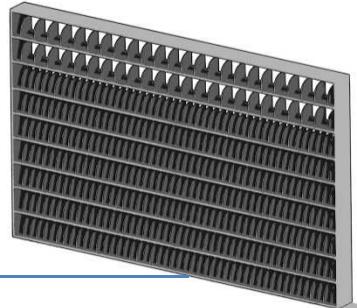


Dossier TPE : Bibliothèque Robotisée

*GUILLAMOT Vincent, PHELIPOT Pascal,
BONNET Guillaume, BOURDON William.*

Thème : Avancées scientifique et réalisation technique.

SOMMAIRE



I/ Le projet

1.1 Le questionnement :	5
1.2 Les débuts.....	5
1.3 Quelques explications	5

II/ Cahier des charges

1.1 Fonction d'usage :	7
1.2 Bêtes à cornes :	7
1.3 Diagramme pieuvre.....	8
1.4 Chaîne d'information et d'énergie.....	10
La Bibliothèque.....	11

ELECTROTECHNIQUE ET INFORMATIQUE:.....13

Explication ARDUINO :.....	14
Explication Electronique matériel:	16
Alimentation.....	42
Explication Explication Electronique logiciel :.....	24
Le programme arduino sans fonction retour	42
Le programme arduino avec fonction retour.....	42
Explication Informatique:.....	33
Programme C++	35
Annexes	46

MECANIQUE:

Introduction :.....	41
Calculs et explications	42
La pince.....	42
Force nécessaire pour maintenir le livre dans la pince :	43
Couple nécessaire pour serrer la pince à l'aide d'un système de vis-écrou :	43

Vitesse d'avancement de la pince pour un tour de moteur (pseudo-vérin).....	44
Le chariot horizontal.....	44
Vitesse de déplacement horizontal :	44
Le chariot vertical	45
Vitesse et couple de déplacement du chariot vertical	45
Annexes	46
 SYNTHESE ET CARNET DE BORD:.....	41
 BIBLIOGRAPHIE	
Partie mécanique :	52
Partie électrotechnique :.....	52
Globalement :.....	52
 CONCLUSION :.....	53

Introduction

Etant en classe de 1^{ère}S°SI au lycée Pablo Picasso, nous avons été amenés à effectuer un TPE (Travail Personnel Encadré), d'une durée de 6-7 mois (de septembre à mi-février).

Les TPE a pour objectifs, entre autres de motiver les élèves par un travail dont définissent eux même le sujet, de les entraîner à surmonter différente problèmes techniques, et de leurs faire découvrir les méthodes de travail vis-à-vis d'un projet concret.

Pour notre part, étant dans une classe de Science de l'ingénieur, nous avons orienté notre projet sur un sujet liés à la fois à la mécanique, à l'électronique et à l'informatique.

REMERCIEMENTS

Tout au long de l'élaboration de notre projet, nous avons eu la chance de bénéficier de l'expérience et de l'aide de nos professeurs de Science de l'Ingénieur, pour la partie mécanique : M. Rieusset et pour la partie électrotechnique : M. Vitasse. Ces professeurs nous ont soutenu et nous ont permis de progresser dans le TPE.

I/ Le Projet

1. 1 Le questionnement.

Le projet est naît suite à plusieurs réflexion sur le mécanisme que nous allons traiter au cours de cette année scolaire en classe première S°SI.

Les premières idées concernant le projet ont été :

- Un créateur de cocktails automatique (Boisson), personnalisable.
- Une souris scanneuse qui permettrait de numériser des documents rapidement.

Ces projets ont vite été abandonnés, car l'idée de la bibliothèque robotisée nous a paru être la plus intéressante, qui nous proposait un défi de taille que nous voulions relever.

1.2 Les débuts.

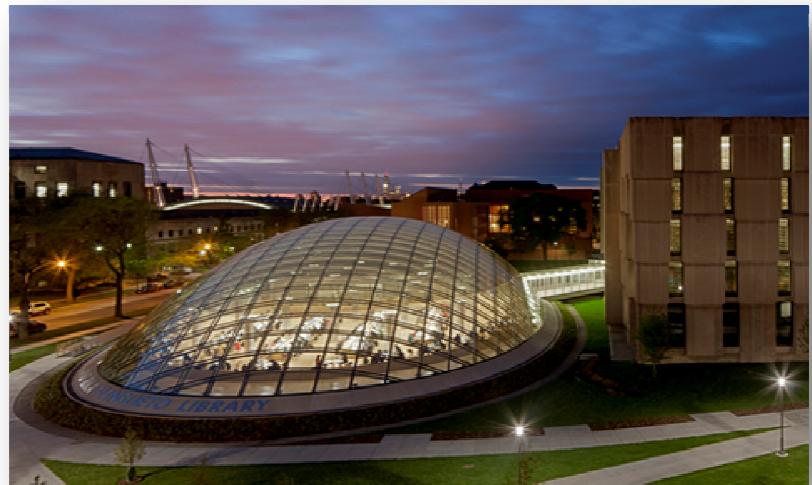
A ses débuts, le projet n'était à l'état que de réflexion, ce n'était qu'une idée, avec de nombreux questionnement sur le fonctionnement de ce robot, de l'organisation qu'il devait présenter. Notamment le traitement de l'information, le type de pince que nous devrions utiliser, la communication avec l'utilisateur, ainsi que les cinématiques de la machine.

1.3 Quelques explications

Pour ce TPE, nous vous proposons de créer une bibliothèque robotisée. Avant d'aborder les notions techniques de cet appareil, les paragraphes suivants vont vous exposer quelques informations clés sur la bibliothèque en relation avec notre TPE

Rechercher un livre est souvent une tâche complexe, quand il est questionnement de très grande bibliothèque, où les livres sont alignés à perte de vue. Pour avoir une recherche un peu plus harmonieuse, des projets de ce type ont déjà été réalisé, à une échelle beaucoup plus grande.

Suite à une recherche sur le moteur de recherche de Google, nous avons pu voir l'existence de ce mécanisme à l'université de Chicago (Etats-Unis d'Amérique) à une échelle beaucoup plus grande :

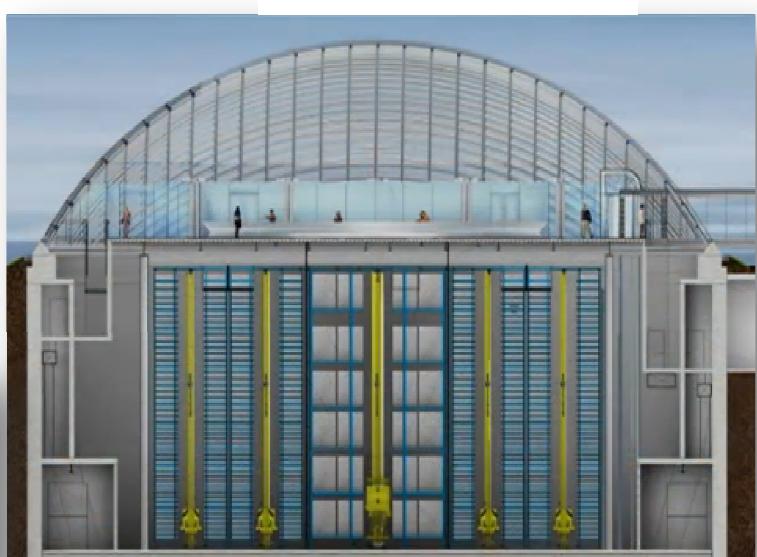


Ceci est l'extérieur de la bibliothèque de Chicago

Le mécanisme ainsi que le bâtiment vu de profil, nous pouvons observer sur le dessus l'espace de travail des étudiants et des documentalistes, et en dessous la bibliothèque ainsi que le robot se chargeant des livres.



Ici se trouve le robot vue de plus près. C'est lui qui se charge du retrait et de la descente es cases de livres.



Ici se trouve la bibliothèque et le robot. Les livres sont rangés dans des cases métalliques.

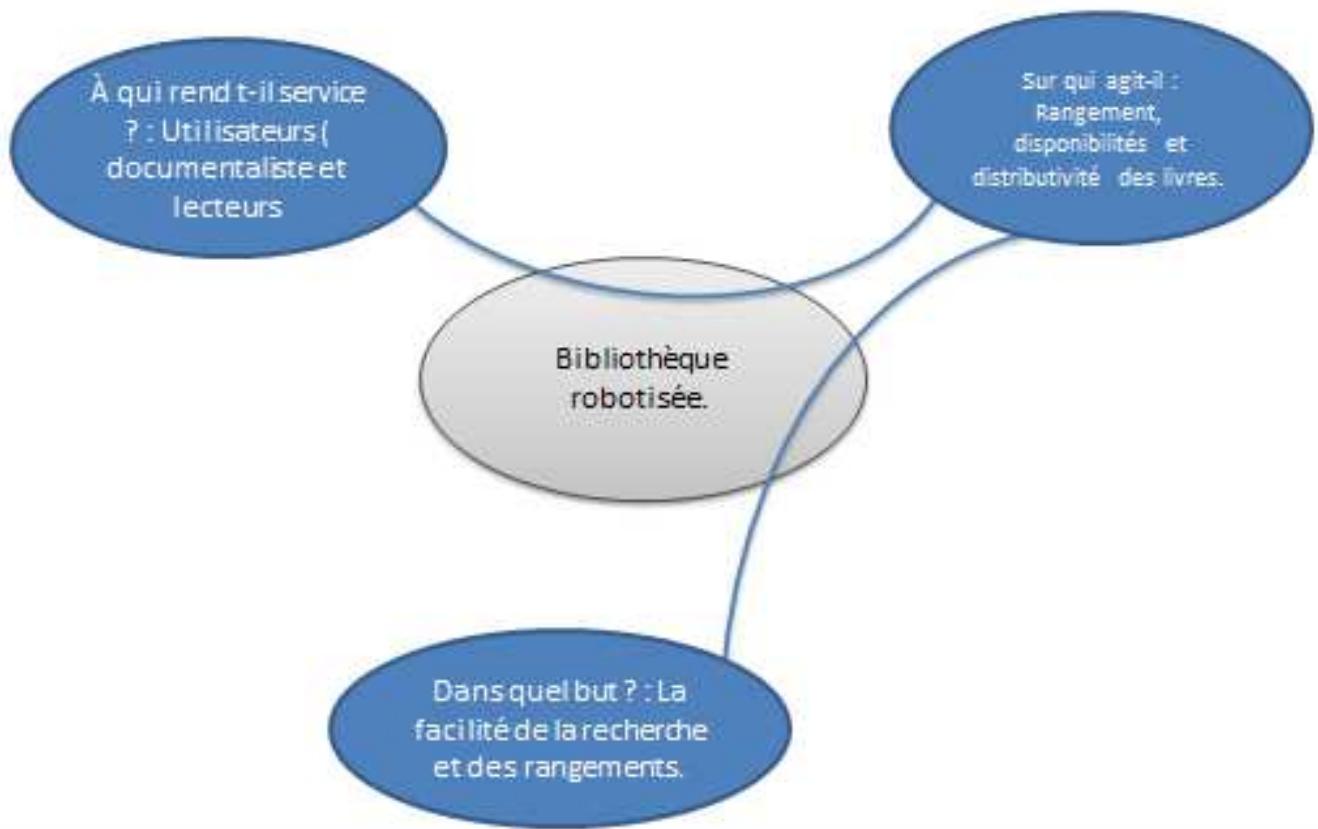


II/ Cahier des charges

1.1 Fonction d'usage :

Elle se résume à : Récupérer et ranger des livres plus facilement.

