Université Paris 8

Université Paris-Ouest
Nanterre

Développement de module de contrôle d'une main robot

Rapport de stage

Auteur :
ABAK-KALI Nizar
Numéro d'étudiant :
11290569
Numéro LIT :
C621

Université Paris 8

Université Paris-Ouest
Nanterre

Développement de module de contrôle d'une main robot

Rapport de stage

Auteur :
ABAK-KALI Nizar
Numéro d'étudiant :
11290569
Numéro LIT :
C621

Responsables :
M. Nicolas JOUANDEAU
M. Adrien REVAULT D'ALLONES
Numéro du Laboratoire :
01.49.40.70.44

Remerciements

Il n'est pas aisé pour un étudiants de 3ème année de licence de trouver un stage, c'est pourquoi je remercie le laboratoire d'informatique avancé de Saint-Denis de m'avoir accueillie durant ce mois et demi.

Je tiens à remercier tout particulièrement M. Nicolas JOUANDEAU ainsi que M. Adrien REVAULT D'ALLONES, qui m'ont accordés leurs confiances et attribués une tâche aussi importante durant ce stage.

J'aimerais aussi les remercier d'avoir pris le temps de me superviser, ainsi que de m'avoir tant appris durant mon stage.

Faire mon stage de dernière année dans mon université a été un bonheur, j'ai pu apprendre beaucoup grâce à vous, et surtout, je me suis beaucoup amusé ce qui est, à mes yeux, le plus important.

Table des matières

Introduction **4**

1 Problématique **5**

1.1 Objectif 5

1.2 Liaisons avec les deux autres projets 5

2 Organisation du temps de travail **6**

2.1 Diagramme 6

2.2 Explications 7

3 Solutions existantes **9**

3.1 Projet : InmoovHand 9

3.2 Plateforme ROS et Arduino 10

3.2.1 ROS : Robot Operating System 10

3.2.2 Arduino 10

3.3 Autre type de main : la shadow-harm 11

4 Solutions proposées **12**

4.1 Sous problématiques 12

4.1.1 Contrôle d'un servomoteur classique 12

4.1.2 Contrôle d'un servomoteur série 13

4.1.3 Module de contrôle ROS 13

4.1.4 Communication inter-module 15

4.2 Application pratique 15

4.2.1 Comparaison servomoteur classique/servomoteur série . 15

4.2.2 Architecture matériel du servomoteur série 16

5 Expérimentations **18**

5.1 Montage 18

5.2 Protocole 20

6 Conclusion **22**

Introduction

Le stage m'as été proposé par un professeur de l'université de Paris 8, M. JOUANDEAU Nicolas dans le cadre d'une collaboration entre deux laboratoire, celui de paris 8 le LIASD [1], et le LEME [2] un laboratoire de l'université de Paris Ouest.

Étudiant en troisième année de Licence Informatique , j'ai effectué mon stage de fin d'étude dans le LIASD, avec pour mission de développer des modules pour contrôler une main robot .

Ce stage m'as permis de découvrir le travail au sein d'un laboratoire d'expertise en robotique. Pour renforcer mon choix d'étude future, j'ai souhaité découvrir l'environnement au sein d'un laboratoire de recherche en informatique.

J'ai choisi en priorité le LIASD car il est réputé dans le secteur de la robotique, domaine qui m'a toujours intéressé .

L'enjeu de ce stage était de découvrir l'univers de la recherche, mais aussi dans un second temps, il était d'élargir mes compétences en incluant cette expérience avec la robotique .

Au sein du LIASD, j'ai tenu le poste de développeur et mes missions au cours du stage ont été les suivantes :

— me former pour pouvoir développer sur les plateformes ROS [3] et Arduino [4]
— trouver quel moteur conviendrait le mieux à la main robot
— développer des modules de contrôles pour ces moteurs
— développer une interface de contrôle des moteurs pour lier ce projet et un autre projet

Le but final du stage est de pouvoir téléopérer une main robot.

Ce rapport explique dans un premier lieu le déroulement du stage, puis dans un second temps les solutions déjà existantes. Puis le rapport présente les solutions proposé pour répondre à la problématique .

