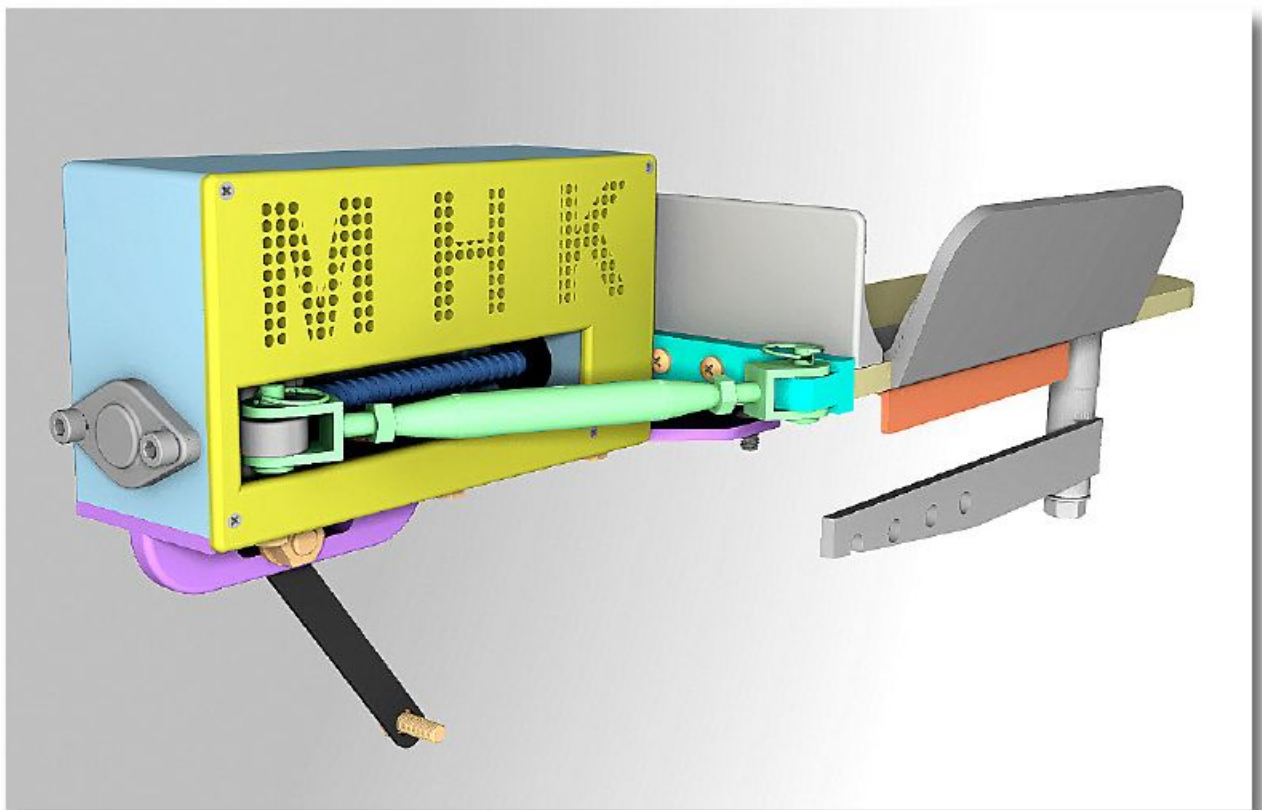


Accoudeoir pivotant motorisé

V 1.42



La version 1.41 annule et remplace toutes les documents précédents afférents à la version motorisée de ce système.

La version 1.42 modifie la version 1.41 seulement pour la partie électronique (schéma et programme Arduino).

Accoudoir pivotant motorisé

V1.42

Cette version motorisée fait suite au projet de développement d'un accoudoir pivotant expérimental à commande manuelle (CF : V.0)

Elle se substitue à la V1.3 à entraînement direct par pignons et tige filetée, jugée trop faible.

Objectif : Concevoir un accoudoir motorisé pivotant approximativement au niveau du poignet afin de pouvoir écarter le coude pour changer de position lorsque celle-ci devient inconfortable. Les mouvements de cet accoudoir doivent être pilotés par le biais du dispositif électrique de commandes en place sur le fauteuil et disposant de 4 relais dont 2 seulement sont nécessaires.

Impératifs : Le système doit pouvoir supporter une partie du poids du haut du corps et être verrouillé dans la position choisie sans risque pour l'utilisateur tout en étant suffisamment puissant pour fonctionner avec le poids du corps en appui dessus.

Fonctionnement : Suite à l'expérience faite avec la version manuelle, on reprend le même principe en y ajoutant un système d'actionneur motorisé piloté par l'utilisateur du fauteuil, lui-même.

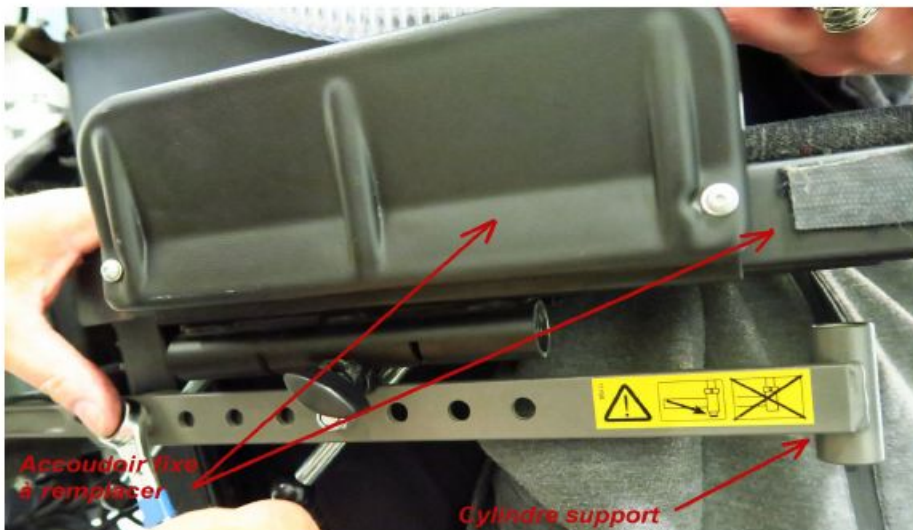
La plaque crantée de la version manuelle n'est plus justifiée puisque c'est le moteur qui maintient la position et c'est donc une plaque de glissement lisse sur laquelle reposer l'accoudoir. (ABS)

Fabrication de l'accoudoir *(Sa fabrication est la même que pour la version manuelle.)*

Le dessus en plastique de l'accoudoir fixe, actuellement en place, est retiré ainsi que la pièce de blocage latéral du bras.

On garde la base métallique sur laquelle on viendra poser le pont de glissement.

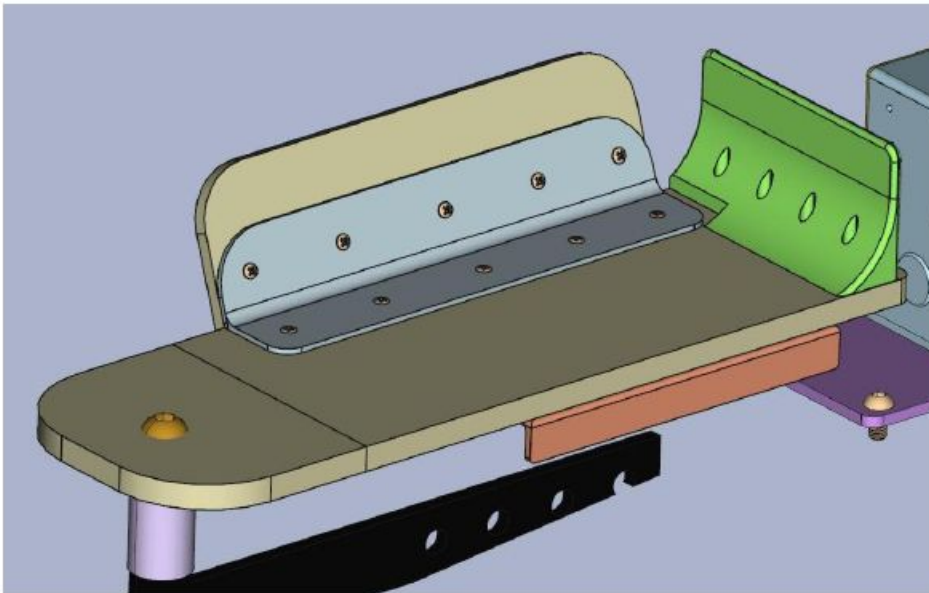
Le tube de 50 mm soudé en bout du bras métallique situé sous l'accoudoir servira de support pour le nouveau système (*cylindre support*).



Un boulon de 10 mm sur 105 mm de long sert de liaison avec ce cylindre support et fait office d'axe de rotation. Il est passé dans un fourreau cylindrique pour rattraper le diamètre intérieur du cylindre support (15 mm).

Une entretoise cylindrique, imprimée en 3D, en appui sur le cylindre support permet de maintenir l'accoudoir au niveau souhaité. (env. 40 mm au dessus du cylindre support).