1.2 Bêtes à cornes :



La bibliothèque automatisée vise particulièrement les utilisateurs concernés par le stockage et la distribution de livre, c'est-à-dire :

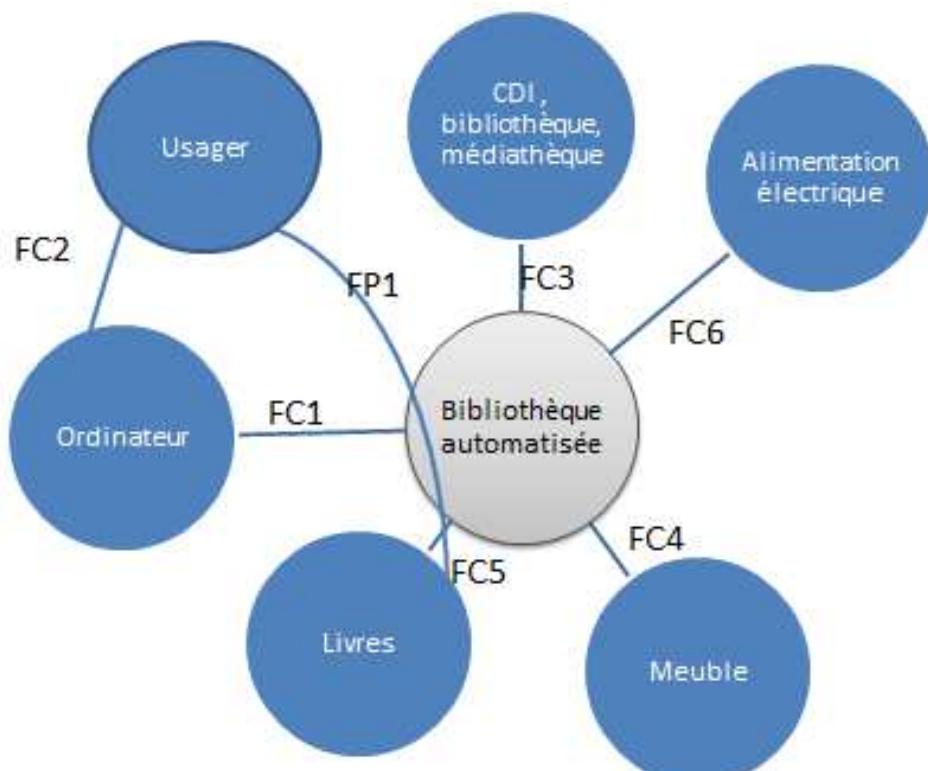
- Les bibliothécaires qui veulent avoir une recherche plus efficiente de leurs produits.
- Les lecteurs qui aiment la lecture et qui conservent d'importante quantité de livres chez eux.

Elle agit sur les bibliothèques, le produit vise notamment à améliorer la recherche d'un livre, et d'avoir un rangement conséquent. Cela permet en effet d'avoir une accessibilité aux livres permanente. Notre projet facilite donc la recherche et les rangements des bibliothèques.

1.3 Diagramme pieuvre.

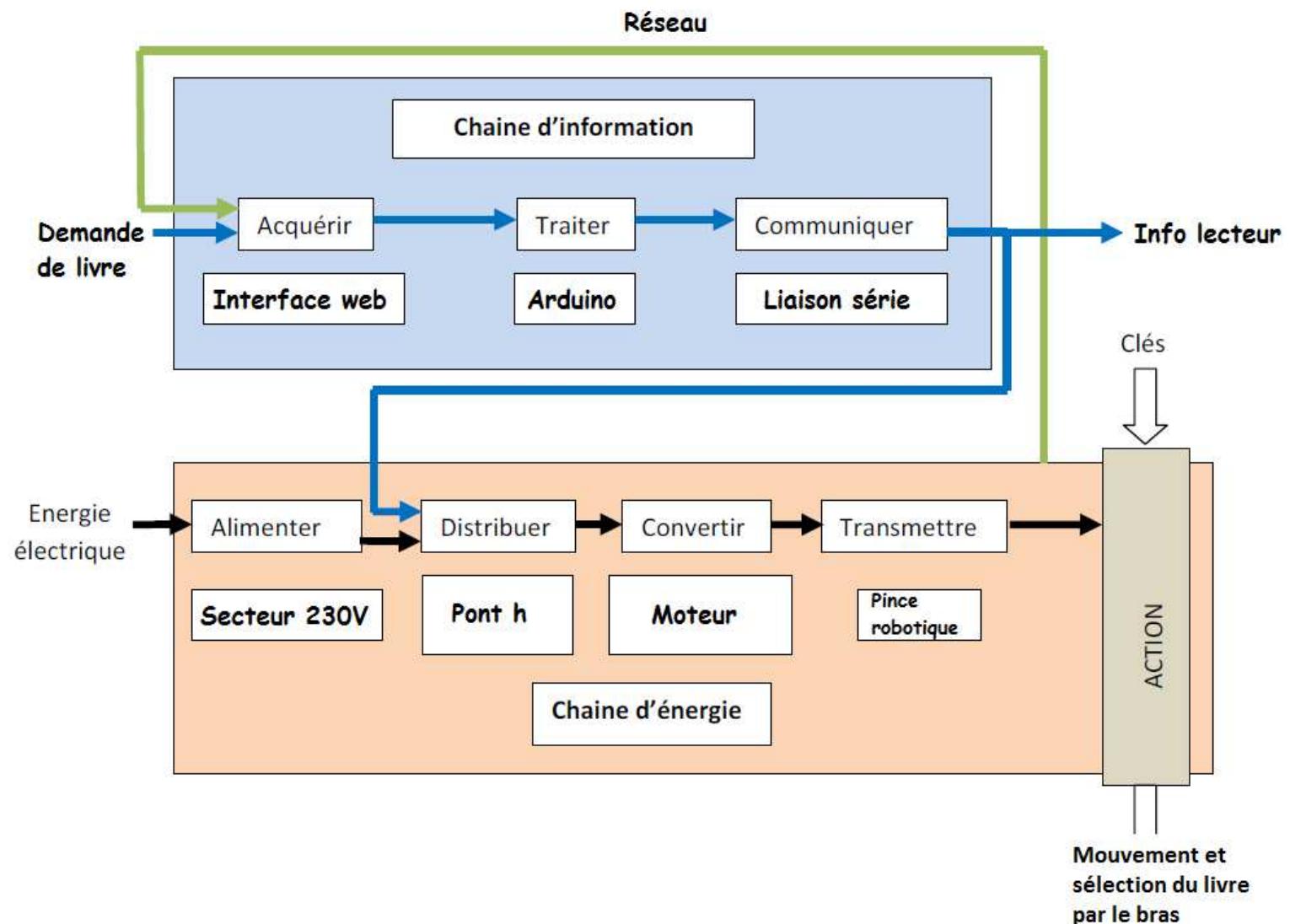
Nous vous proposons donc maintenant d'étudier les contraintes de notre bibliothèque robotisée :

« Pieuvre »



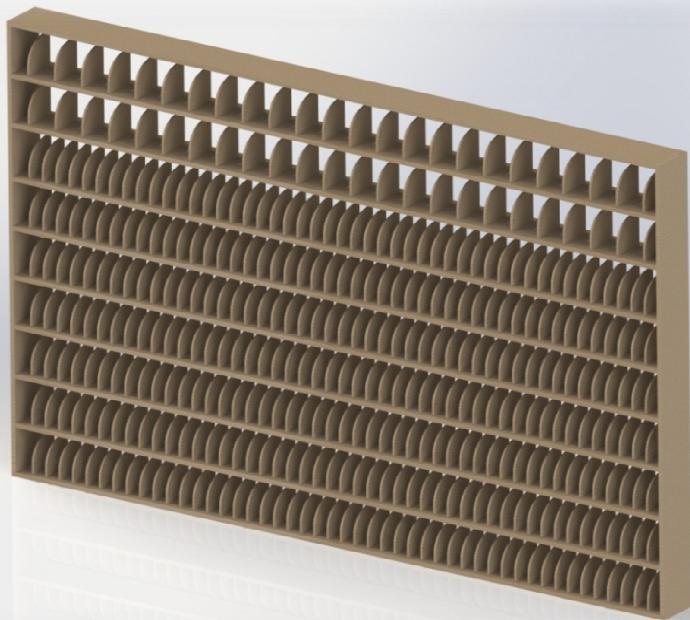
Fonctions	Critères	Flexibilité
FP1	-Permettre aux utilisateurs de pouvoir trouver un livre plus facilement . -Ranger les livres plus facilement .	/
FC1	-Ordinateur envoie l'ordre a la bibliothèque.	/
FC2	-Interface homme-machine.	/
FC3	-Construit pour les CDI, etc.	/
FC4	-Stabilité de la bibliothèque	Surface au sol à calculer.
FC4	-Place / Dimension	Largeur : 3.130m Hauteur : 20cm Profondeur : 1.940m
FC5	-Supporter le poids, la largeur, la taille de chaque livres	Max 1 kg (1 livres) 30cm de hauteur(1 livre) 7cm de largeur tranche
FC6	Alimentation de l'appareil.	230V.

1.4 Chaîne d'information et d'énergie.



La bibliothèque

Notre produit doit être accessible à tout type de bibliothèque, doit pouvoir déplacer tout type de livre, et être efficace dans sa tâche. Malgré cela nous avons rencontré de nombreux problèmes avec l'adoptions du robot sur les bibliothèques et nous en avons donc conclus que la bibliothèque en question devras avoir des caractéristique précise pour pouvoir être fonctionnel le robot.



Nous avons dû modéliser cette bibliothèque. Les caractéristiques de cette bibliothèque :

NOMBRE

PETITE CASE	336
-------------	-----

GRANDE CASE	50
-------------	----

Les petites cases sont pour des petits livres tels que des romans, ou tout autre livre plus mince en épaisseur. Tout au contraire, la grande case sert à mettre des livres plus épais et plus lourd. Quant aux poids, généralement, les romans pèse de 100 à 250g, un livre plus épais atteint les 700g pour les plus épais.

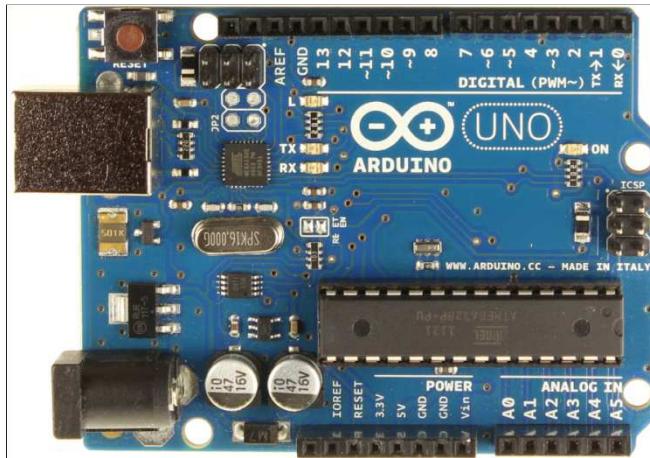
Point faibles : Les encyclopédies, ainsi que des livres « lourds » ne peuvent être mis dans cette bibliothèque. Le poids maximum d'un livre doit être de 1kg.



ELECTROTECHNIQUE ET INFORMATIQUE

Explication ARDUINO

Arduino est un projet open-hardware c'est-à-dire que c'est une platine de test libre et open-source, libre de droit.



Composé d'un microcontrôleur ATmega328 cadencé à 16 MHz, d'origine il a déjà un programme à l'intérieur (boot loader) permettant sa programmation par le port USB, alimenté avec un voltage de 5V.

Pour fonctionner, la carte a besoin d'une alimentation. Le microcontrôleur fonctionne sous 5V, la carte peut être alimentée en 5V par le port USB ou bien par une alimentation externe qui est comprise entre 7Vet 12V. Cette tension doit être continue et peut par exemple être fournie par une pile 9V. Un régulateur se charge ensuite de réduire la tension à 5V pour le fonctionnement de la carte. Le Régulateur de la carte accepte entre 7V et 15V (même si le régulateur peut supporter plus, sa durée de vie risque de diminuer). Elle dispose de 13 pin (broche) de commande dont 6 pouvant envoyer des signaux pwm (digital simulé) et 6 broche analogique.

Elle a aussi un pin d'alimentation de 3,3V et d'un de 5V et 3 GND (masse).

La programmation de l'Arduino ce fait par un langage propre à eux, qui est en fait une modification du C pour le rendre plus simple à écrire pour la carte. Une fois le code écrit proprement, le compilateur `avr-gcc` va compiler le code (le mettre en binaire) et l'envoyer par le port USB dans l'ATMega328.

