

LABESSE Louis
Candidat : 50188

LE SYSTÈME DE PRETRAITEMENT DANS LES CENTRALES D'EPURATION

TIPE 2022/2023

CONTEXTE



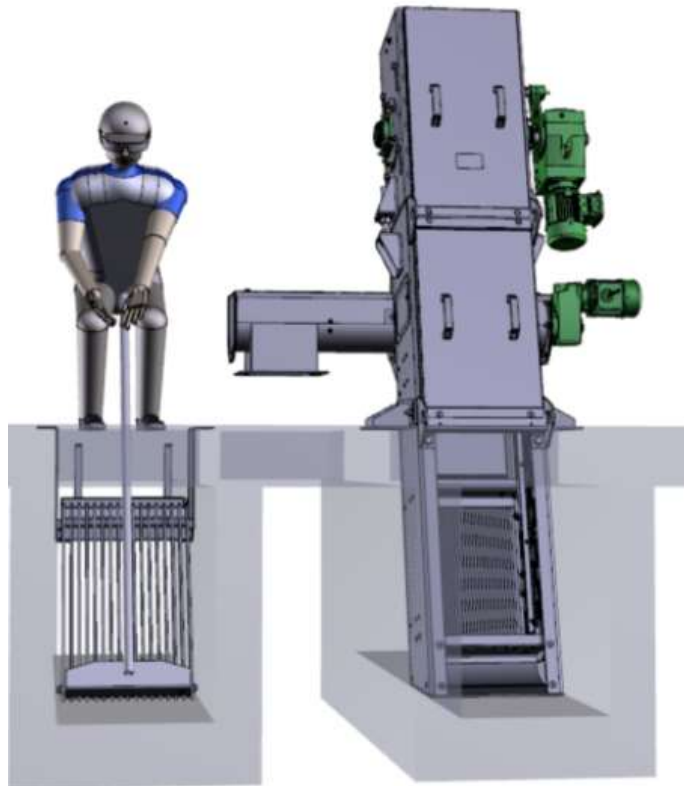
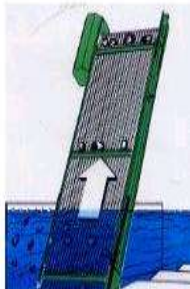
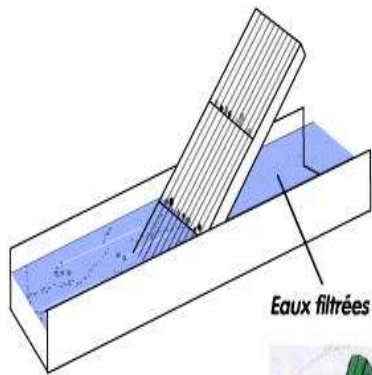
- Série de grilles à taille variable
- Absence d'automatisation dans la plupart des dégrilleurs
- Ramassage de déchets souvent manuel



OBJECTIFS

- Modéliser et construire un dégrilleur fonctionnel le plus rapproché possible d'un modèle réel
- Créer un cahier des charges du système exploitable lors des expériences
- Automatiser le ramassage à l'aide du pilotage d'un moteur
- Déterminer la masse maximale soulevable et la vitesse de levage de cette masse.
- Calcul du rendement du moteur

PRESENTATION DU DÉGRILLEUR AUTOMATIQUE INCLINÉ



CAHIER DES CHARGES

(seine aval)

Exigences du système

Raclage fréquent :
Passage du râteau
toutes les 30min, en
période non fluviale.

Rendement de 75,6%
(en puissance) du
moteur

Ramasser un refus de
3,26kg/min

Conception simple =
fiabilité durable dans
le temps



PROBLÉMATIQUE RETENUE

Comment optimiser le prétraitement dans une centrale d'épuration à l'aide d'un dégrilleur automatique?

DÉROULEMENT DU PROJET

PARTIE I : Modélisation

- I.a) Présentation de la maquette
- I.b) Mesures des frottements secs
- I.c) Mise en équation

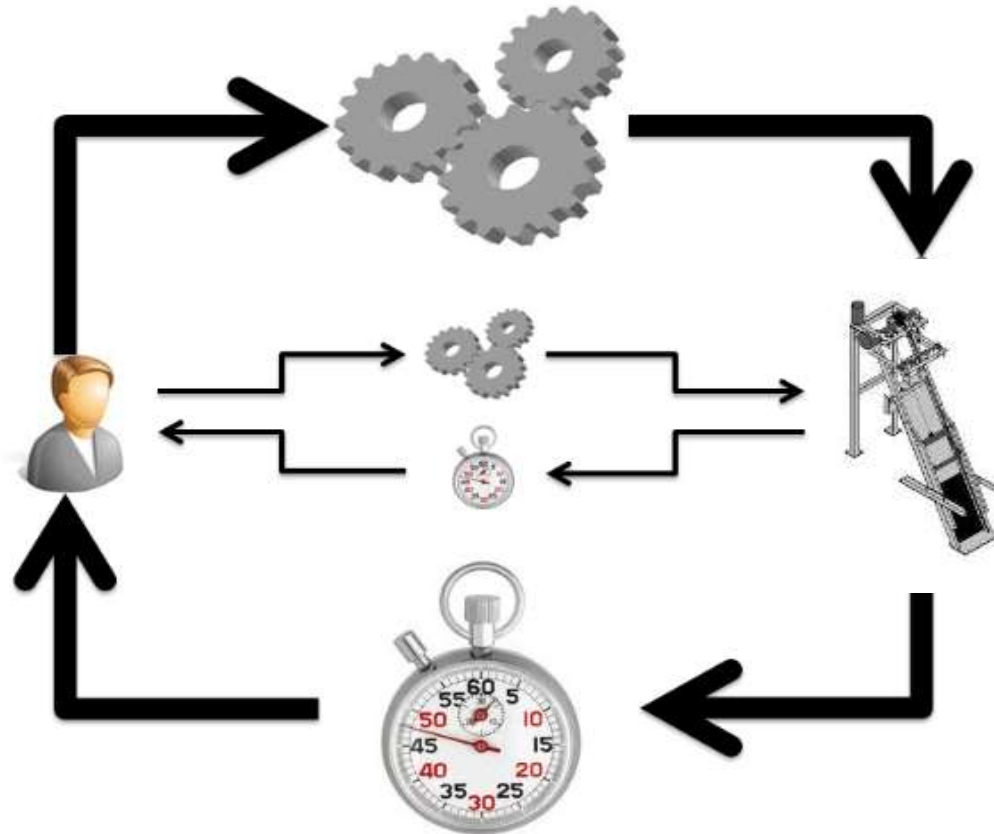
PARTIE II : Expérience du rendement

- II.a) Mesure de grandeurs caractéristiques du moteur
- II.b) Expérience de levage de masse
- II.c) Représentation de l'entrée et de la sortie
- II.d) Calcul du rendement expérimental



PARTIE III : Conclusion

- III.a) Améliorations possibles
- III.b) Ouverture sur d'autres expériences

PARTIE I: MODÉLISATION



MOTEUR ET HACHEUR

Moteur:	Hacheur:
<p>Motoréducteur 6V avec encodeur (1:53) Référence : TRENZ - 24142</p>	<p>Module PmodHB5 Diligent 12V max</p>
	

ROULEMENTS

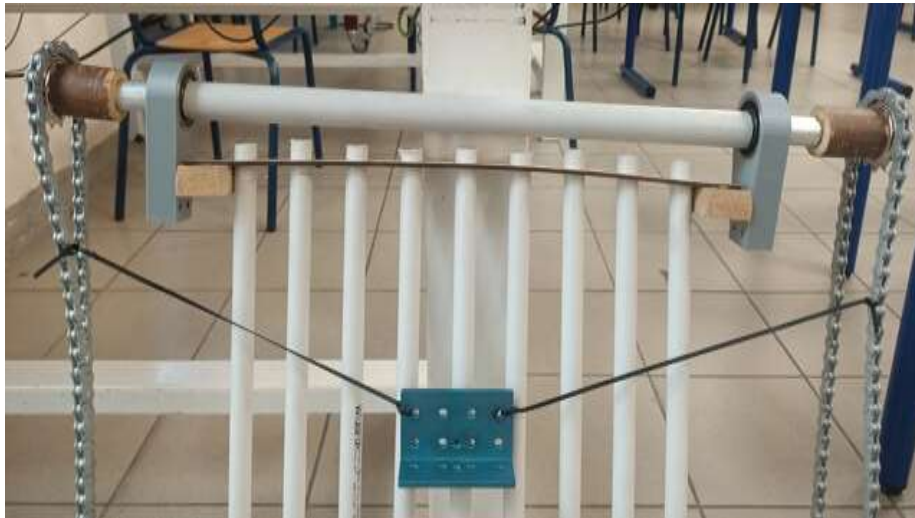


Référence: 686

Coefficient de résistance au roulements :