La pièce d'appui est faite dans une planchette de bois de 8 ou 9 mm d'épaisseur. Elle est complétée par une autre planchette de même épaisseur servant de butée latérale, à laquelle elle est fixée par collage-vissage selon un angle d'environ 110 °.



NB : suite aux test de la version manuelle, on a augmenté la hauteur de la butée latérale et ajouté une butée de coude pour éviter que le bras glisse vers l'arrière.

L'ensemble de l'accoudoir, y compris la butée latérale et la butée de coude, est recouvert de mousse autocollante épaisse pour un meilleur confort.

Le pont de glissement reste fixe et se positionne sur le support métallique. Il sert aussi de butée aux axes en T de soutien de l'ensemble.

Fabrication de l'actionneur

L'actionneur est muni d'un moteur (*Nema 17 de 40 mm*) qui fait tourner une tige filetée trapézoïdale par l'intermédiaire d'une courroie crantée GT2. Les poulies ont un rapport de 2,5 afin d'accroître la puissance de rotation de la tige filetée, donnant ainsi un couple d'environ 1 Nm. Sur celle-ci un chariot va et vient selon le sens de rotation dans des limites délimitées par des contacts de fin de course.

Le chariot est relié en biais à l'accoudoir par un ridoir à chapes M6 (*pièce d'accastillage inox réglable en longueur de 140mm à 210 mm*). L'autre extrémité du ridoir est fixée à la plaque d'appui de l'accoudoir par un palonnier vissé sur son champ. Il est réglé à 153 mm de long.

Le déplacement du chariot sur la tige filetée provoque donc l'écartement de l'accoudoir dans le plan horizontal selon un axe perpendiculaire à celui de la tige filetée.

La tige filetée est coupée à 154 mm de long et les extrémités sont arrondies à la meuleuse. Elle est supportée par des roulements à billes et maintenue en place à chaque extrémité par des billes prisonnières en butée.

NB : Cette tige filetée ne nécessitera pas d'être lubrifiée car la surface de frottement du chariot est assurée par un écrou trapézoïdal à très faible friction (Iglidur).

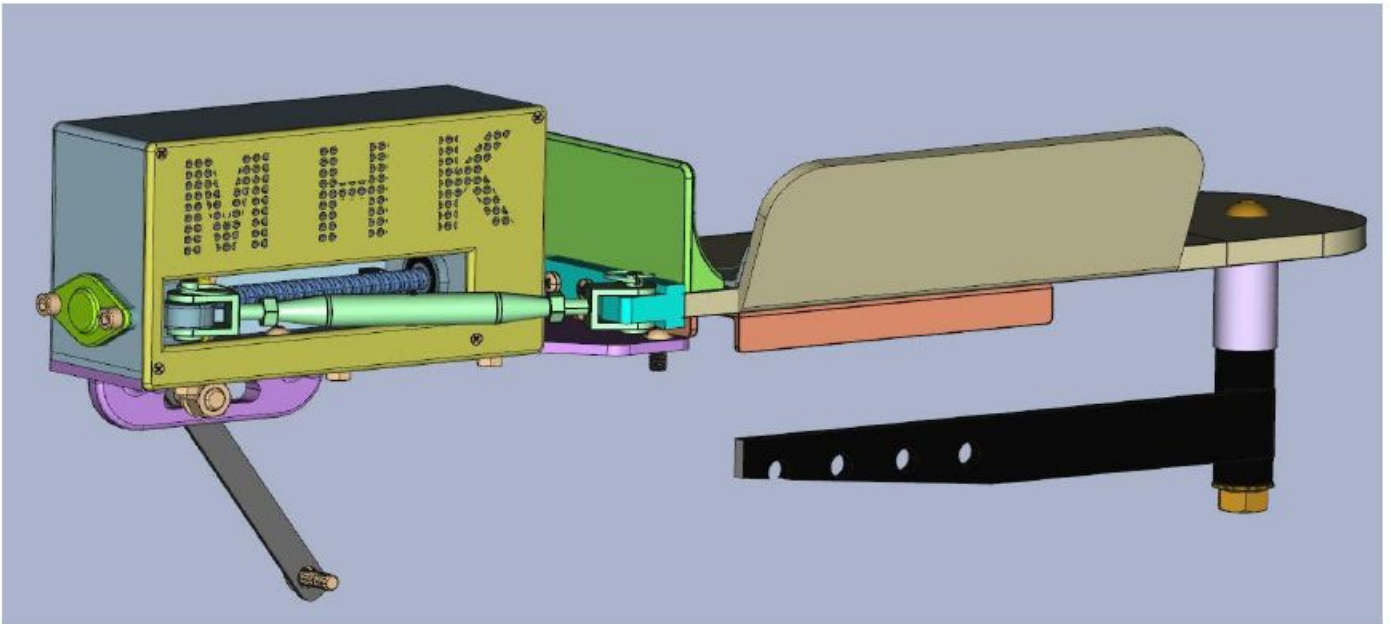
Note sur le choix de la disposition du système.

Lorsque l'accoudoir est en position fermée (au plus près du corps), la force d'appui du bras sur l'accoudoir s'exerce à 95% de façon verticale. Par contre, plus l'accoudoir pivote vers l'extérieur, plus la force d'appui du bras a une composante horizontale importante.

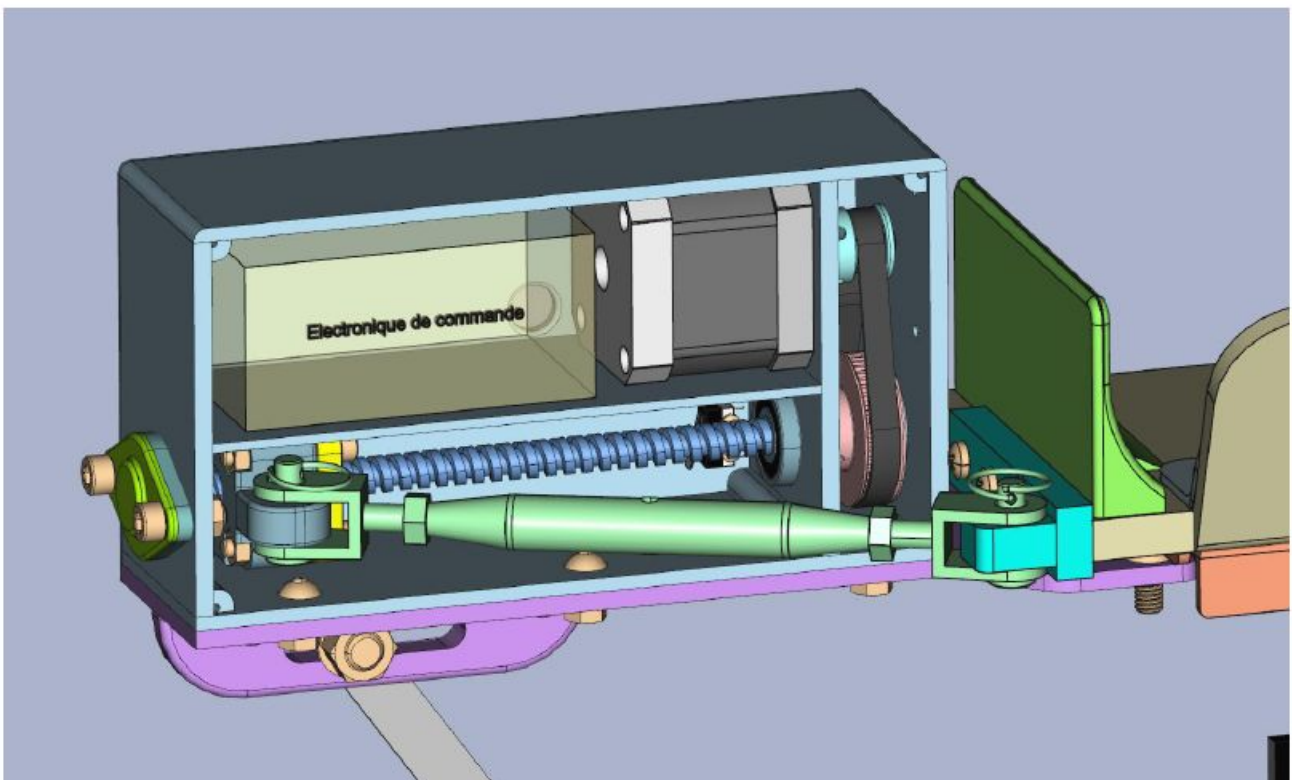
L'utilisation d'une poussée triangulée permet d'exercer une force augmentant avec l'ouverture de l'accoudoir. Ceci est surtout important lors de la fermeture de l'accoudoir où il faut compenser la poussée latérale contraire du bras.

En outre cette disposition permet d'avoir un réglage plus fin aux grandes ouvertures de l'accoudoir et de permettre un positionnement plus précis.