Le code Arduino a une particularité au niveau de la syntaxe minimal : c'est la présence de boucle `setup` et `loop`

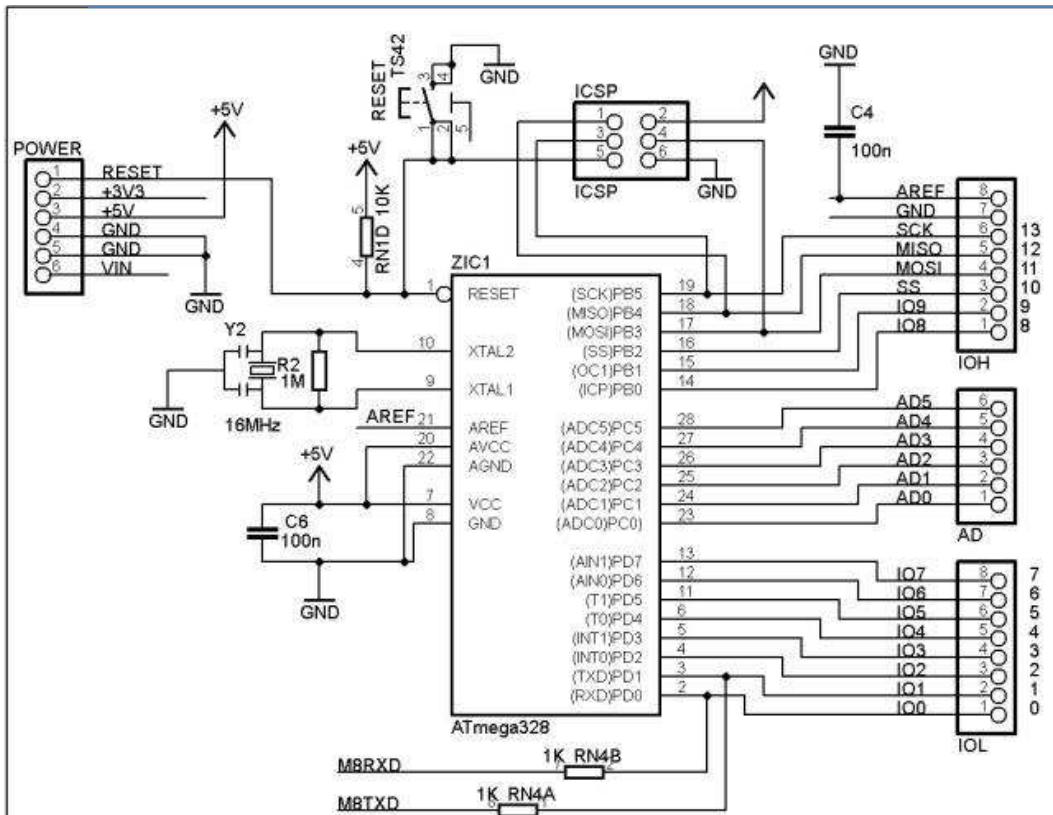
```

void setup() //fonction d'initialisation de la carte
{
    //contenu de l'initialisation
}

void loop() //fonction principale, elle se répète (s'exécute) à l'infini
{
    //contenu du programme
}

```

Schéma électronique de l'ARDUINO UNO



Pourquoi avoir choisi ARDUINO ?

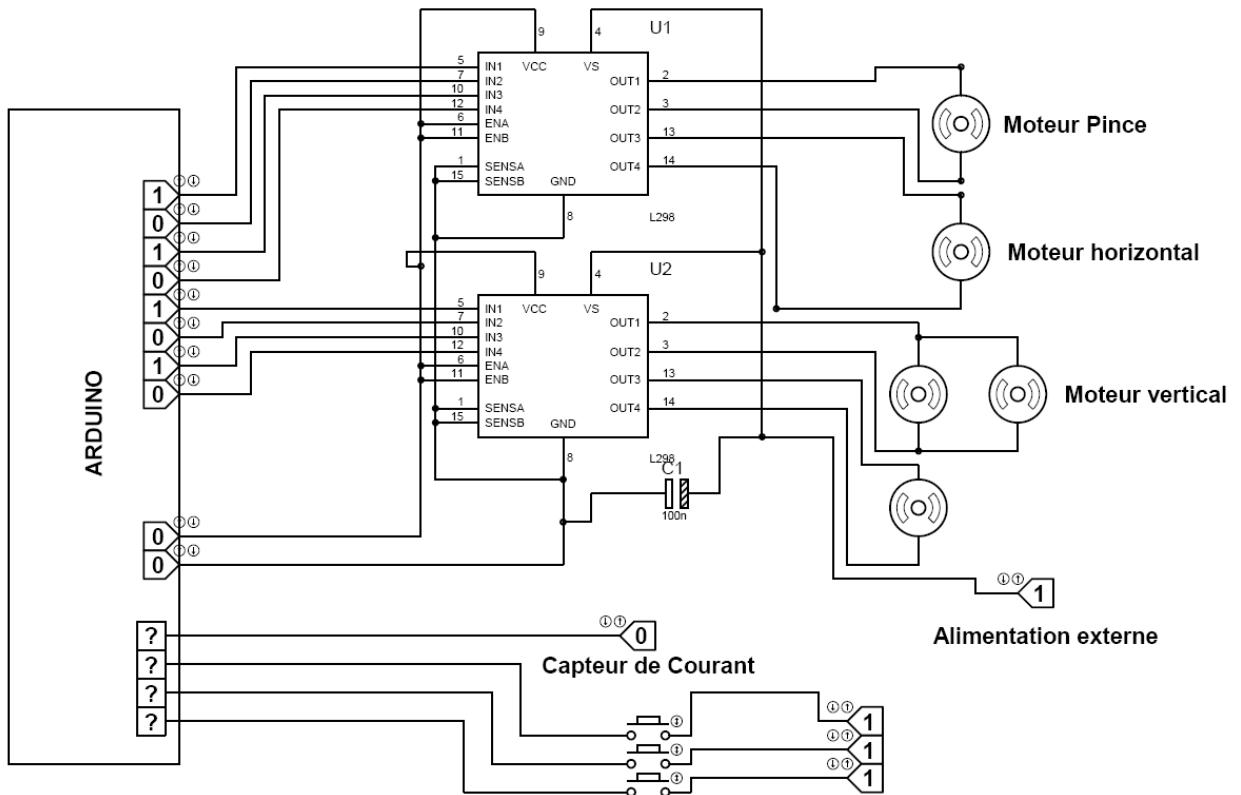
Nous avons choisi ARDUINO car c'est un système libre et peu coûteux et d'une puissance suffisante pour la bibliothèque robotisé. Il a aussi pour atout de simplifier grandement le schéma électronique.

Il met aussi à disposition un nombre important d'extension utile (carte contrôle moteur, carte Ethernet) permettant d'augmenter le rayon de possibilité déjà offert par cette carte.

Explication Electronique matériel

L'objectif du circuit électrique est simple, il est de gérer cinq moteurs, que ça soit au niveau des erreurs, de la vitesse, de la durée de fonctionnement, de la protection des moteurs.

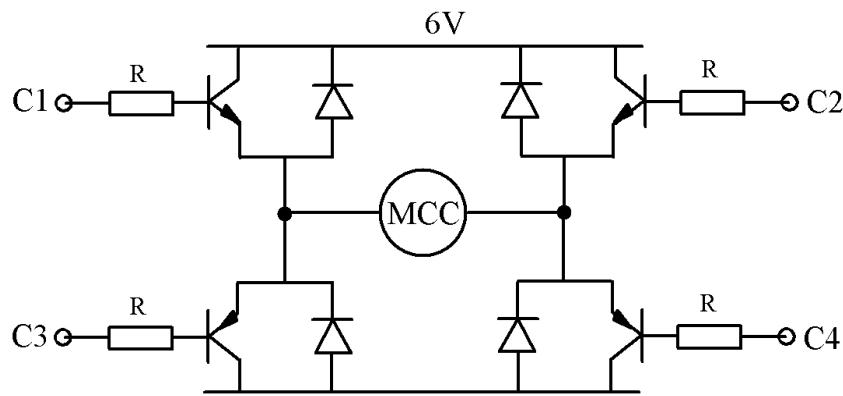
Le circuit



(créer avec ISIS proteus)

Les ponts H

Les ponts H sont des circuits à base de transistor et de diode permettant à un moteur (ou autre) de changer de sens de rotation.

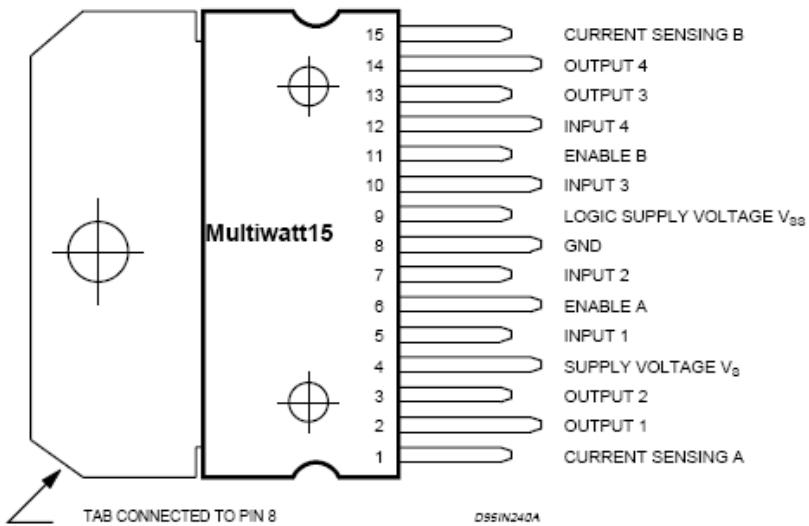


L298 –Pont H

Nous utilisons le double pont H L298 du fait que nous avons 5 moteurs dont deux en parallèle (donc 4 commandes) nous avons besoin de 2 L298. Ces pont H en plus de disposer d'une alimentation « normal » de 7V elle dispose d'une alimentation supplémentaire d'un maximum de 50V. L'intensité maximal en sorti par pont H (rappelons qu'il y en a deux par circuit intégré) est de 2A.

Les broches 5,7 servent au commandement du moteur 1, les broches 10, 12 au moteur 2.

Les broches 6 et 11 servent à l'activation des ponts H ENABLE A au moteur 1 (broche 6) et ENABLE B au moteur 2 (broche 11).



Symbol	Paramètres	Valeur	Unité
V_s	Alimentation supplémentaire	50	V
V_{ss}	Voltage supplémentaire pour la logique	7	V
V_i, V_{en}	Tension d'entrée et Activer	-0.3 à 7	V
V_{sens}	Tension de détection	-1 à 2.3	V
P_{tot}	Dissipation énergétique total (Tcase = 75°C)	25	W
T_{op}	Température de fonctionnement	-25 à 130	°C
T_{stg}, T_j	Température de stockage et de jonction	-40 à 150	°C

Durée de fonctionnement

Pour gérer la durée de fonctionnement l'Arduino va recevoir par liaisons série (simulé part l'USB) le temps exacte (millisecondes près) de fonctionnement du moteur (*1 et 2 ou 3*). Le calcul du temps est fait comme cela :

X = temps pour que le moteur fasse Xcm ; D = Distance à parcourir ; Dinit = Distance initial pour calibrer à la première case.

$$T = D * x + D_{init}$$

Les erreurs imprévues et décalage

Il se peut que lors du fonctionnement de la machine, un facteur extérieur perturbe la machine ou que pour une raison non connue, un décalage au niveau des moteurs se créé. Pour pallier à ce problème et réduire le risque de blocage de la machine, un système de remise à zéro a été mis en place, en effet lorsque le livre sera pris la pince ne reviendra pas à ses coordonnées ultérieur pour un traitement du temps de fonctionnement mais par un système de buté, lorsque le chariot du moteur

arrivera en fin de rail un bouton poussoir va être pressé et signalera à la carte Arduino la buté, de ce fait la carte gérera l'extinction du moteur le mettant en même temps au 0 absolu de la machine.

La protection des moteurs

Pour les moteurs verticaux et horizontaux (cf. dossier mécanique) le système de résolution d'erreur par les boutons poussoir remplit aussi cette fonction, aucun obstacle est présent sur les rails et le 0 au départ est absolue, aucun blocage des chariots n'est possible donc aucune surcharge d'intensité à gérer.

Contrairement au moteur cité ci-dessus pour le moteur de la pince il est obligatoire pour gérer le serrage des livres. Un capteur de courant mesura en temps réel l'intensité de courant demandé par le moteur, lorsque que le livre sera bien serré le capteur va capter la surcharge demandé par le moteur et le stoppé. Pour déposer le livre le principe sera le même mais lorsque la vis ne pourra plus est dévissé le moteur va forcer et donc demander plus de courant, à ce moment le capteur refera la

Même action que pour le serrage. Pour choisir le capteur nous avons dû choisir un courant primaire relativement faible pour avoir une précision assez haute. Nous avons choisi HX 03-P/SP2 disposant d'un courant primaire de 3A et pouvant mesurer ± 9 A, sachant que d'une part les moteur ne consomme pas plus de 300mA mais nous savons qu'à leur démarrage il consomme plus mais le capteur est suffisamment précis pour gérer les surcharges de courant.

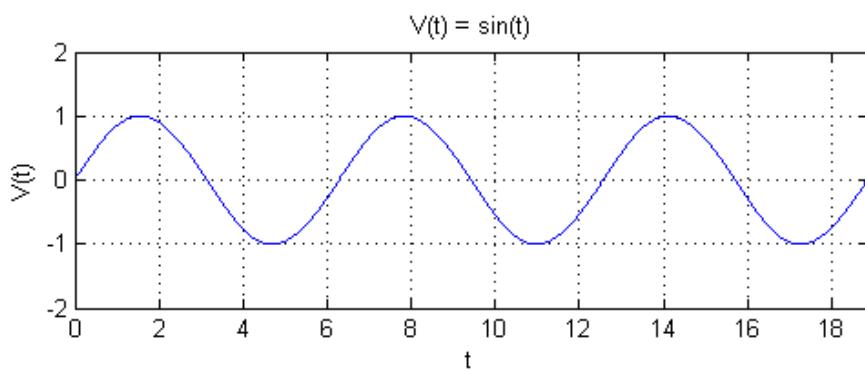
Alimentation 230V/12V

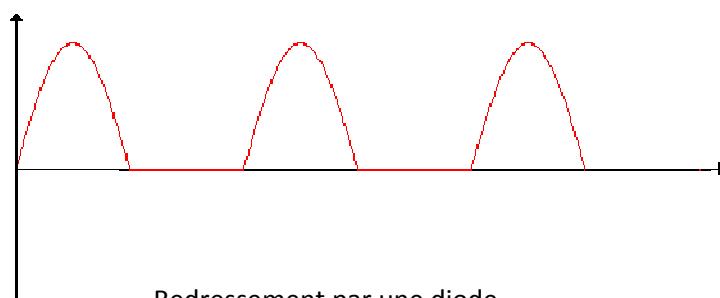
Nous avons choisi d'alimenter le projet Bibliothèque robotisé par le secteur (230V). Nous avons opté pour cette solution car pour la taille du projet et l'emplacement prévu pour celui-ci met facilement à disposition des prises de courant. En effet il aurait été peu judicieux de mettre une alimentation sur batterie, sachant que le système serait placé en intérieur, aura plusieurs moteurs à faire fonctionner, toute la journée.