$$C_{RR} = 0,01$$

MAQUETTE



$$m_{chaînes} + m_{râteau} = 0,192\text{kg}$$

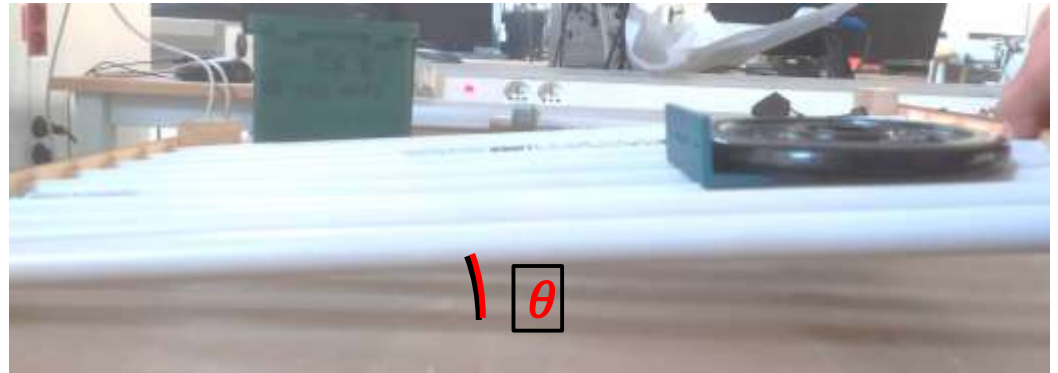
$$\alpha = 60^\circ$$

$$R_{reduc} = \frac{15}{72}$$



MESURE DES FROTTEMENTS SECS

$$F_{f,secs} = \mu * m * g * \cos(\theta)$$



Masse glissante, subie au frottements secs :

$$m = 1,042 \text{ kg}$$

$$\theta = 19^\circ$$

$$\mu = \tan(\theta) = 0,34$$

$$\text{A.N : } F_{f,secs} = 34,9 \text{ N}$$

BILAN DE PUISSANCES

Système isolé : {courroies + râteau + masse de déchets soulevée par le râteau}

$$M_{tot} = m_{chaîne} + m_{râteau} + m_{déchets}$$

Bilan de Puissances:

- Puissance du moteur :

$$P_{mot} = U_{mot} * I$$

- Puissance de pesanteur:

$$P_{pes} = M_{tot} * g * \cos(\alpha) * V_{moy}$$

- Frottements fluides du moteur :

$$P_{f,mot} = -k_{mot} * V_{moy}^2$$

- Puissance de la force de résistance au roulement :

$$P_{f,roul} = -C_{RR} * P_{pes}$$

- Frottements secs des déchets sur la grille inclinée :

$$P_{f,secs} = -F_{f,secs} * V_{moy}$$

THÉORÈME DE L' ÉNERGIE CINÉTIQUE

$$\frac{dEc}{dt} = \sum_{i=0}^n P_{i,int} + P_{i,ext}$$

Conditions: - Moteur fonctionnant en régime établi
 - Référentiel terrestre supposé galiléen

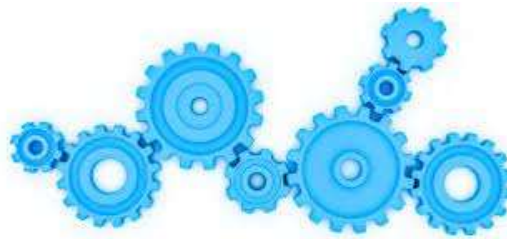
$$\begin{aligned} \frac{dEc}{dt} &= \frac{d}{dt} \left(\frac{1}{2} * M_{tot} * V_{moy}^2 + \frac{1}{2} * I * (\omega_{moy} * R_{réduc})^2 \right) \\ &= M_{tot} * \frac{dV_{moy}}{dt} * V_{moy} + I * \frac{d\omega_{moy}}{dt} * \omega_{moy} * R_{réduc}^2 = 0 \end{aligned}$$

$$P_{mot} + P_{pes} + P_{f,secs} + P_{f,roul} + P_{f,mot} = 0$$

$$U_{mot} * I + P_{pes} * (1 - C_{RR}) - F_{f,secs} * V_{moy} - k_{mot} * V_{moy}^2 = 0$$

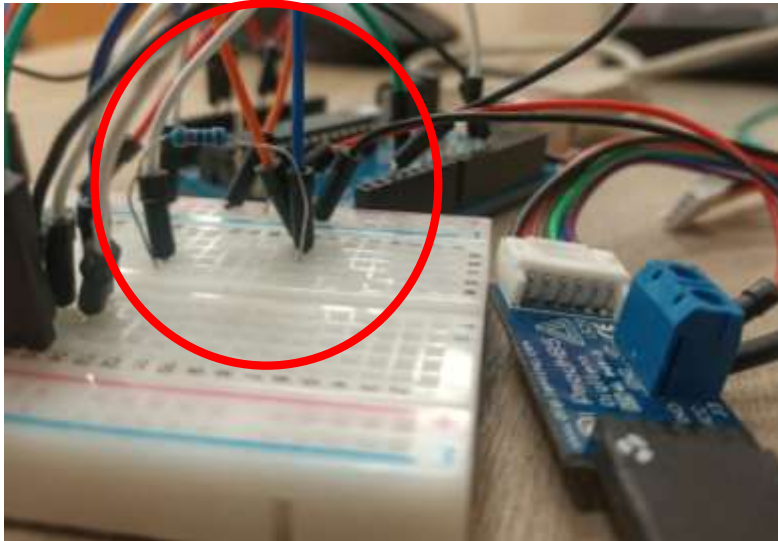
$$U_{mot} * I + P_{pes} * (1 - C_{RR}) - F_{f,secs} * V_{moy} - k_{mot} * V_{moy}^2 = 0$$

PARTIE II: EXPERIENCE DU RENDEMENT



GRANDEURS CARACTÉRISTIQUES DU MOTEUR

Résistance en série



Tension aux bornes de la résistance
et du moteur sur Arduino



Mesures:

$$R = 100 \, \Omega$$

$$U_R = 4,98V$$

$$U_{mot} = 2,21V$$

Loi d'Ohm du Moteur:

$$I = 49,8 \, mA$$

$$R_{mot} = \frac{U_R - U_{mot}}{I} = 55,62 \, \Omega$$

EXPÉRIENCE DE LEVAGE DE MASSE

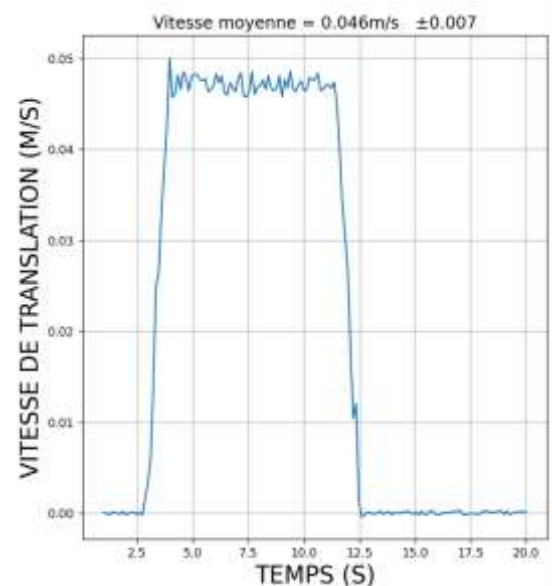
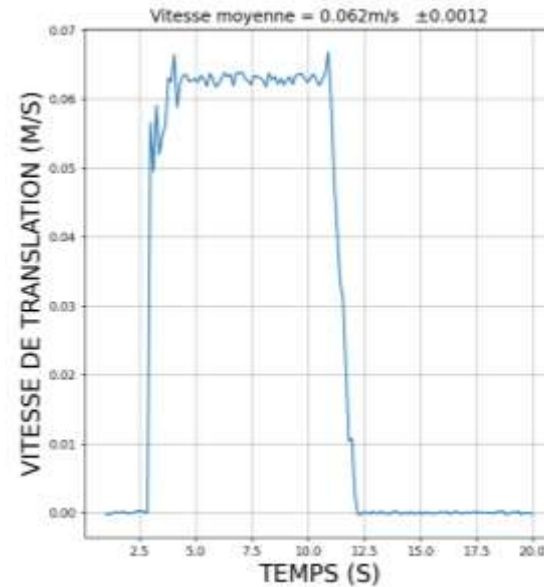
Le moteur est supposé fonctionner en régime établi pendant les expériences