Vue d'ensemble :

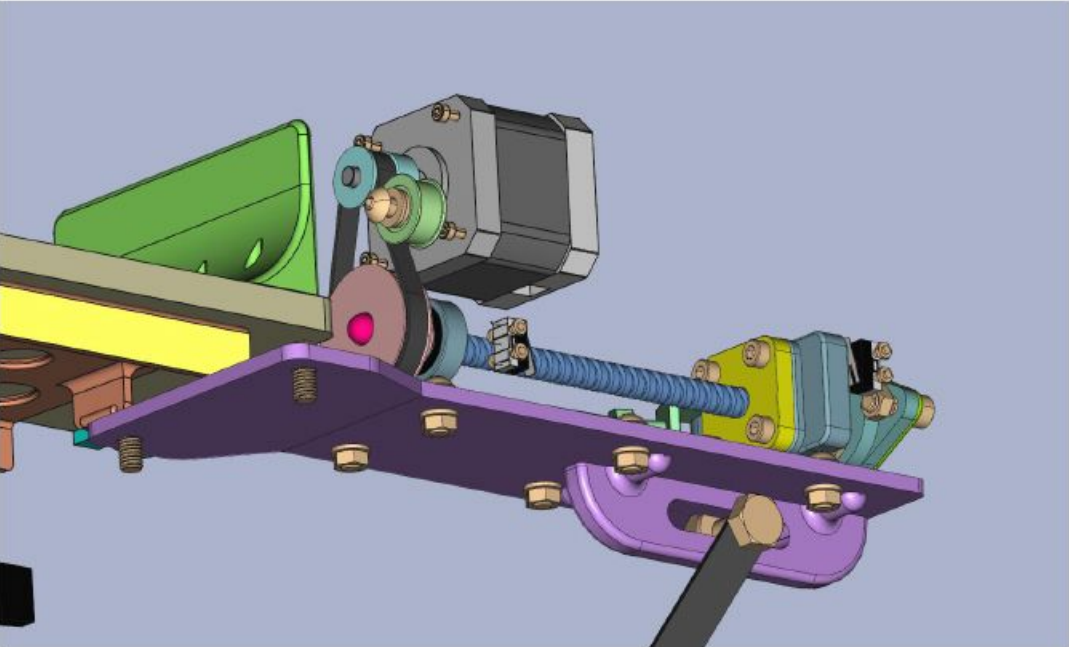
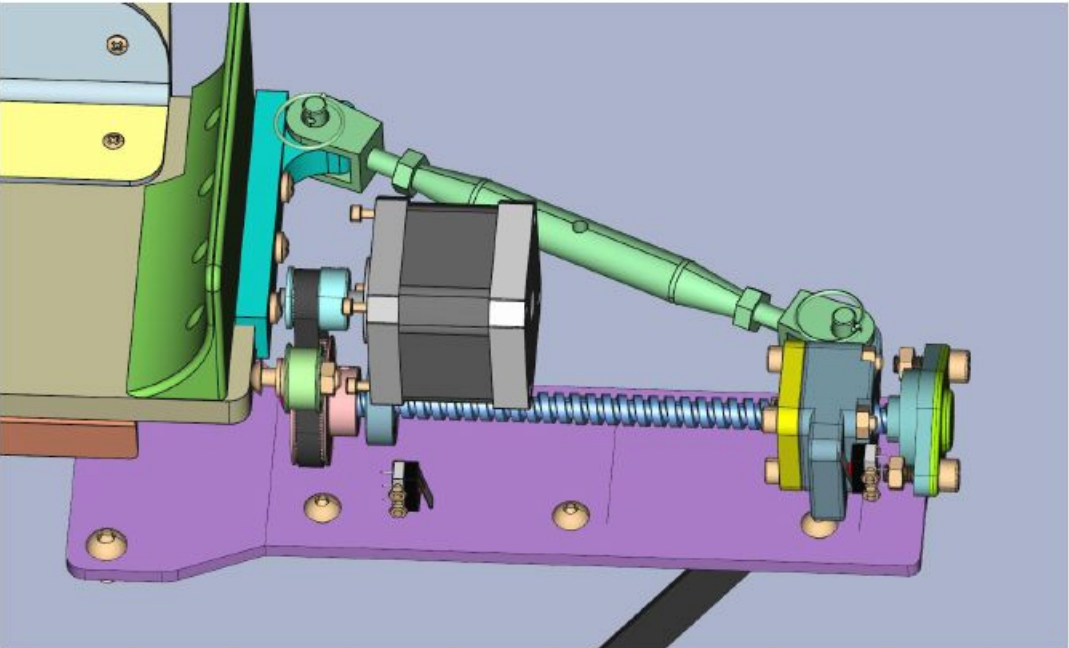
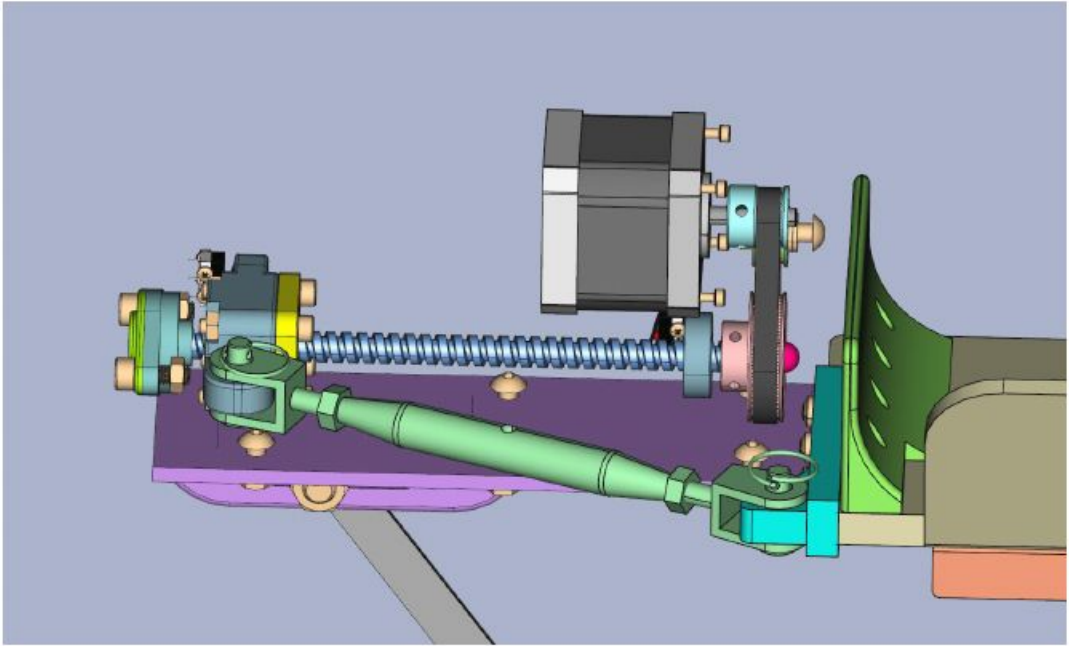


