

# TELESCOPE Meade ETX90 : ASSERVISSEMENT.



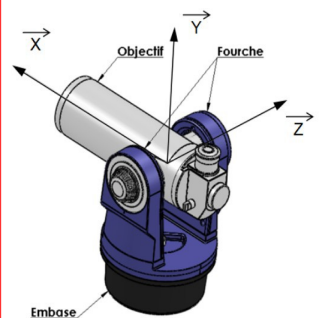
## A. Objectif de l'activité.

Il s'agit d'étudier l'**asservissement en vitesse du mouvement azimuthal du télescope** et de préparer une **présentation orale** qui reprend les différents paragraphes de ce texte.

Vous disposez du fichier de présentation '**trame presentation - tp asservissement - astrolab.ppt**' que vous allez compléter.

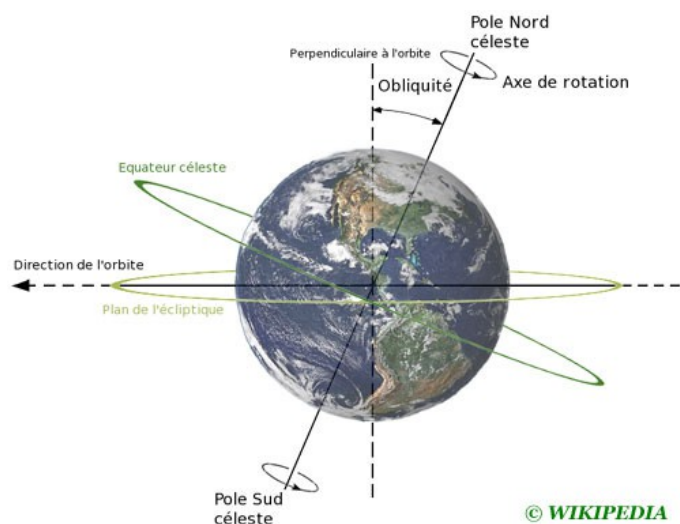


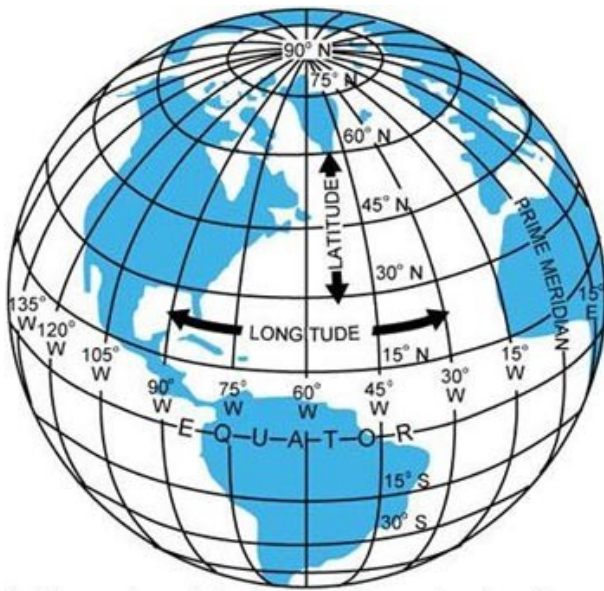
**ATTENTION.** La base et la fourche ont une rotation limitée. Dans le plan horizontal des butées limitent la rotation de la fourche à  $360^\circ$  pour éviter d'endommager les liaisons électriques. Dans le plan vertical, une butée empêche l'objectif de venir frapper la fourche vers le haut, au delà de  $90^\circ$  et de venir frapper l'embase vers le bas à plus de  $30^\circ$ . **Ne forcez pas car des dégâts pourraient sinon en résulter.**



## B. Hypothèse de l'étude, le montage équatorial.

Le montage équatorial aligne l'axe polaire du télescope avec le pôle céleste Nord (ou Sud si vous êtes dans l'hémisphère Sud). Avec un télescope monté en mode équatorial, les objets astronomiques peuvent être suivis en déplaçant simplement le télescope sur son axe polaire (la rotation azimuthale de sa fourche). Il suffit d'allumer le moteur de votre ETX et il compensera la rotation de la Terre autour de son axe et maintiendra les objets dans le champ de vision de l'oculaire. Si le télescope est bien aligné sur le pôle, les réglages de la déclinaison seront à peine nécessaires.



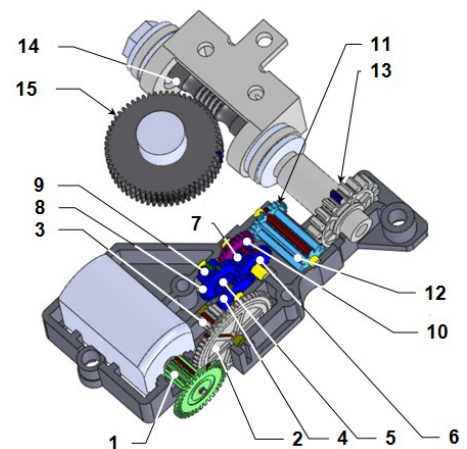


**Travail B.1 :** Expliquer le montage équatorial. En particulier vous donnerez la valeur de l'angle d'inclinaison du plateau sur lequel repose l'embase par rapport à l'horizontale ; et vous donnerez aussi la valeur, ainsi que le sens, de la vitesse de rotation azimuthale qui permet le suivi d'un astre.