- À vide : $m_{déchets} = 0\text{kg}$

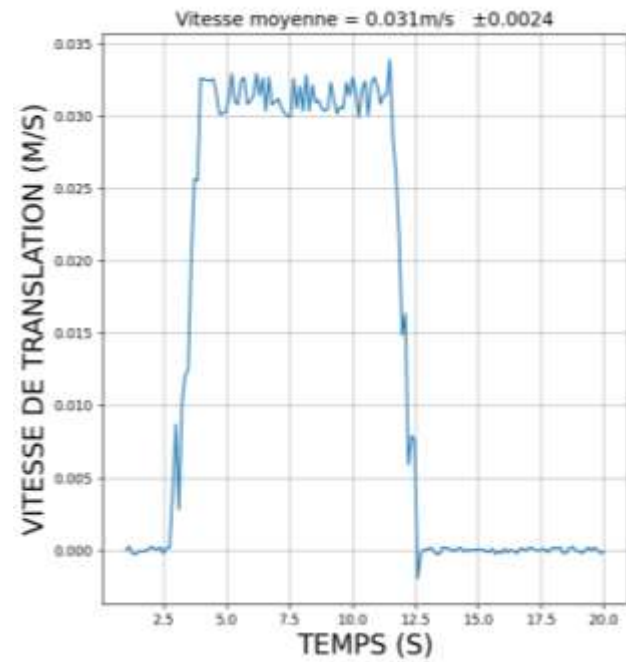
- Masse déplacée :

$$m_{chaînes} + m_{râteau} = 0,192\text{kg}$$

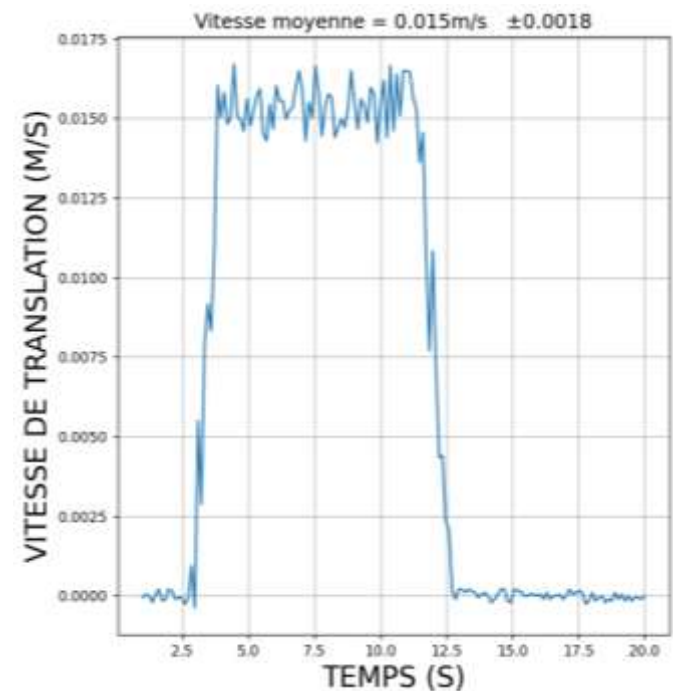
$$m_{déchets} = 100\text{kg}$$



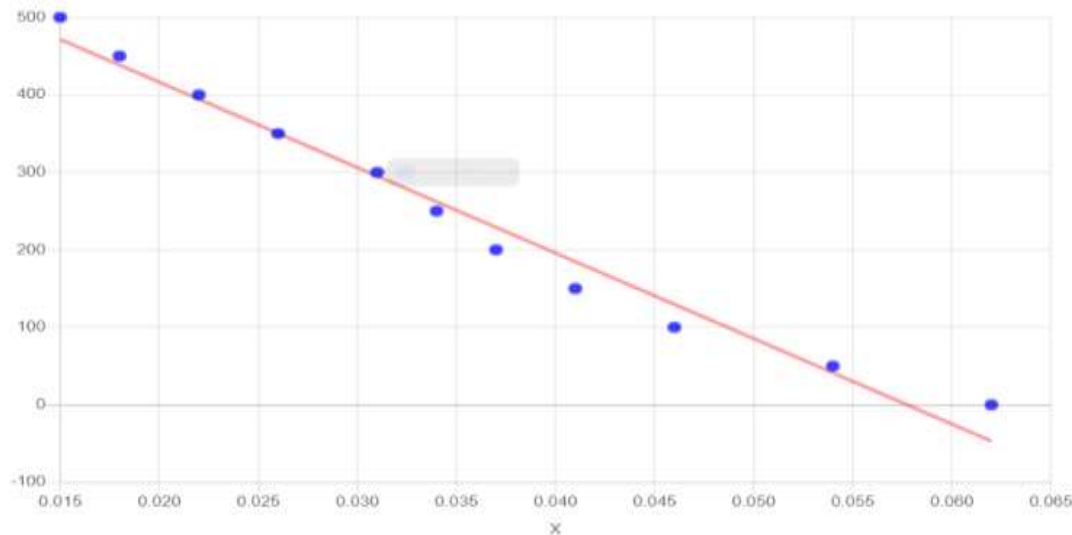
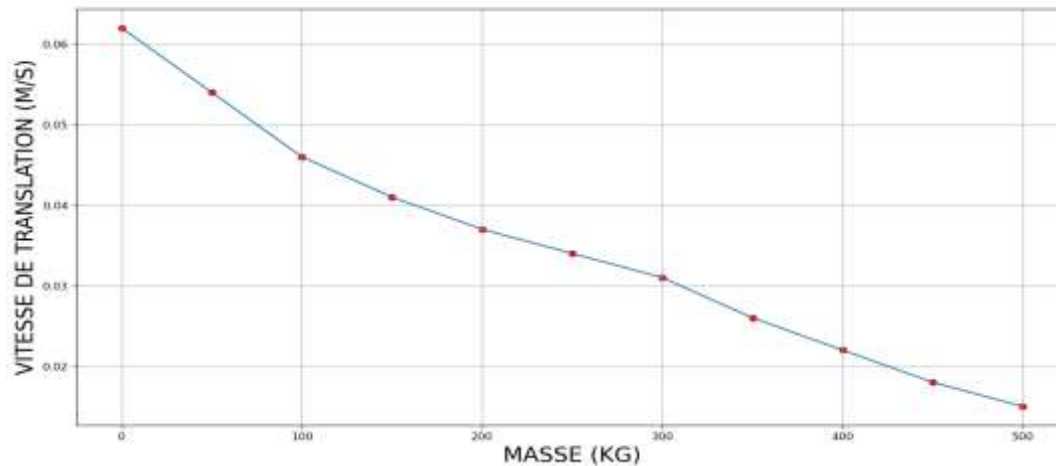
$$m_{\text{déchets}} = 300\text{kg}$$



$$m_{\text{déchets}} = 0,5\text{kg}$$

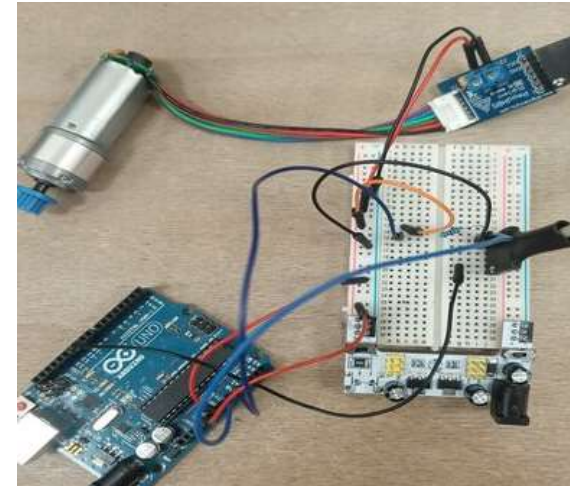
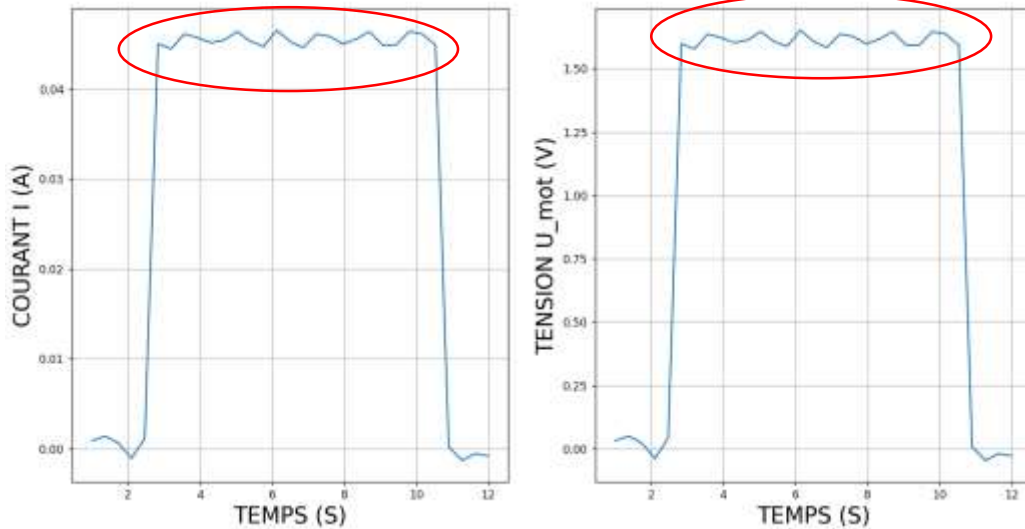


