

ROYAUME DU MAROC



Ministère de l'Éducation Nationale, de la Formation Professionnelle,
de l'Enseignement Supérieur et de la Recherche Scientifique

C.P.G.E
CLASSES PRÉPARATOIRES
AUX GRANDES ÉCOLES

SPHERO ROBOT

Entre Divertissement et Éducation Technologique

Réalisé par : **CHAATOUF Abdessamad**

CODE CNC : **OJ044T**

Session : **2024**



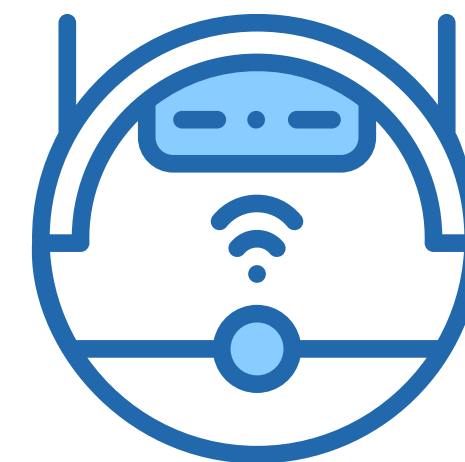
01

Plan de la présentation

- I** *Description globale du système étudié : Sphero robot*
- II** *Conservation du module interne.*
- III** *Dimensionnement et choix du moteur.*
- IV** *Choix des capteurs.*
- V** *Réalisation d'une expérience sur un capteur en vue d'explorer ses fonctionnalités et d'analyser ses performances.*

Problématique

Comment aider les enfants qui ont des difficultés à planifier et à enchaîner des gestes pour réaliser les activités de la vie quotidienne ?

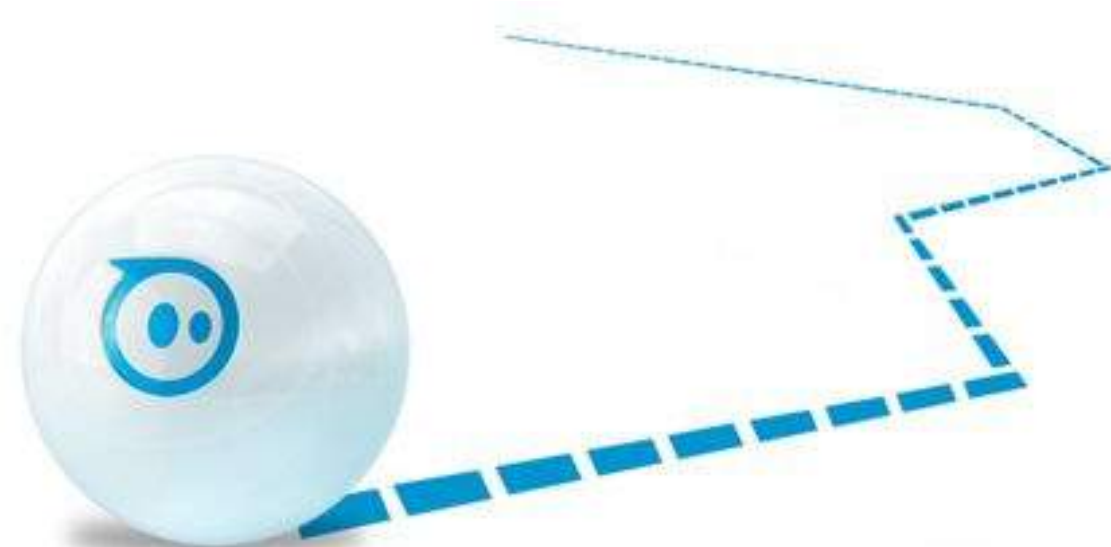


Présentation du robot



I. Introduction

- **Sphero** est un robot sphérique pilotable via une application mobile. Sa programmation simple introduit les enfants à la logique algorithmique.



- Sa conception attractif stimulent la créativité et incitent les enfants à explorer la programmation et la robotique de manière amusante.

I. Introduction

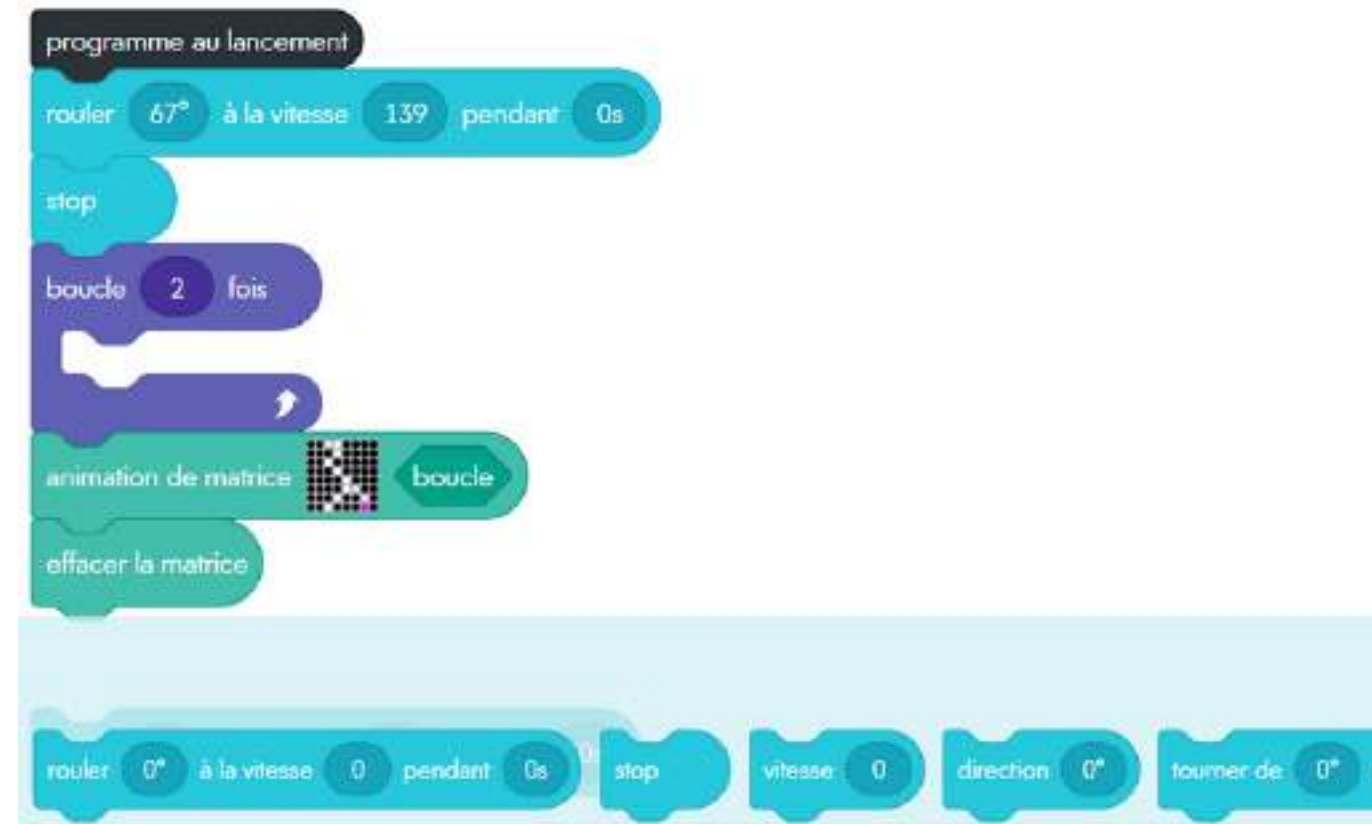
- Pour créer de petits programmes qui contrôlent le robot et qui peuvent également être utilisés par des enfants, on utilise l'un des trois langages interprétés offerts par le Sphero.

Langages du Sphero :

Macros



LAB Lightning



OrbBsic



Figure 1

I. Introduction

LAB Lightning :

- **LAB Lightning**, basé sur Scratch, est idéal pour les enfants grâce à ses commandes simples. Le robot atteint une vitesse de 2.55 m/s.

```
Code JavaScript
1 async function startProgram() {
2   await roll(176, 255, 1);
3   stopRoll();
4   for (var _i0 = 0; _i0 < 2; ++_i0) {
5     await delay(0.025);
6   }
7   playMatrixAnimation(0, true);
8   clearMatrix();
9   await roll(112, 0, 0);
10 }
11
12 async function onCollision() {
13 }
14 registerEvent(EventType.onCollision, onCollision);
15
16 async function blockName() {
17 }
```

Figure 3

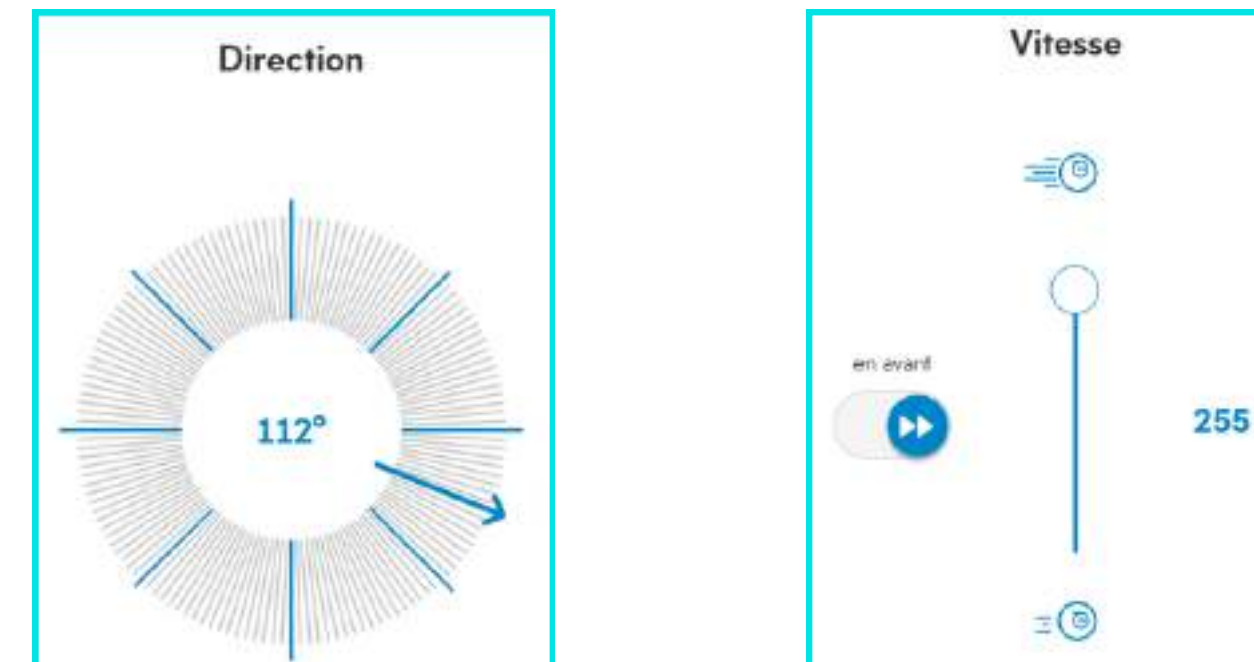


Figure 2

- **LAB Lightning** réside dans sa capacité à permettre aux enfants de transformer leurs programmes en syntaxe JavaScript.