1. Laboratoire d'Informatique Avancée de Saint-Denis
2. Laboratoire Énergétique Mécanique Électromagnétique
3. Robot Operating System cf. partie 3 Solutions existantes pour plus d'info
4. cf. partie 3 Solutions existantes pour plus d'info

1 Problématique

1.1 Objectif

Lors de ce stage l'un des objectif est de mettre en place tout les outils qui permettrons de télé-opérer une main robot.
Or le sujet en question s'articule en deux sous-questions : "comment téléoperer une main robot ?", et ensuite dans un second temps :"Quels outils donner un utilisateur pour qu'il puisse utiliser la main robot ?".
Ces deux problématique donnerons lieu à deux stage de fin d'étude, celui présenté par ce rapport, et un autre fait par un collègue de 3e année .
Dans ce rapport nous traiterons de la manière de commander la main robot. Plusieurs sous-problématique résulte de cela, dans un premier temps il est primordial de ce poser la question suivante : "quelle type de main robot utiliser ?". Ensuite dans un seconds temps, on viens à se poser la question de "comment la fabriquer ?", c'est à dire quel matériel , quel composant. Enfin, comment contrôler cette main ?

1.2 Liaisons avec les deux autres projets

Ce projet est couplé à deux autres projet : le premier consistant à porter tout paquet qui peut être nécessaire, le second est complémentaire aux projet présenté dans ce rapport, il s'agit de contrôle de main robot par reconnaissances de mouvement de la main par caméra RGB-D [5]. Ainsi la finalité du projet est utilisateur qui fait des gestes de la main devant une caméra RGB-D, ces mouvements sont interpréter, puis envoyer à la main robot qui mime les mouvements de l'utilisateur.

5. caméra qui en plus de capté la lumière et les couleurs peut aussi retranscrire la profondeur d'où le "D" dans RGB-D, ici le "D" correspond "depth" qui signifie profondeur