REPRÉSENTATION DE LA VITESSE DE SORTIE EN FONCTION DE LA MASSE



$$Y = 637.1766 - 11033.5311X$$

OBSERVATION DE L'ENTRÉE

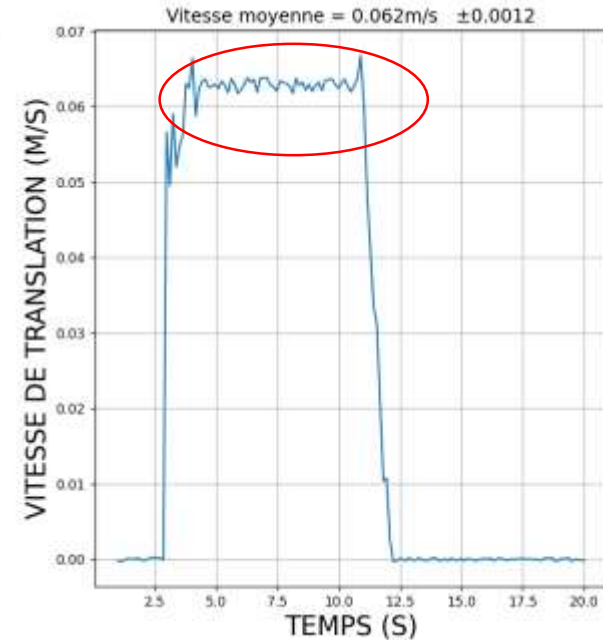
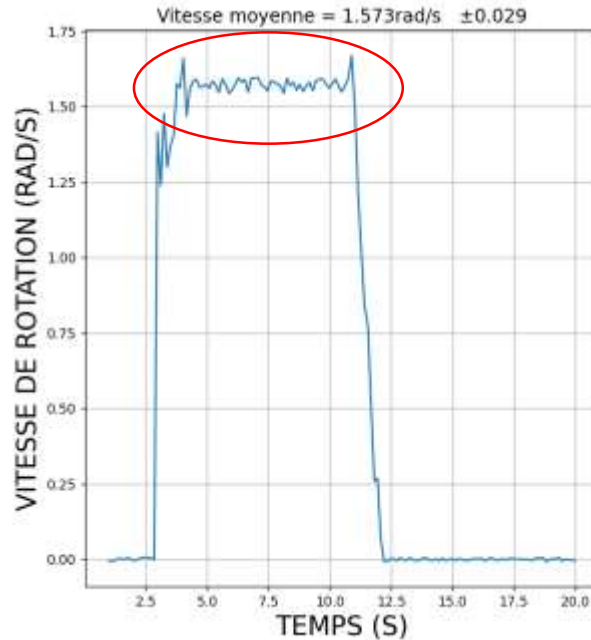


- $I = 49,8 \text{ mA}$
- $U_{mot} = 2,21 \text{ V}$
- $P_{mot} = U_{mot} * I = 0,11 \text{ Watt}$

OBSERVATION DE LA SORTIE

Système à vide:

- $m_{déchets} = 0\text{kg}$
- $M_{tot} = 0,192\text{kg}$
- $\omega_{moy} = 2,248 \text{ rad/s}$
- $R_{arbre_sortie} = 0,04\text{m}$
- $V_{moy} = 0,062 \text{ m/s}$
- $R_{reduc} = \frac{15}{72}$
- $m_{chaînes} + m_{râteau} = 0,372\text{kg}$



- $P_{pes} = M_{tot} * g * \cos(\alpha) * V_{moy} = 0,058\text{Watt}$

CALCUL DU RENDEMENT

$$\eta = \frac{P_{mot}}{P_{pes}} = \frac{0,058}{0,11} = 53\%$$

- Source d'alimentation électrique limitée
- Vitesse très faible en régime établi (0,62m/s sans charge supplémentaire)
- Rendement correct : beaucoup de pertes énergétiques sont négligeables par rapport au modèle réel en station d'épuration.

(Pertes par effet Joule, de nombreux frottements fluides et visqueux dans le canal où est entreposé le dégrilleur...)

CONCLUSION

- Choix des matériaux: modèle réel : Acier inox plus solide que le bois.
- Conception de la maquette: encastrement du râteau sur les chaînes à revoir.
- Choix du moteur : MCC (semblable au moteur réel)
- Maquette non prévue à un étude dans un canal (imperméabilité)
- Axes d'ouvertures sur d'autres études concernant la maquette :
 - Asservissement: en fonction des périodes fluviales, une perte de charge est observée dans le canal entre l'amont du dégrilleur et à l'aval traduisant une grosse accumulation de déchets sur la grille.
 - RDM : Information pouvant être rajoutée sur la cahier des charges : les déchets arrivent en moyennent à des vitesse de 1,4m/s sur la grille, qui doit supporter des charges appliquées a la partie immergée allant jusqu'à 890N en période de pluie.

FIN DE LA PRESENTATION

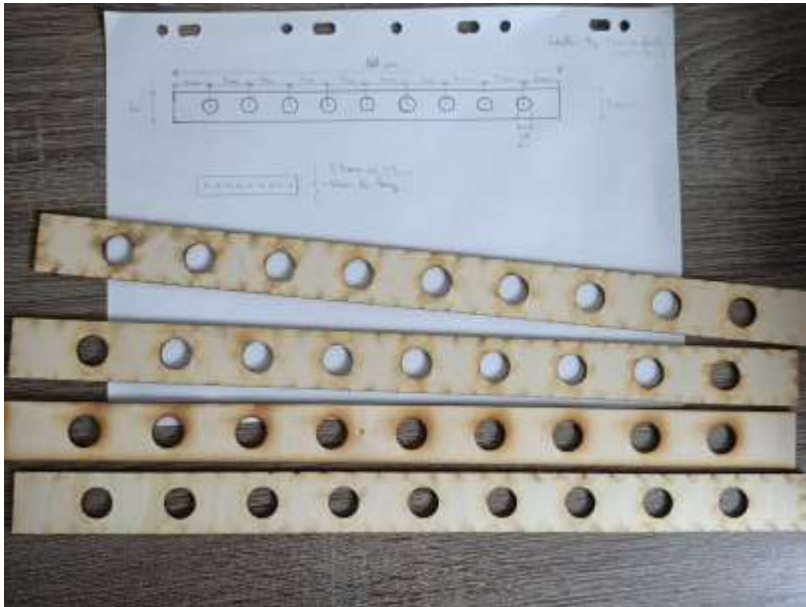
Merci de m'avoir écouté



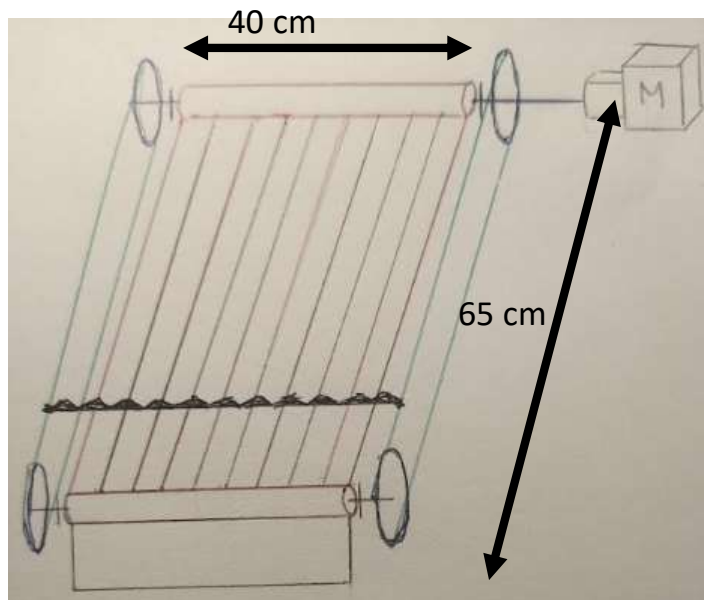
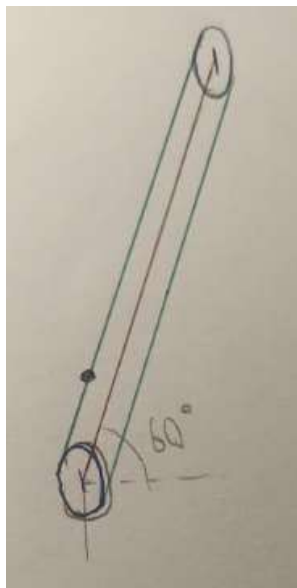
Annexes (1)



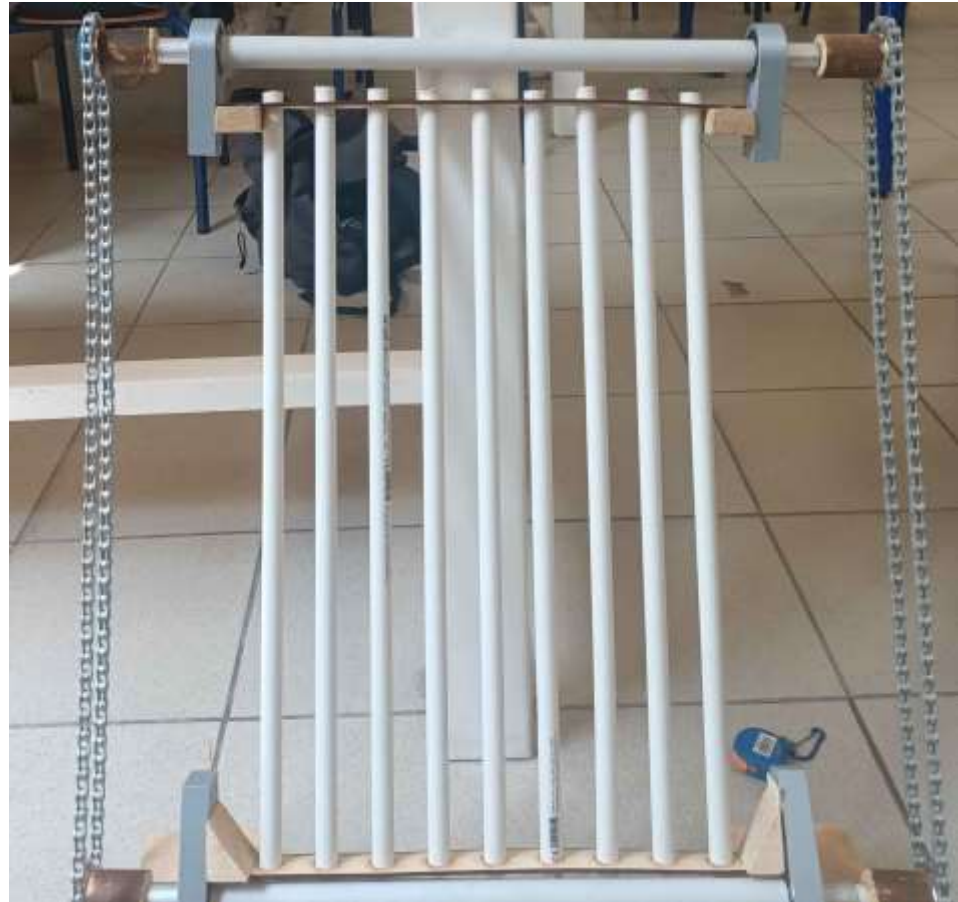
Annexes (2)