### C. Performances attendues.

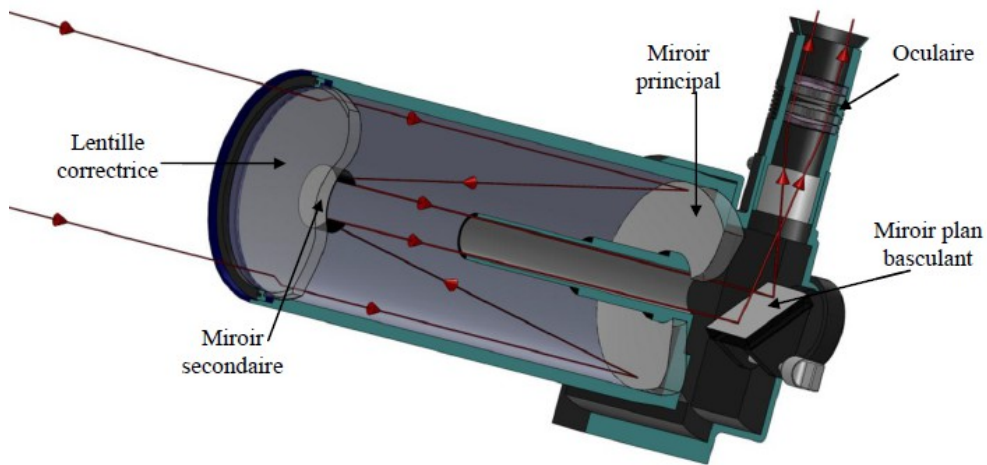
On veut conserver un objet céleste dans le champ du télescope pendant **au moins deux heures**. Pour simplifier l'étude on se place dans le cas d'une monture équatoriale.

**Travail C.1:** Sachant que le rapport de réduction entre l'axe de sortie (15) de l'embase et l'arbre du moteur (1) d'azimut vaut **1/12320**, à quelle vitesse doit tourner ce moteur pour assurer le suivi d'un objet céleste ?



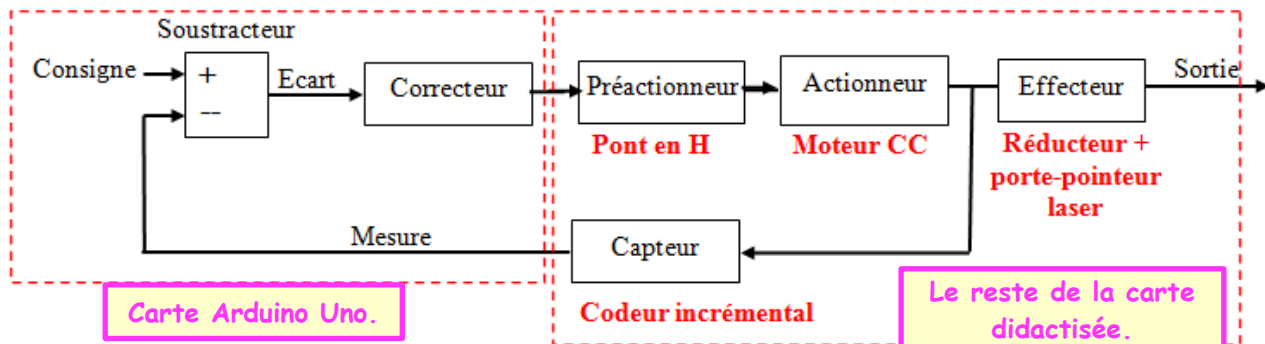
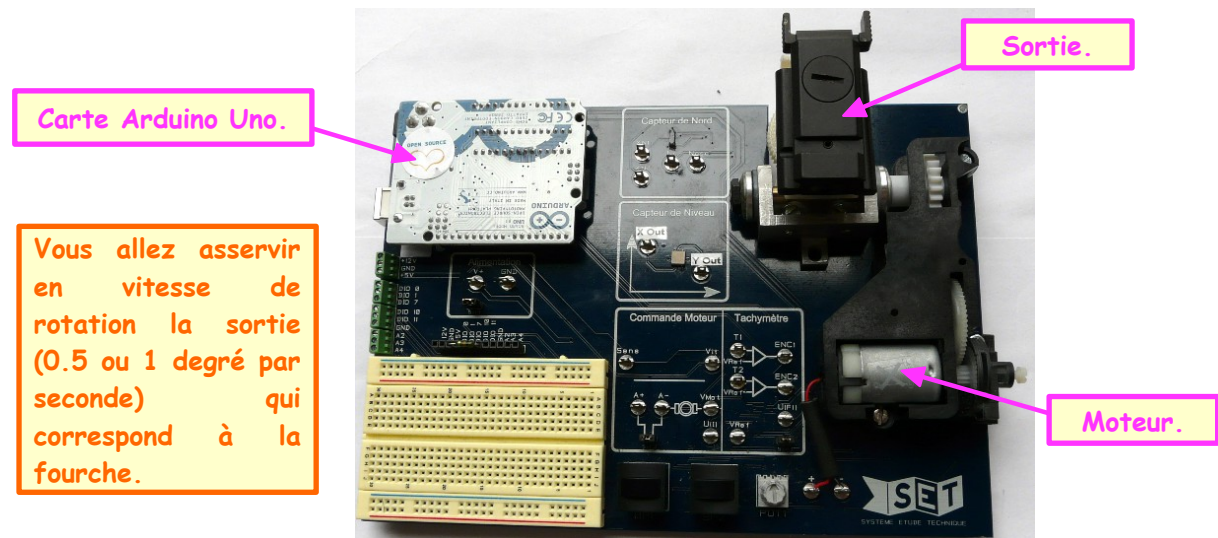
**Travail C.2:** Sachant qu'un système de commande de moteur à courant continu en boucle ouverte permet de stabiliser la vitesse au mieux à  $\pm 5\%$  de la valeur de consigne, est-ce que la commande en boucle fermée est justifiée. On se placera dans le cas extrême où la vitesse à toujours une erreur de  $+5\%$  par exemple.