Déroulement du stage

Cette partie présente le déroulement du stage

2 Organisation du temps de travail

2.1 Diagramme

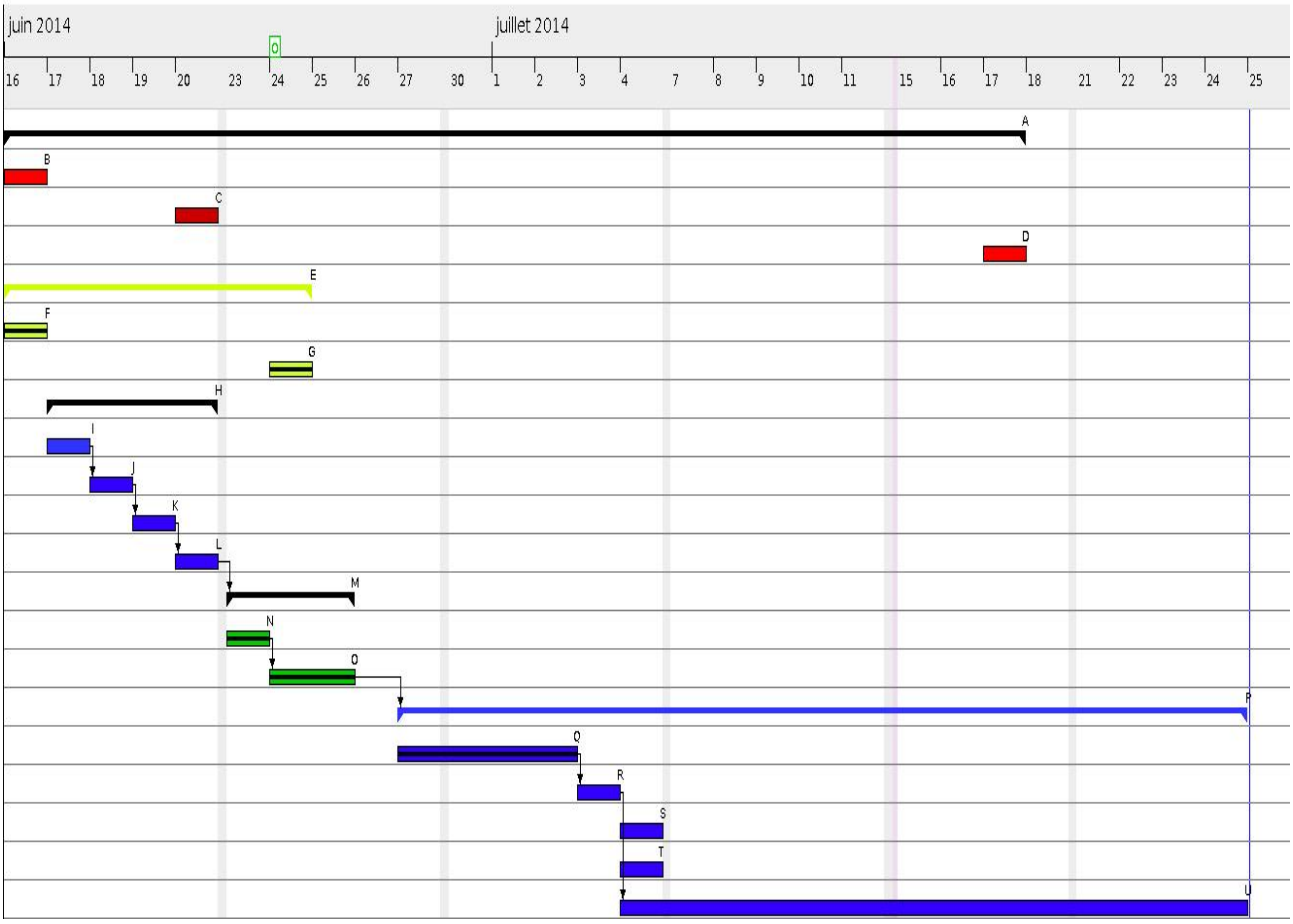

FIGURE 1 – Diagramme de Gantt présentant le déroulement du stage

2.2 Explications

cf. page suivantes pour info

Lettre	Nom	Date de début	Date de fin
A	Réunions	16/06/14	17/07/14
B	rencontre avec l'équipe	16/06/14	16/06/14
C	première réunion avec les clients	20/06/14	20/06/14
D	deuxièmere réunion : présentation des résultats aux clients	17/07/14	17/07/14
E	Installations	16/06/14	24/06/14
F	installation de ROS Fuerte	16/06/14	16/06/14
G	Installation IDE Arduino	24/06/14	24/06/14
H	Prise en main de ROS Fuerte	17/06/14	20/06/14
I	création des premiers noeud	17/06/14	17/06/14
J	création des premiers services	18/06/14	18/06/14
K	création des premiers publisher/subscriber	19/06/14	19/06/14
L	rédaction de la documentation sur ce qui a été fait	20/06/14	20/06/14
M	Etude de solutions possibles	23/06/14	25/06/14
N	étude de la InmoovHand	23/06/14	23/06/14
O	prise en main d' Arduino	24/06/14	25/06/14
P	Phase de développement	27/06/14	24/07/14
Q	développement des premiers modules pour servomoteur classique	27/06/14	02/07/14
R	réception de nouveaux matériels : servomoteur série AX-12	03/07/14	03/07/14
S	réception du prototypes experimentales de main robot	04/07/14	04/07/14
T	réception de deux capteurs	04/07/14	04/07/14
U	Développement de module Arduino pour les servos moteurs series	04/07/14	24/07/14

TABLE 1 – Tableau explicatif du diagramme ci-dessus

3 Solutions existantes

Cette partie présente les solutions déjà existantes .

3.1 Projet : InmoovHand

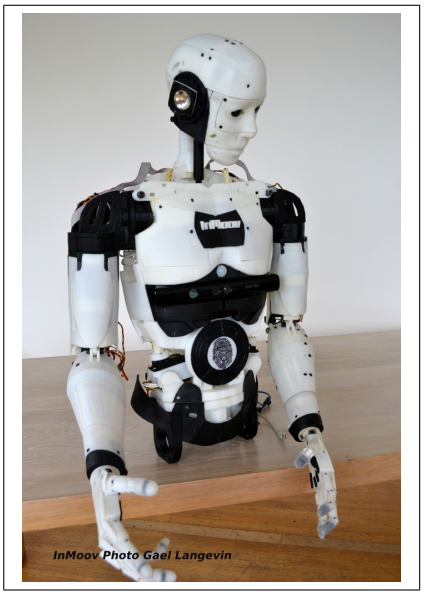

FIGURE 2 – Image du robot Inmoov

Le projet "InMoov" est le premier robot Open Source à taille humaine entièrement imprimable en 3D . Le projet présent en ligne donne tout les outils pour fabriquer chaque partie du robot (modèle 3D, tutoriels ,programmes pour faire marcher le robot , outils de développement ...) . Lors de notre projet nous nous intéresserons particulièrement à la main du robot .