Annexes (3)

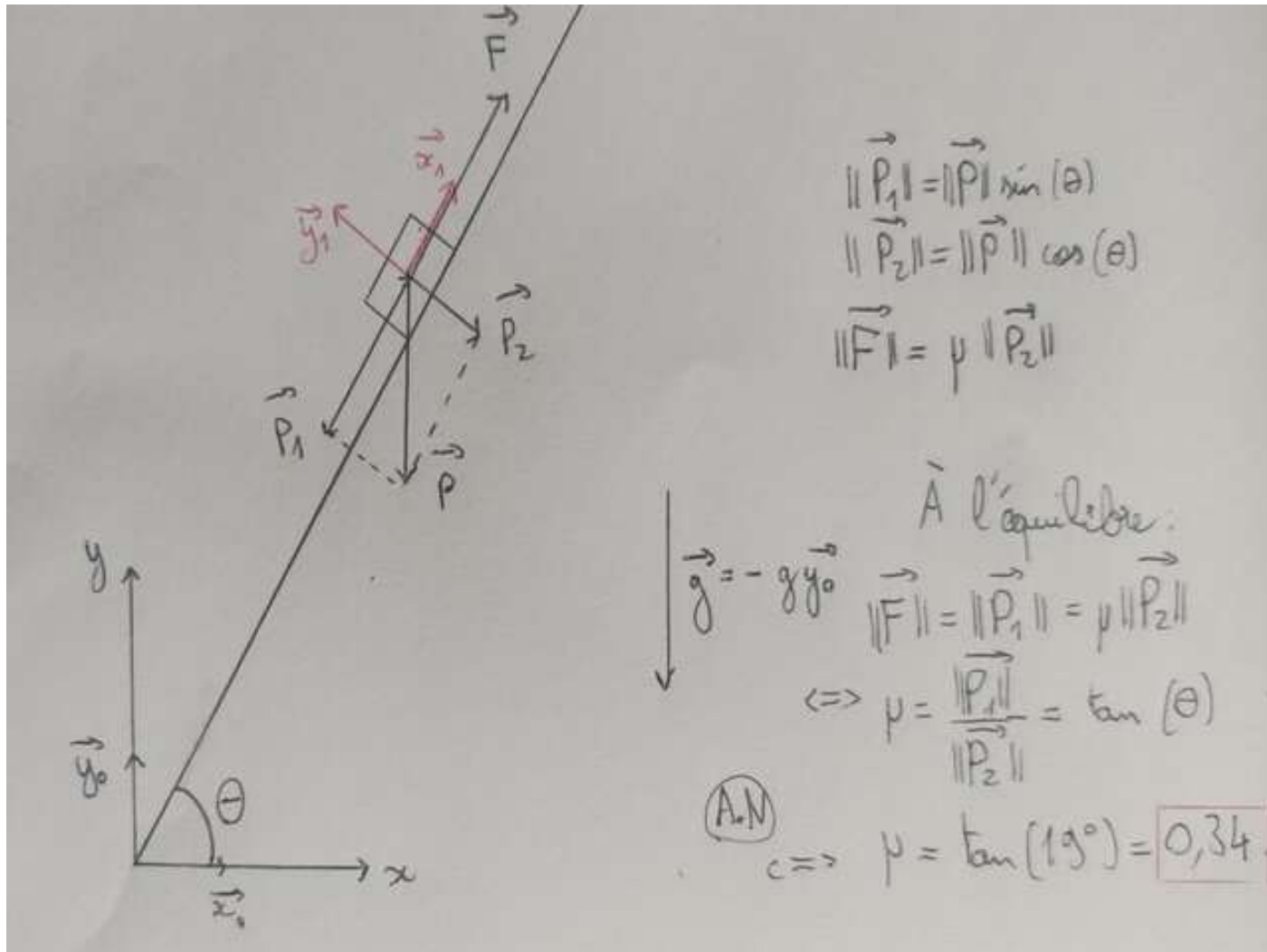


Annexes (4)



Annexes (5)

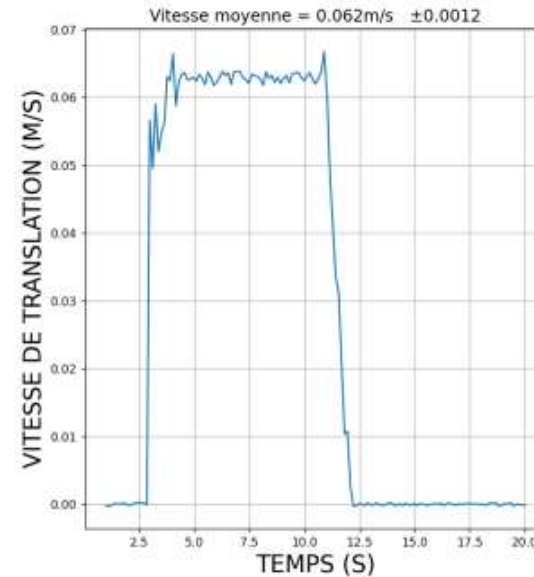
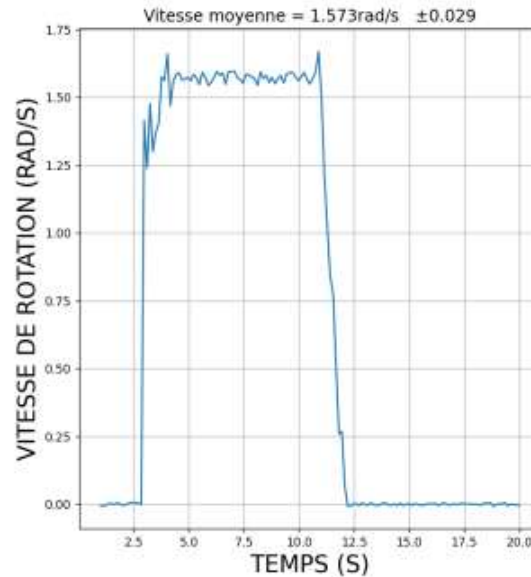
Schéma des projections lors du calcul des frottements secs



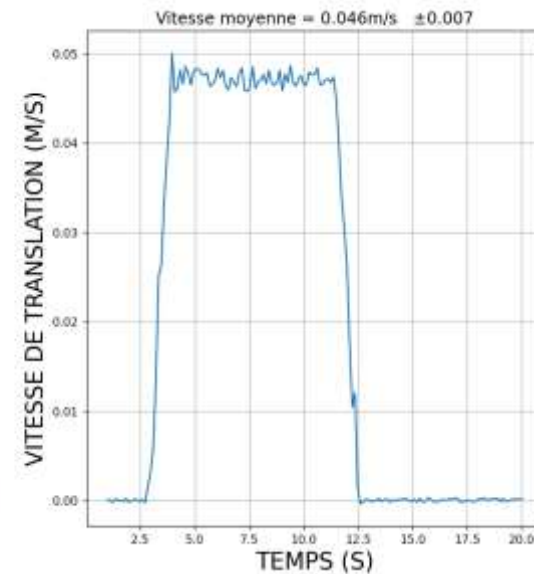
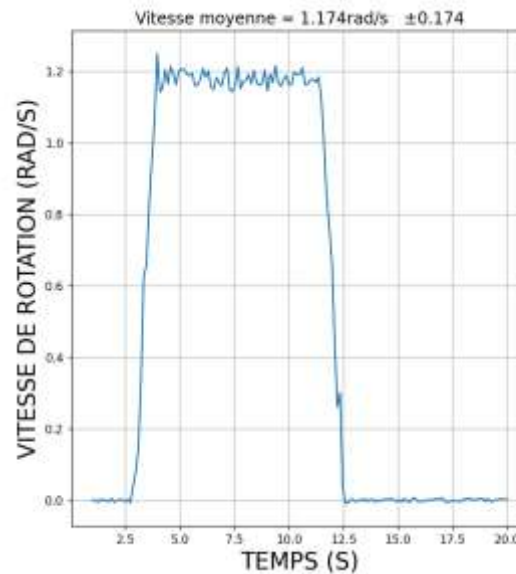
Annexes (6)