I. Introduction

- Diagramme des cas d'utilisation :

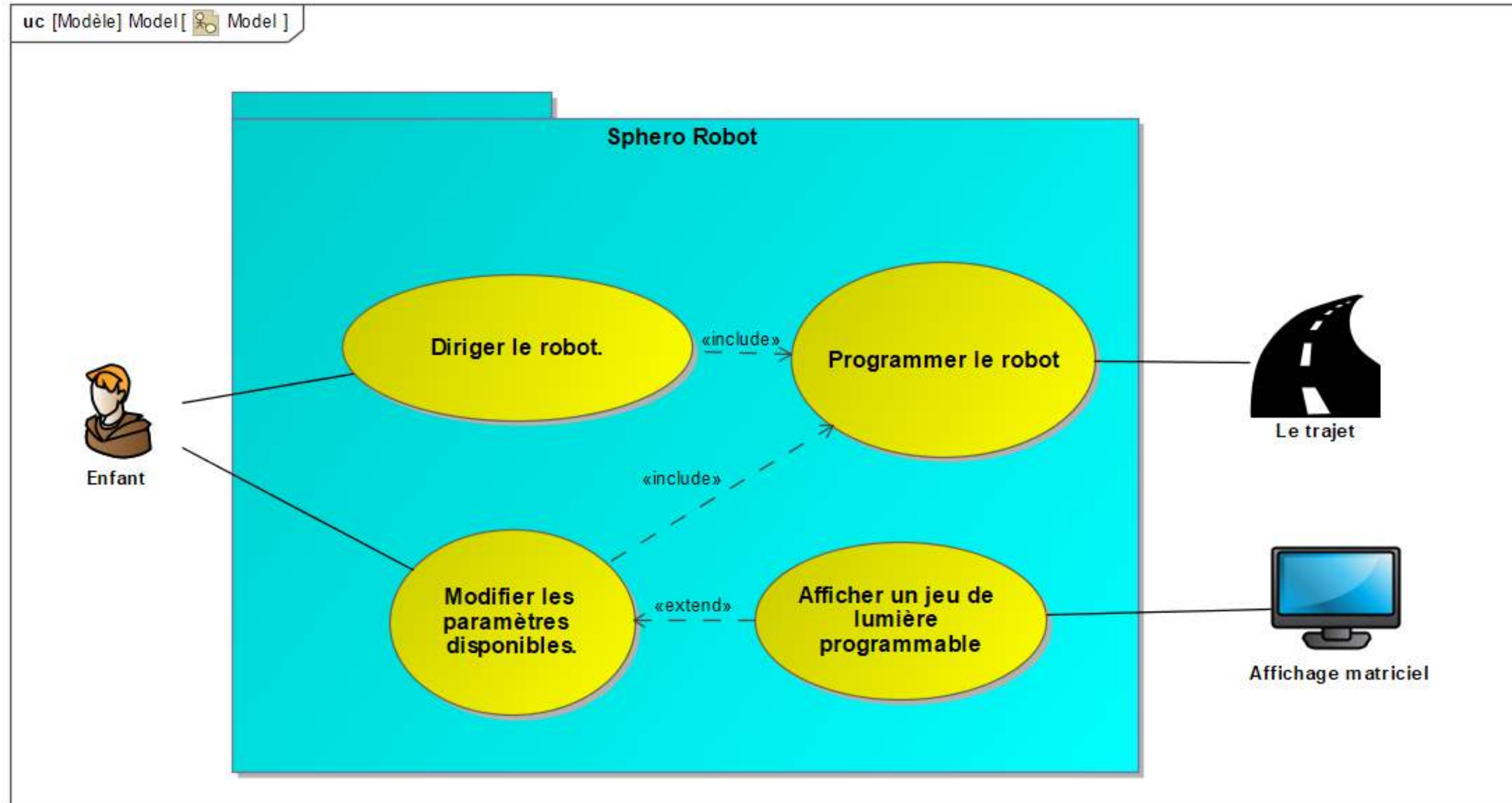


Figure 4

I. Introduction

- Diagramme de définition de blocs :

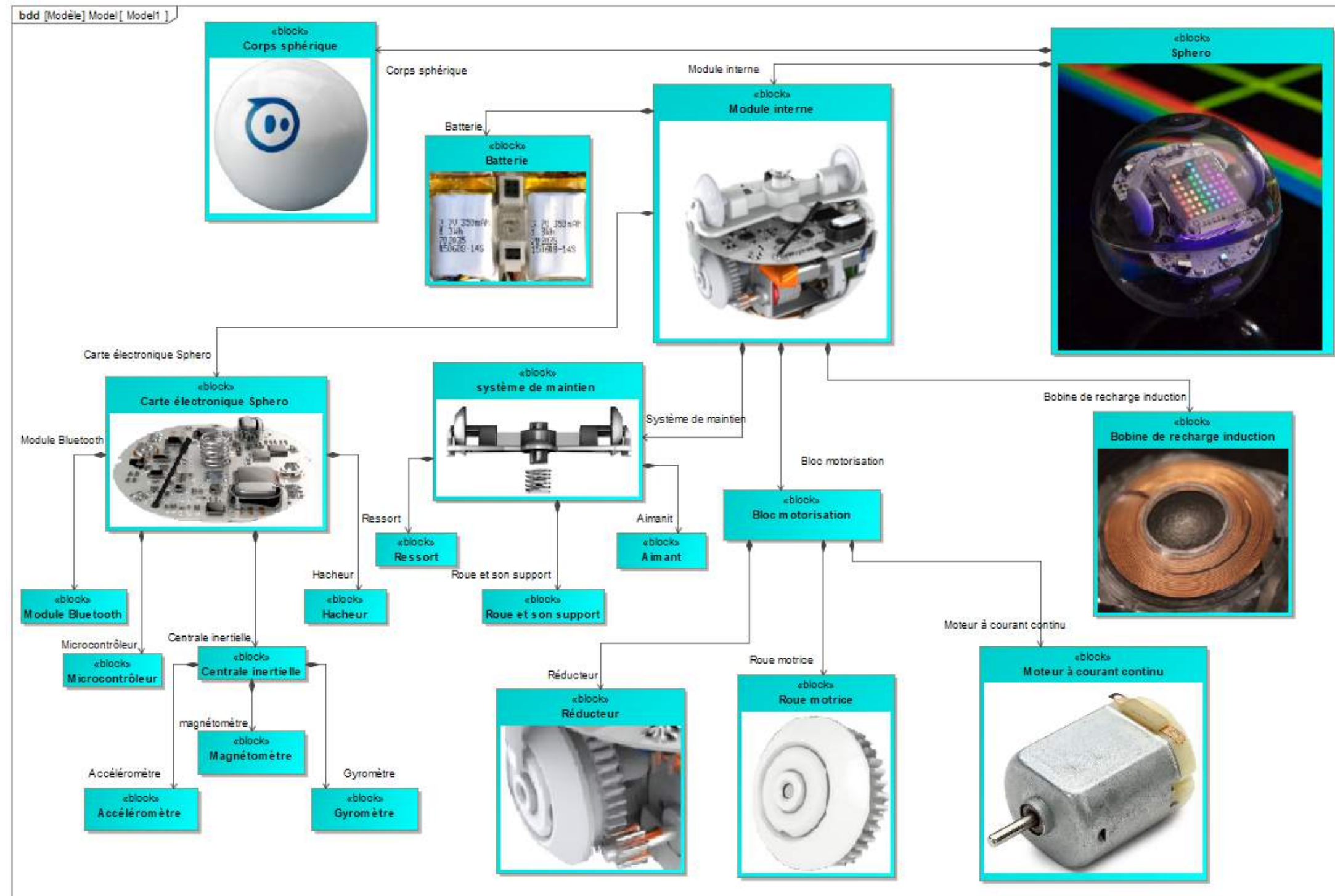


Figure 5

I. Introduction

- Diagramme des exigences :

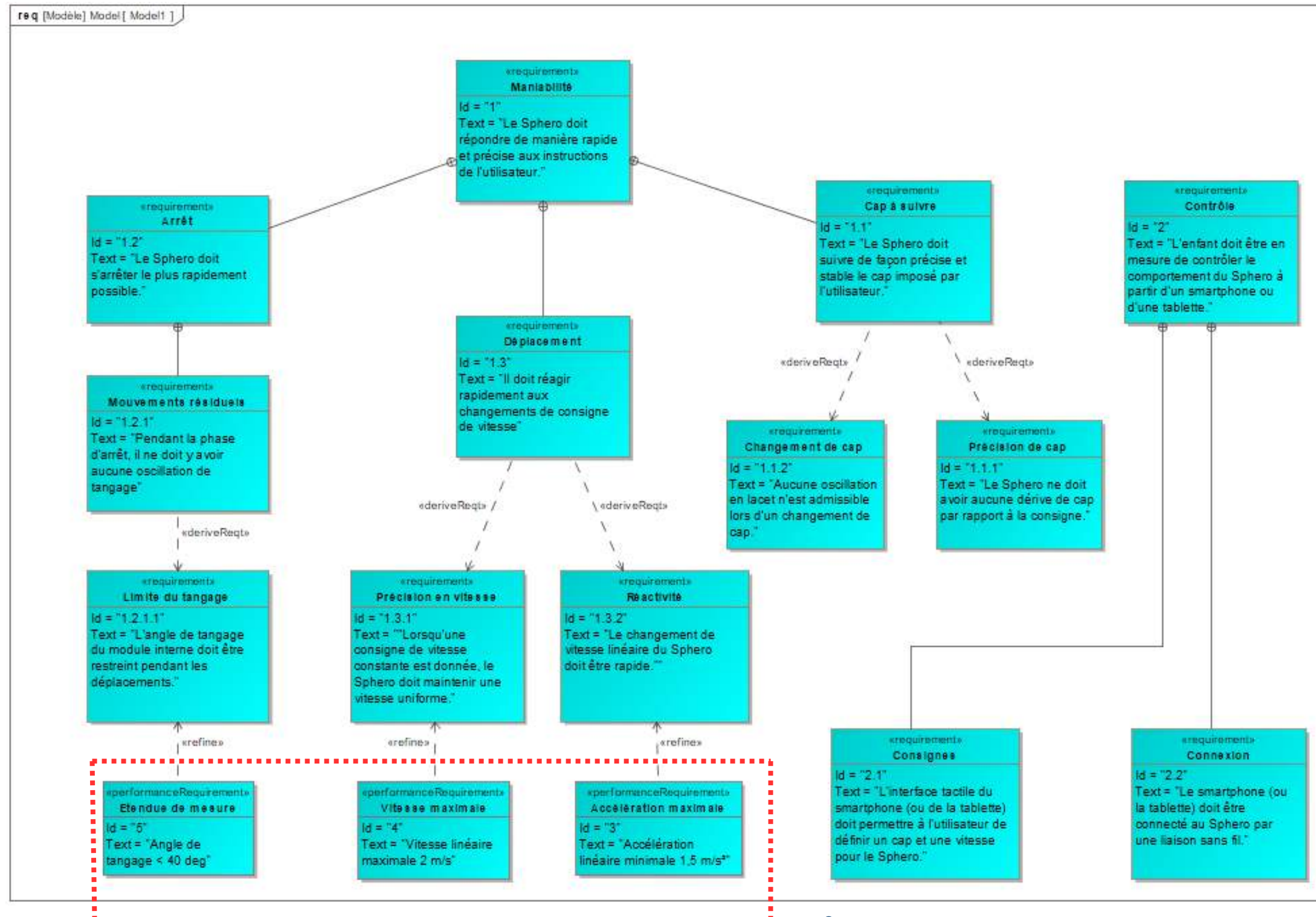
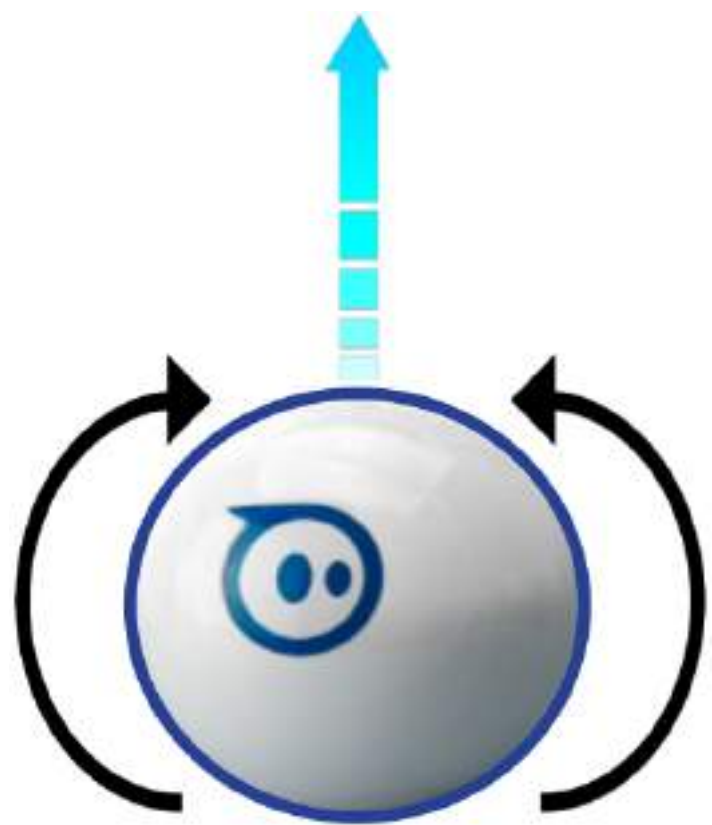