3.2 Plateforme ROS et Arduino

3.2.1 ROS : Robot Operating System

FIGURE 3 – Ci-dessus logo de la dernière version de ROS , INDIGO

ROS ou "robot operating system" est un système d'exploitation pour robot sous licence open source permettant de contrôler un robot . Il donne par la même occasion tout les outils nécessaires aux développement de modules pour ces derniers .

3.2.2 Arduino

FIGURE 4 – Logo Arduino

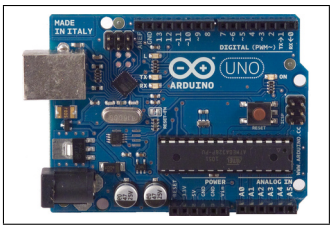

FIGURE 5 – carte Arduino Uno

Arduino est un circuit imprimé en matériel libre [6] sur lequel se trouve un microcontrôleur qui peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique [7], le pilotage d'un robot, etc. C'est une plateforme basée sur une interface entrée/sortie simple. Arduino peut être utilisé pour construire

6. dont les plans de la carte elle-même sont publiés en licence libre mais dont certains composants sur la carte, comme le microcontrôleur par exemple, ne sont pas en licence libre

7. le contrôle des appareils domestiques - éclairage, chauffage ...

des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur pour communiquer avec ses logiciels (ex. : Macromedia Flash, Processing, Max/MSP, Usine Hollyhock, Pure Data, SuperCollider).

3.3 Autre type de main : la shadow-harm

FIGURE 6 – La Shadow-Hand

FIGURE 7 – Comparaison Shadow Hand/main humaine

La Shadow Dexterous Hand est une main robot à forme humaine développer par la compagnie The Shadow Robot Company présente à Londres. La main est comparable à une main humaine aussi bien dans la forme, que dans la taille, et reproduit tout les degrés de liberté de mouvement d'une main normale.

4 Solutions proposées

4.1 Sous problématiques

4.1.1 Contrôle d'un servomoteur classique

Un servomoteur [8] 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.

FIGURE 8 – Image d'un servo classique

Pour le contrôler ce dernier il faut le raccorder à un microcontrôleur, comme ceci :

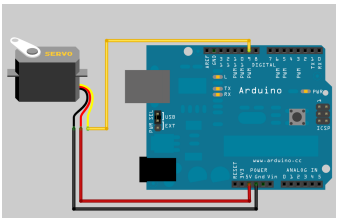

FIGURE 9 – Image d'un servo classique relié à un microcontrôleur Arduino

8. ou aussi appelé "servo"

4.1.2 Contrôle d'un servomoteur série

Un servomoteur série est un servo moteur que l'on peut connecter à d'autres servomoteur série créant ainsi une chaîne .Chacun de ces servos, une fois connecté, a un identifiant propre, et peut recevoir des données par le bus.

FIGURE 10 – Servo série AX-12A de face

FIGURE 11 – AX-12 vue de dos avec les ports entrés et sortie

Et voici un schéma montrant à quoi ressemble l'interconnections entres servos séries :
L'ordre dans lequel sont reliés les servos n'a pas de rapport avec leurs iden-

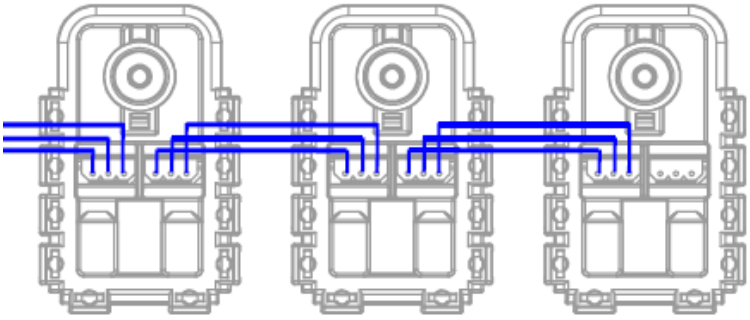

FIGURE 12 – Schéma de servos séries relier entre eux

tifiants. Chaque servo est configuré avec son propre identifiant propre.

4.1.3 Module de contrôle ROS

Cette partie explique comment le module ROS contrôle une carte Arduino et active les moteurs.

Dans premier temps, on utilise l'IDE de Arduino auquel on ajoute les librairie ROS. Ensuite on programme directement sur l'IDE Arduino.

