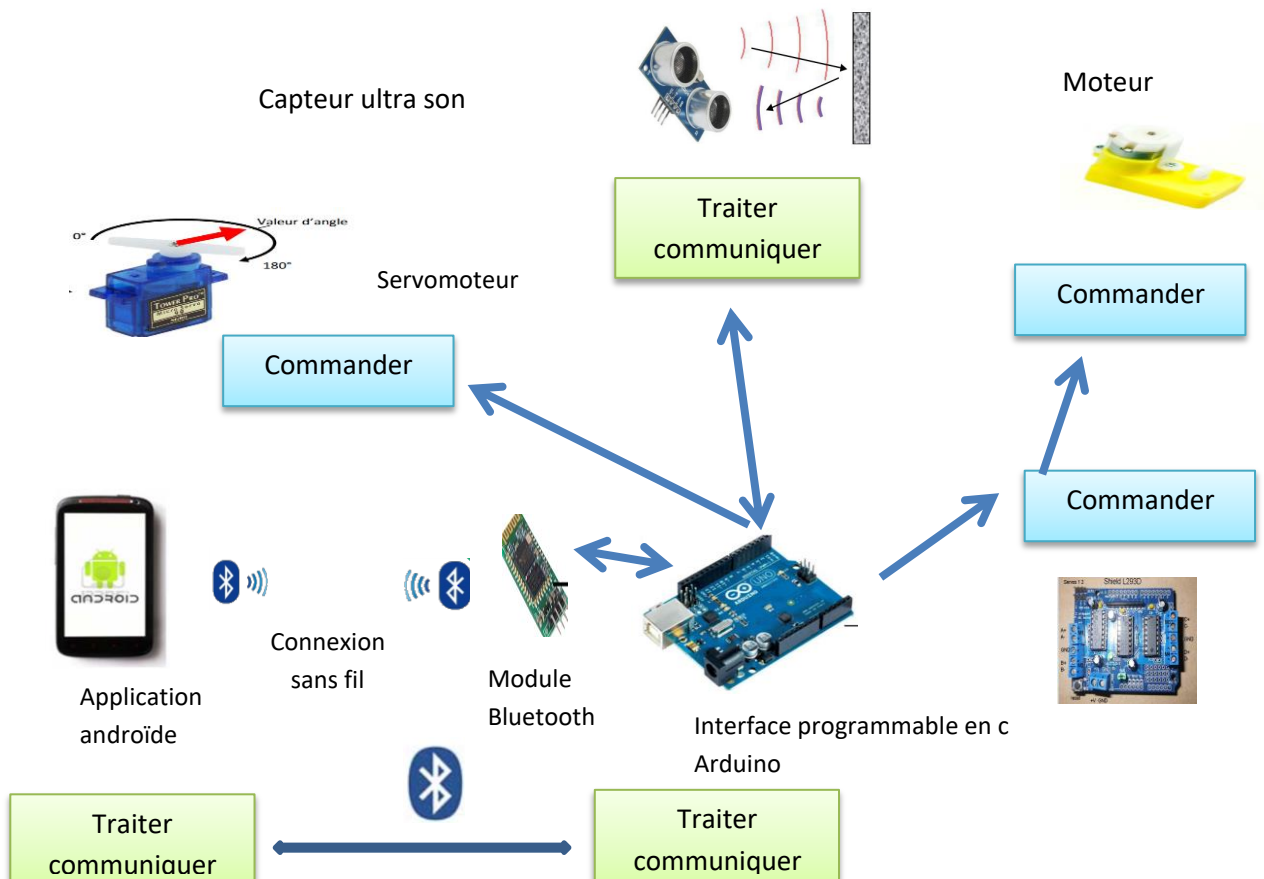
Projet robot détecteur d'obstacle

3.2 Architecture matérielle

Le diagramme ci-dessous résume notre paragraphe précédent sous la forme d'un schéma :



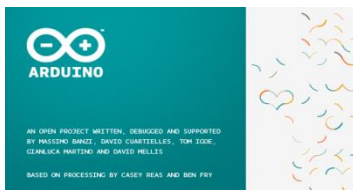
3.3 Architecture logicielle



3.4 Les solutions logicielles



- App Inventor est un logiciel de création d'applications Android (système d'exploitation pour smartphones et tablettes) . Les informations sont stockées sur des serveurs distants, ce qui nécessite une connexion Internet pendant tout le développement de l'application. Ce logiciel fonctionne par l'assemblage de blocs ce qui permet une prise en main instantanée facile mais se révèle être fastidieux avec de la pratique. Le gros avantage est que ce programme est gratuit et utilisable par tous à la maison qu'ils aient ou non une tablette ou Smartphone Androïde car un émulateur Androïde est présent. Le programme réalisé est testable immédiatement sur l'appareil Androïde de 2 façons possibles : connexion USB ou Wifi. Sans appareil Androïde, un émulateur Androïde est disponible sur le PC ou Mac.