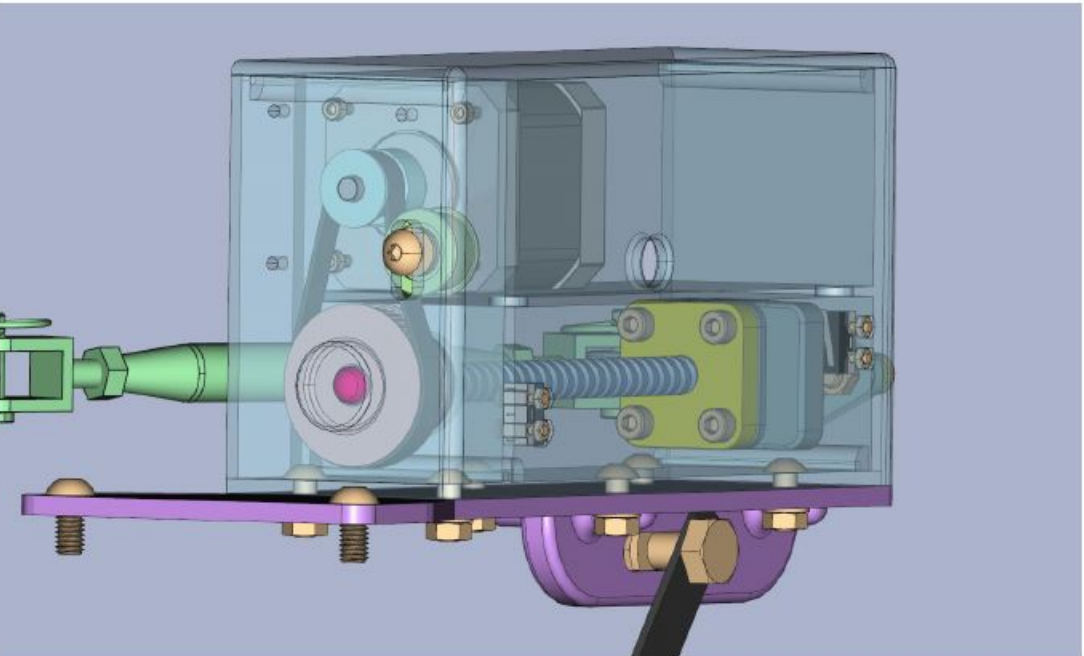
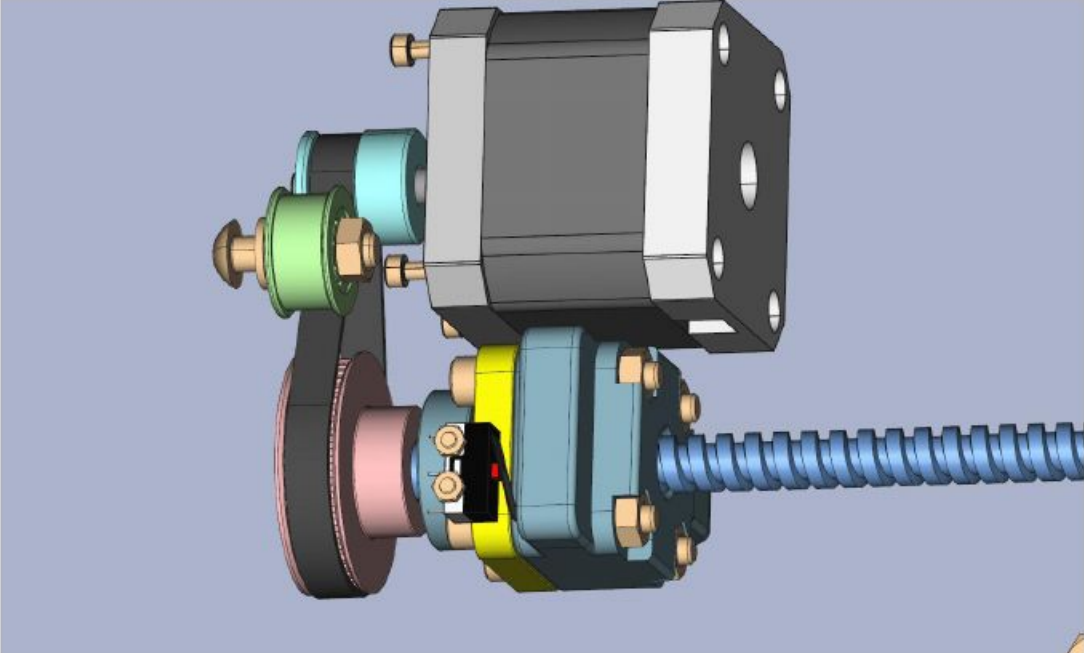
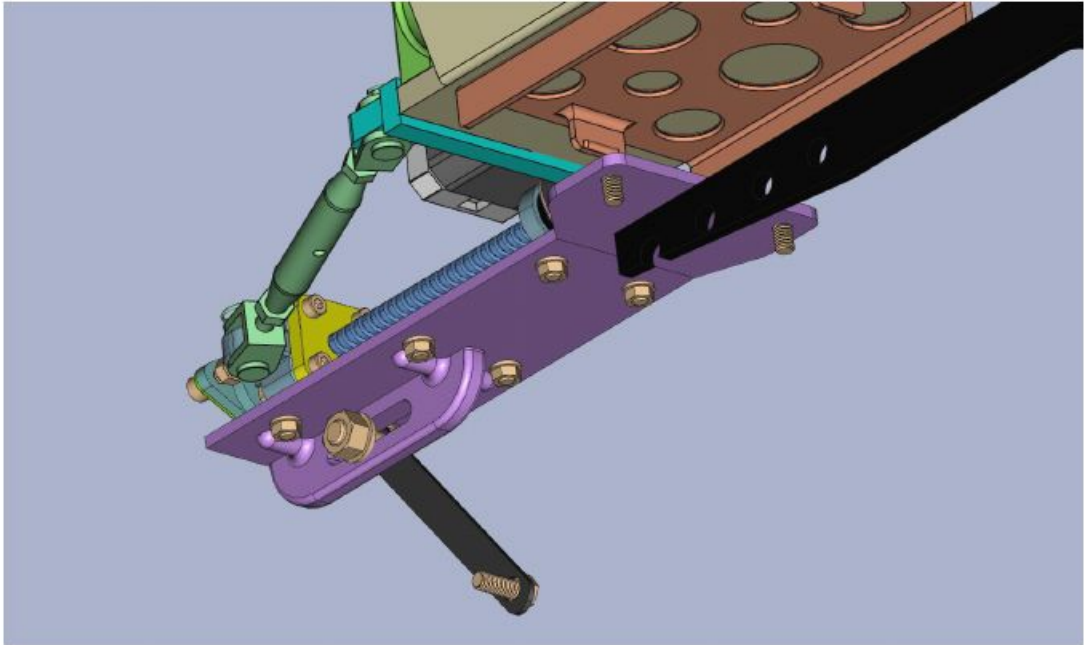
Vue de l'actionneur, capot ouvert :

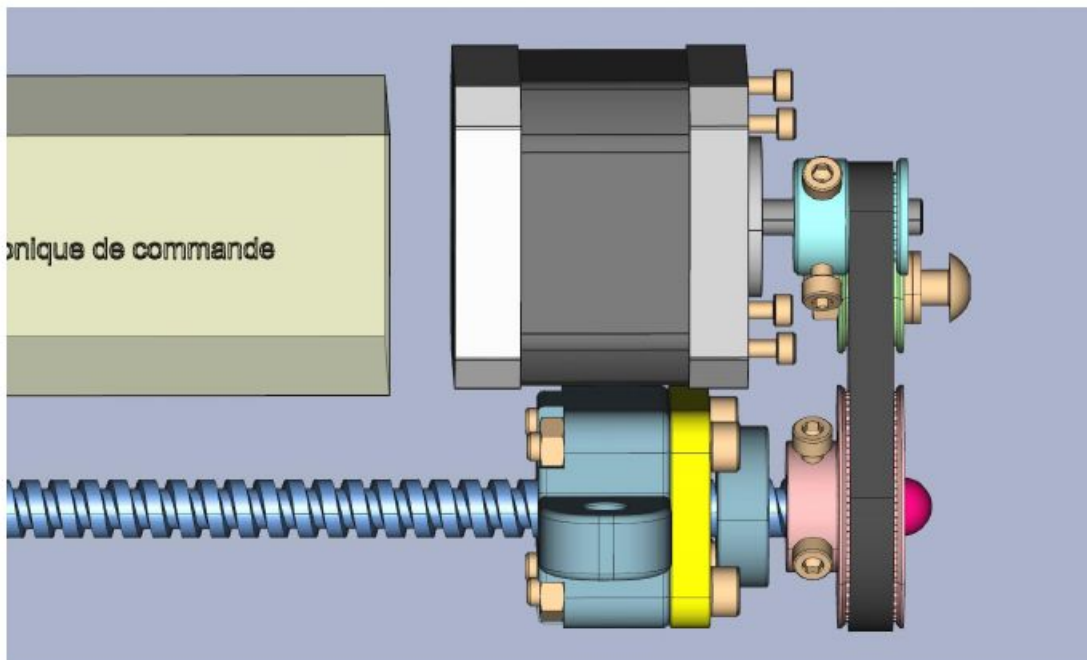
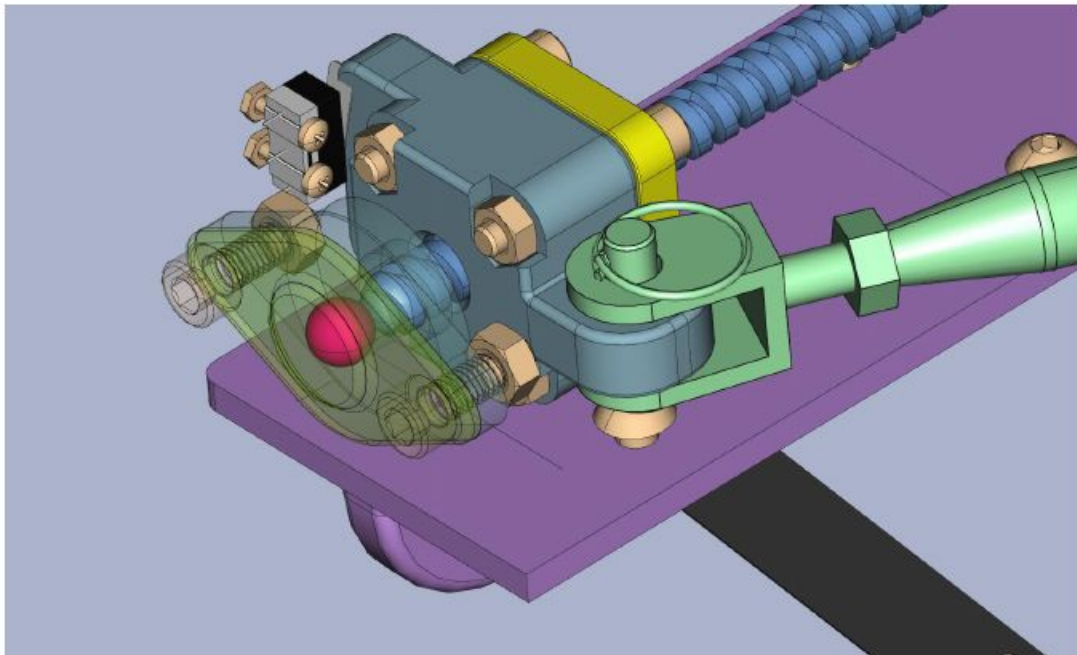


NB : Le volume situé derrière le moteur, comme indiqué sur l'image, représente le volume disponible pour recevoir l'électronique de commande.