Ci-dessous un module général de contrôle de servo avec ROS/Arduino

```
#include <Servo.h>
#include <ros.h>
#include <std_msgs/UInt16.h>

ros::NodeHandle  nh;

Servo servo;

void servo_cb( const std_msgs::UInt16& cmd_msg){
  servo.write(cmd_msg.data); //positionner l'angle du servo , l'angle se situe d
  digitalWrite(13, HIGH-digitalRead(13));  //allumé la led
}


ros::Subscriber<std_msgs::UInt16> sub("servo", servo_cb);


void setup(){

  pinMode(13, OUTPUT);

  nh.initNode();
  nh.subscribe(sub);

  servo.attach(9); //attacher le servo au pin 9 de la carte Arduino
}

void loop(){
  nh.spinOnce();
  delay(1);
}
```

4.1.4 Communication inter-module

Lors de ce stage deux projets sont destinés à être couplés, celui-ci qui consiste à contrôler les moteur à partir de module ROS/Arduino, et un autre qui consiste à de la reconnaissance des mouvement de la main à l'aide d'une caméra RGB-D [9]. Ainsi les deux modules communiquent grâce à ROS. Au final, on contrôle la main robot à l'aide d'une caméra.
Dans le système ROS, deux modules peuvent communiquer grâce a une interface de communication spécifique à ROS. Ce système s'organise autour de deux structures : "les publishers", modules qui écrivent des messages, et "les subsribers" modules qui lise ces messages. Pour que ces deux modules communiquent, il faut qu'il s'inscrivent à un "topic" qui est un lieu ou tout les messages sont publiés .
Donc dans le cas de notre projet le module récupération des gestes de la main publie sont interprétations puis le module de controle de la main récupère les messages les interprètes puis fait bouger la main en conséquences .

4.2 Application pratique

4.2.1 Comparaison servomoteur classique/servomoteur série

Cette partie présente les différences entres un servomoteur classique et un servomoteur série. Les deux types de servos ont déjà été présentés dans des parties précédentes.
Comme expliqué dans les parties précédentes, les servos séries peuvent êtres reliés ensemble, et ainsi il n'ont besoin que d'un port de sortie de donnés PWM [10].Contrairement au servo série, le servo classique à besoin d'autant de port qu'il y a de servos à commandé .
Donc :

pour N servo classique => N port entré, N port sortie et une seul alimentation en courant
pour N servo série => 1 port entré, 1 port sortie et une seul alimentation en courant

9. Red Green Blue - Depth, une caméra qui possède une matrice de profondeur
10. PWM = pulse width modulation , ou autrement appelé signal carré est un signal discret utilisé par les microcontrôleur pour communiqué avec les composants

	Servomoteur série AX-12A	Servomoteur Classique HSR-1422 CR
POIDS (g)	55	41,7
VOLTAGE DE FONCTIONNEMENT(V)	7-10	4,8-6,0
COUPLE MAX(kgf.cm)	7V->12 || 10V->16,5	2,8
TEMPÉRATURE DE FONCTIONNEMENT (Celcius)	-5 to +85	-20 to +60

TABLE 2 – tableau comparatif entre un servo série et un classique

Après expérimentation, nous avons préféré le servo série pour sa capacité à se mettre en série, pour sa vitesse très élevé, et pour sa force de pression .

4.2.2 Architecture matériel du servomoteur série

Cette partie présente l'architecture du servo série AX-12. Pour ce faire nous allons donner une liste :

1. Contrôle de la précision : la position et la vitesse peut être controler avec une résolution de 1024 pas.
2. Feedback : le retour d'information pour la position angulaire, la vitesse angulaire, et le couple de charge est disponible.
3. Communication : la communication est très aisé grâce à la structure en chaîne et cette communication peut aller jusqu'à 1M BPS [11]
4. la LED de status : cette LED permet d'indiquer les erreurs de statut à l'utilisateur.

11. Bauds per second : nombres de caractères transmissible par seconde

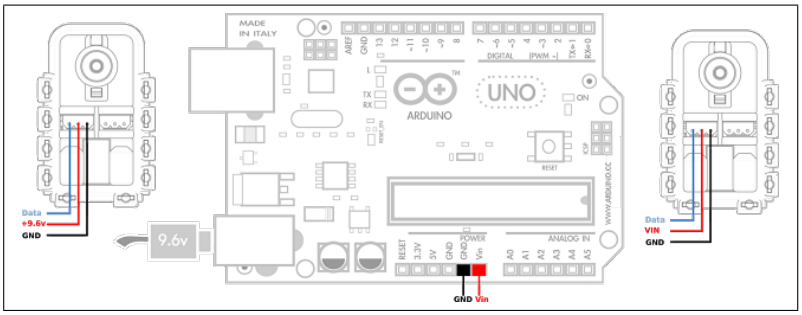

FIGURE 13 – Schéma qui représente une la connexion d'un servo série AX-12 à une carte Arduino Uno

Le servo AX-12 possède trois entres par port, le bleu qui contient les données ,le rouge la tension et le noir est la terre . Le second port de droite est celui de transmission à un autre servo série.