- Un programme utilisateur Arduino est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne. La carte lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code. Une fois la dernière ligne exécutée, la carte revient au début et recommence sa lecture et son exécution des instructions successives. Et ainsi de suite dans une boucle infini appeler void loop(). Cette boucle se déroule des milliers de fois par seconde et anime la carte.



- Fritzing est d'un logicielle de conception des maquettes des circuits électroniques réel et fonctionnant correctement.

4. L'électronique embarquée

Dans cette partie, nous avons pour but de commander les moteurs et par conséquent les roues de façon automatique et de façon manuelle en utilisation un téléphone androïde. Nous nous sommes aussi attardés à rendre le robot sensible à son environnement à l'aide de capteur et de servomoteur. Nous présenterons le fonctionnement de chaque partie en quelque sorte et donnerons également leurs schémas électriques de branchement.

4.2 Pont en H avec la L293D

Dans un premier temps, pour permettre au robot d'avoir une grande mobilité (avancer, reculer, stop, tourner à droite ou à gauche), nous devons assurer la rotation des moteurs dans les deux sens, commandable par l'Arduino. C'est pourquoi nous avons utilisé la L293D. Le schéma électrique de branchement est le suivant :

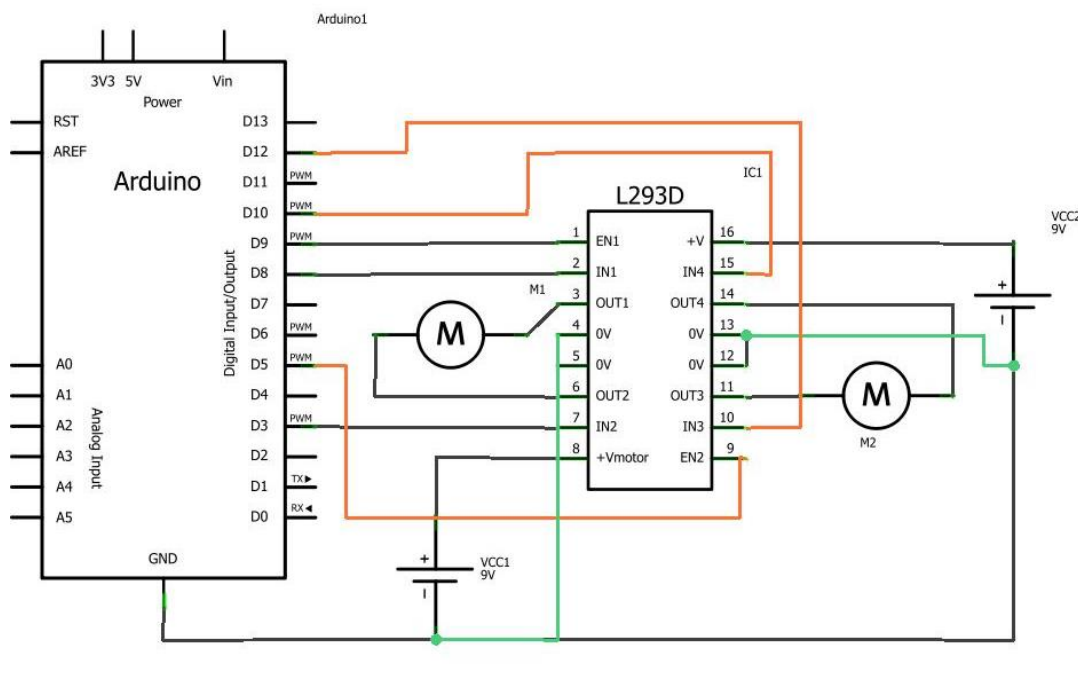


Figure 8:Schéma de branchement des moteurs

Explication du fonctionnement du schéma :