Pour la fabrication de l'alimentation nous avons donc recours à un transformateur 230V/15V, à 4 Diode de redressement 1N4004 acceptant un courant de 1A, à un circuit intégré LM7812 pour réguler la tension à 12V pour alimenter l'Arduino et les moteurs. Après le passage des Diode de redressement la tension va légèrement augmenter vers les 19V. Deux condensateurs seront aussi nécessaires pour le filtrage du courant, un de $2200\mu F$ et acceptant une tension de 40V et un de $22\mu F$ acceptant une tension de 25V.

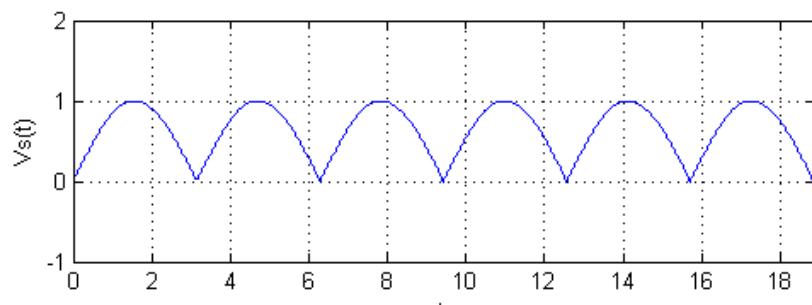
1) Redressement

Le redressement sert à « couper » le courant alternatif pour avoir que du courant positif. Le redressement par une diode va donner une demi-sinusoïde, pour avoir donc du courant tout le temps, nous avons fait un pont de diode. Ceci va générer une légère augmentation de la tension de l'ordre de quelques Volt. Pour le projet la tension augmentera au alentour de 19V.

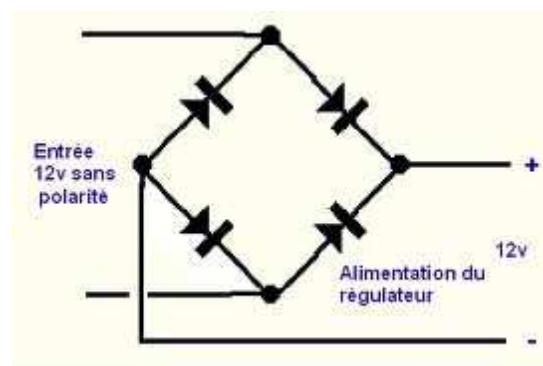




Redressement par une diode.



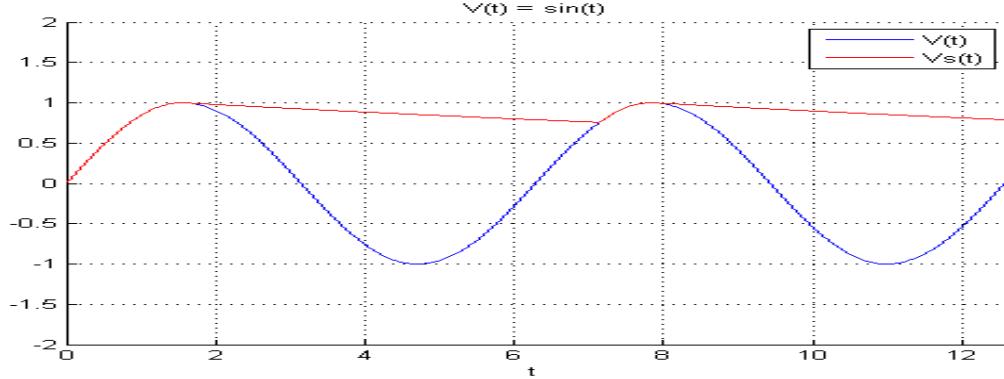
Courant redressé avec un pont de diode



Pont de diode

2) Filtrage

Pour le filtrage et le filtrage nous avons opté pour des condensateurs capacitifs de $2200\mu F$ et de $22\mu F$. Cela permettra de lisser le courant en lissant le courant nous obtenons un courant presque continu qui suffit largement à notre projet.



3) Régulateur de courant

Le Régulateur de courant utilisé est le LM7812 qui fait en sorti 12V et jusqu'à 1 Ampère. En entré il demande une tension auxalentours de 19V, nous aurons donc environ 19V en entrée grâce au redressement. Les régulateurs de tension ont souvent besoin de radiateur, en effet ils chauffent beaucoup, avoisinant les $125^{\circ}C$ lors de la régulation.



La formule suivante permet de calculer la puissance maximale dissipée :

$$PD = (Vin - Vout) * I$$

$$PD = (19 - 12) * 1 = 7$$

Donc la puissance dissipée maximale de 7W

Nous calculons maintenant la température maximale, on part sur -25% de la température maximale accepté par le composant.

$$T_{max} = 125 * 0.75 = 100$$

Pour finir un fait :

$$\frac{T_{max}}{PD} = 14.28^{\circ}C/W$$

Nous devons donc avoir un dissipateur capable de dissipé $14.28^{\circ}C/W$.

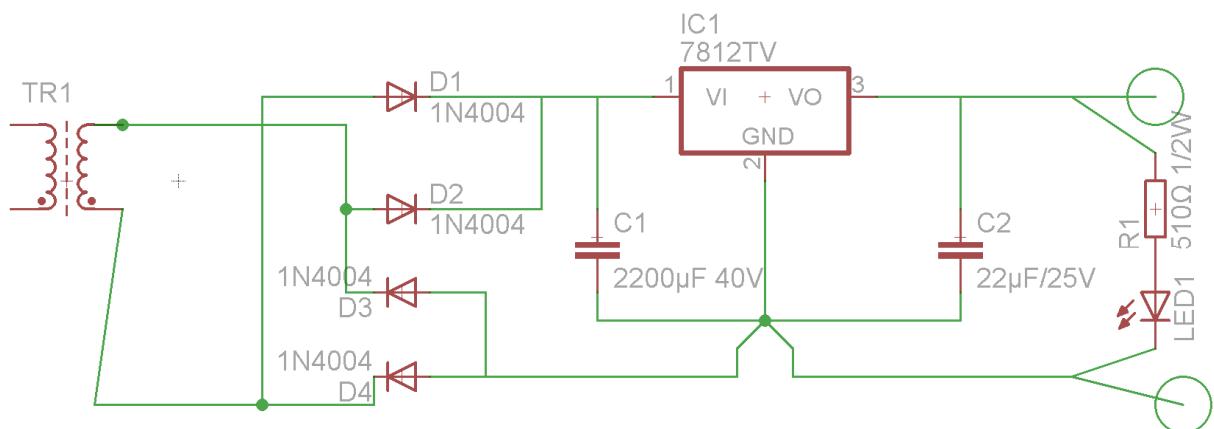
4) Calcul résistance pour la LED

Voici les calculs qui ont permis de déterminer la résistance nécessaire pour la LED rouge (standard)

$$R1 = \frac{(12 - 1.6)}{0.02} = 520$$

La résistance choisie doit être de 510Ω et d'une puissance de $1/2W$.

5) Le schéma électronique de l'alimentation



Dossier Explication Electronique Logiciel

Protocole de communication pour la liaison série

Le protocole permet le transfert des données liées au fonctionnement de la bibliothèque. Les liaisons série permet d'envoyer uniquement octet par octet soit chiffre par chiffre ou lettre par lettre. Pour pallier ce problème nous avons donc mis un ordre précis dans l'envoi de ces données. Grâce à cette ordre d'envoi bien établi, nous pourrons récupérer dans un tableau de char les octets et les convertir par un traitement en int ou long suivant le cas.

Du Type :

000000000000000000000000 (Ce n'est pas du binaire mais du Décimal)

Explication :

les 3 premiers octets déterminent les moteurs à activer.

Les octets 4, 5 et 6 déterminent le sens des moteurs (l'octet 4 détermine le sens du moteur, l'octet 5 détermine le sens du moteur 2, l'octet 6 détermine le sens du moteur 3).

Les 5 octets suivants définissent le temps de fonctionnement du moteur 1.

Les octets 7, 8, 9, 10, 11, 12 définissent le temps de fonctionnement du moteur 2.

Les octets 13, 14, 15, 16 définissent le temps de fonctionnement du moteur 3.

Tableau :

moteur1	moteur2	moteur3	sens1	sens2	sens3	temps1	temps1	temps1	temps1
temps1	temps2	temps2	temps2	temps2	temps2	temps3	temps3	temps3	temps3

Exemple de donnée transmise :

110100050000300000000

Traduction :

Moteur 1 et 2 sélectionné, le moteur 3 n'est pas sollicité, le moteur 1 doit tourner dans le sens inverse des aiguilles d'une montre, le moteur 2 dans le sens des aiguilles d'une montre, le moteur 1 tournera pendant 5000ms (5seconde) et le moteur 2 pendant 3000ms (3seconde).

Logiciel ARDUINO

A savoir :

- Les octets reçus par la liaison série sont en ASCII

- En ASCII 0 = 48 et n = 48+n
- (les tableaux dans la plus part des langages informatiques commencent à 0).
- Les types de variable principalement utilisées :

Type de variable	Nombre en bits
int entier	16 bits
long entier	32 bits
char entier	8 bits
Byte entier non négatif	8 bits

- millis() est capable de mesurer une durée allant jusqu'à 50 jours, la valeur renournée est en milliseconde et doit être stockée dans une variable de type "long".

Etape par Etape

- 1- Dans un premier temps on déclare les variables contenant les numéros des broches utilisées et les variables liées au fonctionnement du programme (`char i = 0; char communication [] = {};` `byteaccess`) dont un tableau de char nommé communication où les donnée reçus vont être mises.
- 2- Puis on déclare dans la boucle `setup()` toutes les variables moteurs en OUTPUT (sorti) et la variable du capteur (`sensor_flow`) en INPUT, ainsi que les 4 boutons poussoir (`button_ «nombre»`). On déclare aussi le baud (cf. ANNEXE) de la communication série.
- 3- Dans la boucle `loop()` on va commencer le programme séparé en plusieurs sous parties
 - 1- On créer la boucle de blocage du script principal en testant la variable access si il est égal à 1 on rentre dans la condition mais juste après on lui attribue la valeur de 0.
 - 2- On teste la réception de donnée, si l'objet `Serial.available()` renvoie 0 c'est qu'il n'y a aucune donnée reçus. Après cela on créer une boucle `while` (on aurait pu faire une boucle `for`) avec comme conditions le fait qu'on reçoit des données et que i soit inférieur ou égal à 21, dans cette boucle grâce à l'indice i qu'on incrémente à chaque tour de boucle, on peut mettre dans une case différente du tableau « communication » toutes les données reçus (octet par octet) par l'objet `Serial.read()`.
 - 3- On regarde si dans la case 0 il y a un 1 si oui on exécute le code de fonctionnement du moteur 1. On reforme le temps de fonctionnement en effet les octets étant séparés mais dans le bon ordre et avec un nombre près défini de chiffre, cette opération est simple on remet la valeur ASCII en valeur normal en enlevant 48. Pour les valeurs entre crochets se référer au dossier sur le protocole de communication p11 (`communication[6]-48)*10000+(communication[7]-48)*1000+(communication[8]-48)*100+(communication[9]-48)*10+communication[10]-48;` Ensuite on ajoute ce temps à l'heure précise du moment (en ms).
 - 4- Pour finir le programme du moteur1 on regarde l'octet envoyé et reçu par la communication série désignant le sens dans l'octet n°3 si il est égal à 0 c'est le sens des aiguilles d'une montre, et si c'est 1 c'est le sens inverse.
 - 5- On exécute les autres moteurs.