5 Expérimentations

Cette partie présente les montages ainsi que les résultats d'expérimentation.

5.1 Montage

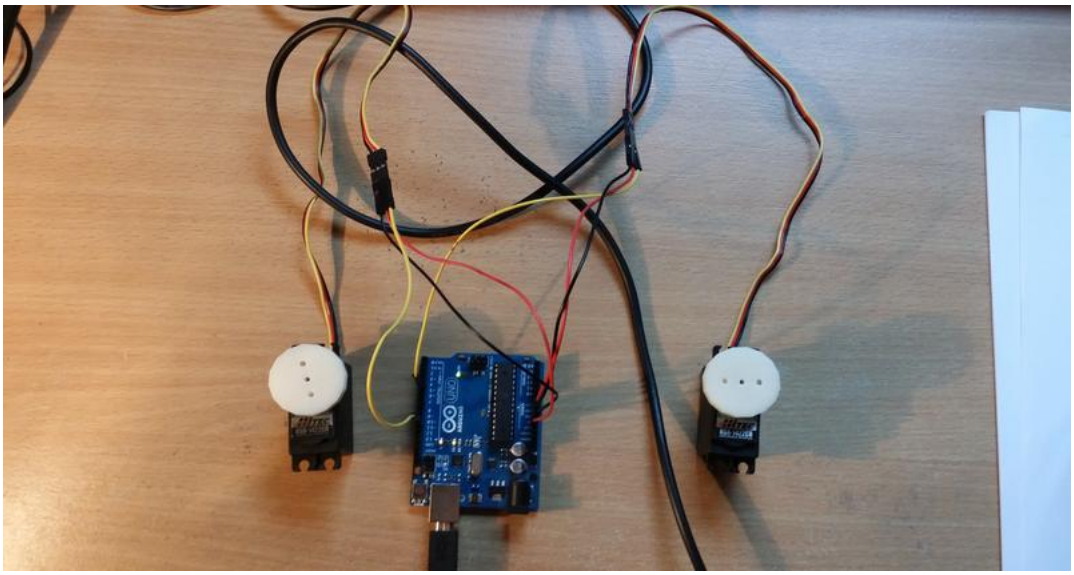

FIGURE 14 – Montage d'une carte arduino UNO et d'un servo moteur simple

Le premier montage consiste en deux servomoteurs classiques relié à une carte Arduino