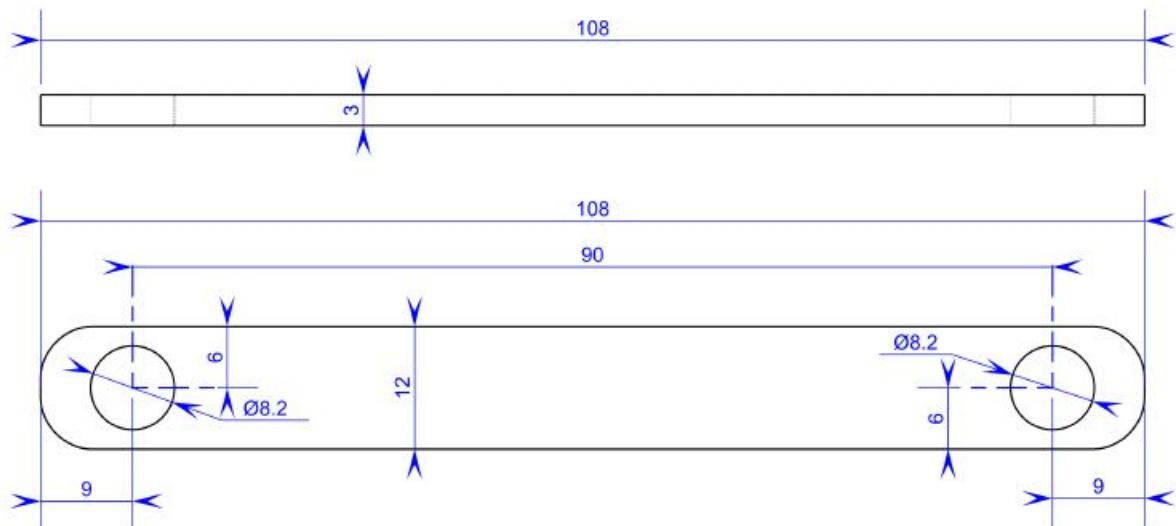
L'actionneur est fixé au fauteuil par l'intermédiaire d'une solide plaque support imprimée en 3D. Cette plaque est maintenue dans l'axe latéral par deux boulons vissés dans le support métallique sous l'accoudoir. Dans le plan vertical elle est maintenue par une patte en acier boulonnée sur le bras métallique supportant l'ensemble de l'accoudoir. En jouant sur la longueur du ridoir et sur le boulonnage de la patte support, on pourra ajuster avec précision la position de l'actionneur.



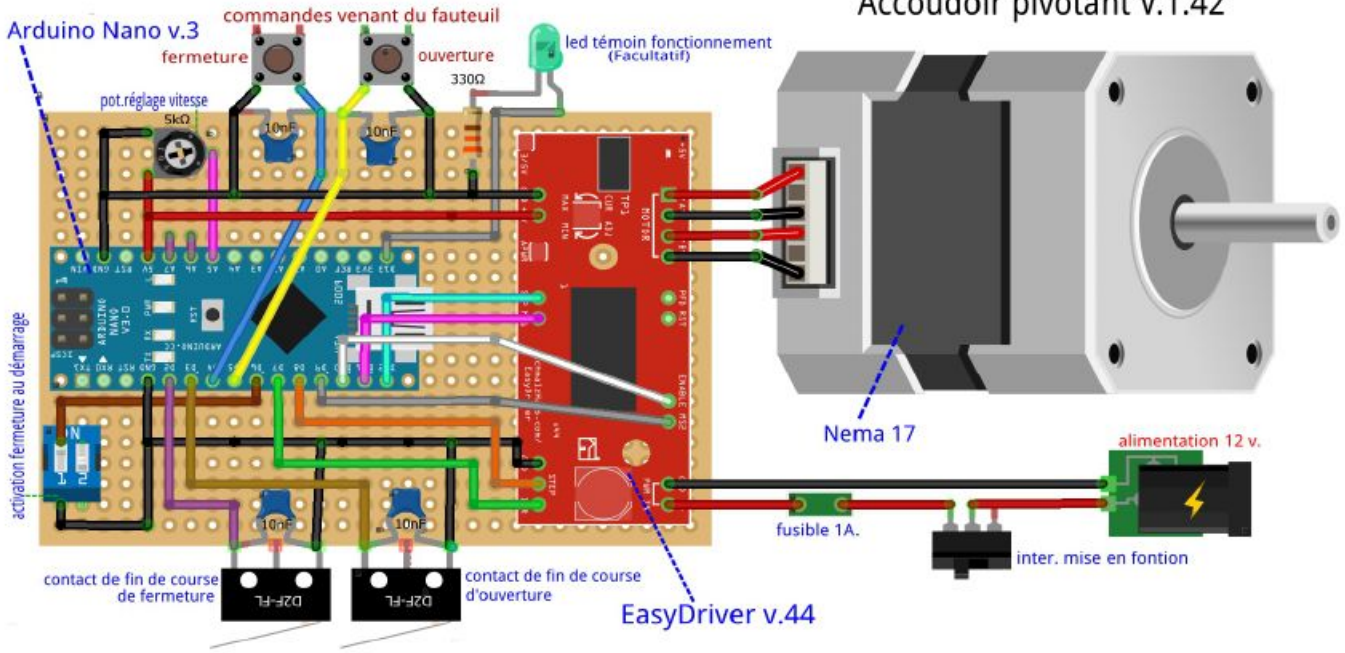




Annexe 2 (Bras métal support vertical du boîtier)



Accoudoir pivotant V.1.42



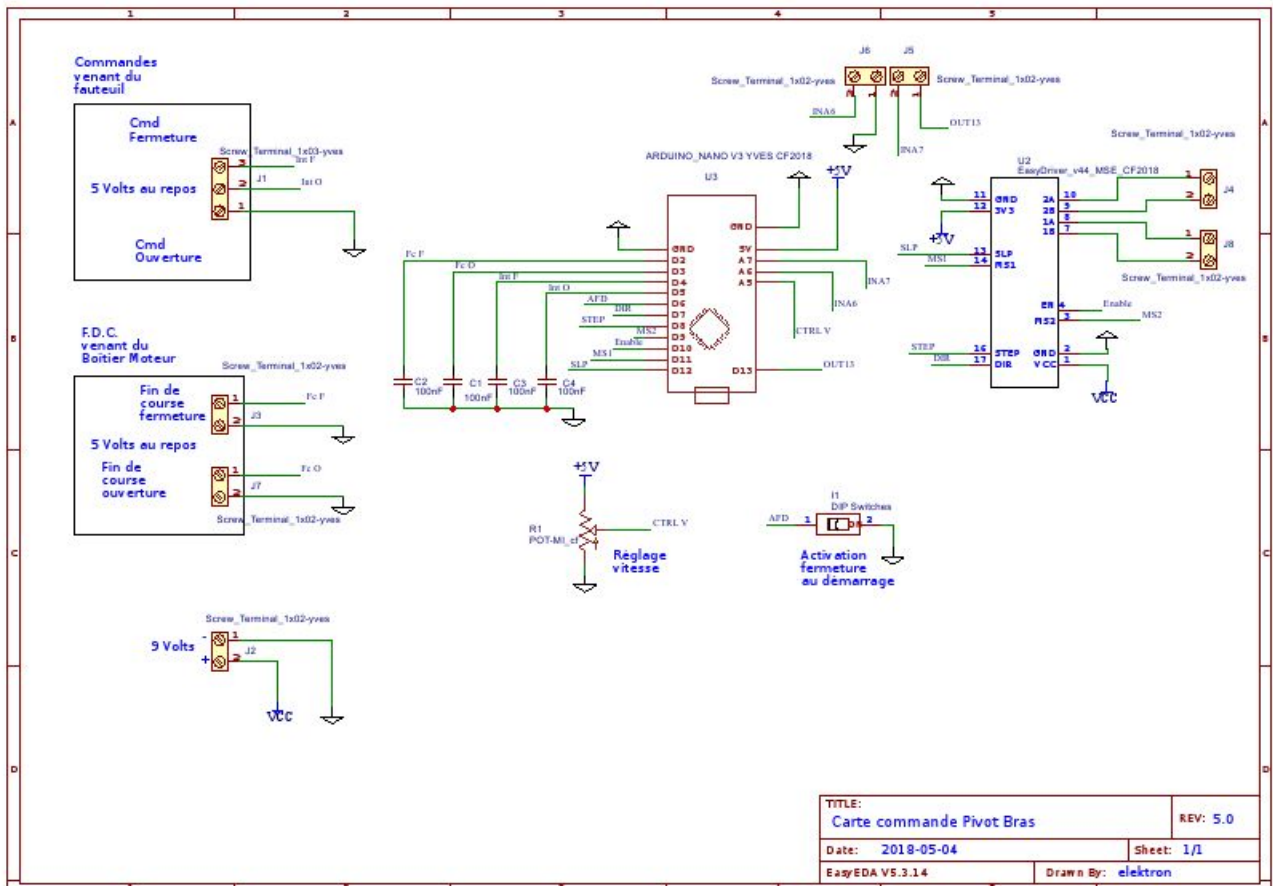
Remarque :

Pour éviter tout incident de fonctionnement en cas de rupture d'un fil, les entrées des commandes sont par défaut à 0 volts au repos et la déconnexion d'un fil ne provoquera pas la mise en route du moteur.