On se place dans le cas de l'oculaire de distance focale 26mm fourni en standard, qui donne un champ de vision apparent de  $52^\circ$ . Comme le champ réel d'observation est égal au champ apparent de l'oculaire divisé par le grossissement qui est ici de 48; le champ réel d'observation est donc réduit à  $52/48 = 1.08^\circ$ .



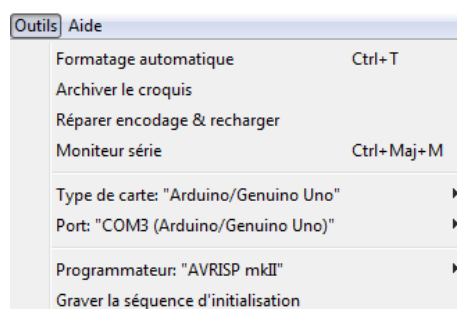
**N'oubliez pas votre présentation, utilisez des photos!**

## D. Etude expérimentale de l'asservissement en vitesse d'azimut.



### Travail D.1: Lancer le logiciel arduino

Vérifier si la carte arduino et le port com sont bien sélectionnés.



Cartes Arduino AVR

- Arduino Yún
- Arduino/Genuino Uno
- Arduino Duemilanove or Diecimila
- Arduino Nano
- Arduino/Genuino Mega or Mega 2560

Serial ports

- ☒ COM3 (Arduino/Genuino Uno)



**Travail D.2:** Ouvrir le fichier 'asservissement\_azimut.ino' du dossier '3b - astrolab - tp asservissement\asservissement\_azimut'.

```

float nc = 1; // consigne de vitesse de la fourche en deg/s
float kp = ; // Coefficient proportionnel
float ki = 0; // Coefficient intégrateur
float kd = 0; // Coefficient dérivateur
float vref = 11.42; // tension de ref
const int pot = A5; // potentiometre connecte a A5
const int mot = 9; // moteur controle vitesse avec pwm
const int sens = 8; // moteur controle sens
const int nmu = A1; // tension de vitesse moteur
    
```

N'oubliez pas votre présentation, utilisez des photos!

**Travail D.3:** Etude de l'influence de la correction proportionnelle  $K_p$  seule, avec une consigne de vitesse  $N_c = 1 \text{ deg/s}$  de rotation de la fourche.

Vous allez faire des essais avec

- $K_p = 50$
- $K_p = 100$
- $K_p = 200$
- $K_p = 300$
- $K_p = 500$

```

float nc = 1; // consigne de vitesse de la fourche en deg/s
float kp = ; // Coefficient proportionnel
float ki = 0; // Coefficient intégrateur
float kd = 0; // Coefficient dérivateur
float vref = 11.42; // tension de ref
const int pot = A5; // potentiometre connecte a A5
const int mot = 9; // moteur controle vitesse avec pwm
const int sens = 8; // moteur controle sens
const int nmu = A1; // tension de vitesse moteur
    
```

Après chaque modification.

1) Enregistrer CTRL + S.

2) Téléverser le prog. dans l'Arduino.

3) Visualiser l'enregistrement en vitesse (deg/s).

```

float kp = ; // Coefficient proportionnel
float ki = 0; // Coefficient intégrateur
float kd = 0; // Coefficient dérivateur
float vref = 11.42; // tension de ref
const int pot = A5; // potentiometre connecte a A5
const int mot = 9; // moteur controle vitesse avec pwm
//const int sens = 8; // moteur controle sens
const int nmu = A1; // tension de vitesse moteur
    
```

Temps (s)      N (deg/s)

0.010	0.071	11.420	255	278.577
0.020	0.077	9.270	207	207.882
0.030	0.885	1.523	34	34.357
0.040	1.128	0.000	0	-38.481
0.050	0.900	1.344	30	30.072
0.060	0.686	4.210	94	94.341
0.070	0.686	4.210	94	94.341
0.080	0.800	2.687	60	60.064
0.090	0.828	2.284	51	51.495
0.100	0.778	2.956	66	66.491
0.110	0.728	3.628	81	81.487
0.120	0.757	3.224	72	72.918
0.130	0.793	2.777	62	62.206

Cette valeur doit être 9600 baud (bits/s)

Arrêter le défilement pour copier des données.

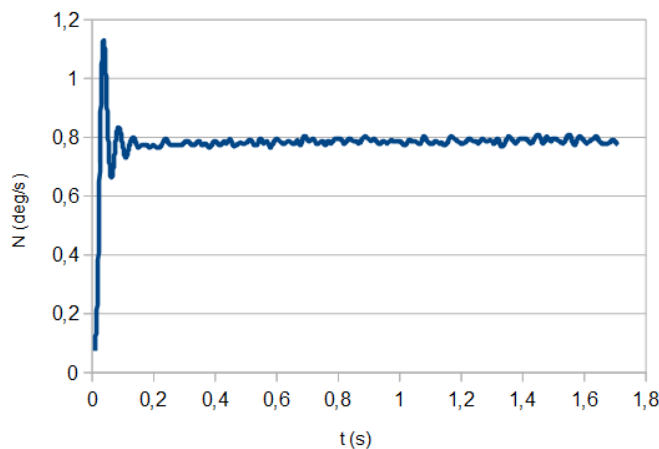
☐ Défilement automatique      Pas de fin de ligne      9600 baud

**Travail D.4:** Tracé du graphe avec un tableur, par exemple ici avec OpenOffice (j'ai un petit faible pour cette suite bureautique).