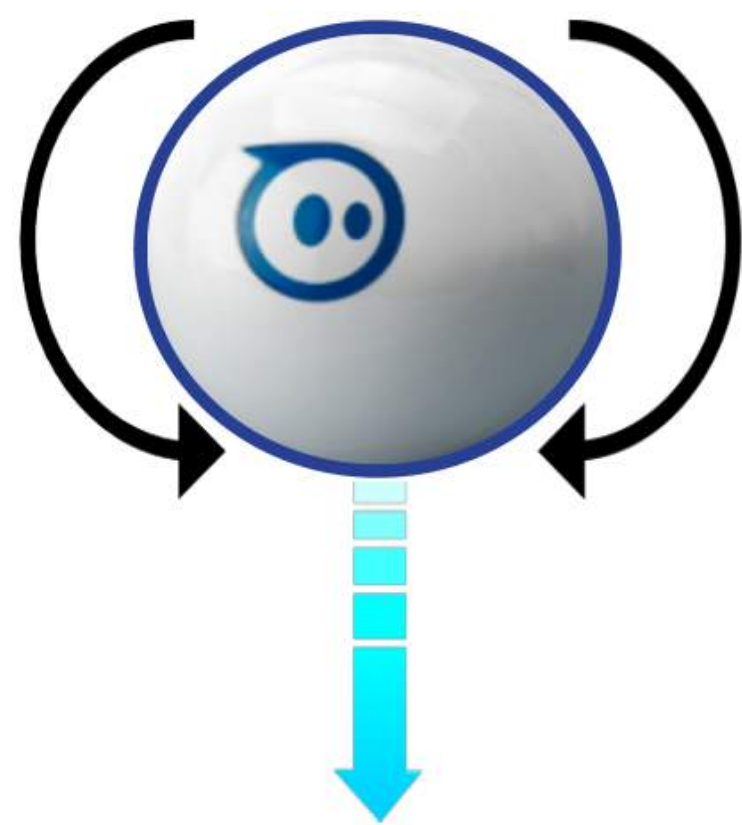
Figure 6

I. Introduction

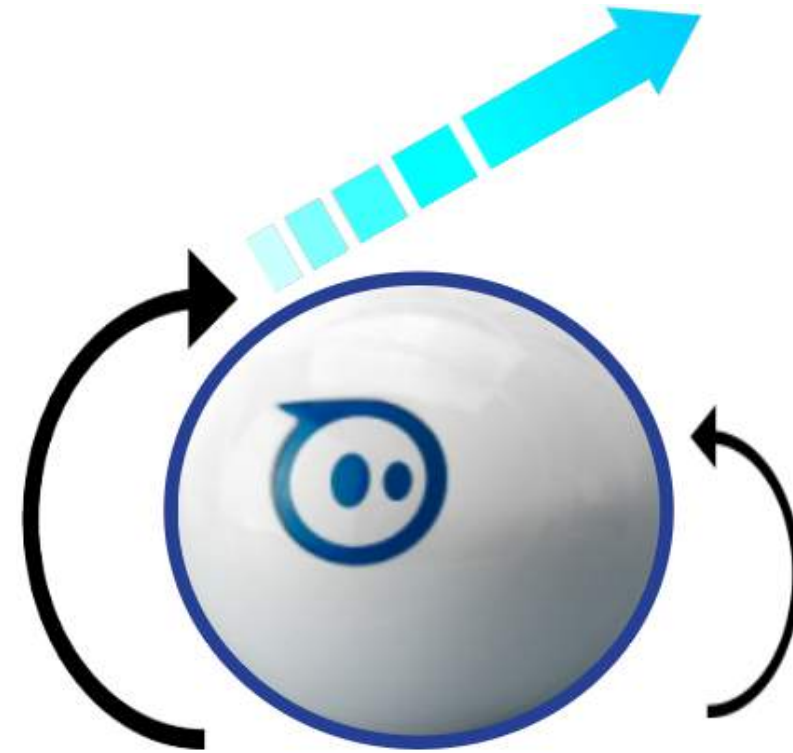
- Les mouvements du robot :