La L293D est circuit intégré contenant deux ponts à H permettant de commander un moteur. Ce circuit mobilise 2 pins de l'Arduino par moteur qui peuvent consommer chacun jusqu'à 1 ampère. Les circuits dédiés au contrôle des moteurs (motor shields) sont plus performants et plus pratiques à utiliser mais nettement plus chers. A noter que ce circuit permet aussi de commander des moteurs pas à pas. La reconnaissance de son brochage est essentielle mais notez qu'il n'est pas si compliqué dans la mesure où chaque rangée est symétrique et correspond à un pont en H, pour le contrôle d'un moteur. IN1 est la patte en entrée qui

Projet robot détecteur d'obstacle

contrôle la patte OUT1 en sortie qui est reliée à la borne + d'un moteur. IN2 commande la patte OUT2 reliée à la borne – du même moteur. Même principe pour le second moteur avec IN3 et OUT3 et IN4 et OUT4. La commande de vitesse des moteurs est réalisée en envoyant une valeur comprise entre 0 et 255 sur les pates EN du circuit (patte 1 pour les moteurs 1 et 9 pour le moteur 2). La commande de la vitesse des moteur s'effectue avec les broche PWM d'Arduino qui sont les broches 3, 5, 6, 9, 10, 11. Le programme de commandes des moteurs implémente le tableau de vérité du circuit suivant :

EN	IN1 (IN2)	IN3 (IN4)	Fonction
HIGH	LOW	HIGH	Tourne à droite
HIGH	HIGH	LOW	Tourne à gauche
HIGH	LOW	LOW	Arrêt rapide
HIGH	HIGH	HIGH	Arrêt rapide
LOW	Indifférent	Indifférent	Arrêt

Un programme correspondant à la commande des moteurs et implémentant la table de vérité précédente est le suivant :

```
#define EN2 5
#define A3 8
#define A4 9
#define EN1 6
#define A1 10
#define A2 11
void setup() {
  pinMode(EN2, OUTPUT);
  pinMode(A3, OUTPUT);
  pinMode(A4, OUTPUT);
  digitalWrite(A3, HIGH);
  digitalWrite(A4, LOW);
  analogWrite(EN2, 255);
  pinMode(EN1, OUTPUT);
  pinMode(A1, OUTPUT);
  pinMode(A2, OUTPUT);
  digitalWrite(A1, HIGH);
  digitalWrite(A2, LOW);
  analogWrite(EN1, 255);
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

. Maintenant que notre robot peut se déplacer, nous devons le rendre sensible à son environnement.

4.3 Schéma de branchement du servomoteur et principe de fonctionnement

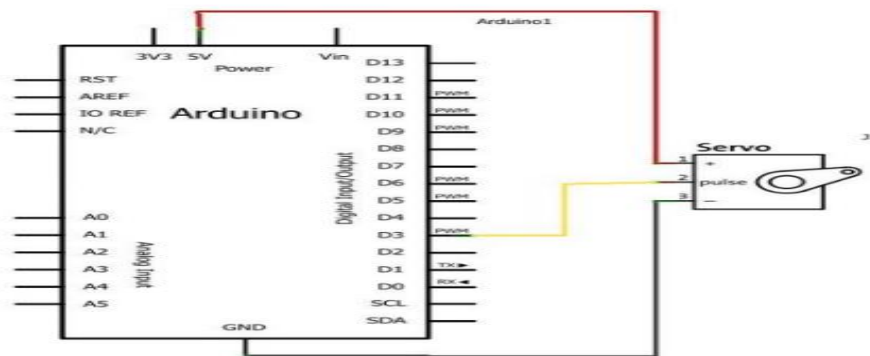


Figure 9: schéma de branchement du servomoteur

➤ Description

Le servomoteur est alimenté par le +5V et la masse de la carte Arduino, le pin D3 connecté à la patte 2(pulse) génère une impulsion dans le servomoteur d'où sa rotation. Le capteur ultrason est fixé sur notre servomoteur afin de roter suivant la direction que prendra le servomoteur. Le programme de commande du servomoteur est le suivant :

```
#include <Servo.h>
int pinservo=3;
Servo cerv;
void setup() {
  // put your setup code here, to run once:
  cerv.attach(pinservo); //définir le pin de commande du cerveau moteur
  // put your setup code here, to run once:
}

void loop() {
  servoMoteur(0); // mettre le cerveau moteur a la position 90 degré
  delay(500);
  servoMoteur(90);
  delay(500);
  servoMoteur(180);
  delay(500);
  servoMoteur(90);
  delay(500);