**Copier Coller**

**Sélectionner la zone puis  
CTRL + F  
Pour changer les points en virgules.**

	A	B	C	D	E
1	0.010	0.071	11.420	255	278.577
2	0.020	0.307	9.270	207	207.882
3	0.030	0.885	1.523	34	34.357
4	0.040	1.128	0.000	0	-38.481
5	0.050	0.900	1.344	30	30.072
6	0.060	0.686	4.210	94	94.341
7	0.070	0.686	4.210	94	94.341
8	0.080	0.800	2.687	60	60.064
9	0.090	0.828	2.284	51	51.495
10	0.100	0.778	2.956	66	66.491
11	0.110	0.728	3.628	81	81.487
12	0.120	0.757	3.224	72	72.918
13	0.130	0.793	2.777	62	62.206



**Faire un graphe avec  
l'outil diagramme,**

Choisissez un type de diagramme

- Colonnes
- Barres
- Secteur
- Zone
- Ligne
- XY (dispersion)**
- Bulle
- Toile
- Cours
- Colonnes et lignes

Lignes seules

☒ Lignes lisses

☐ Trier par valeurs X

Propriétés...

**Travail D.5:** Sachant qu'on souhaite une stabilité optimale, choisir la valeur la plus adaptée du coefficient de correction proportionnelle  $K_p$ . Vous apporterez des justifications.

**N'oubliez pas votre présentation, utilisez des photos!**

**Travail D.6:** Etude de l'influence de la correction intégrale  $K_i$  seule, avec une consigne de vitesse  $N_c = 1 \text{ deg/s}$  de rotation de la fourche.

Pour la valeur choisie du coefficient de correction proportionnelle  $K_p$  ; vous ferez des expériences pour des valeurs de  $K_i = 50 ; 100 ; 500 ; 1000 ; 1500 ; 2000$

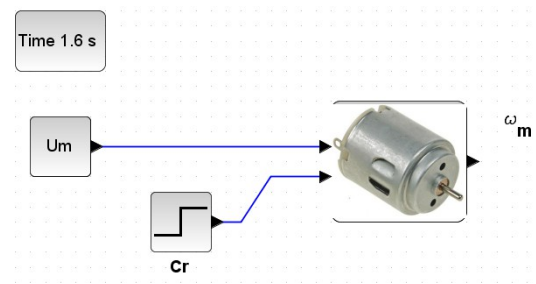
**Travail D.7:** Sachant qu'on souhaite une stabilité et une précision optimale, choisir les valeurs la plus adaptées des coefficients de correction intégrale  $K_i$  et proportionnelle  $K_p$  . Vous apporterez des justifications.

**N'oubliez pas votre présentation, utilisez des photos!**

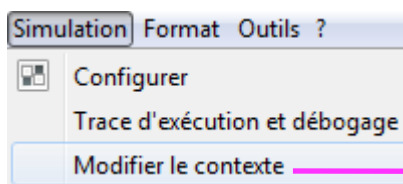
**Travail D. 8:** Si vous avez le temps vous pouvez étudier l'influence de la correction dérivée avec  $K_d$  , et imposer une autre consigne de vitesse de rotation de la fourche, par exemple  $N_c = 0.5 \text{ deg/s}$  .

## E. Simulation de l'asservissement en vitesse d'azimut.

Ouvrir le fichier '**moteur.zcos**' du dossier '**3b - astrolab - tp asservissement**'.



**Travail E.1 :** Compléter les valeurs des paramètres (les données du moteur et du réducteur sont fournies ci-dessous).



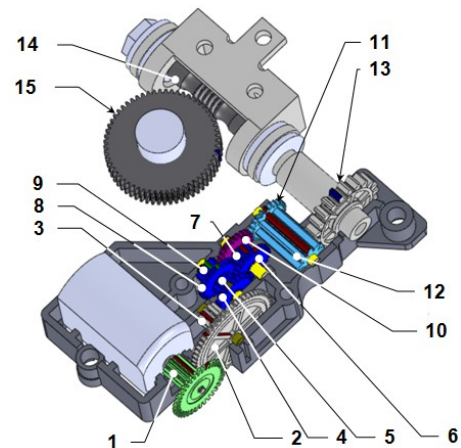
```
Um=5
Cr=
f=0
J=
Kc=
Ke=
R=
L=
C0=0.0008
Rred=
```

Vous définirez toutes les variables utiles avec leurs valeurs données au fur et à mesure de la construction de votre schéma.