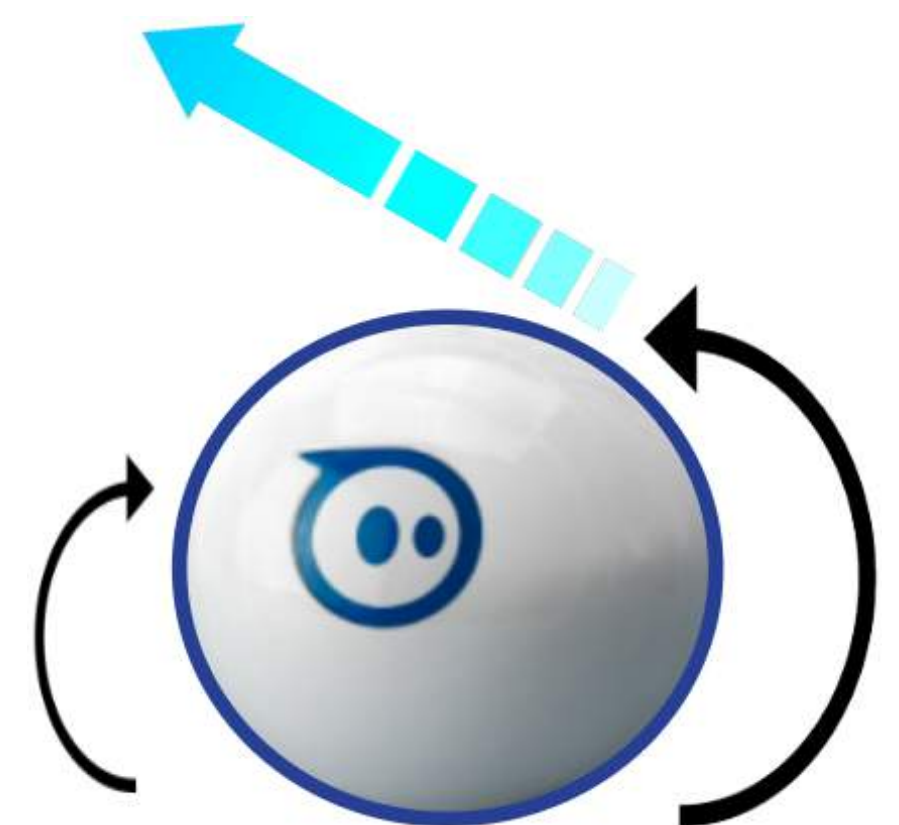
Le robot avance



Le robot recule



Le robot tourne à droite



Le robot tourne à gauche

Figure 7

I. Introduction

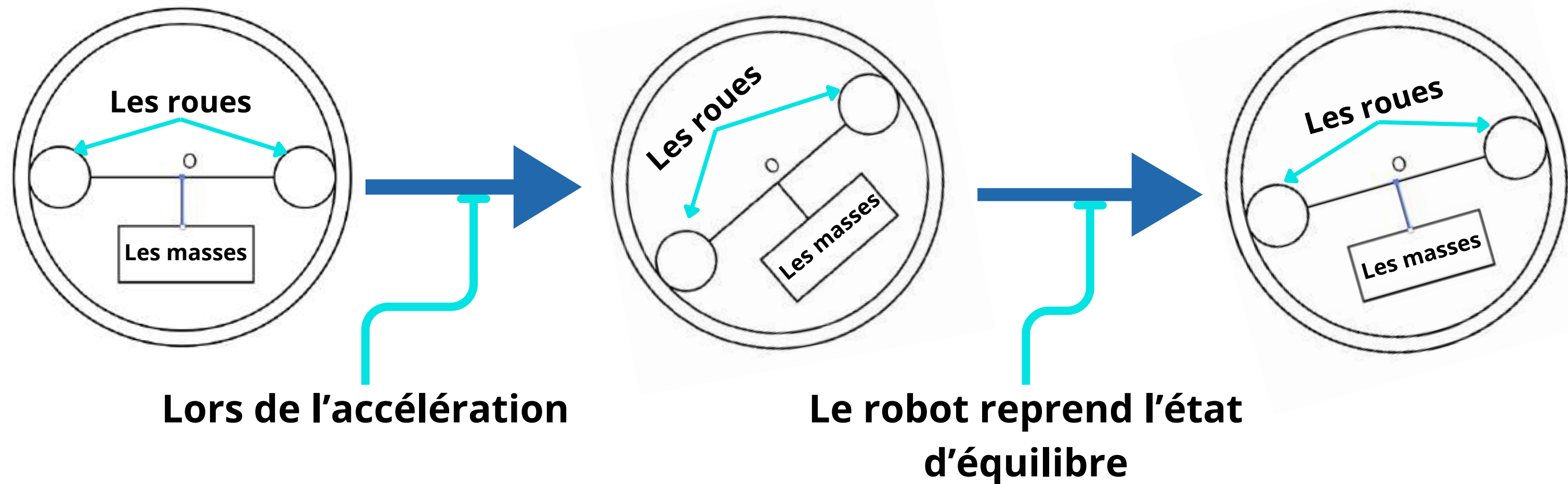


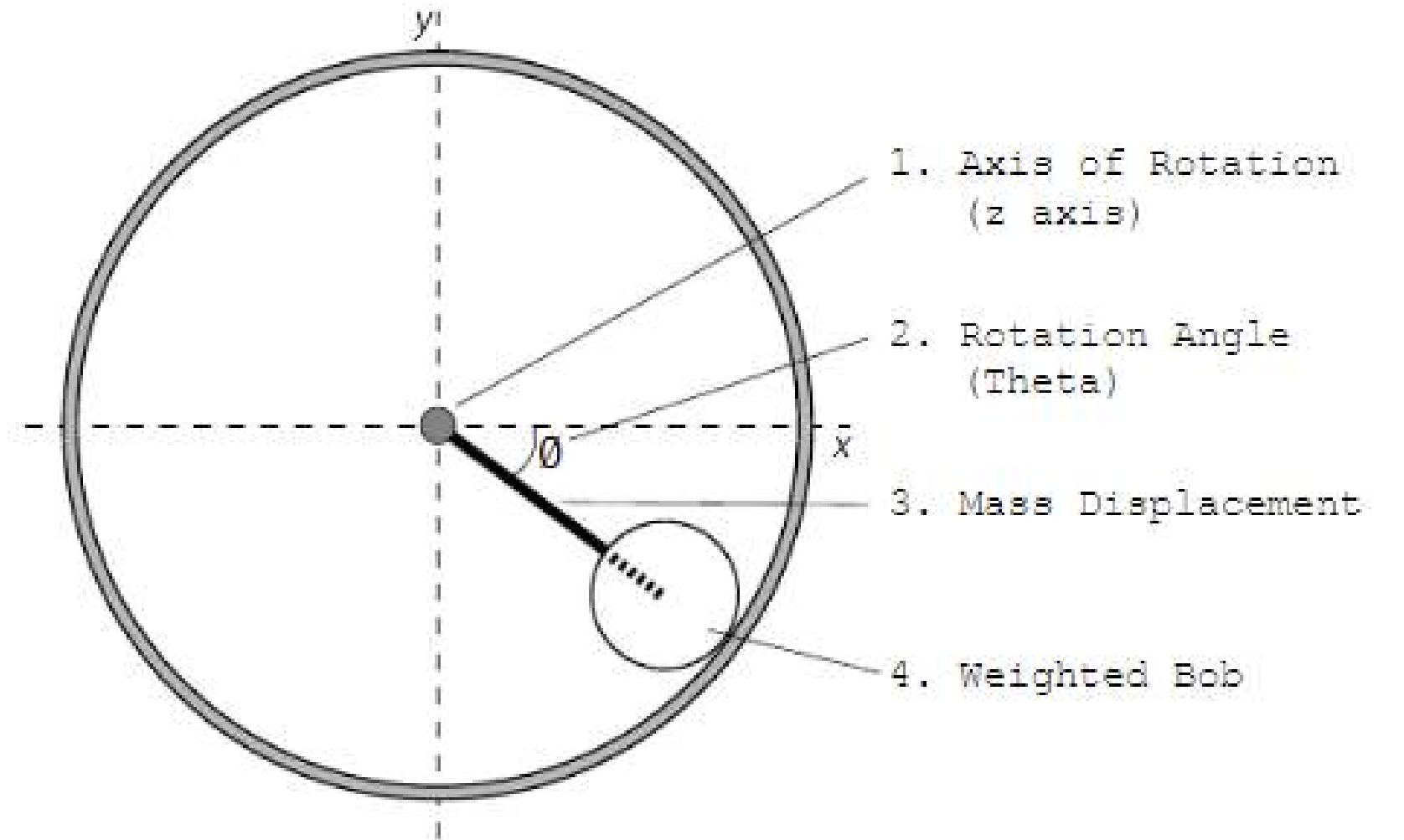
Figure 8

Question :

Comment le robot reprend son état d'équilibre sans aucun système d'asservissement de la position d'équilibre du module interne ?

II

Conservation du module interne



II Conservation du module interne

Solution 1

- La force de pression des roues secondaires

Les roues 234 et 236 en contact avec le corps permettent aux roues 220 et 222 de déplacer le robot en empêchant de tourner en rond à l'intérieur de la corps sphérique.

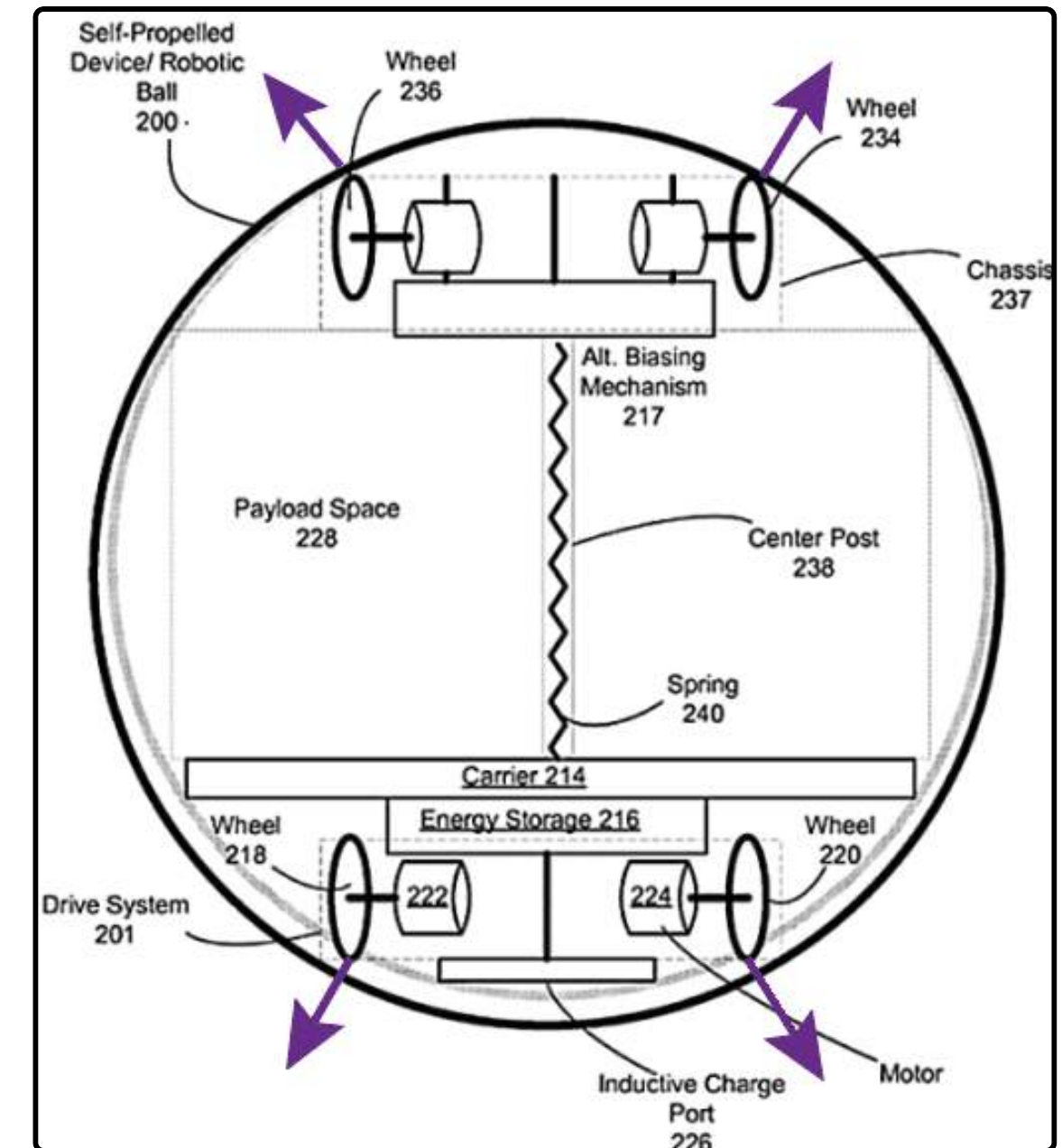


Figure 9.1

II Conservation du module interne

Solution 2

- Centre de masses bas

Sphero minimisent le risque de basculement pendant des mouvements dynamiques résultant en une performance plus fiable.

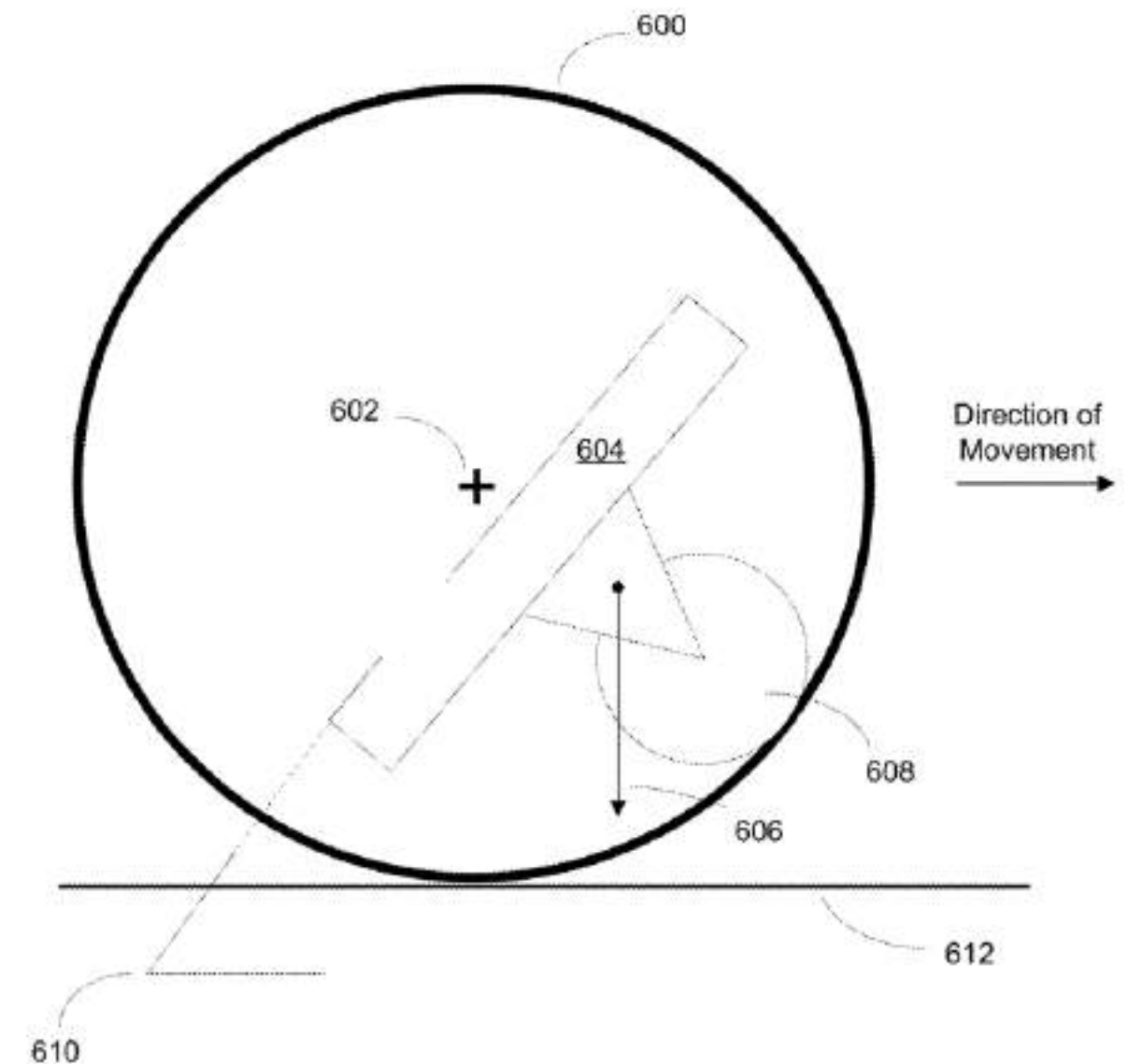


Figure 9.2

II Conservation du module interne

► Positionnement du centre de masse :