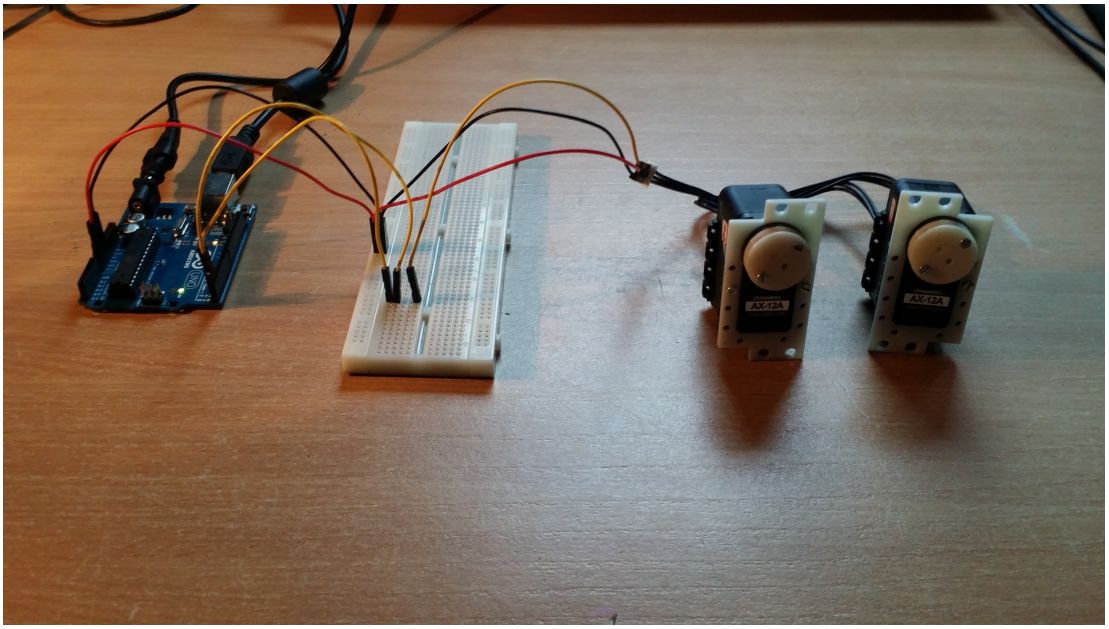

FIGURE 15 – Montage deux servomoteurs série AX-12 vue de face

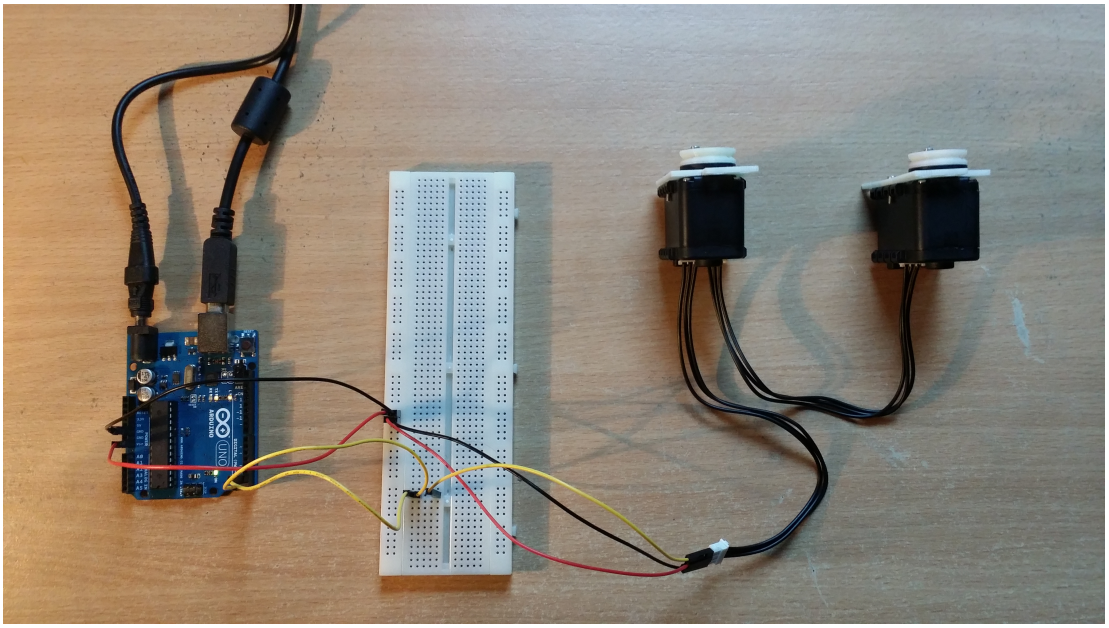

FIGURE 16 – Montage deux servomoteurs série AX-12 vue du dessus

Le montage ci-dessus montre comment on monte les deux servos séries ensemble

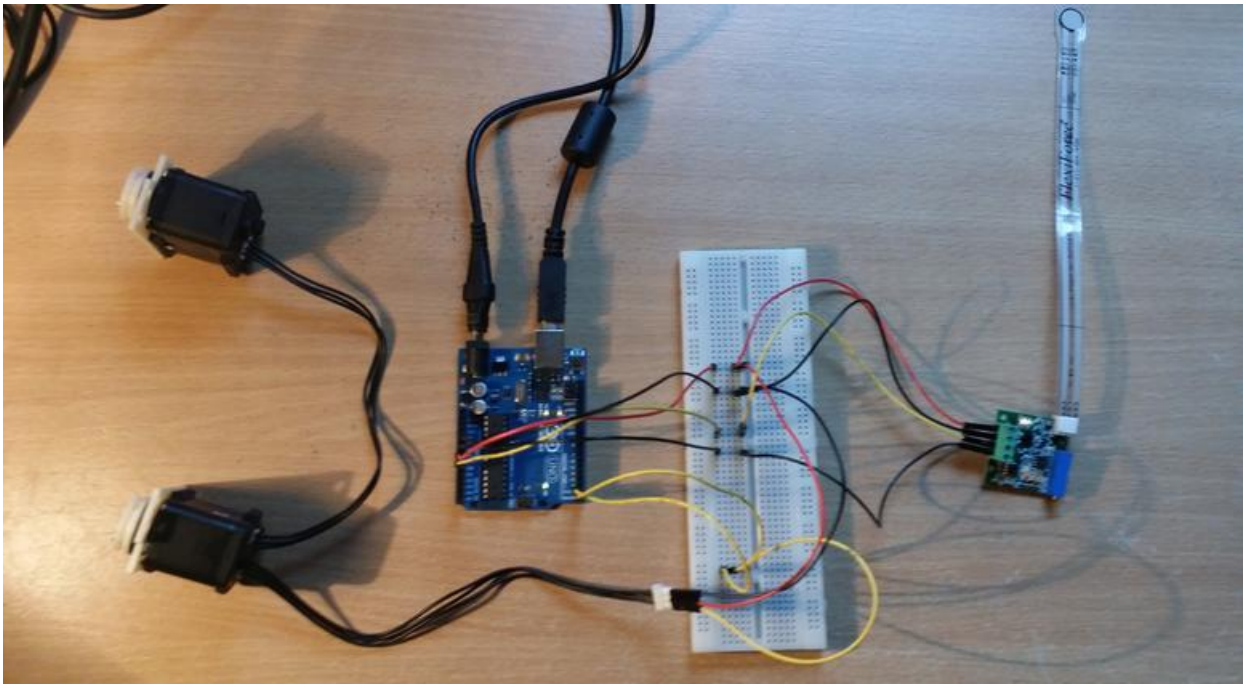

FIGURE 17 – Montage deux servomoteurs série AX-12 et d'un capteur de pression FlexiForce

Les montages ci-dessus montre deux servomoteurs séries montés, plus un capteur de pression.

5.2 Protocole

Cette partie explique le protocole de montage pour le dernier montage présenté :

1. Brancher les deux servomoteurs
2. Brancher un câble jaune à l'entrée des datas des servos, un rouge pour le courant, et un noir au dernier port.
3. Connecter ces 3 câbles à la plaque de travail, chacun sur une colonne .
4. Connecter deux fils jaune à la carte Arduino sur les port RX et TX puis brancher les à la plaque sur la même colonne que le fil jaune précèdent.
5. Brancher un fil noir sur le port GND de la carte puis ajouter le sur la même colonne que l'autre fil noir.

6. Brancher un fil rouge sur le port Vin de la carte puis ajouter le sur la même colonne que l'autre fil rouge.
   A ce moment les deux servos doivent être connectés à la carte Arduino de par la plaque.
   Maintenant nous allons connecté le capteur à la carte Arduino .