### SPECIFICATIONS TECHNIQUES

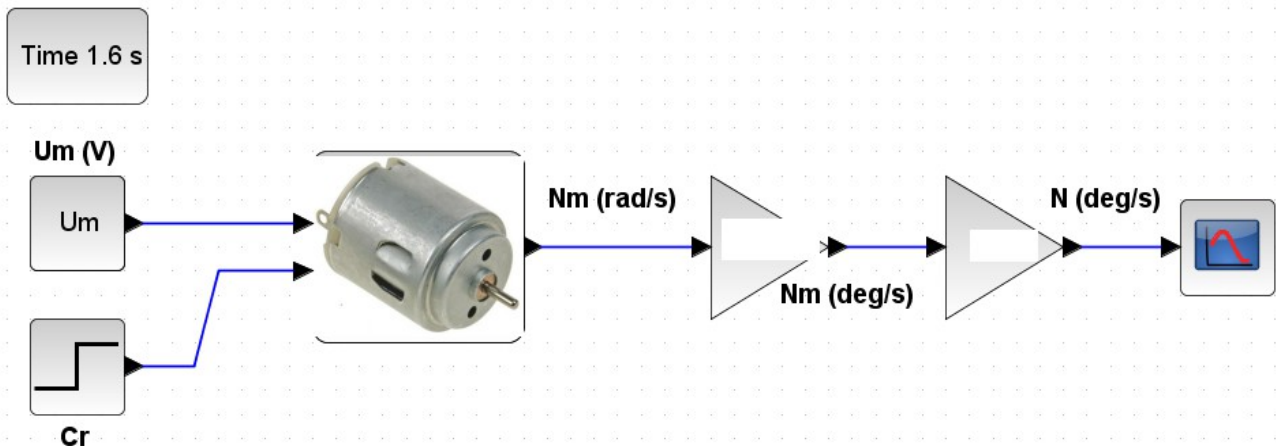
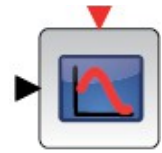
		12V
Tension d'alimentation (Ua)	V	12
Vitesse au courant In	tr/mn	8572
Couple au courant In	mNm	4
Courant max permanent (In)	mA	620
Vitesse à vide à Ua à +/- 10%	tr/mn	12232
Courant à vide à +/- 50%	mA	92
Couple de démarrage à Ua	mNm	13
Courant de démarrage à Ua	mA	1691
Constante de couple	mNm/A	8.6
Constante de vitesse	tr/mn/V	1108
Pente vitesse/couple	tr/mn/mNm	915
Vitesse limite	tr/mn	15000
Puissance utile max. à Ua	W	4.3
Rendement maximum	%	53
Constante de temps électromécanique	ms	24
Inertie	gcm²	3.5
Résistance aux bornes	Ohm	7.1
Inductivité	mH	5.3

Le rapport de réduction entre l'entrée (1) et la sortie (15),  
 $R = 1/12320$ .





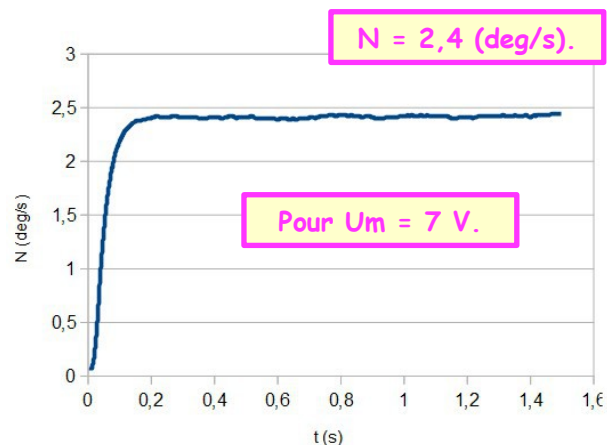
**Travail E.2 :** Compléter ce schéma bloc sous **Xcos**, il correspond à une alimentation du moteur sous une tension constante  $U_m = 7$  (V). Vous prendrez un **couple résistant**  $C_r = 0.003$  Nm constant et une inertie totale des masses en mouvement ramenée sur l'axe moteur  $J = 0.00000035$  (Kg m<sup>2</sup>). Le **temps de calcul** total sera de **1,6 s**.



Vous comparerez le résultat de votre simulation à la courbe expérimentale ci-contre.

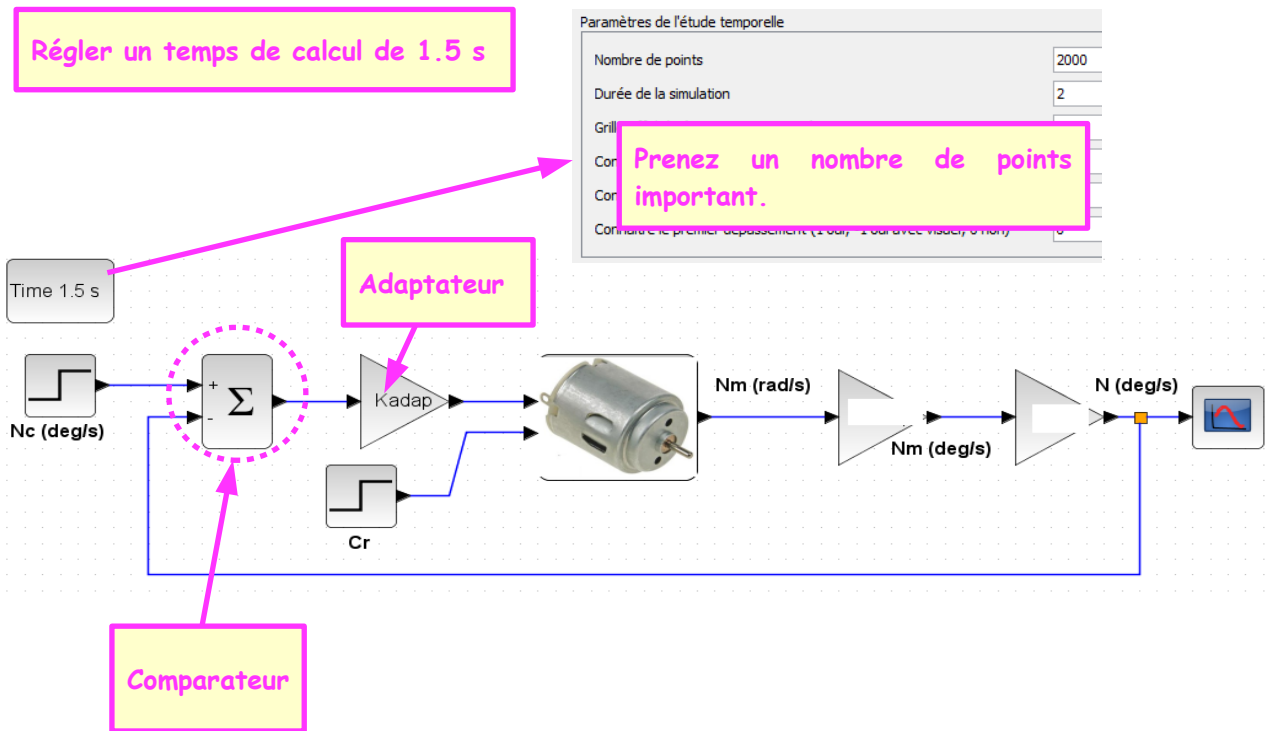
Ajuster la valeur du couple résistant dans vos simulation pour obtenir la valeur de la vitesse de rotation aux temps long.