  // put your main code here, to run repeatedly:
}

// fonction permettant de faire tourner le cerveau moteur d'un angle donné
void servoMoteur(int angle) {
  cerv.write(angle); //pour faire tourner le cerveau moteur d'un angle a
}
```


4.5 Schéma de câblage final

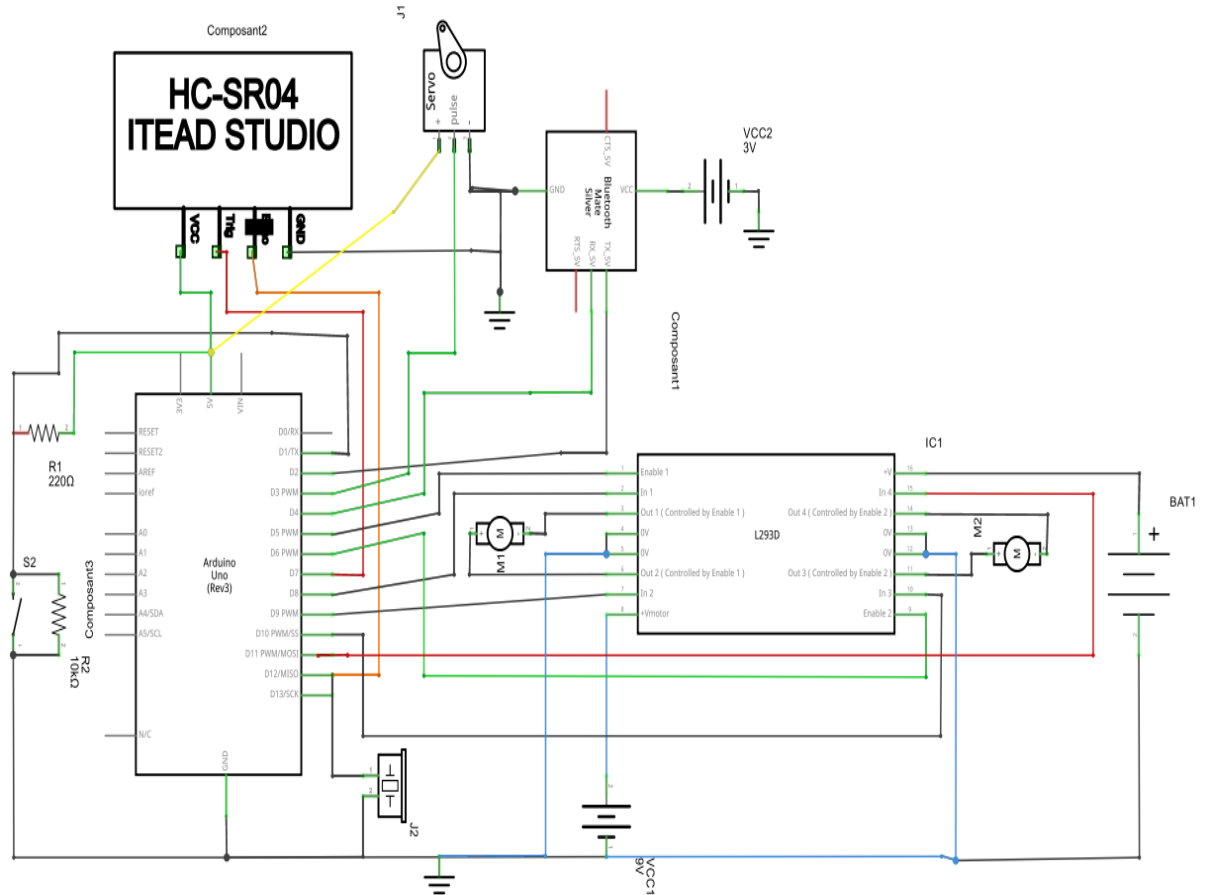


Figure 11: schéma de câblage final de la voiture

- Description

La carte Arduino ici est le chef d'orchestre car c'est elle qui pilote tous les autres modules. Le capteur ultrason accroché sur le servomoteur permet de mesurer la distance de détection d'obstacle sur un angle de 180 degrés et d'envoyer une information à la carte de commande. Suit au traitement de cette information par la carte de commande, le servomoteur adoptera un comportement spécifique (statique ou rotatif) afin de permettre au véhicule de ne pas entrer en collision avec un obstacle. Le circuit L293D permet le pilotage des deux moteurs en continu suivant une direction bien précise (défini par le résultat de traitement de l'information reçue par la carte de commande). L'interrupteur S2 permet d'activer le fonctionnement du mode automatique.

4.6 Test de fonctionnement

Une fois que toutes les parties essentielles ont été complètes, nous sommes passés à la phase de test. D'abord, nous avons décidés de tester chaque partie individuellement. Les moteurs, dans un premier temps, en essayant de les faire tourner ensemble, avec des vitesses différentes et dans les sens différents par rapport au réglage sur le programme. Ensuite, nous avons travaillés sur la commande du servomoteur en définissant des différents angles de rotation dans le programma de commande. Par la suite nous avons branchés le capteur ultrason à la carte Arduino pour effectuer des prises de distances par rapport à un obstacle. Nous avons synchronisés le mouvement rotation du servomoteur à celui de l'ultrason puis à celui du mouvement des moteurs. Et, finalement, le fonctionnement du capteur de distance synchronisé aux LED rouge en arrière pour gérer la proximité d'un obstacle. Enfin, nous avons essayé de simuler le mouvement complet du robot, avec toutes ses fonctions actives et entrain de travailler ensemble. De cette façon, nous avons eu la

Possibilité d'observer et d'analyser la performance du véhicule, trouver les problèmes éventuels et les résoudre.

4.7 Observation lors du fonctionnement de la maquette

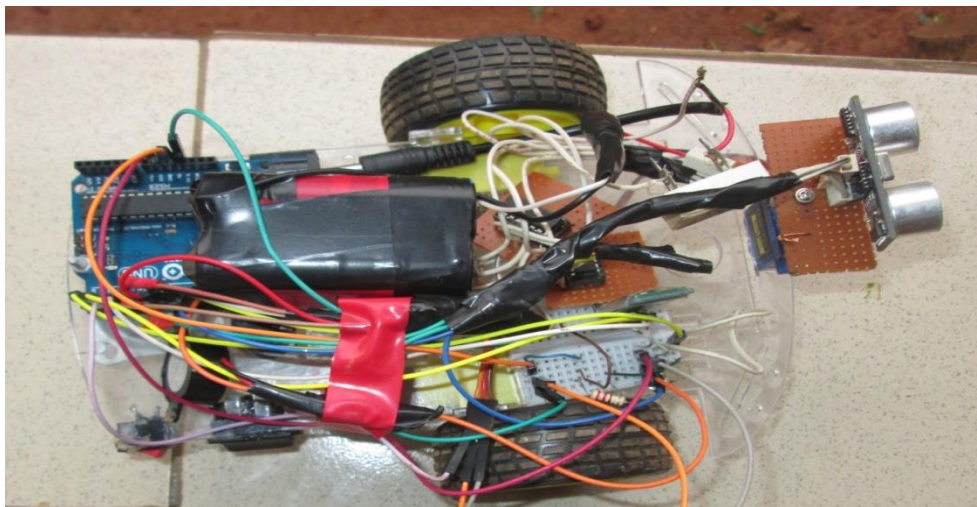


Figure 12: maquette de la voiture

Nous avons constaté que la commande via du Bluetooth n'est pas instantané et qu'il y'a un légère retard au niveau de l'envoi de la commande et de l'exécution de la commande. de même il se déconnecte constamment.

Lorsque la carte Arduino n'est pas suffisamment alimentée, elle exécute les instructions difficilement et saute d'autre. Ses composants embarques ont du mal à exécuter leurs tâches.

Lorsqu'on alimente les deux moteurs par une source de tension commune, soit un accumulateur il y'a une différence de vitesse de rotation des deux moteur lorsqu'on demande