Courbes obtenues lors des expériences de soulèvement de la masse

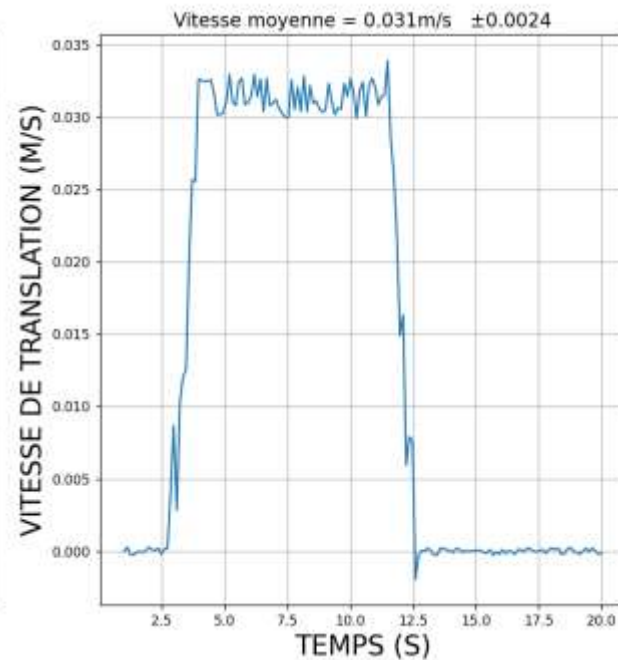
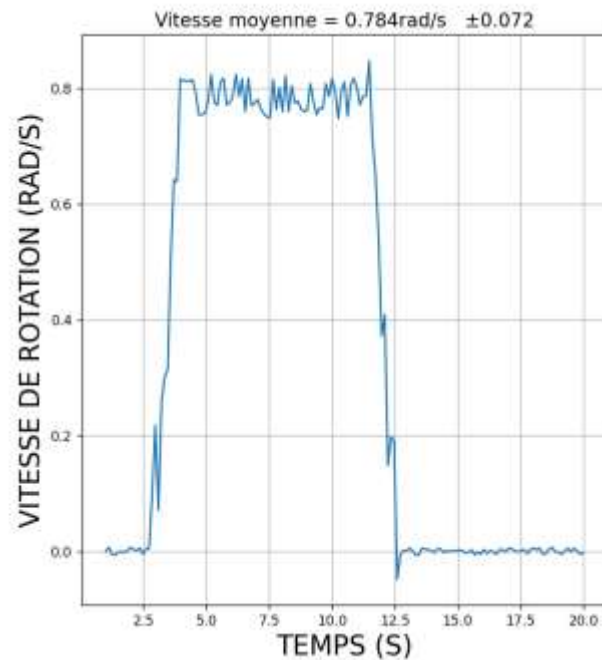
$$m_{déchets} = 0\text{kg}$$



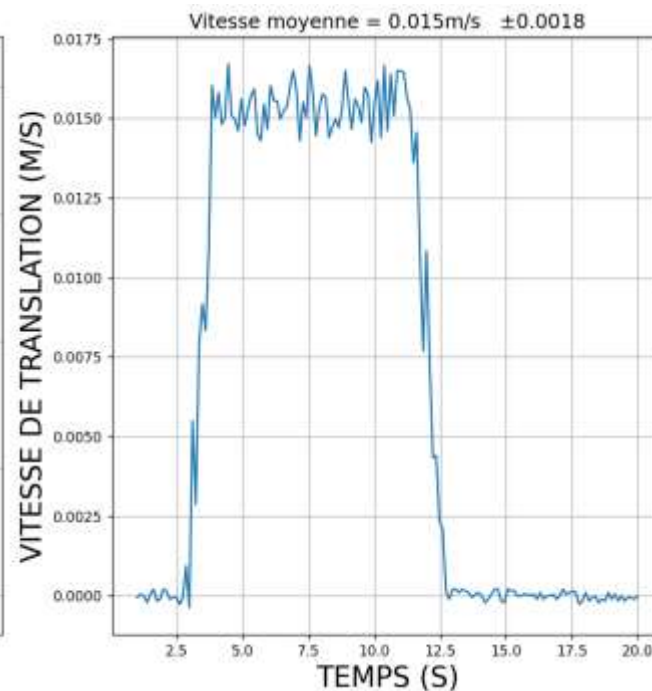
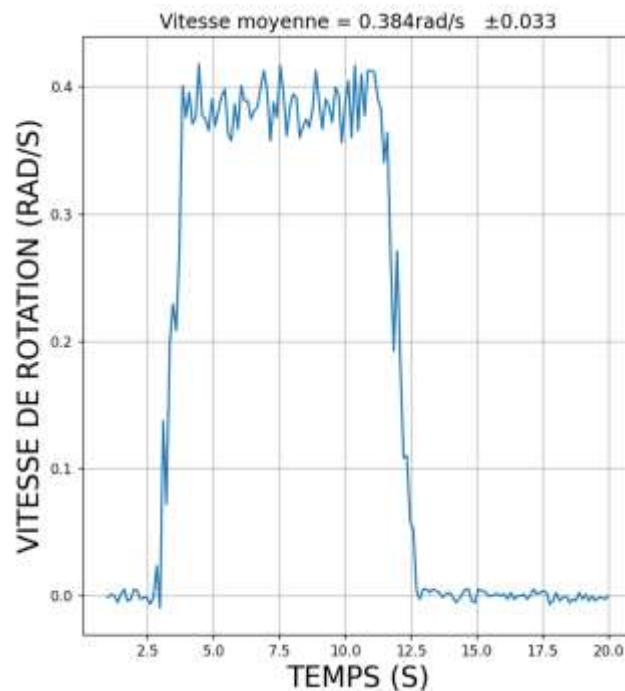
$$m_{déchets} = 0,1\text{kg}$$



$$m_{\text{déchets}} = 300\text{kg}$$



$$m_{\text{déchets}} = 0,5\text{kg}$$



Annexes (7)

CODES ARDUINO

Codes pour afficher la tension du moteur de la résistance en série et du courant

```
ObserverTensionetCourant.ino
1  const int motorPin = 11; // Broche de contrôle du moteur DC
2  const float referenceVoltage = 5.0; // Tension de référence (5V pour la plupart des Arduinos)
3  const float resistorValue = 100.0; // Valeur de la résistance en ohms
4
5  void setup() {
6      pinMode(motorPin, OUTPUT); // Définir la broche de contrôle du moteur DC en sortie
7      Serial.begin(115200); // Démarrer la communication série à une vitesse de 9600 bauds
8  }
9
10 void loop() {
11     // Activer le moteur dans une direction spécifique (dans cet exemple, le moteur tourne dans le sens horaire)
12     digitalWrite(motorPin, LOW);
13
14     // Mesurer la tension aux bornes du moteur
15     int sensorValue = analogRead(A0); // Lire la tension analogique à partir d'une broche analogique (A0 dans cet exemple)
16     float voltage = sensorValue * (referenceVoltage / 1023.0); // Convertir la valeur analogique en tension réelle
17
18     // Calculer le courant à travers la résistance
19     float current = voltage / resistorValue;
20
21     // Afficher la tension et le courant sur le moniteur série
22     Serial.print("Tension Mot : ");
23     Serial.print(voltage);
24     Serial.println(" V");
25
26     Serial.print("Tension R : ");
27     Serial.print(current);
28     Serial.println(" A");
29
30     delay(1000); // Attendre 1 seconde
31 }
```

```

1  const int pinTensionR = A0;
2  const int pinTensionM = A1; // Broche analogique utilisée pour l'acquisition de tension
3  #define EN 6;
4  int k = 0; // Exemple de broche de contrôle pour la MCC
5
6  void setup() {
7      Serial.begin(9600); // Initialise la communication série à une vitesse de 9600 bauds
8      pinMode(11, OUTPUT); // Configure la broche de contrôle en tant que sortie
9      analogReference(5);
10 }
11
12 void loop() {
13     // Contrôler la MCC selon vos besoins
14     // Exemple : activer le sens de rotation souhaité
15
16     // Effectuez l'acquisition de tension
17
18     // Affichez les données
19     digitalWrite(11, HIGH);
20     while(k<100){
21         int tensionR = analogRead(pinTensionR);
22         int tensionM = analogRead(pinTensionM); // lit la valeur analogique de la broche
23         // float tensionEnVolts = map(tension, 0, 1023, 0, 5.0); // Convertit la valeur en volts (0-1023 correspond à une plage de 0-5V)
24
25         //Serial.print(millis()); // Affiche le temps écoulé en millisecondes
26         Serial.print("\t");
27         Serial.println(tensionR);
28
29         Serial.print(tensionM);
30
31         // Serial.println(tensionEnVolts);
32
33         k=k+1;
34     }
35     // Affiche la tension en volts
36
37     k=0;
38     digitalWrite(11, LOW);
39     // delay(4000); // Attente d'une seconde avant de lire à nouveau la tension
40 }
41