Ajuster ensuite si nécessaire la valeur de l'inertie  $J$  pour obtenir la bonne 'rapidité' (demander conseil à votre enseignant).



**Travail E.3 :** Création de la chaine de retour 'brute' et observer le signal de sortie ; la vitesse de rotation de l'axe de la fourche  $N(t)$  (**exprimés en deg/s**) pour une consigne représentée par un échelon  $N_c(t) = 1 \times u(t)$  (**exprimés en deg/s**).

Il faut intégrer un comparateur et un adaptateur qui transforme l'écart entre la vitesse de rotation de consigne et la vitesse de rotation de l'axe de la fourche en une tension de commande du moteur. Cet adaptateur est modélisé par un bloc de gain de valeur  $K_{adap} = \frac{11.42}{255}$  (11.42 représente la tension d'alimentation maximale du moteur et 255 la valeur maximale du signal numérique de la sortie de la carte Arduino qui commande le moteur).



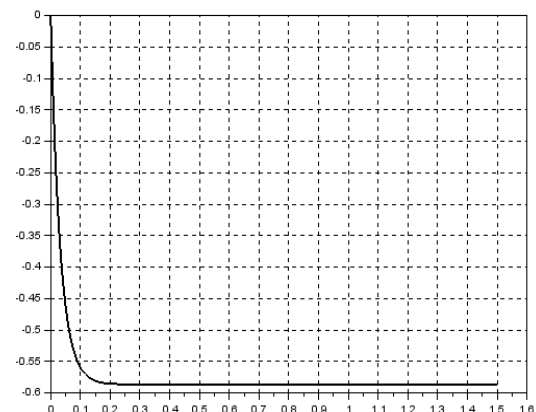
**Travail E.4 :** Vous ferez une simulation avec les valeurs suivantes et pour des conditions initiales  $N(t=0) = 0$ .

La consigne de vitesse de rotation est un échelon un échelon  $N_c(t) = 1 \times u(t)$  (**exprimés en deg/s**).

Le couple résistant est  $C_r = ?$  Nm déterminé au travail (E.2).

L'inertie totale  $J = 3.5 \cdot 10^{-7}$  des masses en mouvement ramenée sur l'axe moteur en **Kg m²**.

Expliquer ce que vous observez.



## Utilisation d'un P.I.D. (un correcteur Proportionnel Intégral Dérivé).

**Travail E.5 :** Vous allez insérer un PID après le comparateur. Cet élément module le signal de commande du moteur en fonction de l'écart  $\varepsilon = N_c - N$ .

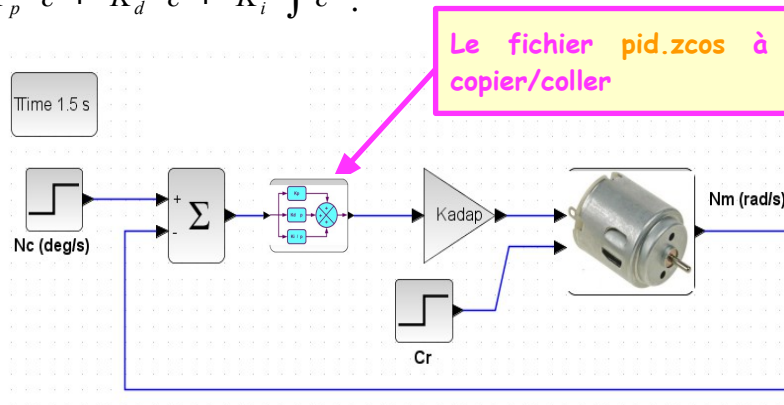
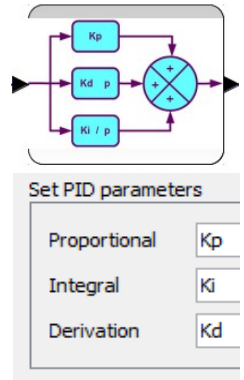
Le coefficient  $K_p$  multiplie l'écart.

Le coefficient  $K_d$  multiplie la dérivée de l'écart.

Le coefficient  $K_i$  multiplie l'intégration de l'écart.