aux deux moteurs de tournée à la même vitesse, par contre avec une alimentation externe de 12V en continu les deux ont les mêmes vitesses.

Au niveau du fonctionnement du capteur ultra son. Il se comporte parfois comme si il n'était pas en fonctionnement active dans le mode automatique et pourtant dans la programmation le fonctionnement de la distance est mesuré à chaque instant, ce qui nous amène à nous poser des questions sur l'inconvénient de l'utilisation d'un seul capteur ultra son pour le fonctionnement d'un tel dispositif et son inconvénient en elle-même.

Au niveau du fonctionnement du buzzer, nous avons remarqué que lorsque la voiture est mise en fonctionnement automatique, il crie normalement à la rencontre du choc, mais en fonctionnement manuelle on n'entend presque pas le son dégager, la question que nous nous sommes posés est de savoir si le Bluetooth impacte sur le fonctionnement de celui-ci car il est clair que ça joue au moins un rôle.

Lors du fonctionnement de la carte Arduino elle chauffe et on se pose la question de savoir comment faire pour limiter la tension ou le courant fourni pas la carte en utilisant une solution plus adéquate pour limiter ce problème.

4.8 Difficultés rencontrées

- Comportement inattendu du capteur ultra-son : lors des premiers tests de fonction de la voiture, le capteur détectait des obstacles à tous les niveaux et pourtant il n'y avait pas d'obstacle, pour pallier à ce problème nous avons choisi d'augmenter la distance seuil et de faire des tests dans un environnement parfaitement lisse, horizontal et sans bavure.
- Problème d'alimentation : les batteries à notre disposition ne parviennent pas à alimenter suffisamment la carte pour le fonctionnement des composants qui prennent leur source d'énergie sur la carte puis fonctionnent normalement, pour faire face à ce problème nous avons utilisé les batteries au lithium provenant des batteries de laptop que nous avons mises en série pour avoir une bonne tension à près de 14V.
- Les batteries 9V prissent sur le marché n'alimentent pas la carte Arduino pour longtemps et se déchargent complètement après une utilisation d'en moyenne 5min. pour y faire nous avons adopté la solution précédente.
- Récupération des données reçues par la carte et traitement. Lors de la réception du message avec la carte Arduino, le code que nous avons rédigé au préalable récupérait le mot envoyé avec les caractères supplémentaires ce qui nous empêchait de faire des tests afin d'exécuter l'instruction correspondante au message reçu. Pour pallier à ce problème nous avons demandé de l'aide pour le code, la solution était d'effectuer une concaténation à l'or de la réception. après une parfaite récupération, un autre problème est encore survenu, celui lié au test d'instruction, lors de la concaténation il y'avait le caractère de fin de phrase qui s'ajoutait au mot et une fois de plus nous étions encore bloqué, pour contourner cette nouvelle difficulté, nous avons effectué un test sur le caractère marquant la fin de la phrase avant de faire une concaténation dans le cas où le test était faux.