De la même façon, les entrées de détection de fin de course sont à 0 volts au repos (circuit fermé) et la déconnexion d'un fil est alors assimilée à une détection de fin de course (ouverture du circuit) qui empêchera la mise en route du moteur dans cette direction.

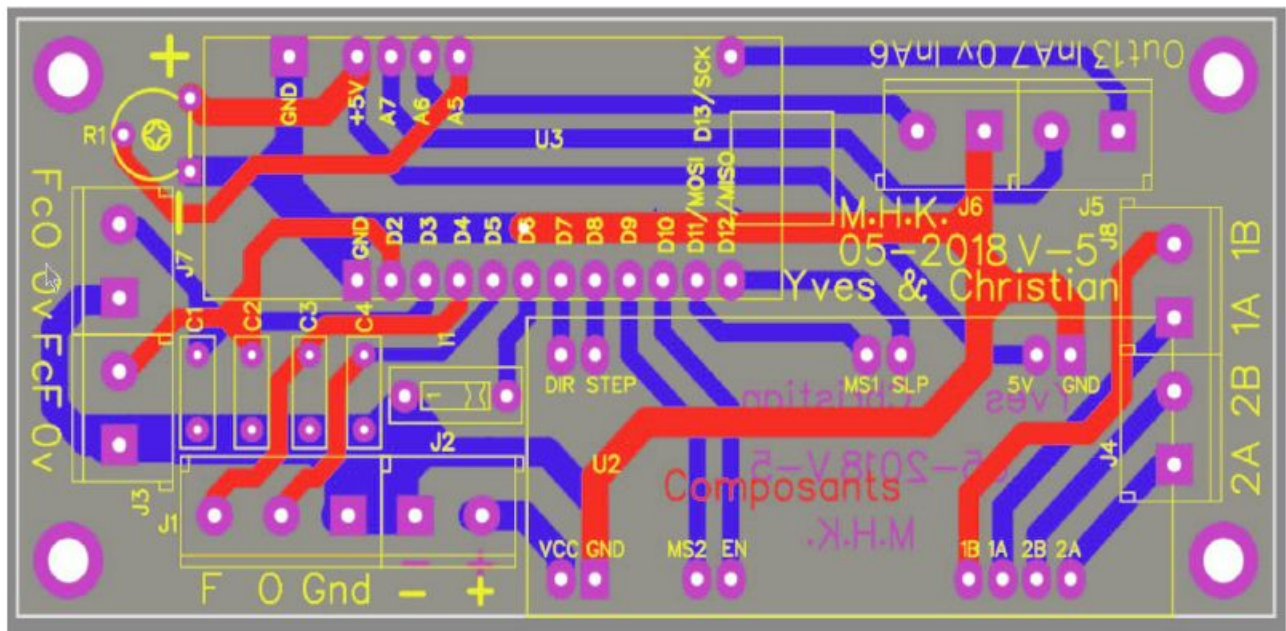
Lorsque l'on appuie sur les deux boutons de commande en même temps, rien ne se passe.

Le switch de fermeture automatique à la mise sous tension du système ainsi que le potentiomètre de réglage de la vitesse de rotation de l'accoudoir sont à régler en fonction des souhaits de l'utilisateur.



TITLE: Carte commande Pivot Bras		REV: 5.0
Date: 2018-05-04	Sheet: 1/1	
EasyEDA V5.3.14	Drawn By: elektron	

PCB



Ce PCB est équipé de bornier à vis pour faciliter les branchements et en assurer la solidité. Il dispose de deux entrées analogiques A6 et A7 non utilisées ici et d'une sortie D13 pour allumer une led facultative de contrôle du fonctionnement moteur.

Le fichier "Gerber_Pivot_v1,42.zip" est prêt a l'emploi pour demander la création de ce PCB chez EasyEDA à l'adresse suivante : <https://easyeda.com/order>

Programme à téléverser sur l'Arduino

```
/* Pilotage du bras rotatif (compilé et testé sous Arduino 1.8,1)
```

```
YLC + CF 08/05/2018 pour Seb L. chez MHK
```

```
Fonctionnalités :
```

```
A la mise sous tension : (en option) remise en position fermée de l'accoudeur s'il est ouvert.
```

```
Le programme reçoit une commande d'ouverture ou de fermeture.
```

```
Celles-ci est activée si la fin de course correspondante n'est pas détectée.
```

```
L'ouverture ou la fermeture se poursuit tant que l'ordre est maintenu ou jusqu'à fin de course.
```

```
La vitesse d'ouverture ou fermeture est ajustable par potentiomètre.
```

```
Les MicroSteps sont mis en position par défaut à HIGH (5V) soit 8 micropas/pas.
```

```
La démultiplication de la transmission permet de ne pas bloquer le moteur dans une position donnée en l'absence d'ordre ;
```

```
donc le moteur est coupé en l'absence de commande de rotation. (enable = HIGH) (=> Plus faible consommation)
```

```
Affectation des broches de l'arduino :
```

```
Dir Moteur (DIR) =>D7 (Commande sens de rotation moteur)
```

```
Step Moteur (STEP) =>D8 (Commande de pas moteur)
```

```
Activation Fermeture automatique =>D6
```

```
Commande Fermeture =>D4
```

```
Commande Ouverture =>D5
```

```
Fin Course Fermeture =>D2
```

```
Fin Course Ouverture =>D3
```

```
MicroStep Mode(MS2) =>D9 full step (0,0), half step (1,0), 1/4 step (0,1),
```

```
MicroStep Mode(MS1) =>D11 and 1/8 step (1,1 : default)
```

```
Enable (EN) =>D10 Permet de couper l'alimentation moteur Low = Out Enable ; HIGH = Out Disable
```

```
Pot. Vitesse =>A5
```

```
Sleep (SLP) =>D12 Permet de couper la carte EasyDriver pour réduire la consommation. Laisser à HIGH ici.
```

```
Entrée Analogique (INA6) =>A6 Pour un usage éventuel ultérieur. Non utilisé ici.
```

```
Entrée Analogique (INA7) =>A7 Pour un usage éventuel ultérieur. Non utilisé ici.
```

```
Led témoin marche moteur =>D13 facultatif (permet de visualiser les commandes moteur)
```

```
*/
```

```
#include <AccelStepper.h>
```

```
AccelStepper nema17(AccelStepper::DRIVER, 3, 2);
```

```
const int vitmax = 8000; // vitesse moteur maxi
```

```
const int vitmin = 200; // vitesse moteur mini
```

```
const int vitauto = 500; // vitesse moteur lors de la fermeture automatique initiale
```

```

// Switch Activation de la fermeture auto
const int p_auto = 6; // pin activation de fermeture automatique

// Commandes de rotation l'accouoir
const int com_fer = 4; // pin de reception commande de fermeture
const int com_ouv = 5; // pin de reception commande d-ouverture

// Fin de course
const int lim_fer = 2; // pin de reception contact limite de fermeture
const int lim_ouv = 3; // pin de reception contact limite d-ouverture

// Commandes microstepping
const int MS2 = 9;
const int MS1 = 11;

// Commandes Gestion Puissance
const int EN = 10;
const int SLP = 12;

// Visualisation Moteur ON
const int LED = 13;

// Potentiomètre contrôle de vitesse
const int p_pot = A5; // pin potentiomètre réglage vitesse moteur

// Variable de temps
unsigned long oldmilli = 0; // stockage temps de la mesure précédente
unsigned long milli = 0; // stockage du temps présent
const unsigned long delai = 1000; // intervalle des lectures du potentiomètre (1 sec)

// Affectation des variables
int pot = 0; // valeur actuelle du potentiomètre
int oldpot = 0; // valeur précédente du potentiomètre
static float vit = 0.0; // vitesse du moteur (pas/sec)
static int dir = 0; // sens de rotation moteur (-1=arrière, 0=arrêt, 1=avant)
static float rota = 0.0; // valeur positive ou négative de rotation moteur

//-----

void setup() {
  pinMode(MS1,OUTPUT);
  pinMode(MS2,OUTPUT);
  pinMode(EN,OUTPUT);
  pinMode(SLP,OUTPUT);
  pinMode(LED,OUTPUT);
  digitalWrite(MS1,HIGH); // MS1 = 1 + MS2 = 1 => Avance par 1/8 de pas
  digitalWrite(MS2,HIGH);
  digitalWrite(EN,HIGH); // Par défaut le moteur n'est pas sous tension
  digitalWrite(SLP,HIGH); // La carte EasyDriver reste activée en permanence
  digitalWrite(LED,LOW); // par défaut led témoin éteinte

  nema17.setMaxSpeed(8000);
  nema17.setAcceleration(500);
  // les entrées des commandes de mouvement sont à 5v au repos (pullup interne).
  // Une action sur un bouton de commande met l'entrée à 0v (GND)
  pinMode(com_ouv,INPUT_PULLUP);
  pinMode(com_fer,INPUT_PULLUP);
  // les entrées des switches de fin de course sont à 0v au repos. Entrée à GND (0v).
  // une détection de fin de course ouvre le circuit et met l'entrée à 1; (pullup interne).
  pinMode(lim_fer,INPUT_PULLUP);
  pinMode(lim_ouv,INPUT_PULLUP);
  // l'entrée d'activation de fermeture auto est à 5v si non activée (pullup interne)
  pinMode(p_auto,INPUT_PULLUP);

  // mise sous tension : si demandé, on met l'accouoir en position fermée si ouvert
  if (digitalRead(p_auto) == false) { // si fermeture auto activée...
    vit = vitauto; // on met la vitesse de fermeture auto
    if (digitalRead(lim_fer) == false) { // si fin de course non détectée (=0v.)
      dir = -1; // sens de rotation moteur
      sens_moteur();
      digitalWrite(EN,LOW); // mise sous tension du moteur
      digitalWrite(LED,HIGH); // allumage led témoin
      do { nema17.runSpeed(); } // on ferme l'accouoir
      while (digitalRead(lim_fer) == false); // tant que pas de contact de fin de course
      digitalWrite(EN,HIGH); // coupure tension moteur
      digitalWrite(LED,LOW); // extinction led témoin
    }
  }
}