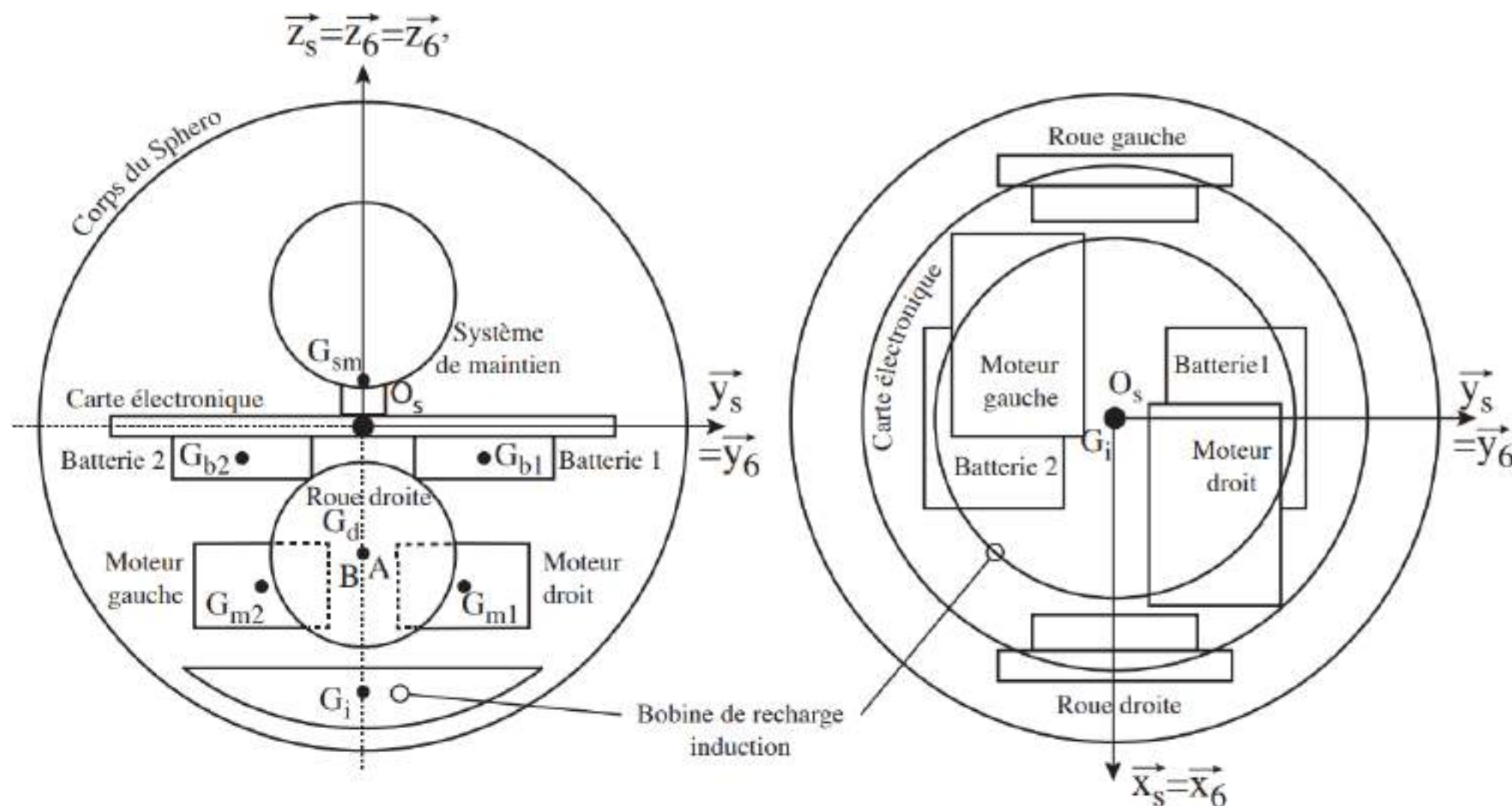


Figure 9.3

La conservation du centre de masse est effectuée à l'aide des masses de la partie inférieure du robot.

$$\begin{aligned}
 & - \overrightarrow{O_s G_{b1}} \cdot \vec{z}_{6'} = \overrightarrow{O_s G_{b2}} \cdot \vec{z}_{6'} = -h_b ; \\
 & - \overrightarrow{O_s G_{m1}} \cdot \vec{z}_{6'} = \overrightarrow{O_s G_{m2}} \cdot \vec{z}_{6'} = -h_m ; \\
 & - \overrightarrow{O_s A} \cdot \vec{z}_{6'} = \overrightarrow{O_s B} \cdot \vec{z}_{6'} = -h_r ; \\
 & - \overrightarrow{O_s G_{sm}} \cdot \vec{z}_{6'} = h_s ; \\
 & - \overrightarrow{O_s G_i} \cdot \vec{z}_{6'} = -h_i .
 \end{aligned}$$

II Conservation du module interne

► Positionnement du centre de masse :

Désignation	Position	Masse
Carte électronique	en O_s	$m_c = 12 \text{ g}$
Batterie 1	$h_b = 9 \text{ mm}$	$m_b = 7 \text{ g}$
Batterie 2	$h_b = 9 \text{ mm}$	$m_b = 7 \text{ g}$
Bobine de recharge	$h_i = 35 \text{ mm}$	$m_i = 27 \text{ g}$
Système de maintien	$h_s = 9 \text{ mm}$	$m_s = 8 \text{ g}$

Désignation	Position	Masse
Roue droite	$h_r = 15 \text{ mm}$	$m_r = 1 \text{ g}$
Roue gauche	$h_r = 15 \text{ mm}$	$m_r = 1 \text{ g}$
Moteur droit	$h_m = 20 \text{ mm}$	$m = 26 \text{ g}$
Moteur gauche	$h_m = 20 \text{ mm}$	$m = 26 \text{ g}$

Centre de masse d'ensemble de solides :

$$\overrightarrow{O_s G} = \frac{\sum m_k \overrightarrow{O_s G_k}}{\sum m_k}$$

$$Z_G = \frac{(m_c \vec{0} + m_{b1} \overrightarrow{O_s G_{b1}} + m_{b2} \overrightarrow{O_s G_{b2}} + m_i \overrightarrow{O_s G_i} + m_s \overrightarrow{O_s G_{sm}} + m_r \overrightarrow{O_s A} + m_r \overrightarrow{O_s B} + m \overrightarrow{O_s G_{m1}} + m \overrightarrow{O_s G_{m2}}) \cdot \vec{z}_{G'}}{2m + 2m_r + m_s + m_c + m_{b1} + m_{b2} + m_i}$$

$$Z_G = \frac{-2069}{115}$$



$$Z_G = -17,99 \text{ mm}$$

II Conservation du module interne

► Positionnement du centre de masse :

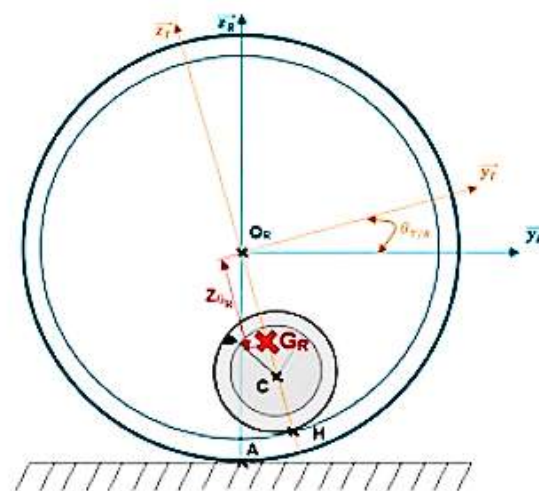
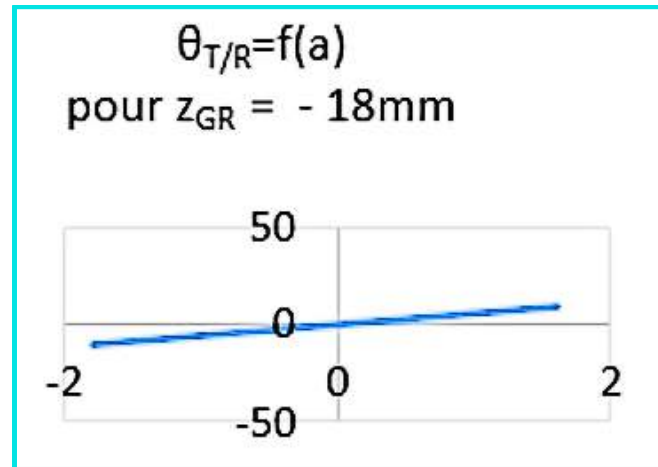


Figure 9.6

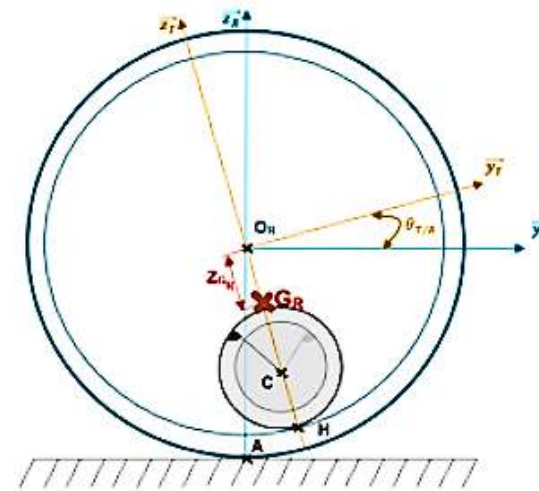
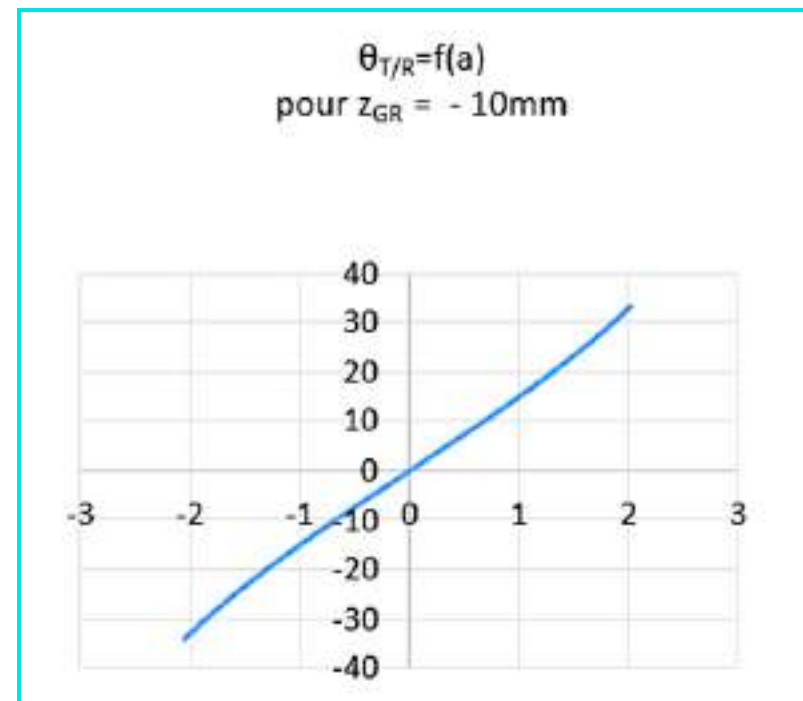


Figure 9.5

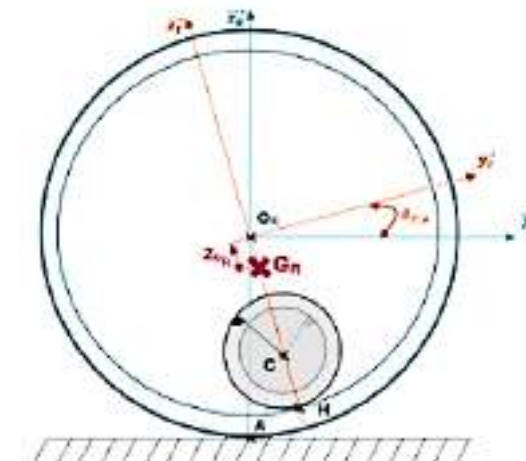
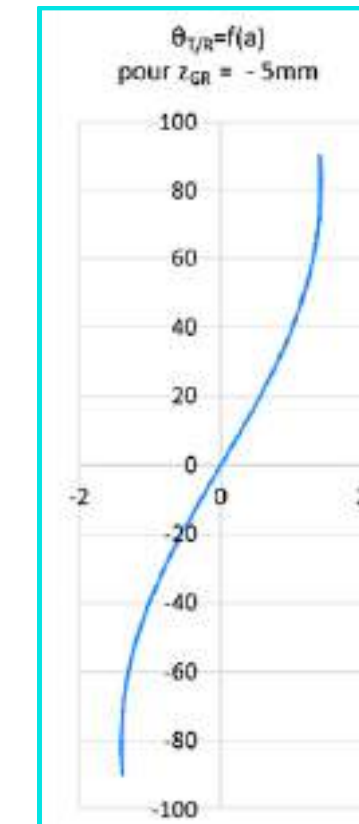


Figure 9.4

**Quelle est
l'angle
maximale peut
atteint le robot
lors de
roulement?**



II Conservation du module interne

► Angle maximale de roulement

- Les hypothèses :

Mouvement de roulement sans glissement doit adoptée au point I (point de contact 2/1), au point J (point de contact 4/1) et au point K (point de contact 1/0).