- 6- Vu que « access » est toujours bloqué on exécute le *else* du premier *if* et on teste si le temps de fonctionnement est écoulé si oui on éteint les moteurs on fait avancer la pince, on pince jusqu'à ce que le capteur de courant renvoie une surcharge et à ce moment on stoppe le moteur, fait reculer et on fait tourner les moteur 1 et 2 dans le sens inverse de ce qui est demandé par la voie série pour revenir au début. Un système de butté par bouton poussoir dira au moteur qu'ils sont arrivé et stoppera par la même occasion les moteurs.

Le programme ARDUINO sans fonction retour (testé)

```

7- constintmotor_one_one=2;//moteur 1
8- constintmotor_one_two=3;//moteur 1
9- constintmotor_two_one=4;//moteur 2
10- constintmotor_two_two=5;//moteur 2
11- constintmotor_three_one=6;//moteur 3
12- constintmotor_three_two=7;//moteur 3
13- constintsensor_flow=8;//capteur de courant
14- constintmotor_one=9;//voyant moteur 1
15- constintmotor_two=10;//voyantmoteur 2
16- constintmotor_three=11;//voyantmoteur 3
17- char i =0;
18- longtime_one;
19- longtime_two;
20- longtime_three;
21- longtime_one_add;
22- longtime_two_add;
23- longtime_three_add;
24- char communication []={};
25- byte access=1;
26- voidsetup()
27- {
28- pinMode(motor_one_one, OUTPUT);
29- pinMode(motor_one_two, OUTPUT);
30- pinMode(motor_two_one, OUTPUT);
31- pinMode(motor_two_two, OUTPUT);
32- pinMode(motor_three_one, OUTPUT);
33- pinMode(motor_three_two, OUTPUT);
34- pinMode(motor_one, OUTPUT);
35- pinMode(motor_two, OUTPUT);
36- pinMode(motor_three, OUTPUT);
37- pinMode(sensor_flow, INPUT);
38- Serial.begin(19200);
39- }
40- voidloop()
41- {
42- if(access==1){
43- if(Serial.available()>48){
44- while(Serial.available()>0&& i <=24)
45- {
46- communication[i]=Serial.read();
47- i++;//on passe à l'indice suivant
48- }
49- /*****Moteur 1 *****/

```

```

50- if(communication[0]=='1'){
51- access=0;
52- time_one=(communication[6]-48)*10000+(communication[7]-
48)*1000+(communication[8]-48)*100+(communication[9]-
48)*10+communication[10]-48;
53- time_one_add=millis()+time_one;
54- Serial.println(time_one);
55- Serial.write("/");
56- Serial.println(time_one_add);
57- Serial.write("/");
58- Serial.println("Moteur 1 Réussi");
59- digitalWrite(motor_one, HIGH);
60- if(communication[3]=='0'){
61- digitalWrite(motor_one_one, HIGH);
62- digitalWrite(motor_one_two, LOW);
63- Serial.println("Sens aiguille d'une montre");
64- }
65- else{
66- Serial.println("Sens inverse aiguille d'une montre");
67- digitalWrite(motor_one_one, LOW);
68- digitalWrite(motor_one_two, HIGH);
69- }
70- }
71-
72-
73- ****Moteur 2****
74- if(communication[1]=='1'){
75- access=0;
76- time_two=(communication[11]-48)*10000+(communication[12]-
48)*1000+(communication[13]-48)*100+(communication[14]-48)*10+
communication[15]-48;
77- time_two_add=millis()+time_two;
78- Serial.println(time_two);
79- Serial.write("/");
80- Serial.println(time_two_add);
81- Serial.write("/");
82- Serial.println("Moteur 2 Réussi");
83- digitalWrite(motor_two, HIGH);
84- if(communication[4]=='0'){
85- digitalWrite(motor_two_one, HIGH);
86- digitalWrite(motor_two_two, LOW);
87- Serial.println("Sens aiguille d'une montre");
88- }
89- else{
90- Serial.println("Sens inverse aiguille d'une montre");
91- digitalWrite(motor_two_one, LOW);
92- digitalWrite(motor_two_two, HIGH);
93- }
94- }
95- ****Moteur 3****
96- if(communication[2]=='1'){
97- access=0;
98-
99- time_three=(communication[16]-48)*10000+(communication[17]-
48)*1000+(communication[18]-48)*100+(communication[19]-48)*10+
communication[20]-48;
100- time_three_add=millis()+time_three;
101- Serial.println(time_three);
102- Serial.write("/");
103- Serial.println(time_three_add);
104- Serial.write("/");

```

```

105-     Serial.println("Moteur 3 Réussi");
106-     digitalWrite(motor_three, HIGH);
107-     if(communication[4]=='0'){
108-         digitalWrite(motor_three_one, HIGH);
109-         digitalWrite(motor_three_two, LOW);
110-         Serial.println("Sens aiguille d'une montre");
111-     }
112-     else{
113-         Serial.println("Sens inverse aiguille d'une montre");
114-         digitalWrite(motor_three_one, LOW);
115-         digitalWrite(motor_three_two, HIGH);
116-     }
117- }
118-
119- }
120- }
121- else
122- {
123-     Serial.println("on attend");
124-     Serial.println("/");
125-     Serial.print(time_two_add-millis());
126-     Serial.print(">");
127-     Serial.print(time_two);
128-
129-     if(digitalRead(motor_one)== HIGH){
130-         if((time_one_add-
131-             millis())>time_one){digitalWrite(motor_one,LOW);Serial.println("motor 1 eteint");}
131-     }
132-     if(digitalRead(motor_two)== HIGH){
133-         if((time_two_add-
134-             millis())>time_two){digitalWrite(motor_two,LOW);Serial.println("motor 2 eteint");}
134-     }
135-     if(digitalRead(motor_three)== HIGH){
136-         if((time_three_add-
137-             millis())>time_three){digitalWrite(motor_three,LOW);Serial.print
138-             ln("motor 3 eteint");}
138-     }
139- }
140-
141-

```

Le programme ARDUINO avec fonction retour (non testé)

```
const int motor_one_one=2; //moteur 1
const int motor_one_two=3; //moteur 1
const int motor_two_one=4; //moteur 2
const int motor_two_two=5; //moteur 2
const int motor_three_one=6; //moteur 3
const int motor_three_two=7; //moteur 3
const int sensor_flow=8; //capteur de courant
const int motor_four_one=9; //moteur 4
const int motor_four_two=10; // moteur 4
const int button_one=11; //bouton 1
const int button_two=12; //bouton 2
const int button_three=13; //bouton 3
byte motor_one=0;
byte motor_two=0;
char i=0;
long time_one;
long time_two;
long time_three;
long time_one_add;
long time_two_add;
long time_three_add;
char communication []={};
byte access =1;
void setup()
{
pinMode(motor_one_one, OUTPUT);
pinMode(motor_one_two, OUTPUT);
pinMode(motor_two_one, OUTPUT);
pinMode(motor_two_two, OUTPUT);
pinMode(motor_three_one, OUTPUT);
pinMode(motor_three_two, OUTPUT);
pinMode(motor_four_one, OUTPUT);
pinMode(motor_four_two, OUTPUT);
pinMode(button_one, INPUT);
pinMode(button_two, INPUT);
pinMode(button_three, INPUT);
pinMode(sensor_flow, INPUT);
Serial.begin(19200);
}
void loop()
{
if(access ==1){
if(Serial.available()>48){

while(Serial.available()>0&&i<=24)
{
communication[i]=Serial.read();
i++; //on passe à l'indice suivant
}

*****Moteur 1 *****/
if(communication[0]=='1'){
access=0;
motor_one=1;
```

```

time_one=(communication[6]-48)*10000+(communication[7]-
48)*1000+(communication[8]-48)*100+(communication[9]-
48)*10+communication[10]-48;
time_one_add=millis()+time_one;
Serial.println(time_one);
Serial.write("/");
Serial.println(time_one_add);
Serial.write("/");
Serial.println("Moteur 1 Réussi");
if(communication[3]=='0'){
digitalWrite(motor_one_one, HIGH);
digitalWrite(motor_one_two, LOW);
Serial.println("Sens aiguille d'une montre");
}
else{
Serial.println("Sens inverse aiguille d'une montre");
digitalWrite(motor_one_one, LOW);
digitalWrite(motor_one_two, HIGH);
}
}

/*****Moteur 2*****/
if(communication[1]=='1'){
access=0;
motor_two=1;
time_two=(communication[11]-48)*10000+(communication[12]-
48)*1000+(communication[13]-48)*100+(communication[14]-48)*10+
communication[15]-48;
time_two_add=millis()+time_two;
Serial.println(time_two);
Serial.write("/");
Serial.println(time_two_add);
Serial.write("/");
Serial.println("Moteur 2 Réussi");
if(communication[4]=='0'){
digitalWrite(motor_two_one, HIGH);
digitalWrite(motor_two_two, LOW);
Serial.println("Sens aiguille d'une montre");
}
else{
Serial.println("Sens inverse aiguille d'une montre");
digitalWrite(motor_two_one, LOW);
digitalWrite(motor_two_two, HIGH);
}
}
}

else
{

Serial.println("on attend");
Serial.println("/");
Serial.print(time_two_add-millis());
Serial.print(">");

Serial.print(time_two);

```

```

if(motor_one=='1'){//pour éteindre moteur 1
if((time_one_add-millis())>time_one){Serial.println("motor 1
eteint");motor_one==0;}
}
if(motor_two=='1'){//pour éteindre moteur 2
if((time_two_add-millis())>time_two){Serial.println("motor 2
eteint");motor_two==0;}
}

/** MoteurPince (3) ***/
if(communication[2]=='1'&&motor_one=='0'&&motor_two=='0'){
Serial.println("Moteur Pince (3) Réussi");
digitalWrite(motor_four_one, HIGH);//fait tourner le pseudo vérin (avance)
digitalWrite(motor_four_two, LOW);
delay(1000);
digitalWrite(motor_three_one, HIGH);//on fait tourner le moteur de la pince
digitalWrite(motor_three_two, LOW);
}
if(digitalRead(sensor_flow)=='TRUE')// Si surcharge d'intensité
{
digitalWrite(motor_three_one, LOW);// on éteint le moteur de la pince
digitalWrite(motor_three_two, LOW);
digitalWrite(motor_four_one, LOW);//fait tourner le pseudo vérin (recule)
digitalWrite(motor_four_two, HIGH);
delay(1000);
digitalWrite(motor_four_two, LOW);// stop le vérin
digitalWrite(motor_two_one, LOW);// on allume le moteur 1 et 2
digitalWrite(motor_two_two, HIGH);
digitalWrite(motor_one_one, LOW);
digitalWrite(motor_one_two, HIGH);
}

/** Gestion bouton poussoir sécurité **/
if(digitalRead(button_one)== HIGH)// retour déplacement horizontal éteint
{
digitalWrite(motor_one_one, LOW);

digitalWrite(motor_one_two, LOW);
}
if(digitalRead(button_two)== HIGH)// déplacement vertical côté gauche
{
digitalWrite(motor_two_one, LOW);
digitalWrite(motor_two_two, LOW);
}
if(digitalRead(button_three)== HIGH)// déplacement vertical côté droit
{
digitalWrite(motor_two_one, LOW);
digitalWrite(motor_two_two, LOW);
}

if(digitalRead(button_three)== HIGH &&digitalRead(button_two)== HIGH)// Si
les 2 sont appuyé on fait avancé pendant 27ms la pince puis relache le livre
{
digitalWrite(motor_one_one, HIGH);// moteur vertical (1) tourne
digitalWrite(motor_one_two, LOW);
delay(27);
digitalWrite(motor_one_one, LOW);// moteur vertical (1) éteint
digitalWrite(motor_four_one, HIGH);//fait tourner le pseudo vérin (avance)
digitalWrite(motor_four_two, LOW);
}

```

```

digitalWrite(motor_three_one, LOW); // fait tourner le moteur 3 dans le sens
inverse d'une aiguille d'une montre
digitalWrite(motor_three_two, HIGH);
if(digitalRead(sensor_flow)=='TRUE')
{
digitalWrite(motor_three_two, LOW); // on éteint le moteur 3
digitalWrite(motor_four_one, LOW); //fait tourner le pseudo vérin (recul)
digitalWrite(motor_four_two, HIGH);
digitalWrite(motor_one_one, HIGH); // moteur vertical (1) tournesens
aiguille d'unemontre
digitalWrite(motor_one_two, LOW);
if(digitalRead(button_three)== HIGH){
digitalWrite(motor_one_one, LOW);
access=0;
}
}
}
}

}

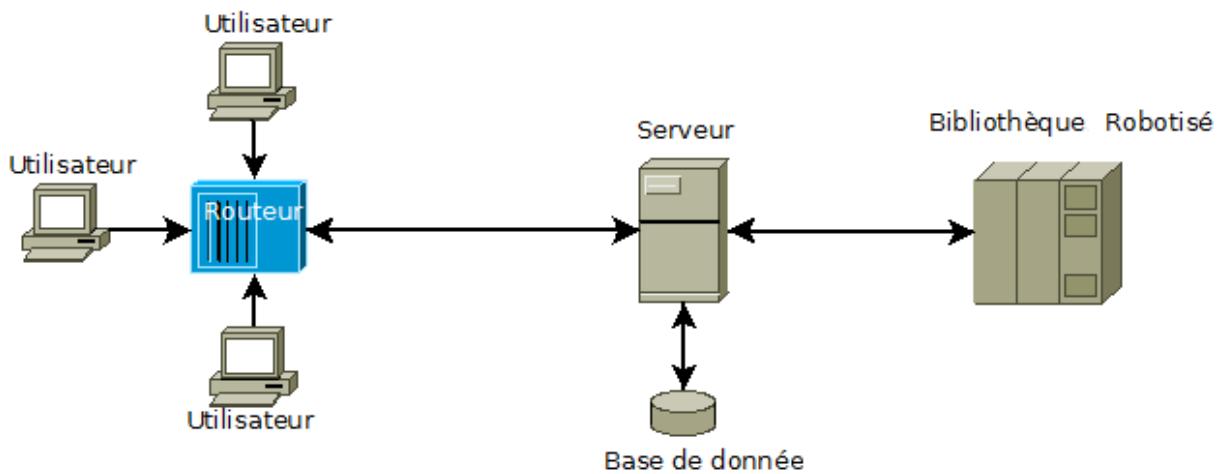
```

Dossier Explication Informatique

Le système

Le Système est prévu pour plusieurs utilisateurs. Le centre du système est le serveur (pouvant être fait avec un PC très basique). Les utilisateurs du système (les ordinateurs) sont connectés à un

routeur qui va rediriger leur connexion internet vers le serveur, on final on va créer un réseau local simple mais permettant un circuit fermé.