```

```

}
}
//-----

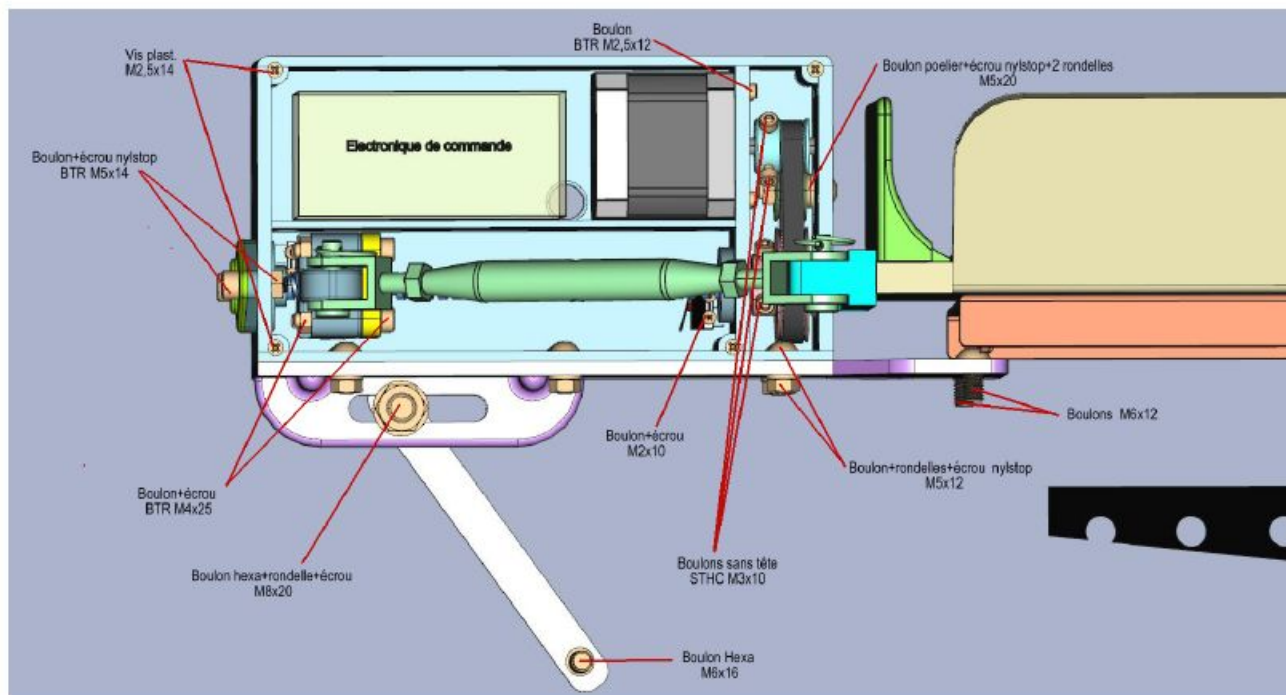
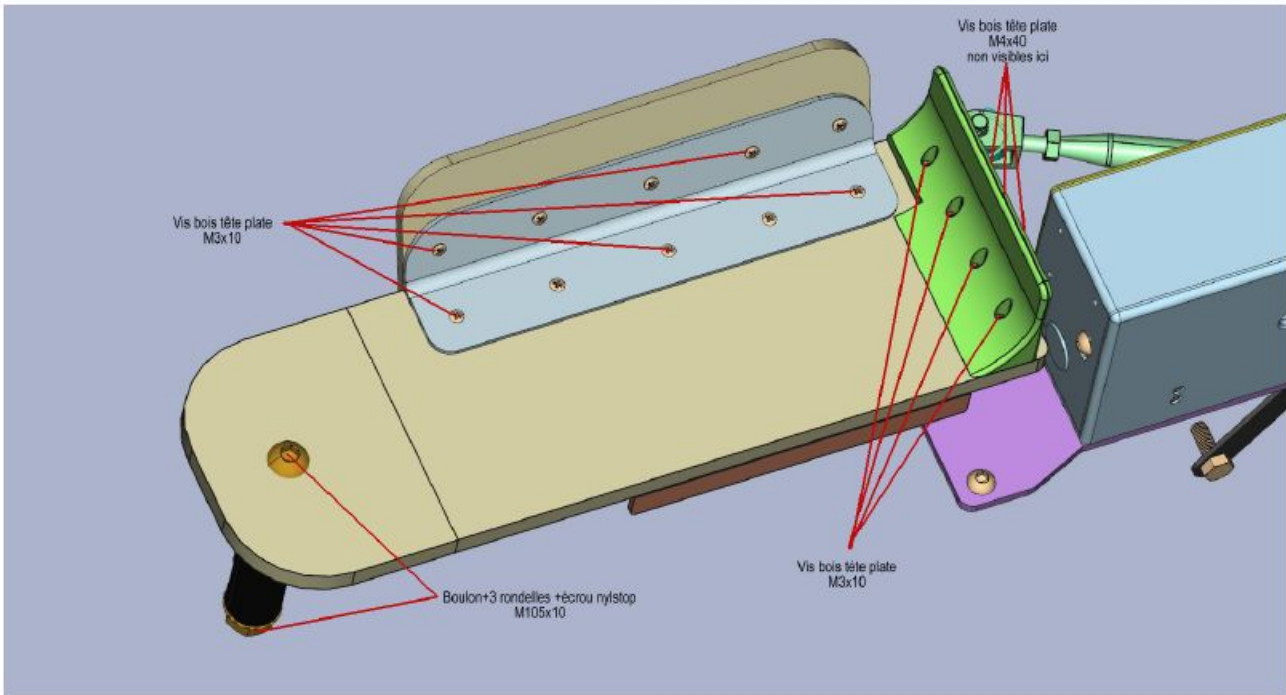
void loop() {
// pour ne pas ralentir la boucle, on lit le potentiomètre une fois par (valeur du délai)
milli = millis(); // récup du temps présent
if (milli - oldmilli > delai) { // si lect.précédente remonte a plus longtemps que délai choisi...
  lect_potar();
}
// test si demande de mouvement
if (digitalRead(com_ouv) == false && digitalRead(com_fer) == false){
  dir = 0; // on ne fait rien car on a appuyé sur les deux boutons en même temps
}
else {
if (digitalRead(com_ouv) == false) {
  dir = 1; // on vient de mettre à la masse l'entrée => commande d'ouverture reçue
  sens_moteur();
  action();
}
else if (digitalRead(com_fer) == false) {
  dir = -1; // on vient de mettre à la masse l'entrée => commande de fermeture reçue
  sens_moteur();
  action();
}
else {
  dir = 0; // pas de commande reçue
  digitalWrite(EN,HIGH); // coupure tension moteur
  digitalWrite(LED,LOW); // extinction led témoin
}
}
}
//-----

void sens_moteur() {
  rota = vit * dir;
  nema17.setSpeed(rota); // initialise le sens de rotation du moteur
}
//-----

void lect_potar() {
  pot = analogRead(p_pot); // ...lecture valeur potentiomètre :
  // pour pallier l'instabilité de la tension en sortie du potentiomètre,
  // on teste s'il y a un écart d'une valeur supérieure à 5 avec la mesure précédente.
  if (pot < oldpot-5 || pot > oldpot+5) { // on ne calcule la vitesse que si changement de valeur
    vit = map(pot, 0, 1023, vitmin, vitmax); // ajuste la vitesse entre valeurs mini et maxi choisies
    oldpot = pot; // stockage valeur potentiomètre
  }
}
//-----

void action() { // rotation moteur
  // // test des switches de fin de course gauche et droite
  // false ; la fin de course n'est pas détectée (circuit fermé) l'entrée est à 0v.
  // true ; la fin de course est détectée (circuit ouvert) l'entrée est a 5V par le pull-up interne.
  if ((dir == 1 && digitalRead(lim_ouv) == false) // selon sens demandé...
    || (dir == -1 && digitalRead(lim_fer) == false)) { // ... action si contact fin de course corresp. non détecté
    digitalWrite(EN,LOW); // mise sous tension du moteur
    digitalWrite(LED,HIGH); // allumage led témoin
    nema17.runSpeed(); // actionne le moteur
  }
}
}
/*****/

```



NB : avant de procéder au montage, visionner la vidéo accompagnant ce dossier. Cette vidéo (AP1,41.mp4) d'une durée de 4 minutes montre l'ordre dans lequel les pièces et composant de ce système doivent être assemblés.