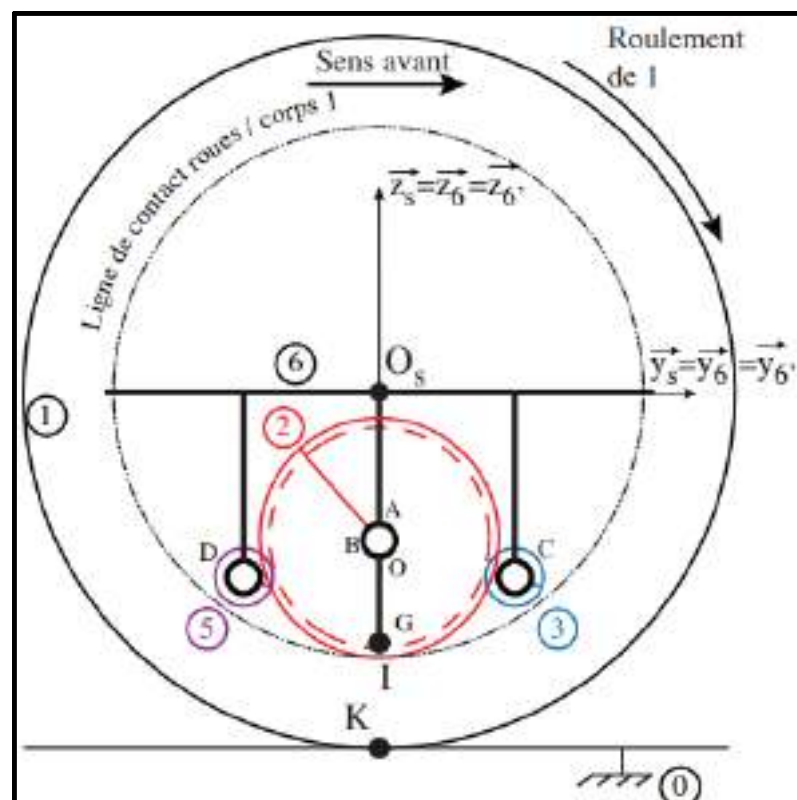


Figure 9.8

- Bilan des efforts :

- Pesenteur :

$$\{\tau_{pes \rightarrow s}\} = \begin{Bmatrix} -Mgz_s \\ \vec{0} \end{Bmatrix}_G$$

- Liaison sphère\plan en K :

$$\{\tau_{0 \rightarrow 1}\} = \begin{Bmatrix} T_{01}\vec{y}_s + N_{01}\vec{z}_s \\ \vec{0} \end{Bmatrix}_K$$

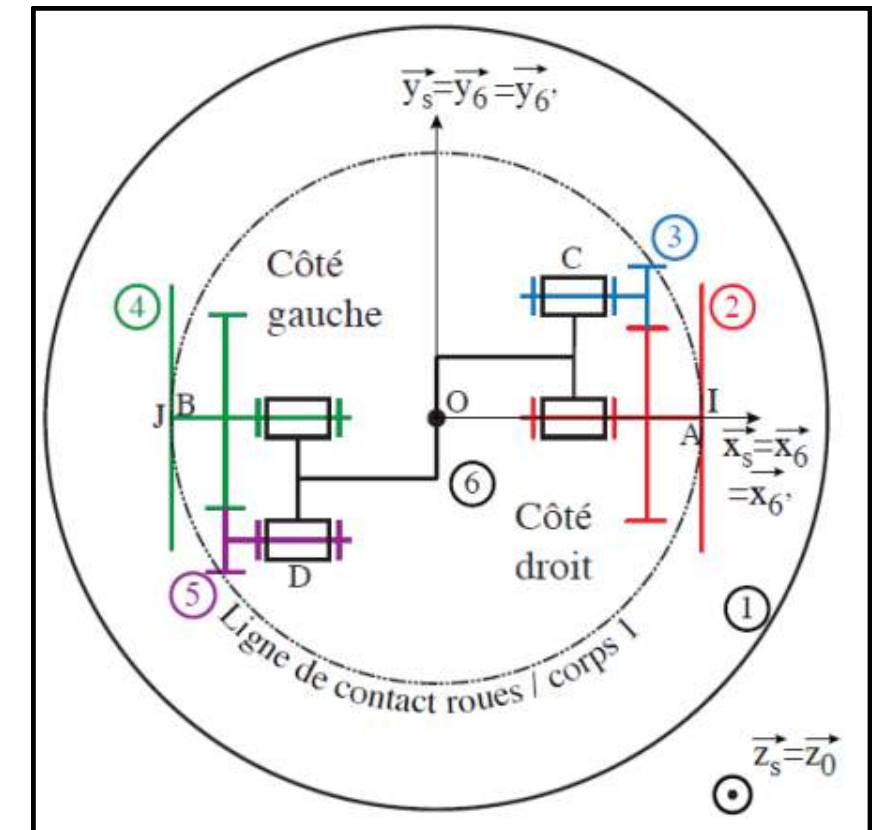


Figure 9.7

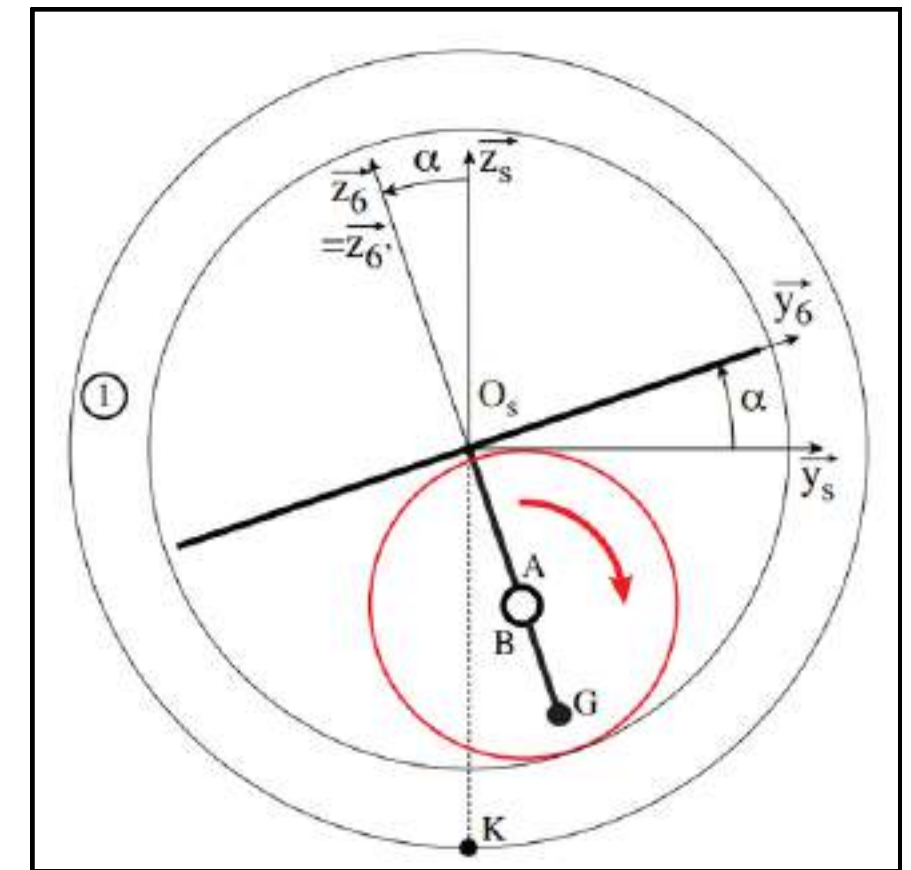
II Conservation du module interne

► Angle maximale de roulement

En considérant un **mouvement rectiligne uniformément accéléré** du cadre du module interne tel que α est un angle constant de tangage de ce dernier.

$$\begin{aligned}
 \overrightarrow{\delta_{K,S/0}} &= \overrightarrow{\delta_{K,1/0}} + \overrightarrow{\delta_{K,2/0}} + \overrightarrow{\delta_{K,3/0}} + \overrightarrow{\delta_{K,4/0}} + \overrightarrow{\delta_{K,5/0}} + \overrightarrow{\delta_{K,6/0}} \\
 &= \overrightarrow{\delta_{G,6/0}} + \overrightarrow{KG} \wedge M \vec{\Gamma}_{G,6/0} \\
 &= \frac{d}{dt} [\vec{\sigma}_{G,6/0}]_0 + (R_s \vec{z}_G + z_G \vec{z}_6) \wedge Ma \vec{y}_s \\
 &= \frac{d}{dt} [\vec{\Omega}_{s/0} I(G,S)]_0 + (R_s \vec{z}_G + z_G \vec{z}_6) \wedge Ma \vec{y}_s
 \end{aligned}$$

$$\overrightarrow{\delta_{K,S/0}} = -Ma(R_s + Z_G \cos(\alpha)) \vec{x}_s$$



**Vue de profil du Sphero avec
 $\alpha \neq 0$
Figure 9.9**

II Conservation du module interne

► Angle maximale de roulement

$$\overrightarrow{\delta_{K,S/0}} = \overline{KG} \wedge -Mg\overline{z}_S$$

$$\overrightarrow{\delta_{K,S/0}} = Mgz_G \sin(\alpha) \overline{x}_S$$

$$a = \frac{gz_G \sin(\alpha)}{R_S + z_G \cos(\alpha)}$$

- La résolution de L'équation:

$$1.5 = \frac{0.17658 \cdot \sin(x)}{74 \cdot 10^{-3} - 18 \cdot 10^{-3} \cdot \cos(x)}$$

Résultat

Voici l'équation demandée : $1.5 = \frac{0.17658 \sin(x)}{\frac{37}{500} - \frac{9 \cos(x)}{500}}$

Figure 9.10

D'où :

$$\alpha = 29,72^\circ$$

Lorsque le robot atteindra l'accélération minimale $a = 1.5m/s^{-2}$, le robot tangue presque 30 degré.