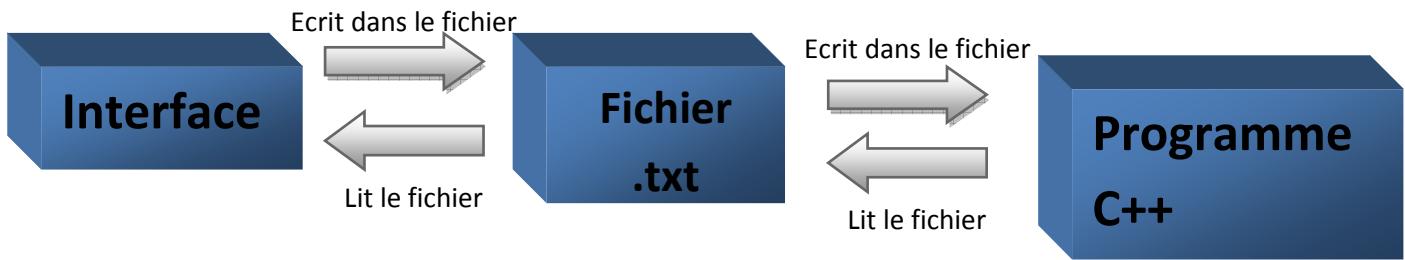
Le serveur

Au niveau de la configuration du serveur la seule spécificité est la présence Obligatoire d'une distribution Linux (Ubuntu, Debian, Fedora ou autre). Au niveau logiciel un serveur LAMP (Linux-Apache-PHP-MySQL) sera installé dessus. Sans aucune modification au niveau des fichiers de configuration. (Tous les logiciels sont gratuits et distribué sous licence Apache, GPL)

- Apache est un serveur http, c'est le plus utilisé au monde avec 65% du marché mondial (HyperText Transfer Protocol) (cf. ANNEXE)
- PHP est un langage informatique interprété, utilisé principalement dans le codage de programme web côté serveur (cf. ANNEXE)
- MySQL est une des bases de données les plus utilisés au monde.

Les programmes de gestion

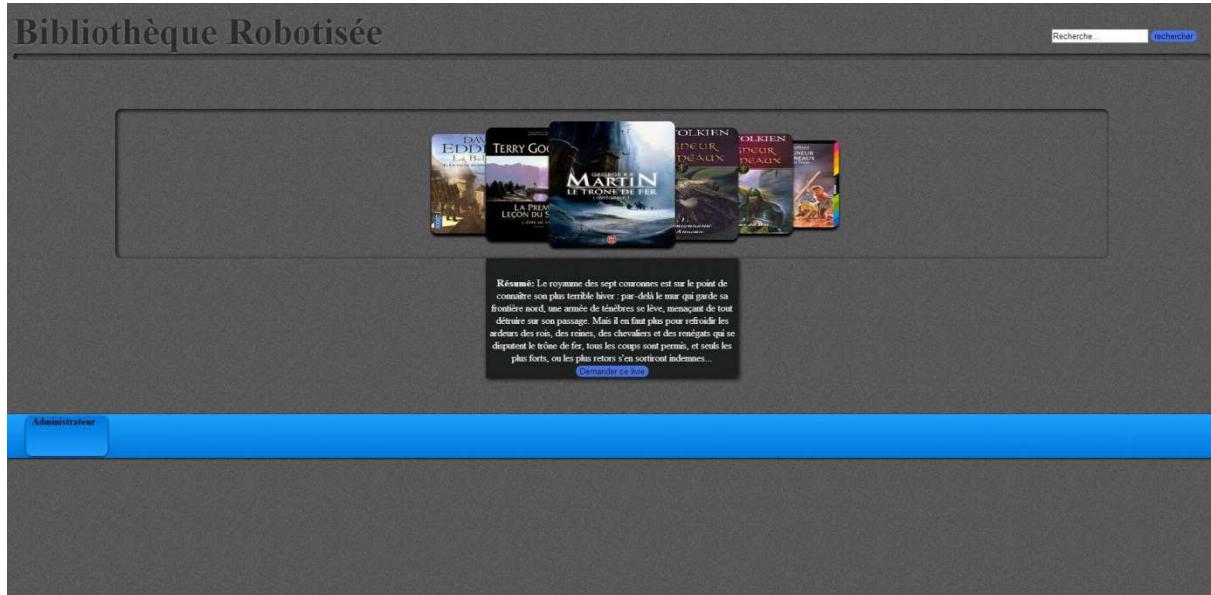
Dans un premier temps l'interface qui sera affiché sur les PC utilisateur sera faite en PHP principalement mais aussi en *Jquery* pour certaines fonctions. Une communication est faite avec le pc.



Nous avons donc l'Interface (php) qui communique avec un programme C++ par le biais d'un fichier .txt. L'interface va écrire dans le .txt les coordonnées du livre demandé que s'il est vide, le programme va lire cela envoyé par le biais de la liaison série à l'ARDUINO toute les informations qu'il en a déduit et laissera le fichier rempli tant que l'ARDUINO n'aura pas par la liaison série renvoyé un octet disant que la tâche est fini cela pour empêcher l'interface de demander un autre livre et de créer des conflits.

L'interface

Cette interface est basé sur la technologie web pour sa simplicité et sa portabilité.



Programme C++ (non terminé)

Le programme est donc en C++ qui est un langage orienté objet. C'est une notion compliquée, une variante de fonction plus souple, facilitant la compréhension du code et la modification. Il permet aussi d'avoir un main relativement léger et aéré. Le code ci-dessous ne possède pas la gestion série. Elle peut et sera faite avec la librairie QextSerialPort.

main.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include "comtxt.h"
using namespace std;

int main()
{
    //****Boucle d'écriture****/
    inttaille;
    ifstream open("C:/Users/vincent/Desktop/tpe.txt");
    comtxtcommu;
    commu.test(open);
}
```

comtxt.h

```
#ifndef COMTXT_H
#define COMTXT_H
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>

class comtxt
{
public:

    //Constructeurs
    comtxt();
    ~comtxt();
    void test(std::ifstream&open);
    void dntxt(std::ifstream&open);
    void rmtxt();
    void traitement();
    void convert();
```

```

void serialDisplay();
private:

int taille;
int open;
std::string dnx;
std::string dny;
int x;
int y;
bool middle = false;

/* *** gestion moteur *** */
int moteur1 = 0;
int moteur2 = 0;
int moteur3 = 1;
int time_1;
int time_2;
int time_3;

};


```

#endif// COMTXT_H

comtxt.cpp

```

#include "comtxt.h"
#include <fstream>
#include <iostream>
#include <sstream>

comtxt::comtxt()
{
    //ctor
}

comtxt::~comtxt()
{
}
void comtxt::test(std::ifstream& open)
{
    if(open){
        /* *** Boucle de test *** */
        open.seekg(0, std::ios::end); //On se déplace à la fin du fichier
        taille = open.tellg();
        if(taille < 1)
        {
            std::cout<<"Fichier vide";
        }
        else
        {


```

```

comtxt::dntxt(open);
comtxt::convert();
comtxt::traitement();
comtxt::serialDisplay();
}

}

}

void comtxt::dntxt(std::ifstream& open)
{
open.seekg(0, std::ios::beg);
std::cout<<"taille = "<<taille<<std::endl;
getline(open, dnx);
getline(open, dny);
}

void comtxt::rmtxt()
{

std::ofstream write("C:/Users/vincent/Desktop/tpe.txt");
if(write){
write<< "";
write.close();
}
else {
}

}

void comtxt::convert()
{

std::istringstream convertx(dnx);
convertx>> x;
std::istringstream converty(dny);
converty>> y;
std::cout<< "x" << dnx << "/" << x << "y" << dny << "/" << y ;
}

void comtxt::traitement()
{
if(y == 1){moteur1 = 0;} else{moteur1 = 1; time_1 = 30*100*(d*);}
if(x == 1){moteur2 = 0;} else{moteur2 = 1; time_2 = 30*100*(d*);}
//30cm/s