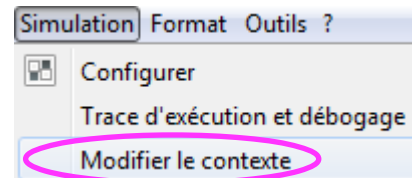
Si on note  $s(t)$  la sortie du PID, elle est reliée à l'écart par :  

$$s(t) = K_p \varepsilon + K_d \dot{\varepsilon} + K_i \int \varepsilon$$



Le fichier pid.zcos à ouvrir puis copier/coller

**Travail D.5 :** Vous allez faire des simulations pour une valeur de  $K_p = 50 ; 100 ; 200 ; 300 ; 500$  (les deux autres coefficients du PID seront nuls  $K_d = K_i = 0$ ); et pour des conditions initiales  $N(t=0) = 0$



Modifier les variables du correcteur PID.

La consigne de vitesse de rotation est un échelon  $N_c(t) = 1 \times u(t)$  (exprimés en deg/s).

Le couple résistant est  $C_r = ? \text{ Nm}$  déterminé précédemment.

Simulation sur 1.5 s, penser à gérer le nombre de points de calcul.

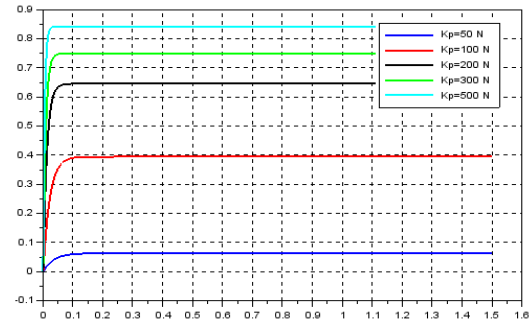
Time 1.5 s

Param. variation

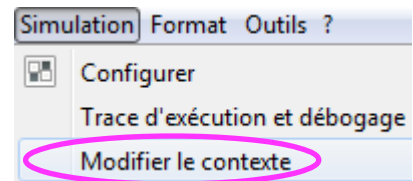
Permet de simuler plusieurs courbes pour différentes valeurs de  $K_p$ .

Vous comparerez vos résultats de simulation avec les courbes expérimentales.

Vous parlerez de la **stabilité** des réponses, déterminerez les **erreur**, la **rapidité à 5%**,  $t_{5\%}$ , et les **dépassements** à l'aide du graphe.



**Travail E.6 :** Vous allez maintenant faire des simulations pour la valeur de  $K_p$  qui correspond aux expériences et  $K_i \in [50 ; 2000]$  (les coefficient dérivé du PID sera nul  $K_d = 0$ ); et pour des conditions initiales  $N(t=0) = 0$ .



La consigne d'angle est un échelon un échelon  $N_c(t) = 1 \times u(t)$  (**exprimés en deg/s**).

Le couple résistant est  $C_r = ? \text{ Nm}$  déterminé précédemment.

Simulation sur 1.5 s, penser à gérer le nombre de points de calcul.

Time 1.5 s

Param. variation

Permet de simuler plusieurs courbes pour différentes valeurs de  $K_i$ .

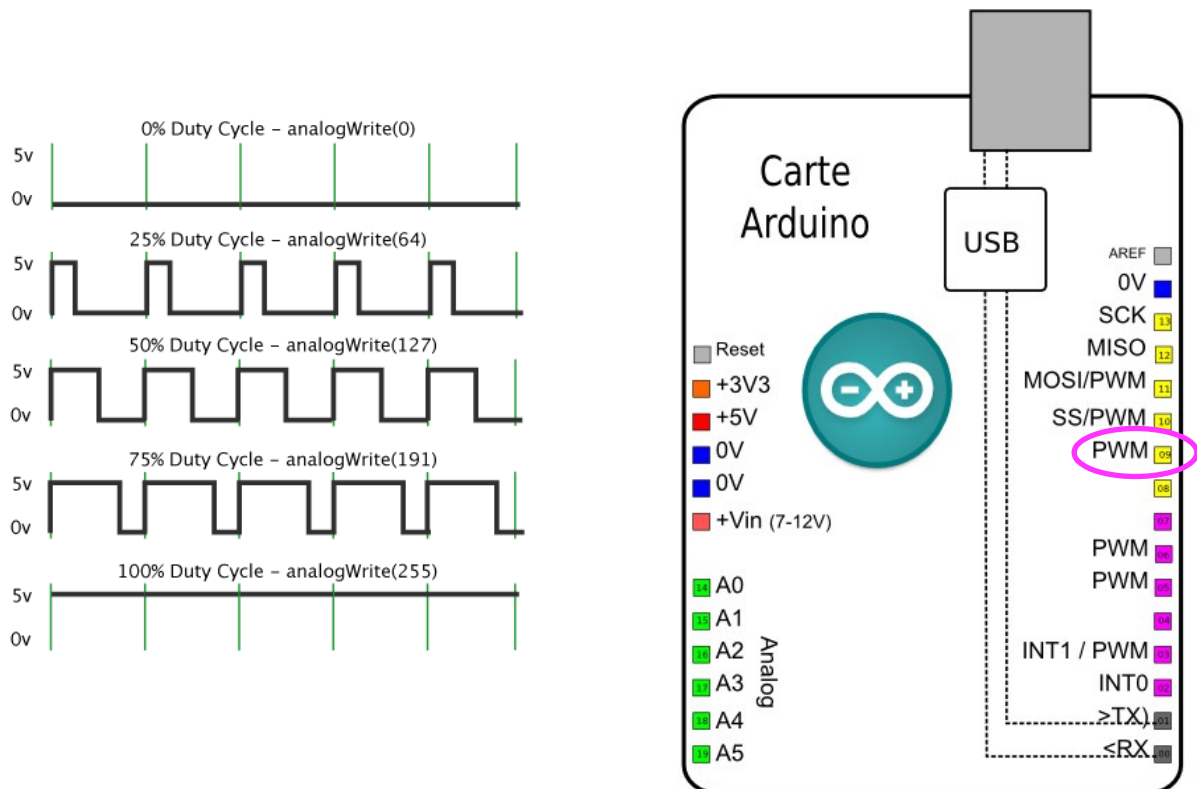
Vous comparerez vos résultats de simulation avec les courbes expérimentales.