Dimensionnement et choix du moteur



III. Choix du moteur

1. Caractérisation du robot

- La masse (Kg) : 200g
- La vitesse (m/s): 2 m/s
- La distance parcourue : pas nécessaire
- Le diamètre des roues : 16 mm
- La pente maximale : $\alpha=8^\circ$

3. Bilan de travail des forces



$$W(P) = -m \cdot g \cdot \sin(\alpha) \cdot d$$



$$W(F) = F \cdot d$$



$$W(R) = 0$$

2. projections des forces

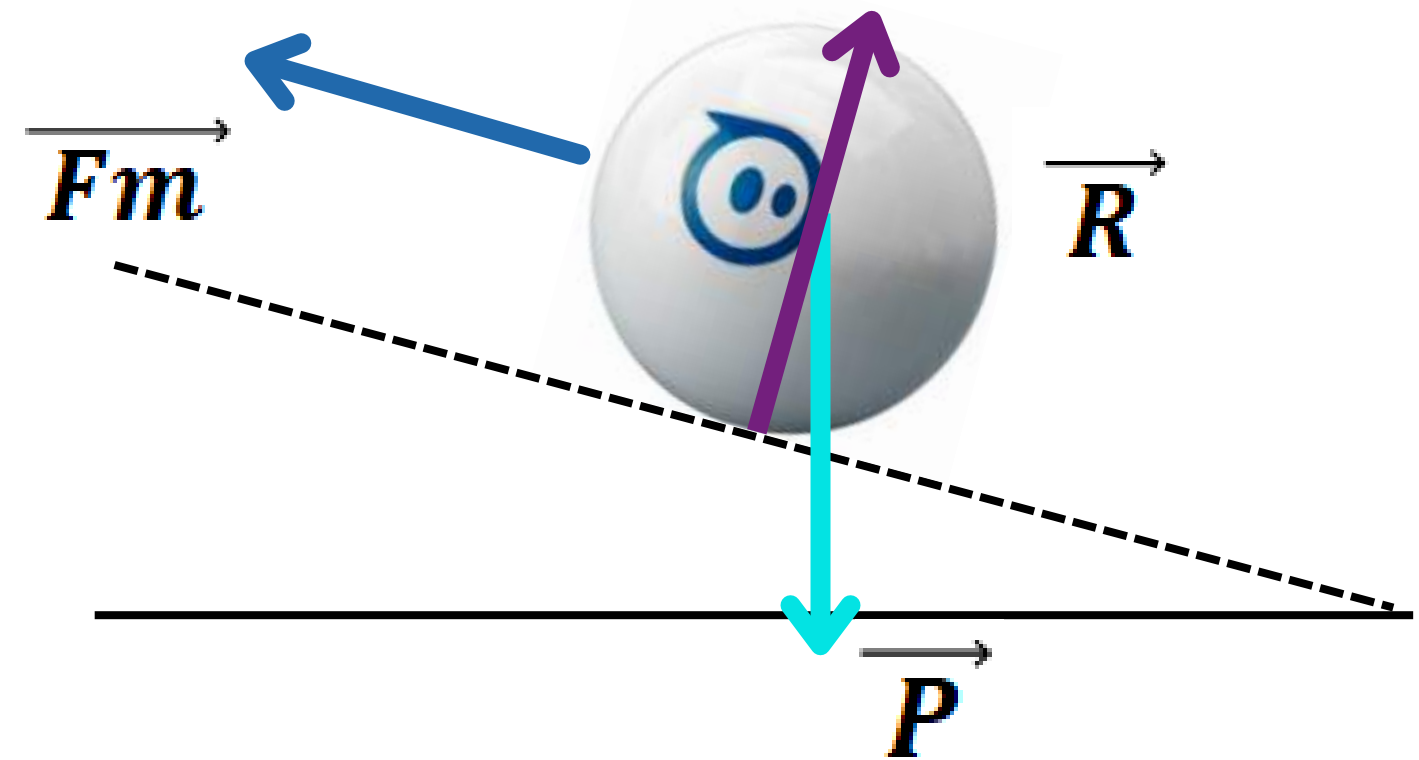


Figure 10.2

4. Théorème de l'énergie cinétique

$$\Delta E_c = \sum W_{F_{EX}} \quad E_c(t) = \frac{1}{2} m v(t)^2$$

III. Choix du moteur

5. Caractérisation du robot

$$\Delta Ec = \sum W_{FEX}$$

$$\frac{1}{2} m (vf^2 - vo^2) = Fd - m.g.\sin(\alpha).d = 0$$

Donc la force moteur : $F = m.g.\sin(\alpha)$

6. Le couple à l'arbre de roue

$$Cr = F \cdot \frac{D}{2}$$

$$Cr = 2.22 \text{ mNm}$$

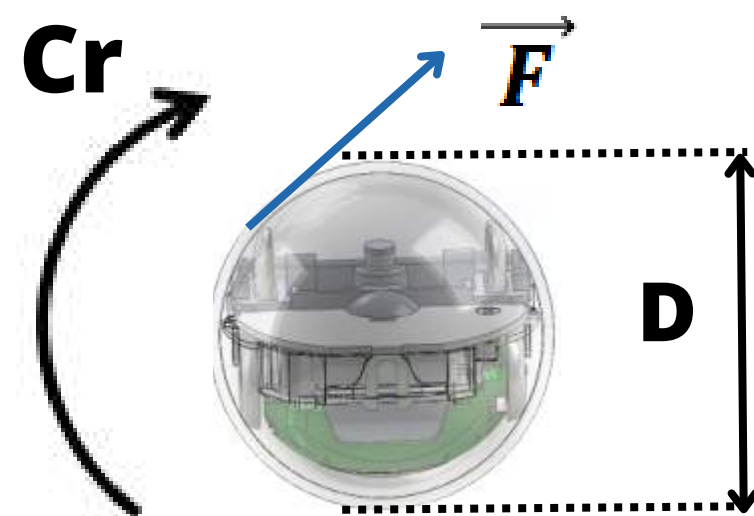
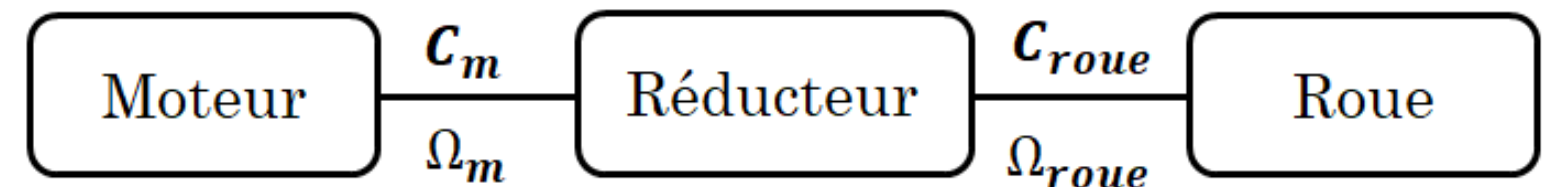


Figure 10.4

7. Chaîne cinématique



Vitesse à l'arbre de roue :

$$v = 2 \text{ m/s} \Rightarrow \Omega_r = 2v/D = 250 \text{ rad/s}$$

$$\Rightarrow Nr = 42 \text{ tr/min}$$

Vitesse à l'arbre moteur :

Les vitesse plus rencontrées au marché :

$$Nm = 5000 \text{ tr/min}$$

Le rapport de réduction :

$$r = \frac{Nm}{Nr} \cong 119$$

$$\text{choix : } r = 120$$

III. Choix du moteur

8- Le couple moteur

$$C_m = \frac{Cr}{r} \approx \mathbf{0.02 \text{ mNm}}$$

9- Le moteur choisi

Nominal voltage	V	3	4.5	→ 6
No load speed	rpm	13900	13900	15600
No load current	mA	51.5	34.7	31.1
Nominal speed	rpm	3950	3620	→ 5240
Nominal torque	mNm	0.734	0.702	→ 0.684
Nominal current	A	0.418	0.27	0.222
Stall torque ¹	mNm	1.09	1.01	1.1
Stall current	A	0.584	0.364	0.332
Max. efficiency	%	51.9	50.4	50.9
Characteristics				
Terminal resistance phase to phase	Ω	5.14	12.4	18.1
Terminal inductance phase to phase	mH	0.127	0.281	0.4
Torque constant	mNm/A	1.86	2.77	3.3
Speed constant	rpm/V	5120	3450	2890
Speed/torque gradient	rpm/mNm	14100	15400	15800
Mechanical time constant	ms	35.1	38.2	39.3
Rotor inertia	gcm ²	0.237	0.237	0.237

On choisit deux moteurs de type
Brushless MAXON EC 9.2 FLAT 624163 :



Figure 10.5

IV

Choix des capteurs



IV. Choix des capteurs

- **Accéléromètre:**

- **L'accélération** résultante tente de déplacer le corps d'épreuve. La force nécessaire pour maintenir le corps en place fournie par le ressort est utilisée pour la lecture accélérométrique.

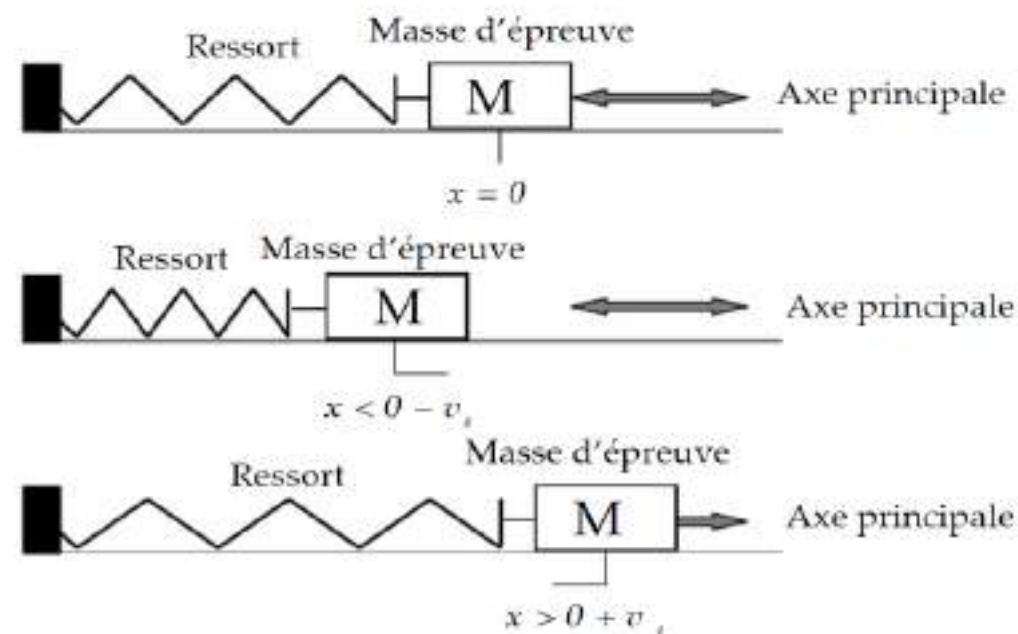


Figure 11.2

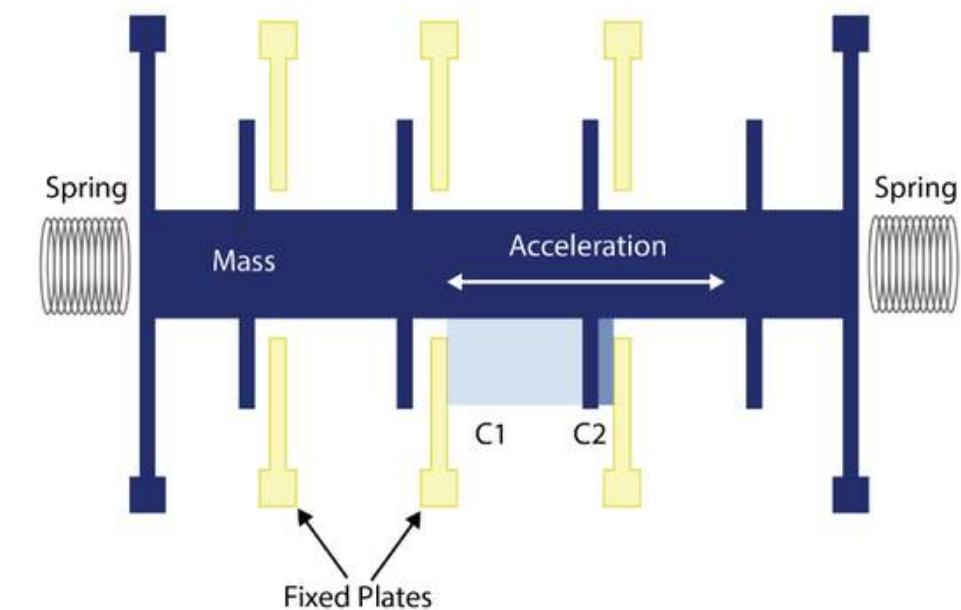
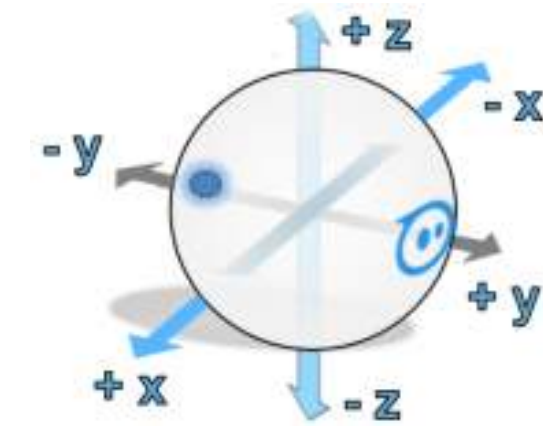


Figure 11.1

- **Les accéléromètres** reposent sur la deuxième loi de Newton. L'interaction entre la force et l'accélération est réciproque. on peut déduire l'accélération comme : $a = -kx/m$.