}

void comtxt::serialDisplay()
{
std::cout<<std::endl<< moteur1 << moteur2 << moteur3 << time_1 << time_2 << time_3;
}

```

ANNEXE

HTTP

L'HyperText Transfer Protocol, plus connu sous l'abréviation HTTP — littéralement « protocole de transfert hypertexte » — est un protocole de communication client-serveur développé pour le World Wide Web.

HTTP est un protocole. Il peut fonctionner sur n'importe quelle connexion fiable, dans les faits on utilise le protocole TCP comme protocole de transport. Un serveur HTTP utilise alors par défaut le port 80 (443 pour HTTPS).

Les clients HTTP les plus connus sont les navigateurs Web (Firefox, Google Chrome, Opera,...) permettant à un utilisateur d'accéder à un serveur contenant les données.

Ces clients se connectent à des serveurs HTTP tel qu'Apache HTTP Server.

Historique

HTTP a été inventé par Tim Berners-Lee avec les adresses Web et le langage HTML pour créer le World Wide Web, l'internet .La première version reste encore partiellement utilisable en 2007, connue sous le nom de HTTP/0.9.

En mai 1996, HTTP/1.0 devient finalement standard de l'IETF et est décrit dans la RFC 1945. Cette version supporte les serveurs HTTP virtuels, la gestion de cache et l'identification.

En janvier 1997, HTTP/1.1 est décrit dans la RFC 2068 de l'IETF, puis dans la RFC 2616 en juin 1999. Cette version ajoute le support du transfert en pipeline et la négociation de type de contenu (format de données, langue).

APACHE

Le logiciel libre Apache HTTP Server (Apache) est un serveur HTTP créé et maintenu au sein de la fondation Apache. C'est le serveur HTTP le plus populaire du World Wide Web. Il est distribué selon les termes de la licence Apache.

Historique

Apache est apparu en avril 1995. Au début, il s'agissait d'une collection de correctifs et d'additions au serveur NCSA HTTPd 1.3, qui était dans le domaine public et le serveur HTTP alors le plus répandu.

De cette origine, de nombreuses personnes affirment que le nom Apache vient de a patchy server, soit « un serveur rafistolé ». Par la suite, Apache a été complètement réécrit, de sorte que, dans la version 2, il ne reste pas de trace de NCSA HTTPd.

Au début, Apache était la seule alternative sérieuse et libre au serveur HTTP de Netscape (iPlanet, maintenant Sun ONE). Depuis avril 1996, selon l'étude permanente de Netcraft, Apache est devenu le serveur HTTP le plus répandu sur Internet.

Part de marché d'Apache :

- En mai 1999, il faisait tourner 57 % des serveurs Web, début 2004, il était à environ 69 % de parts de marché, il ne détient plus que 50,61 % du marché à la fin du mois de janvier 2008.
- En février 2008, Apache représente 50,93 % des parts de marché.
- En novembre 2008, 72,45 % des parts du marché pour Apache.
- En novembre 2011, 65,00 % des parts du marché pour Apache

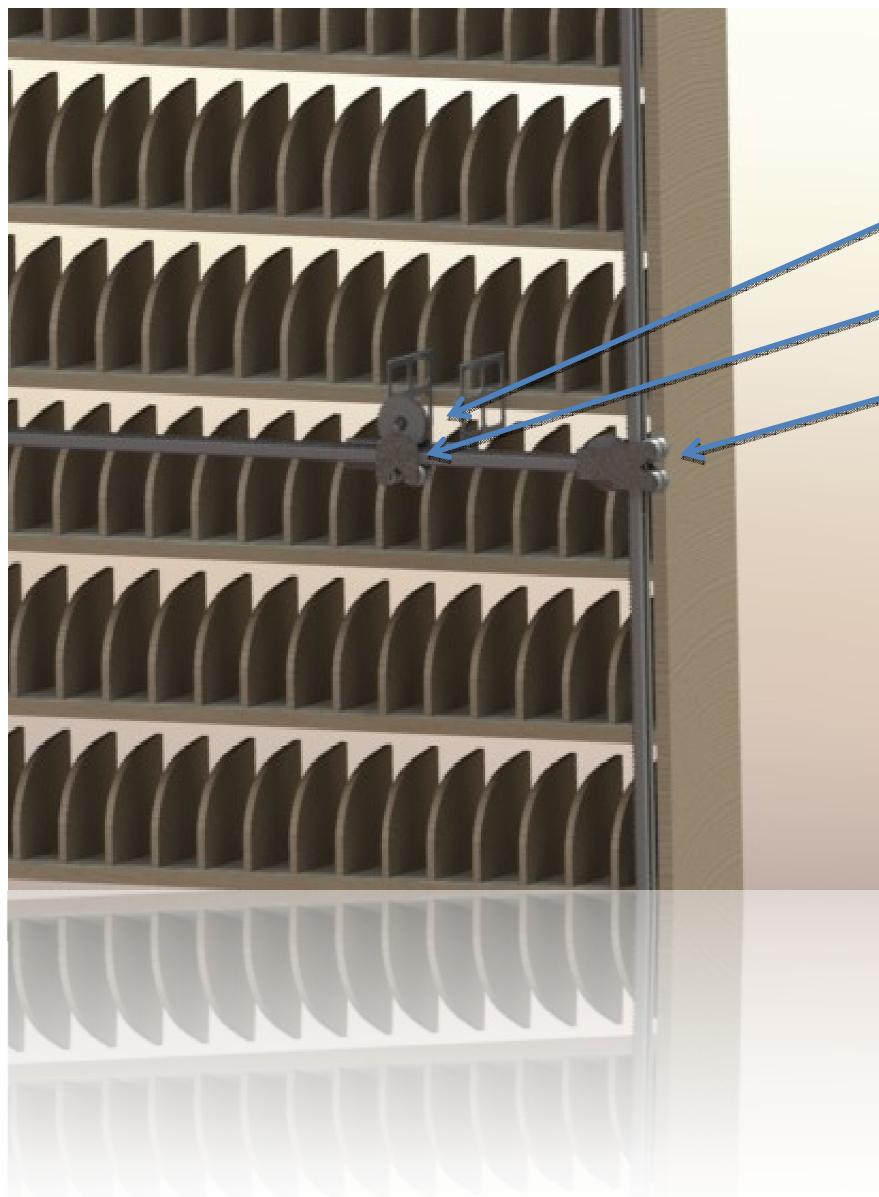
La version 2 d'Apache possède plusieurs avancées majeures par rapport à la version 1, entre autres le support de plusieurs plates-formes (Windows, Linux et UNIX, entre autres), le support de processus légers UNIX, une nouvelle API et le support IPv6.

La fondation Apache (Apache Software Foundation ou ASF) a été créée en 1999 à partir du groupe Apache (Apache Group) à l'origine du serveur en 1995. Depuis, de nombreux autres logiciels utiles au World Wide Web sont développés à côté du serveur HTTP.

MECANIQUE

Introduction :

Dans cette partie du dossier, nous allons expliquer les choix des systèmes mécaniques. Nous fournirons tout d'abord une explication du fonctionnement du système mécanique, puis des calculs pour permettre le choix des pièces et enfin les pièces qui ont été finalement choisies. Les calculs seront accompagnés de schéma, d'un aperçu de l'objet ou bien d'un plan de ce dernier. En annexes, seront placées les fiches techniques des pièces.



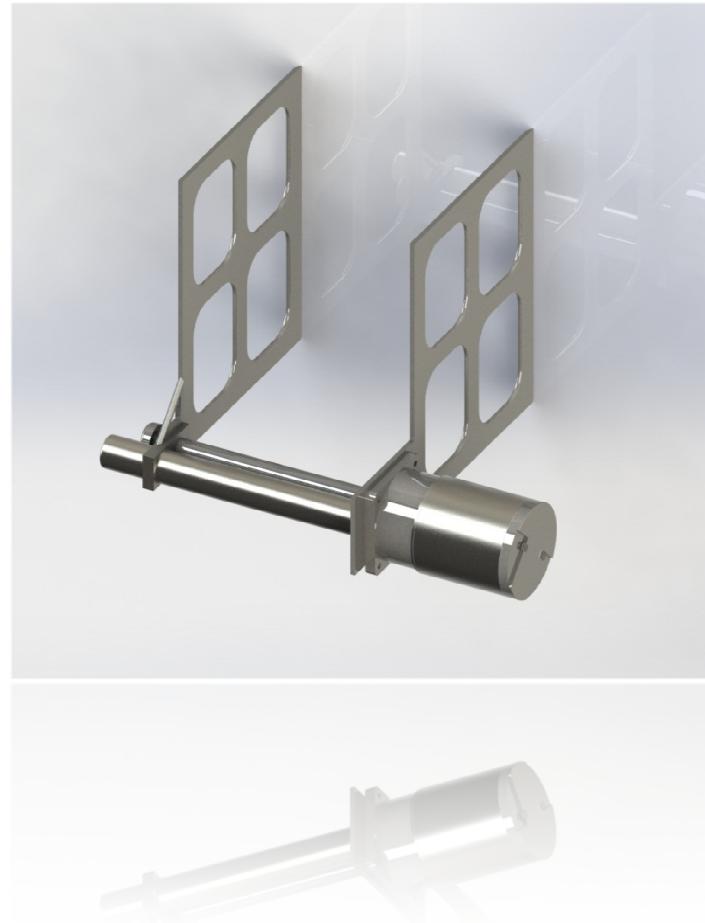
La Pince

Le chariot horizontal

Le chariot vertical

Calculs et explications

La pince



La pince est un élément clé de la bibliothèque robotisée. Elle sert à prendre les livres, les sortir de leurs cases et les déposer dans la case dé réception où l'utilisateur n'aura plus qu'à le prendre. Pour cela, la pince doit être capable de se placer devant la bonne case, d'avancer dans la case (à l'aide d'un pseudo-vérin), de serrer le livre, de reculer et ensuite se rendre à la case où elle déposera le livre pour l'utilisateur. La machine fonctionne bien entendu dans le sens inverse, elle peut ranger les livres que l'on lui donne.

La pince se décompose en plusieurs mouvements :

- La pince en elle-même, qui sert le livre.
- Le vérin qui lui permet de s'engager dans une case puis d'en sortir.

Force nécessaire pour maintenir le livre dans la pince :

Pour maintenir le livre dans la pince sans que ce dernier ne glisse, il nous faut calculer le poids que la gravité génère sur le livre. Ensuite nous calculons la force nécessaire selon le coefficient d'adhérence (celui du métal sur le bois), pour que la pince serre suffisamment pour qu'il tienne.

Masse maximal du livre : 1 kg

Gravité terrestre : 9,81 N/kg

F_{livre} est la force exercée sur le livre par la gravité :

$$F_{\text{livre}} = m * g = 1 * 9,81 = 9,81 \text{ N}$$

$\tan \varphi$ est le coefficient d'adhérence entre l'acier et le livre soit environ : 0,6 (métal sur bois).

F_t est la force tangentielle, ici :

$$F_t = F_{\text{livre}}$$

F_a est la force axiale, la force nécessaire pour maintenir le livre dans la pince.

$$\tan \varphi = \frac{F_t}{F_a}$$

$$Fa = \frac{Ft}{\tan \varphi} = \frac{9,81}{0,6} = 16.35 \text{ N}$$

Il est donc nécessaire que la pince appuie avec une force d'au moins 16,35 Newton sur le livre pour que celui-ci ne glisse pas.

Couple nécessaire pour serrer la pince à l'aide d'un système de vis-écrou :

La pince va serrer à l'aide d'un système de vis écrou, il nous faut donc calculer le couple nécessaire pour le moteur afin de pousser l'écrou suffisamment fort.

F_t est la force tangentielle, F_a la force axiale (ici 16,35 N), r est le rayon de la vis (7 mm), α est l'angle d'inclinaison du pas de la vis (30° ici) et C le couple recherché en N/mm.

$$\frac{Ft}{Fa} = \tan \alpha \text{ or } Ft = \frac{C}{r}$$

$$\text{Donc } \frac{C}{Fa} = \tan \alpha \text{ et } C = \tan \alpha * Fa * r$$

$$= \tan 30^\circ * 16.35 \text{ N} * 7 \text{ mm}$$

$$= 66 \text{ Nmm} \text{ soit } 0,06 \text{ Nm.}$$

Le moteur permettant de serrer la pince doit donc disposer d'un couple de 0,06 Nm pour pouvoir maintenir le livre serré. [cf Annexe : A].

Vitesse d'avancement de la pince pour un tour de moteur (pseudo-vérin)

La pince est poussé par un pseudo-vérin composé d'un système de vis-écrou, il nous faut donc calculer la vitesse à laquelle ce dernier va permettre à la pince d'avancer.

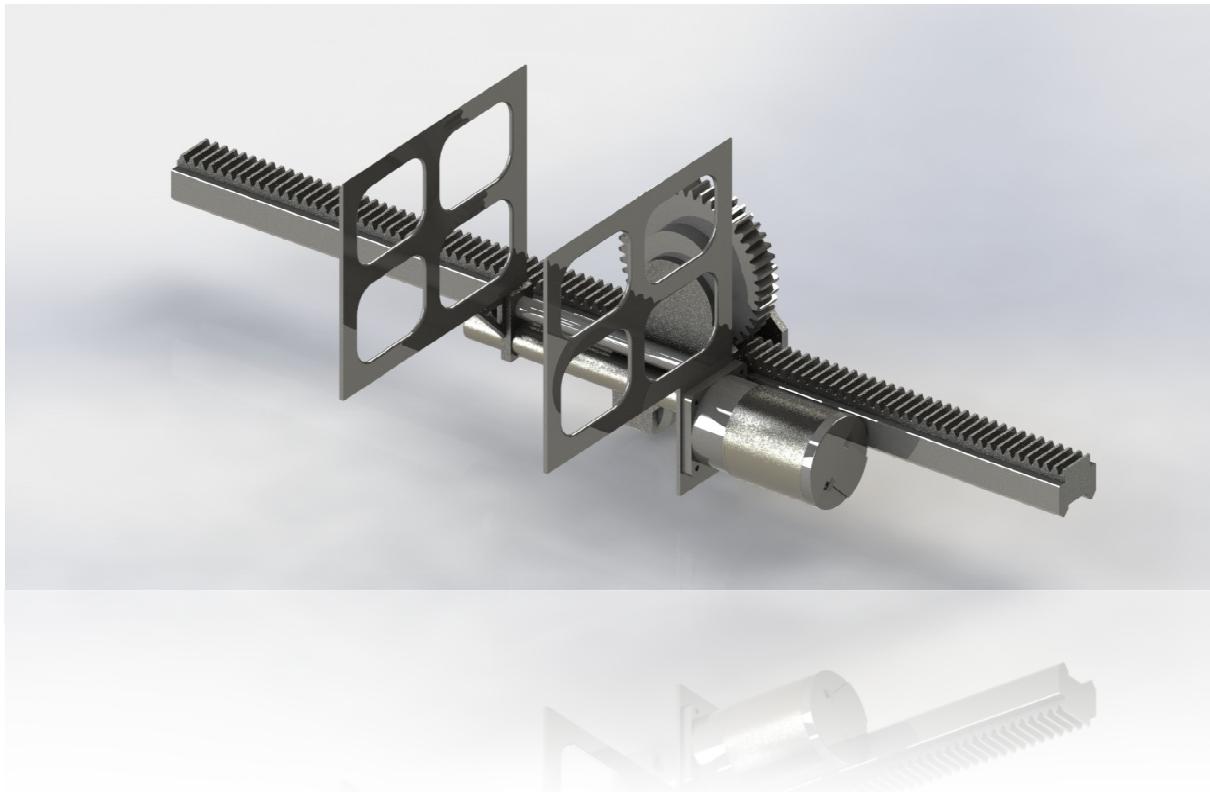
V est la vitesse d'avance, N la vitesse de rotation (en tour/minute) et p le pas de la vis (ici 3). Le moteur tourne à 395 tr/min.

$$V = N * p = 395 * 3 = 1185 \text{ mm/min}$$

$$\frac{1185 \text{ mm/min}}{60} = 2 \text{ cm/s}$$

Le vérin permettra à la pince d'avancer à une vitesse de 2cm/s. Pour les références du moteur, se référer à l'Annexe A.

Le chariot horizontal



Vitesse de déplacement horizontal :

La pince et le système de vérin qui lui est associé sont montés sur un chariot composé de roues guidées par un demi-rail, ce chariot avance grâce à un moteur DC qui fait tourner un engrenage sur la crémaillère de l'axe horizontal. Il nous faut donc calculer la vitesse à laquelle il va se déplacer.

En utilisant un engrenage de 60,0 mm de diamètre, sur une crémaillère de module 1,5 et un moteur tournant à 99,0 tour/minute en charge. [Moteur 2]

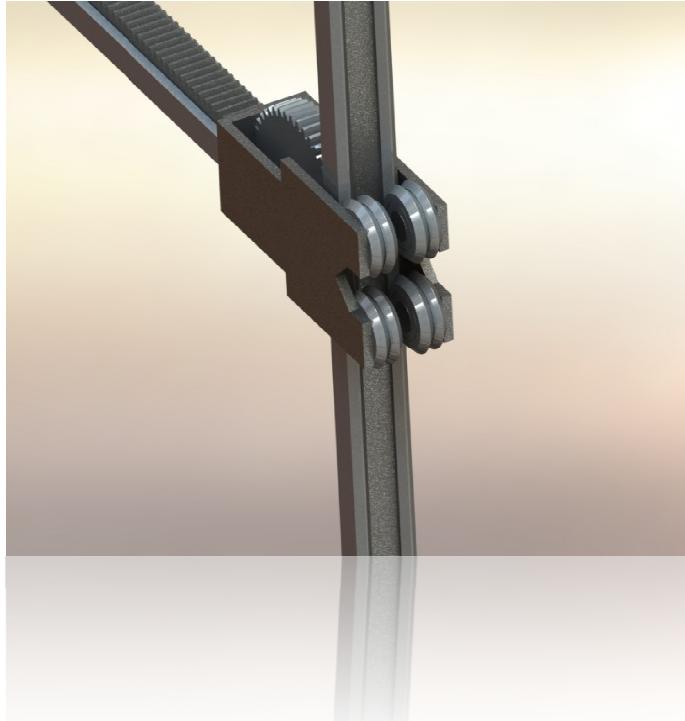
Vitesse linéaire (l'engrenage roule sur la crémaillère):

$$V = \omega * R = 99 * \frac{2\pi}{60} * 30 = 311,0 \text{ mm/s}$$

Soit 31,1 cm/s.

Le chariot portant la pince se déplace donc sur la crémaillère à une vitesse de 31,1 cm/s.

Le chariot vertical



Vitesse et couple de déplacement du chariot vertical

Le chariot vertical se déplace verticalement sur deux crémaillères. Ces chariots doivent avoir un couple plus important car ils doivent vaincre la gravité qui attire l'axe horizontal qui comprend le chariot horizontal et la pince. Ils devront être équipés de freins pour permettre à la pince de manœuvrer et ensuite pour ralentir la chute pour la descente.

$$V = \omega * R = 99 * \frac{2\pi}{60} * 30 = 2970 \text{ mm/s} = 311,0 \text{ mm/s}$$

$$P = C * \omega = 0,200 * 99 * \frac{2\pi}{60} = 2,0 \text{ W}$$

$$F = \frac{P}{V} = \frac{2,0}{311,0 * 10^{-3}} = 6,43 \text{ N}$$

Annexes

Références des moteurs

Les moteurs ont été choisis dans le catalogue de HPC, les modèles 3D sont fournis également par HPC.(www.hpc-europe.com + Ouvrages)