```

Codes pour afficher la position angulaire du moteur, puis de la vitesse angulaire

```
1 #define ENCA 2
2 #define ENCB 3
3
4 void setup() {
5     Serial.begin(9600);
6     pinMode(ENCA, INPUT);
7     pinMode(ENCB, INPUT);
8 }
9
10 void loop() {
11     int a = digitalRead(ENCA);
12     int b = digitalRead(ENCB);
13     Serial.print(a*5);
14     Serial.print(" ");
15     Serial.print(b*5);
16     Serial.println();
17 }
18
19 #include <Encoder.h>
20
21 // Broches de l'encodeur
22 #define PIN_A 2
23 #define PIN_B 3
24
25 // Broche de commande du moteur
26 #define MOTOR_PIN 11
27
28 // Objet Encoder pour lire les impulsions de l'encodeur
29 Encoder encoder(PIN_A, PIN_B);
30
31 // Variables pour le calcul de la vitesse
32 long positionPrecedente = 0;
33 unsigned long tempsPrecedent = 0;
34
35 // Réglages de l'encodeur
36 const int impulsionsParTour = 2; // nombre d'impulsions par tour de l'encodeur
37 const float rapportReduction = 15; // rapport de réduction du moteur
38
39 void setup() {
40     // Configuration des broches
41     pinMode(MOTOR_PIN, OUTPUT);
42
43     // Initialisation de la communication série
44     Serial.begin(9600);
45 }
46
47 void loop() {
48     digitalWrite(MOTOR_PIN, HIGH);
49
50     // lecture de la position de l'encodeur
51     long position = encoder.read();
52
53     // Calcul de la vitesse en tours par seconde
54     unsigned long tempsActuel = millis();
55     float deltaT = (tempsActuel - tempsPrecedent) / 1000.0; // temps écoulé en secondes
56     float vitesse = ((position - positionPrecedente) / impulsionsParTour) / (rapportReduction * deltaT); // calcul de la vitesse en tours par seconde
57
58     // Mise à jour de la position et du temps précédents
59     positionPrecedente = position;
60     tempsPrecedent = tempsActuel;
61
62     // Contrôle du moteur
63     // if (vitesse > 0) {
64     //     digitalWrite(MOTOR_PIN, HIGH); // fait tourner le moteur dans un sens
65     // } else {
66     //     digitalWrite(MOTOR_PIN, LOW); // Arrête le moteur
67     // }
68
69     // Affichage de la vitesse sur le moniteur série
70     Serial.print("Vitesse : ");
71     Serial.print(vitesse);
72     Serial.println(" tours/s");
73
74     // Attendez une petite pause pour éviter un rafraîchissement trop rapide de la vitesse
75     delay(100);
76 }
```

```

1 #include <Encoder.h>
2
3 #define PIN_A 2
4 #define PIN_B 3
5 #define INTERVALLE_MESURE 1000 // Intervalle de mesure en millisecondes
6 #define RAPPORT_REDUK 1.0 / 19.0
7 #define PIN_MOTEUR 4
8 #define DUREE_ROTATION 5000 // Durée de rotation en millisecondes
9
10 Encoder encoder(PIN_A, PIN_B);
11 unsigned long temps_prec = 0;
12 unsigned long temps_debut_rotation = 0;
13
14 void setup() {
15     Serial.begin(9600);
16     pinMode(PIN_MOTEUR, OUTPUT);
17 }
18
19 void loop() {
20     unsigned long temps_actuel = millis();
21
22     if (temps_actuel - temps_debut_rotation <= DUREE_ROTATION) {
23         if (temps_actuel - temps_prec >= INTERVALLE_MESURE) {
24             long delta_position = encoder.readAndReset() * RAPPORT_REDUK;
25
26             float vitesse_tours_par_sec = (float)delta_position / (INTERVALLE_MESURE / 1000.0);
27             float vitesse_tours_par_min = vitesse_tours_par_sec * 60.0;
28
29             Serial.print("Vitesse : ");
30             Serial.print(vitesse_tours_par_min, 3); // Affiche la valeur avec 3 décimales
31             Serial.println(" tours/min");
32
33             temps_prec = temps_actuel;
34
35             // Faire tourner le moteur dans le sens horaire
36             digitalWrite(PIN_MOTEUR, HIGH);
37         }
38     } else {
39         // Arrêter le moteur après la durée de rotation spécifiée
40         digitalWrite(PIN_MOTEUR, LOW);
41     }
42 }

```

Annexes (8)

CODES PYTHON

Codes python pour afficher les données Arduino stockées dans des fichiers Excel:

```
45 fichier1,fichier2,fichier3 = open('Tension.csv'), open('Courant.csv'),open('vitesseangulaire.csv')
46 liste1 = []
47 T1 = []
48 for ligne in fichier1:
49     liste1 = ligne.split(';')
50
51     # print(liste1)
52     T1.append(liste1)
53
54 liste2 = []
55 T2 = []
56
57 for ligne in fichier1:
58     liste2 = ligne.split(';')
59     # print(liste2)
60     T2.append(liste2)
61
62 liste3 = []
63 T3 = []
64 for ligne in fichier1:
65     liste3 = ligne.split(';')
66     # print(liste3)
67     T1.append(liste3)
68
69 #print(len(T))
70 #print(T[0][2])
71
72 def moyenne(a,b,L):
73     M = []
74     for k in range(len(L)):
75         if L[k] > a and L[k] < b:
76             M.append(L[k])
77
78     S = 0
79     n = len(M)
80     for i in M:
81         S += i
82     return S/n
```

Codes python pour afficher tracer les courbes des données stockées dans des fichiers Excel:

```
116 L = [x*2*np.pi*(15/72) for x in L]
117
118 T = np.linspace(1,20,len(L))
119 plt.subplot(1,2,1)
120 plt.axis()
121 plt.grid()
122 plt.xlabel('TEMPS (s)', fontsize=20)
123 plt.ylabel('VITESSE DE ROTATION (RAD/S)', fontsize=20)
124 a,b = 0.35,0.45
125 Moy1 = moyenne(a,b,L)
126 plt.title('Vitesse moyenne = ' + str(Moy1)[:5] + ' rad/s' + ' ± ' + str(arrondi(max(maxi(a,b,L) - Moy1, Moy1 - mini(a,b,L)), n = 3)), fontsize = 14)
127 plt.plot(T,L)
128 plt.subplot(1,2,2)
129 plt.axis()
130 plt.grid()
131 plt.xlabel('TEMPS (s)', fontsize=20)
132 plt.ylabel('VITESSE DE TRANSLATION (M/S)', fontsize=20)
133 L1 = [x*0.04 for x in L]
134 c,d = 0.0125,0.0175
135 Moy2 = moyenne(c,d,L1)
136 plt.title('Vitesse moyenne = ' + str(Moy2)[:5] + ' m/s' + ' ± ' + str(arrondi(max(maxi(c,d,L1) - Moy2, Moy2 - mini(c,d, L1)), n = 4)), fontsize = 14)
137 plt.plot(T,L1)
138 plt.show()
139
140
```

Codes python pour tracer le résultat des expériences de levage de masses

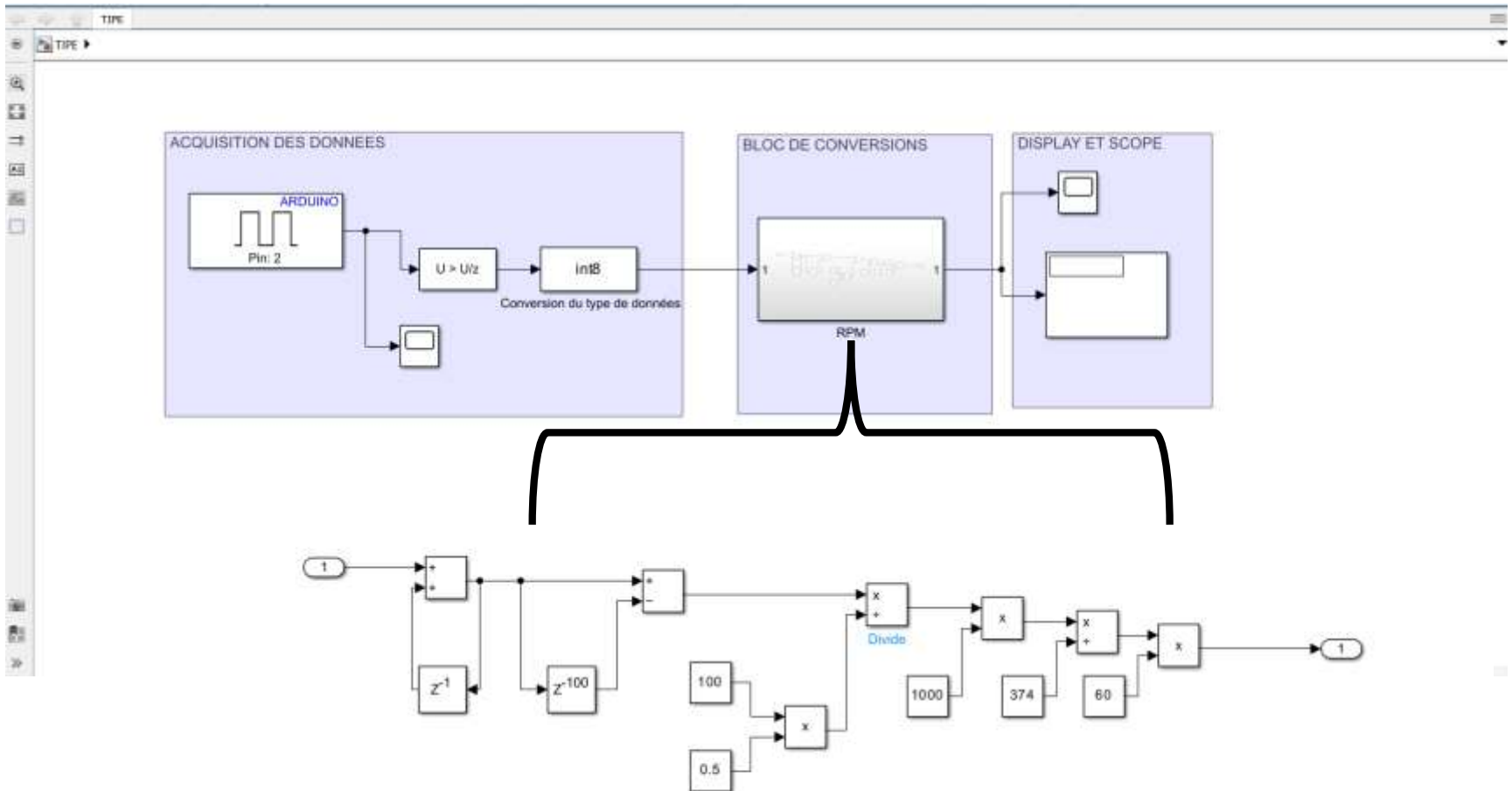
```
145
146
147 masses = [0,50,100,150,200,250,300,350,400,450,500]
148 vitesses = [0.062,0.054,0.046,0.041,0.037,0.034,0.031,0.026,0.022,0.018,0.015]
149
150 plt.axis()
151 plt.grid()
152 plt.xlabel('MASSE (KG)', fontsize=20)
153 plt.ylabel('VITESSE DE TRANSLATION (M/S)', fontsize=20)
154 plt.scatter(masses, vitesses, color='red', label='Points')
155 plt.plot(masses, vitesses)
156 plt.show()
157
158
159
160
161
162
163
```