7. Brancher le capteur à sa plaque
8. Brancher dans l'ordre au port prévue sur la carte un fil rouge, puis un noir, puis un jaune, puis noir.
9. Connecter le fil jaune au port A0 de la carte Arduino, qui est le premier port des input.
10. le fil rouge et le premier fil noir doivent être sur les mêmes colonnes que les rouges et les noirs des servos.
11. le dernier fil noir doit être connecté au deuxième port GND de la carte Arduino.
12. Étape finale, très importantes, brancher la carte Arduino à l'odinateur via USB, brancher l'alimentation 9V à la carte. Ouvrir son IDE Arduino .

6 Conclusion

Lors de ce stage j'avais pour mission de mettre en place les outils qui permettraient de contrôler une main robot. Cela c'est fait en deux temps, d'abord trouver les composants optimaux pour la main robot.Or ce sont les servomoteurs séries qui se sont montrés les plus performants pour la main robot.
Ensuite nous avons programmé les modules en Arduino pour faire marcher ces moteurs.
Le stage s'en est trouvé concluant.
Ce mois dans l'univers de la recherche m'a renforcé dans mon choix d'études avenir .
Cela à été un des projets les plus intéressants auquel il m'a été donné de participer. Même si les débuts ont été laborieux, et qu'il a fallu tout apprendre à zéro, je me sens grandi par cette expérience. Je ne remercierais jamais assez M.Jouandeau de m'avoir accepté pour cette tâche.