- A -

Référence	MAX40-12-006
Tension [V]	12V
Rapports	6:1
Nb.d'étages	2
Rendement	81%
[mm]	64.7
Vitesse à vide [t/min]	682.0
Vitesse en charge [t/min]	395.0
Couple nominal [Nm]	0.069
Courant nominal [mA]	510.0

Moteur servant à serrer la pince.

Moteur servant de vérin pour l'avancée de la pince.

- B -

Tension (V)	12 V
Rapports	30 :1
Vitesse à vide (t/min)	136
Vitesse en charge (t/min)	99
Couple nominal (Nm)	0,200
Courant nominal (mA)	340 mA

Moteur servant au déplacement des chariots (verticaux et horizontaux).

SYNTHESE & CARNET DE BORD

GUILLAMOT
Vincent

I) Choix du sujet

Le Jeudi 13 Septembre nous avons formé les groupes. Je me suis associé avec Pascal PHELIPOT, Guillaume BONNET et William Bourdon.

Nous avons fait des séances de réflexion sur le sujet du TPE, nous avons donc fait des propositions, Souris scanneuse, liseuse mécanique, distributeur de boisson personnalisable, ces projet étant « inutile » et non intéressant au niveau des matières devant être impliqué dans le projet.. Un jour nous avons eu l'idée que nous avons choisi, la Bibliothèque Robotisée un projet jugée innovant, et un défi pour le groupe, avec une bonne quantité de choses à dire.

La bibliothèque a été adopté par tout le monde sauf Guillaume mais au final, ce projet au final a été adopter par tout le groupe.

II) Les Séances

Les Séances de TPE faites au lycée nous ont permis surtout d'affiner la réflexion sur le projet, et de posé les bases pour un travail personnel, par équipe chez soi. Elles ont permis de mettre en place une vrai dynamique de travail dans le groupe pour le TPE. Nous avons donc établit des équipes :

- Electronique : Vincent GUILLAMOT, Guillaume BONNET
- Mécanique : Pascal PHELIPOT, William BOURDON

Ces équipes travaillaient en collaboration étroite.

III) L'apport du TPE

Le TPE m'a apporté personnellement, des connaissances au niveau électronique, informatique et mécanique. Au niveau électronique les principales connaissances qui m'ont été apporté est au niveau du fonctionnement des moteurs, et des alimentations. Le parti mécanique m'a plus apporté des connaissances au niveau des calculs des couples des moteurs, des calculs qui ont comme sujets les moteurs.

Au final ça m'a apporté un savoir rapidement et de façons autonomes.

BONNET

Guillaume

I) Choix du sujet

Le jeudi 13 Septembre, nous devions faire des groupes et partir sur un TPE. Ce choix est important, car il est nécessaire de choisir des camarades sérieux et qui souhaite travailler dans de bonne condition. Au final, je me suis mis en groupe avec GUILLAMOT Vincent, PHELIPOT Pascal et BOURDON William.

Pendant plusieurs semaines, en groupe durant des moments de pose, nous avons réfléchie sur un projet qui pourrait être intéressant. Nous avons fini par nous mettre d'accord sur la Bibliothèque Automatisée en, c'était un projet jugée innovant, et un défi pour le groupe, avec une bonne quantité de choses à dire et à faire ;

Au début, le projet m'as déplu, car je n'y voyais pas vraiment d'utilité, enfin de compte, mes co-équipiers mon convaincu et que ce projet est très bien. Je les ai donc suivit, et je n'en suis pas déçu.

II) Les Séances

Les Séances de TPE faites au lycée nous ont permis surtout d'affiné la réflexion sur le projet, et de posé les bases pour un travail personnel, par équipe chez soi. Elles ont permis de mettre en place une vrai dynamique de travail dans le groupe pour le TPE. Nous avons donc établit des équipes :

- Electronique et INFORMATIQUE: Vincent GUILLAMOT, Guillaume BONNET
- Mécanique : Pascal PHELIPOT, William BOURDON

Ces équipes travaillaient en collaboration étroite.

III) L'apport du TPE

Ce TPE m'as permis d'apprendre beaucoup de chose, au niveau mécanique, électrotechnique et informatique. J'avoue avoir eu plus de plaisir sur le côté informatique que sur le côté mécanique qui m'a paru plus « technique ». Je suis content d'avoir pris part à ce projet et de l'avoir réalisé.

Dossier TPE

Synthèse Personnel



I) Sujet et projet de TPE

Nous avons dans un premier temps commencer à faire les groupes de TPE (Le Jeudi 13 Septembre). Je me suis associer à Vincent GUILLAMOT, Guillaume BONNET, Pascal PHELIPOP. Notre première idée, venant de Vincent GUILLAMOT, était de réaliser un distributeur de boisson personnalisable, notre deuxième idée : Une souris scanneuse, malheureusement nous avons trouver sur internet que cela exister déjà.

Après cela, nous nous sommes orienter vers autres choses, et c'est en parlant de nos activités que nous avons trouver la bases de notre TPE, Pascal PHELIPOP a proposer de crée une machine qui servirait à tourner les pages d'un livre quand on le lit, ensuite nous avons enfin trouver l'idée de notre TPE, proposer par Vincent GUILLAMOT, une bibliothéque robotisée ! Ce projet a directement plus a tous le groupe, et nous avons donc choisi celui-ci.

II) Séances de TPE

Nous avions donc un sujet pour notre TPE, et avons commencés à nous répartir les tâches. Vincent GUILLAMOT a directement choisi de s'occuper de la partie électronique, Pascal PHELIPOP quand à lui a voulu s'occuper de la partie mécanique, j'ai ensuite choisi de rejoindre Pascal PHELIPOP sur la partie mécanique et Guillaume BONNET a rejoint Vincent GUILLAMOT. Notre TPE à évoluer à un rythme régulier chaque Jeudi mais aussi chez nous ou chacun continuer son travail en partageant ces dossier via Dropbox et Skype. Pour la

partie mécanique, nous avons beaucoup sollicité le logiciel Solid Works, et pour la partie électronique le logiciel ISIS.

III) Intérêt pour le sujet et apport personnel

Ce sujet m'a intéressé dès le début, l'idée de travailler en groupe sur un projet utile et innovant me fort plus. Ce projet vise les bibliothèques, médiathèques, les CDI et autres..... Il facilite à la fois la recherche, l'accès et le rangement des livres. J'ai aidé Pascal PHELIPOPOT pour les calculs et aussi la réalisation Solid Works, ou j'ai eu l'idée de faire des rails verticaux pour une meilleure soutenance du rail horizontal.

Dossier TPE

Synthèse Personnel

I) Projet de TPE

Le Jeudi 13 Septembre, nous avons procédé a l'élaboration des groupes. Nous étions 4 avec Guillaume BONNET, William BOURDON, Vincent GUILLAUMOT et moi même. Notre première idée, venant de Vincent GUILLAMOT, était de réaliser un distributeur de boisson personnalisable, notre deuxième idée : Une souris scanneuse, malheureusement nous avons trouver sur internet que cela exister déjà.

Suite à de longues journées de réflexion par rapport aux sujet de TPE que nous pourrions abordé, c'est finalement la Bibliothèque Robotisée qui m'a séduite, par la quantité de travail que l'on devrait fournir du côté mécanique, mais aussi électronique, cela m'a semblait être le plus équitable.

II) Séances de TPE

Je me suis essentiellement préoccuper de la partie mécanique, je me suis mis avec BOURDON William, pour travailler sur Solidworks et modéliser cette pièce. Le plus problématique a été l'application des formules.

III) Apports personnels.

J'ai énormement appris du côté mécanique, avec beaucoup de formules et de fonctionnalités sur Solidworks que je ne connaissais pas. Maintenant je sais comment réaliser un produit complet grâce à mes capacités acquises tout au long de ce projet.

BIBLIOGRAPHIE

Partie mécanique :

Nous remercions le logiciel Solidworks qui est un logiciel propriétaire de conception assistée par ordinateur 3D fonctionnant sous Windows. Site web : <http://www.solidworks.fr/>



Nous tenons aussi à remercier la firme HPC pour ses produits et ses catalogues gratuits. Site web: <http://www.hpc-europe.com>



Partie électrotechnique :

Nous remercions Arduino.



Site web : <http://arduino.cc/fr/>

Nous remercions aussi le site du zéro qui est un site web, qui propose des tutoriels pour l'informatique et l'électronique.

Site web : <http://www.siteduzero.com/>



Globalement :

Nous remercions plus particulièrement Dropbox, qui nous a permis de travailler ensemble à la maison :

Site web : <https://www.dropbox.com/>



Mais aussi skype :

Site web : <http://www.skype.com/fr/>



CONCLUSION

Pour conclure, nous pouvons donc dire que ce projet vise tout une catégorie de personnage, qui va du simple lecteur qui a beaucoup de livre chez lui, jusqu'à des documentalistes et bibliothécaires. Ce projet a un réel potentiel, malgré qu'il n'est pas adaptable sur toutes les bibliothèques et qu'il faut avoir déjà au préalablement une bibliothèque similaire à la nôtre.

La Bibliothèque Robotisée simplifie la vie de ces personnages, et permet une automatisation de la sélection des livres. Grâce à son interface web, la communication de la bibliothèque avec l'utilisateur est harmonieuse.

Nous avons été heureux d'avoir participé au projet.