	A	B
1	0.062	0
2	0.054	50
3	0.046	100
4	0.041	150
5	0.037	200
6	0.034	250
7	0.031	300
8	0.026	350
9	0.022	400
10	0.018	450
11	0.015	500
12		

Annexes (9)

SCHEMA BLOC SIMULINK

Tentative de modélisation sous matlab et simulink, mais les données arduino n'étaient pas correctement transférés



Annexes (10)

FICHE TECHNIQUE DU MOTEUR

2
3
4

參考
REF.

估價
EVALUATE

試作
INITIAL

製作
MASS

4~M2x4.0 dp.

PCD $\phi 18$

$\phi 22^{+0}_{-0.1}$

$4 \sim 90^\circ$

12 ± 0.1

UL1007 AWG24
Length: 100mm

Two Channel Encoder
Connections :

1. Red	+ MOTOR
2. Red	HALL SENSOR Vcc
3. Green	HALL SENSOR GND
4. Blue	HALL SENSOR A Vout
5. Purple	HALL SENSOR B Vout

IST PH-6
P=2.0-8P

REDUCTION RATIO: 1/19.22

OUTPUT SHAFT: STEEL

MOTOR SPECIFICATION: 12V 15000RPM

OUTPUT-789RPM/ $\leq 200\text{mA}$ WITH NO LOAD

核准
APPROVED BY

一般公差表

核對
CHECKED BY

繪圖
DRAWING BY

制圖	孔徑	長度	角度	1	新出圖																																		
<p>一指定端點區分之數值公差</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>1~6</td> <td>6~18</td> <td>18~50</td> <td>50~120</td> <td>120~250</td> <td>250~500</td> <td>500~1000</td> <td>1000~2000</td> <td>2000~5000</td> <td>5000~10000</td> </tr> <tr> <td>0.1</td> <td>0.2</td> <td>0.3</td> <td>0.4</td> <td>0.5</td> <td>0.6</td> <td>0.8</td> <td>1.0</td> <td>1.2</td> <td>1.6</td> </tr> <tr> <td>0.2</td> <td>0.4</td> <td>0.6</td> <td>0.8</td> <td>1.0</td> <td>1.2</td> <td>1.6</td> <td>2.0</td> <td>2.5</td> <td>3.2</td> </tr> </table>										1~6	6~18	18~50	50~120	120~250	250~500	500~1000	1000~2000	2000~5000	5000~10000	0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.6	0.2	0.4	0.6	0.8	1.0	1.2	1.6	2.0	2.5	3.2
1~6	6~18	18~50	50~120	120~250	250~500	500~1000	1000~2000	2000~5000	5000~10000																														
0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.6																														
0.2	0.4	0.6	0.8	1.0	1.2	1.6	2.0	2.5	3.2																														
<p>未指定長度公差</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>±0.1</td> <td>±0.2</td> </tr> </table>										±0.1	±0.2																												
±0.1	±0.2																																						

件名
PART NAME

圖號
DWG.

GEARMOTOR

IG220019X00015R

單位
UNITS

比例
SCALE

mm

9/10

祥儀企業股份有限公司

SHA YANG YE INDUSTRIAL CO., LTD.

2
3
4

核准
APPROVED BY

一般公差表

核對
CHECKED BY

繪圖
DRAWING BY

制圖	孔徑	長度	角度	1	新出圖																																		
<p>一指定端點區分之數值公差</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>1~6</td> <td>6~18</td> <td>18~50</td> <td>50~120</td> <td>120~250</td> <td>250~500</td> <td>500~1000</td> <td>1000~2000</td> <td>2000~5000</td> <td>5000~10000</td> </tr> <tr> <td>0.1</td> <td>0.2</td> <td>0.3</td> <td>0.4</td> <td>0.5</td> <td>0.6</td> <td>0.8</td> <td>1.0</td> <td>1.2</td> <td>1.6</td> </tr> <tr> <td>0.2</td> <td>0.4</td> <td>0.6</td> <td>0.8</td> <td>1.0</td> <td>1.2</td> <td>1.6</td> <td>2.0</td> <td>2.5</td> <td>3.2</td> </tr> </table>										1~6	6~18	18~50	50~120	120~250	250~500	500~1000	1000~2000	2000~5000	5000~10000	0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.6	0.2	0.4	0.6	0.8	1.0	1.2	1.6	2.0	2.5	3.2
1~6	6~18	18~50	50~120	120~250	250~500	500~1000	1000~2000	2000~5000	5000~10000																														
0.1	0.2	0.3	0.4	0.5	0.6	0.8	1.0	1.2	1.6																														
0.2	0.4	0.6	0.8	1.0	1.2	1.6	2.0	2.5	3.2																														
<p>未指定長度公差</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>±0.1</td> <td>±0.2</td> </tr> </table>										±0.1	±0.2																												
±0.1	±0.2																																						

件名
PART NAME

圖號
DWG.

GEARMOTOR

IG220019X00015R

單位
UNITS

比例
SCALE

mm

9/10

祥儀企業股份有限公司

SHA YANG YE INDUSTRIAL CO., LTD.

Technical Data		
Part Number	IG220019X00015R	
Customer P/N		
ITEM	Specifications	Note
1. Operation Status		
1.1 Rated Voltage	12V D.C.	Stable power source 6mm from shaft end
1.2 Rated torque	0.3 kgf.cm	
1.3 Radial load	8N (0.8kg-f)	
1.4 Axial load	6N (0.6kg-f)	
1.5 Turning direction	Shaft horizontal	
1.6 Reverse direction	CW,CCW	
1.7 Using environment	Temperature -10-60 °C Humidity 20-90% RH	
1.8 Preserve environment	Temperature -20-70 °C Humidity 20-90% RH	
1.9 Using voltage range	12V (D.C.) $\pm 10\%$	
2. Electrical Characteristics		
2.1 No Load current	200 mA max.	Motor terminal shell Motor terminal shell Reference Reference Reference
2.2 No Load speed	789 rpm $\pm 15\%$	
2.3 Rated current	620 mA	
2.4 Rated speed	639 rpm $\pm 15\%$	
2.5 Stall current	2.4 A	
2.6 Stall torque	2.2 kgf.cm	
2.7 Insulation	D.C. 100V meg. 1.0 M Ω min	
2.8 Durable voltage	100V (A.C.) * 1 minute min	
2.9 Coil resistance	5 Ω	
2.10 Torque constant	0.91 kgf.cm/A	
2.11 Voltage constant	8.68 mV/r/min	
3. Mechanical characteristic		
3.1 Reduction ratio	1/19.225	By visual judgment
3.2 Thrust play of shaft	0.2 mm max.	
3.3 Radial play of shaft	0.05 mm max.	
3.4 Back lash	3° max.	
3.5 Outside Appearance	No scratch defective....	
2. Life Cycle	72000 cycles min.	After the rated life cycle test current @ rated load must stay within $\pm 30\%$ of the initial value and r.p.m. @ rated load must stay within $\pm 20\%$ of the initial value. However change of mechanical noise level was not considered as part of the testing