➤ Après avoir effectué tous les tests nous avons constaté que les deux moteurs ne tournaient pas à la même vitesse lorsqu'on utilise une même alimentation pour commander les moteurs et la carte Arduino. En provoquant une différence de vitesse dans le programme de commande des moteurs nous avons espéré compenser cette différence sans succès, en diminuant au max la vitesse de rotation des moteurs nous avons observé une légère compensation de différence.

5. Bilan du projet

5.1 Apports individuels et collectifs

➤ NAFACK BAURICE : la réalisation de ce projet ma permit de travailler en groupe et de comprendre qu'un travail en équipe permet de connaitre ces limites en terme de connaissance et de réflexion. Tout au long de ce travail j'ai eu l'occasion d'élargir mes connaissances non seulement dans l'environnement de programmation Arduino mais aussi en termes de composant électronique. ça m'a aussi permis d'élaborer un plan de mise en œuvre, de manipuler ces outils et de comprendre que la question d'un projet n'est pas du tout facile surtout quand il y'a plusieurs facteurs externes qui influencent le bon déroulement du projet. J'ai également remarque qu'un projet sans plan de mise en œuvre peut être un désordre totale.

➤ MATZE LAURA : j'ai eu à découvrir certains matériaux électroniques avec lesquels je me suis familiarisé en quelque sorte. Aujourd'hui, pour moi réaliser un robot n'est plus trop magique, je sais que c'est possible en utilisant des matériaux et des langages nécessaires pour la programmation. En plus de tout ça j'ai aussi gagné en expérience.

➤ LOH BORIS-EDMOND BOHTEH : à partir de ce projet, j'ai eu la possibilité d'étudier de nouveaux composants électroniques, la façon de travailler en groupe ma beaucoup impressionner car ils m'ont montré qu'on peut réaliser un projet en électronique sans faire au préalable des simulations mais en se base sur la data Sheet des composants on fait des montages qui marche, la façons dont nous collaborions pour faire des recherches sur internet ma faire découvrir de nouvelle horizon de recherche sur internet.

➤ FOKOU TCHINDA ADOLPHE HITLER, Ce projet fut pour moi très captivant cars j'ai eu à manipuler de nouvelles composantes électroniques, l'ambiance du déroulement du projet quant à lui était émouvant du fait de la participation active des différents membres. Toujours partant d'ici j'ai eu une vision plus claire de ce qu'est vraiment la domotique et envisage continuer avec la réalisation des projets d'intelligences artificielles.

➤ BASSO TSAKENG FRANCK, à travers ce projet, j'ai appris à piloter les moteurs DC avec le circuit L293D ; à manipuler les nouveaux composants électroniques, à programmer avec la carte Arduino, j'ai également eu l'occasion de découvrir le logicielle FRITZING.

5.2 Liste des composants utilisés pour la réalisation du projet

Matériels	Prix unitaire(en FCFA)	Nombre	Prix total(en FCFA)
Fil de connexion	50 Fr le mètre	8 mètres	400
Batterie de 9V	200	4	800