Pièces à acheter (18 références)

Les adresses URL indiquées ci-dessous, sont purement indicatives et permettent d'avoir pour chaque composant une référence commerciale. Les offres variant dans le temps, il est suggéré de rechercher sur internet s'il n'existe pas d'autres alternatives.

1 moteur Nema 17 dim. 42 x 42 x 40 mm long

<https://prototype3d.fr/284-moteur-pas-a-pas-nema-17-40ncm.html> (12,00 €)

1 carte Arduino nano rev.3

https://www.ebay.fr/itm/1276-Board-upgraded-ATmega328P-For-Nano-V3-contrleur-Board-Compatible-arduino/292016452560?ssPageName=STRK%3AMEBIDX%3AIT&var=591012223048&_trksid=p2057872.m2749.l2649 (5,23 €)

1 carte EasyDriver v44 (pilote moteur pas à pas)

https://www.ebay.fr/itm/1265-A3967-EasyDriver-Stepper-Motor-Driver-V44-for-arduino-development-board/292016622777?ssPageName=STRK%3AMEBIDX%3AIT&var=591012267403&_trksid=p2057872.m2749.l2649 (4,19 €)

4 condensateurs 10 nF

<https://www.gotronic.fr/art-condensateur-lcc-10-nf-3264.htm> (0,60 €)

1 dip switch (ou commutateur permanent)

https://fr.aliexpress.com/item/10Pcs-Slide-Type-Switch-Module-1-Bit-2-54mm-1-Position-Way-DIP-Red-Pitch/32398068489.html?spm=a2g0w.search0306.3.19.391ab5c5eDStZL&ws_ab_test=searchweb0_0,searchweb201602_0_10618_10059_10534_10313_10084_100031_10083_10184_10305_10624_10304_10623_441_10307_442_10306_10341_10065_10142_10340_10068_10343_10342_10103_10620_10344_10622_10302_10621,searchweb201603_0,ppcSwitch_0&algo_pvid=37d81e65-5ff0-4d16-89a3-4a6ab0afef99&algo_expid=37d81e65-5ff0-4d16-89a3-4a6ab0afef99-2 (0,61 €)

1 potentiomètre ajustable 4,7 kΩ

<https://www.gotronic.fr/art-ajustable-t7ya-4-7k-8486-3107.htm> (1,20 €)

2 micro-switches inverseurs miniatures à levier

<https://www.lextronic.fr/micro-switches/1721-micro-switch-miniature-a-levier-3a.html> (3,40 €)

1 lot de bornier à vis 4 broches au pas 2.54

<https://www.ebay.fr/itm/lot-de-5-piece-4-broches-Bornier-a-Vis-Connecteur-PCB-Mount/323243796657?hash=item4b42d4e8b1:g:H6UAAOSw2cVax7ID> (8,50 €)

1 Ridoir à 2 chapes M6 inox 316 de chez Mamutec

en vente chez **Leroy Merlin Betton** réf : Réf 612008043656 (8,40 €)

1 écrou trapézoïdal igus TR 8x1.5

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Broche-filetee-trapezoidale-accessoires/Trapezoidale-plomb-vis-ecrou/Trapezoidale-ecrou-de-vis-bride-Iglidur-J/Trapezoid-nuts-Igus-JFRM-M-01-TR8x15::999994769.html> (5,38 €)

1 tige filetée trapézoïdale TR 8x1,5

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Broche-filetee-trapezoidale-accessoires/Broche-filetee-trapezoidale/Acier-Trapezoidale-tige-filetee/Trapezoidale-tige-filetee-RPTS-droite-TR-8x15-L%3D500mm::999993952.html> (3,01 €)

1 roulement a bille 8x22x7

<http://www.motedis.fr/shop/Dynamic/palier/Roulements-a-billes-a-gorge-profonde/Roulements-a-billes-a-a-gorge-profonde-608-2RS-8x22x7mm::999991457.html> (0,48 €)

1 palier a bride 8 mm

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Flanschlager/Paliers-a-bride-8mm-moule-sous-pression-KFL08::999995213.html> (2,82 €)

1 poulie crantée GT2 20 dents axe 5mm

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Courroies-dentees-et-roues-a-dents/Type-GT2-6mm/GT2-pulley-20-teeth-5mm-bore-for-6mm-belt::999994248.html> (1,26 €)

1 poulie crantée GT2 50 dents axe 8mm

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Courroies-dentees-et-roues-a-dents/Type-GT2-6mm/GT2-pulley-50-teeth-8mm-bore-for-6mm-belt::999994522.html> (4,80 €)

1 galet presseur axe 5mm

<http://www.motedis.fr/shop/Dynamic-unites-lineaire/Courroies-dentees-et-roues-a-dents/Type-GT2-6mm/Roue-intermediaire-pour-courroie-6mm-large::999994523.html> (2,88 €)

1 courroie crantée GT2 fermée 6mm larg x 150 mm long

<http://www.eu.diigiit.com/fr/timing-belt-gt2-pitch-2mm-width-6mm?search=courroie%20gt2> (4,12 €)

2 billes acier 8mm

<http://www.1001roulements.com/lot-de-10-billes-de-diam-8mm.html> (3,60 €)

Visserie :

- 1 boulon poelier M5x20 (galet presseur)
- 1 rondelle M5 (galet presseur)
- 1 écrou nylstop M5 (galet presseur)
- 4 boulons BTR M4x25 (chariot)
- 4 écrous M4 (chariot)
- 2 boulons BTR M5x14 (palier butée)
- 2 boulons M6x12 (base)
- 6 boulons M5x12 (base)
- 6 rondelles M5 (base)
- 8 écrous nylstop M5 (base + palier butée))
- 1 boulon M6x16 (patte support)
- 1 écrou M8 (patte support)
- 1 rondelle M8 (patte support)
- 1 boulon M8x20 (patte support)
- 1 boulon M10 de 105 mm (axe de pivot)
- 1 écrou nylstop M10 (axe de pivot)
- 3 rondelles M10 (axe de pivot)
- 4 vis plast. tête fraisée 2,5 x 14 mm (fixation capot)
- 4 boulons M2 x 10 mm (fixation capteurs de fin de course)
- 4 écrous M2 (fixation capteurs de fin de course)
- 4 Btr M2,5 x 12 (fixation moteur)
- 4 vis sans tête STHC M3 x 10 mm (blocage poulies)
- 3 vis bois tête plate 4mm x 40 mm (fixation palonnier)
- 14 vis bois tête plate 3mm x 10 mm (fixation butée de coude + butée latérale)

Total pièces à acheter

pièces = 72,38 €
+ port ≈ 12 €
+ visserie ≈ 9 €
+ façonnage PCB ≈ 2 €

Total ≈ 95 €

Pièces à imprimer en 3D (CF : 10 fichiers STL)

- Réducteur axe de pivot (le même que version manuelle)
- Entretoise du pivot (la même que version manuelle)
- Pont de glissement (de préférence en ABS)
- Butée de coude
- Palonnier
- Base support (de préférence en ABS)
- corps de chariot
- Boitier
- Capot de boitier

Pièces à découper au laser *(CF : 2 fichiers DXF)*

Planche d'appui de 8 ou 9 mm (3mm + 5mm contrecollés) ou (3 x 3 mm contecollés)
Butée latérale en planche de 5 mm (ou mieux : 2x 3mm contrecollés)

Autres pièces

Bras support vertical du boîtier de l'actionneur :

Plat acier (ou dural) 3 mm d'épaisseur coupé à 108 mm de long sur 12 mm de large avec perforation de 8 mm de diamètre à 9 mm de chacune des extrémités (voir plan).

Liaison accoudoir - appui latéral :

Cornière inox (ou dural) 2mm d'épaisseur x 180 mm de long pliée à 110°.
(non trouvé dans le commerce, à faire à la plieuse).

PCB : à commander chez un façonnier (ex: EasyEDA) ou a faire soi-même.

0-0-0-0-0-0-0-0-0-0-0-0-0-0-0