Vous parlerez de la **stabilité** des réponses, déterminerez les **erreur**, la **rapidité à 5%**,  $t_{5\%}$ , et les **dépassements** à l'aide des graphes de simulation.

## F. Etude de la limitation en tension.

La commande du moteur se fait avec la **sortie 9** de la carte Arduino qui envoie un signal PWM codé sur 8 bits (un nombre entier de 0 à 255).

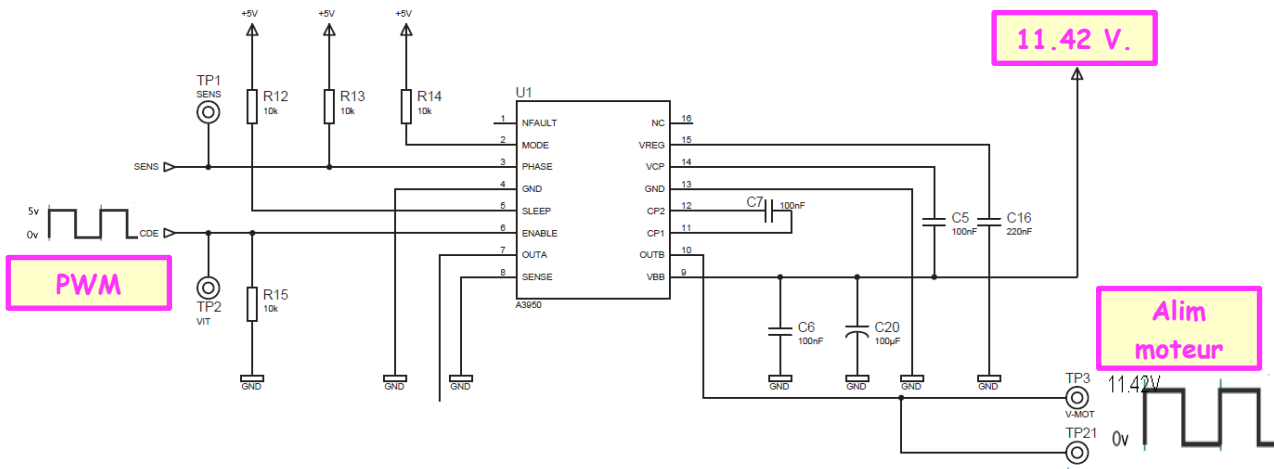
**PWM** signifie **Pulse Width Modulation** et en français cela donne **Modulation à Largeur d'Impulsion** (MLI). La PWM est en fait un signal numérique qui, à une **fréquence** donnée fixe, a un **rapport cyclique** qui change.



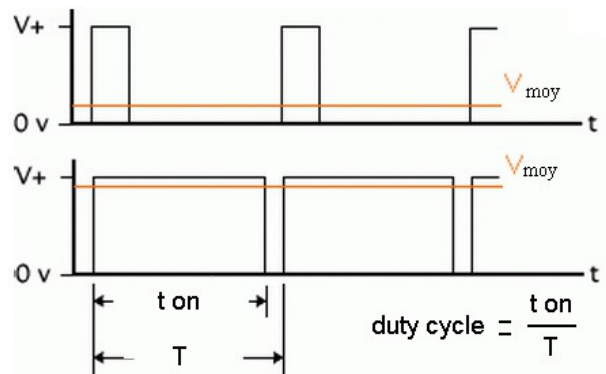
Le **rapport cyclique** est mesuré en pour cent (%). Plus le pourcentage est élevé, plus le niveau logique 1 est présent dans la période et moins le niveau logique 0 l'est. Et inversement. Le rapport cyclique du signal est donc le pourcentage de temps de la période durant lequel le signal est au niveau logique 1.

Dans notre cas, si le PWM = 0 cela donne un signal de tension continue 0V, si le PWM = 255 cela donne un signal de tension continue 5V, pour toute valeur intermédiaire du PWM cela correspond à un signal carré de rapport cyclique associé. <http://eskimon.fr/237-arduino-403-et-les-sorties-analogiques-enfin-presque>.

Il y a un composant (A3950) qui transmet le signal identique au moteur avec une **tension de référence à 11.42 V**. Le moteur interprète ce signal carré comme un signal de tension continue dont la valeur est la valeur moyenne du signal carré périodique.



Le moteur voit le signal carré périodique commandé par la sortie PWM comme un signal continu de valeur moyenne  $V_{moy}$ .



Donc la tension d'alimentation du moteur correspond à :  $U_m = PWM \times \frac{11.42}{255}$

La tension du moteur saturera tout comme le  $PWM \in [0 ; 255]$ .

**Travail F.1 :** Vous chercherez dans les outils '**CPGE/Non linéarité**' le bloc adapté à ce travail.

**N'oubliez pas votre présentation, utilisez des photos!**