Pile de 1.5V	200	6	1200
Plaque à souder	250	1	250
Support pour le L293D	50	4	200
Plaque à essai	1500	1	1500
Tête de batterie	300	2	600
Alimentation de la carte Arduino	300	1	300
Carte Arduino Uno	7000	1	7000
Schild motor (L293D)	7000	1	7000
LED	25	4	100
Chassie de la voiture	8000	1	8000
Kit Bluetooth	4000	1	4000
Plaque à essai	1500	1	1500
Buzzer	500	1	500
Résistances de 220Ω	50	4	200
Scotch	200	1	200
Connexion internet		-//-//-/-	3000
Transport	2000	-//-//-/-	2000
TOTAL	35650	40	41450

Conclusion

La réalisation de ce projet a été une véritable expérience pour nous, ensemble nous avons pu réaliser ensemble un projet de domotique qui pourra à la longue être fait à plus grande échelle pour aider la population du Cameroun en réduisant le taux de mortalité que l'on enregistre dans le triangle de la mort lié à la cause humaine. Nous avons constaté qu'un travail en groupe est plus promoteur qu'un travail individuel lorsque tous les membres du groupe y prennent part avec enthousiasme.

D'autre part, la domotique a cette particularité de rassembler divers corps de métiers : programmation, informatique, électronique, électrotechnique, et bien d'autres. Cette pluridisciplinarité fut un obstacle que nous avons surmonté grâce aux compétences multiples et à la polyvalence de notre équipe. Cette plurivalence a notamment été mise à profit dans l'attribution des tâches afin que chacun tire parti de ses accomplissements.

Enfin, les temps impartis à la réalisation du système furent brefs et il a fallu faire preuve de flexibilité et de persévérance, parfois pour respecter les délais, parfois pour respecter les contraintes technologiques imposées par le projet. Nous avons retrouvé lors de ce mois, les compétences, les contraintes mais aussi l'excitation d'un projet d'entreprise.

BIBLIOGRAPHIE

- Read more: <https://www.neozone.org/auto-moto/les-voitures-autonomes-quels-enjeux/#ixz5Cv1C7mkI>
- <https://www.neozone.org/auto-moto/les-voitures-autonomes-quels-enjeux/>
- <http://idehack.com/blog/configurer-le-module-bluetooth-hc-06/>
- fiche f4 – commandé des servomoteurs classiques et a rotation continue
- fiche f3 – controler des moteurs et diriger un robot
- <https://actu cameroun.com/2017/09/07/cameroun-deja-4190-accidents-de-la-circulation-en-2017-gouvernement/> (consulter le 15/04/2018)
- PROJET DOMOTIQUE DAIO « Domotique All-In-One » *Projet réalisé par* Paul MEGUEULE, Fabien NOIR, Nicolas ROULLEAU, Florian ROUSSEL
- <https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>
- <http://appinventor.mit.edu/explore/ai2/setup-emulator.html>
- <http://ai2.appinventor.mit.edu>
- <http://appinventor.mit.edu/explore/designer-blocks.html>
- <http://idehack.com/blog/configurer-le-module-bluetooth-hc-06/>
- <http://www.instructables.com/id/Arduino-AND-Bluetooth-HC-05-Connecting-easily/>
- Livre Arduino pour bien commencer en électronique et en programmation Par Astalaseven , Eskimon et olyte www.siteduzero.com
- Livre Arduino : Premiers pas en informatique embarquée Auteurs : Hippolyte Weisslinger (olyte)