IV. Choix des capteurs

- **Gyromètre:**

- **Le gyromètre** est un capteur utilisé pour mesurer l'orientation angulaire ou la vitesse angulaire du robot. Il joue un rôle crucial dans la détermination des mouvements d'objets.



Figure 11.4

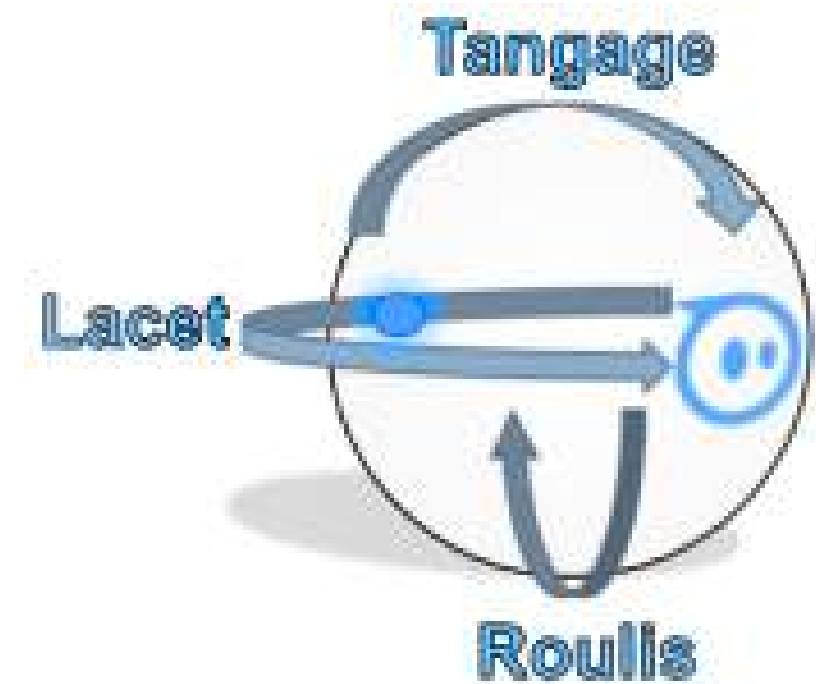


Figure 11.3

- La stabilité du Sphero robot est généralement assurée par une combinaison de capteurs y compris le gyromètre et des algorithmes de contrôle sophistiqués.

IV. Choix des capteurs

- **Magnétomètre:**

- Le champ magnétique terrestre ambiant est présent en permanence, et ce champ magnétique peut être influencé par des objets magnétiques dans l'environnement.

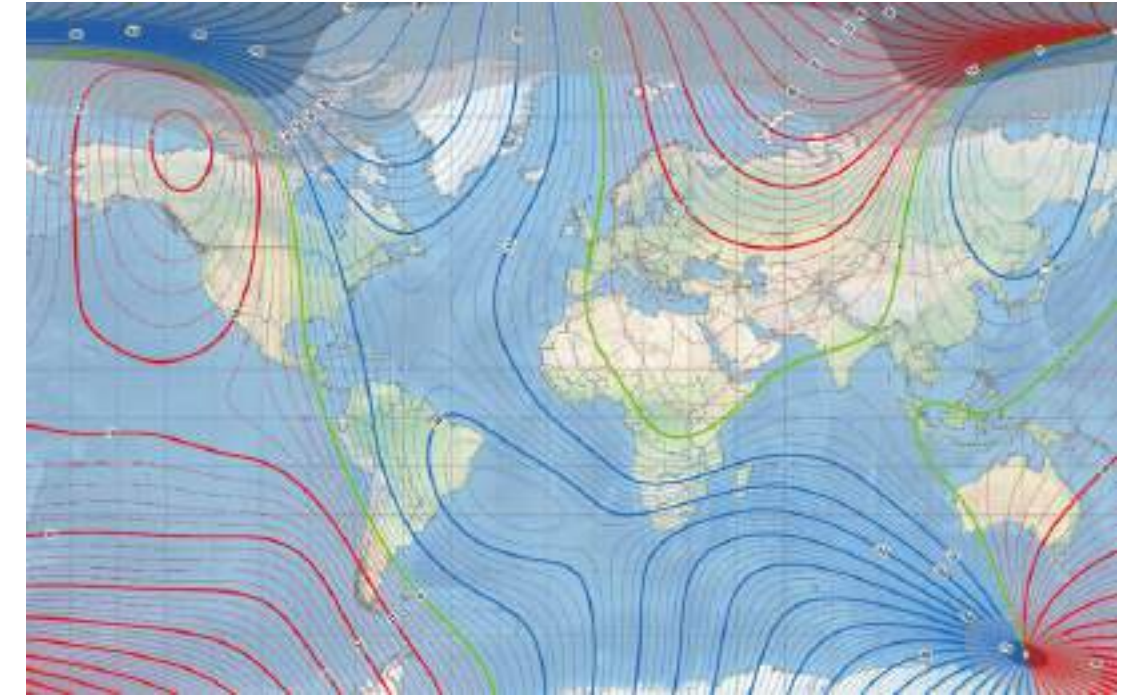


Figure 11.5

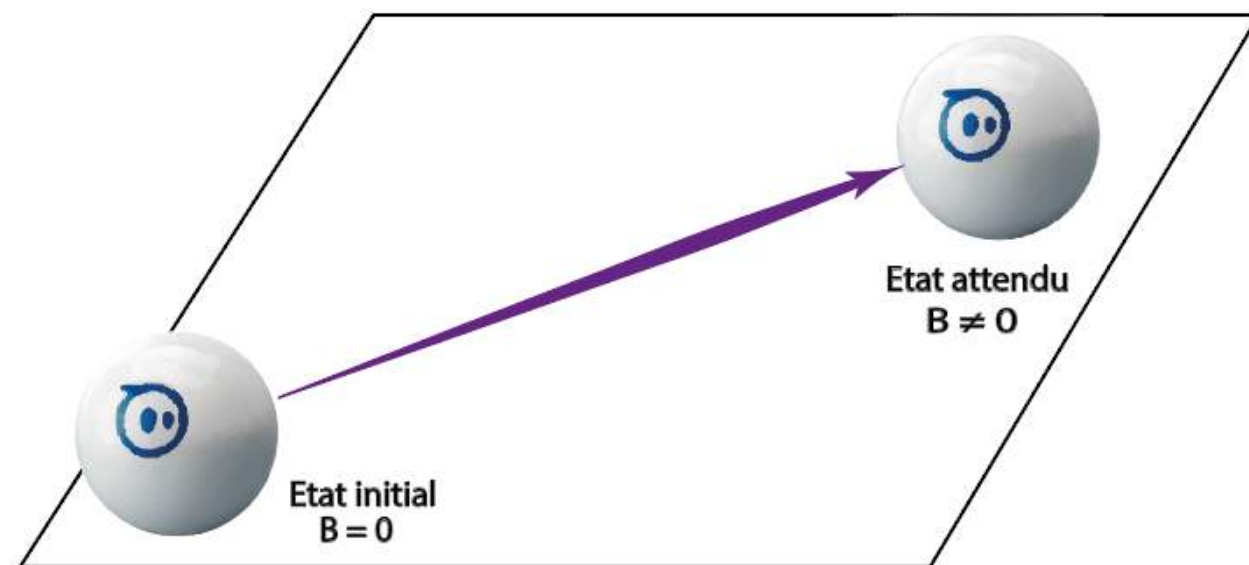


Figure 11.6

- **Le magnétomètre** mesure le champ magnétique autour du robot. Son rôle est de détecter l'orientation magnétique par rapport au champ magnétique terrestre .

IV. Choix des capteurs



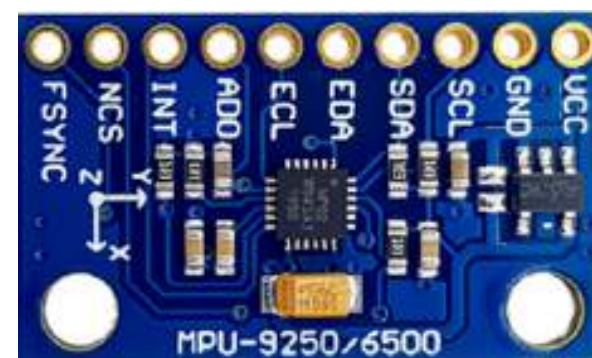
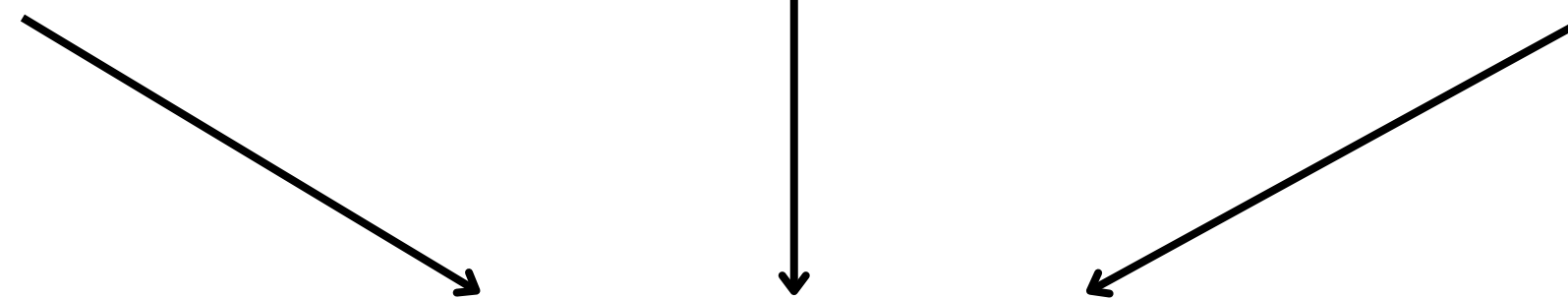
Accéléromètre



Magnétomètre



Gyromètre



Capteur MPU9250

Figure 11.7

intéressé !!



IV. Choix des capteurs

- **IMU (Unité de Mesure Inertielle) :**
 - **IMU** est un dispositif qui combine plusieurs capteurs pour rapporter des informations sur l'accélération, la vitesse angulaire et le champ magnétique environnant.

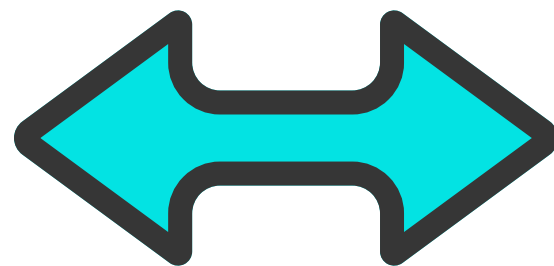


Figure 11.9

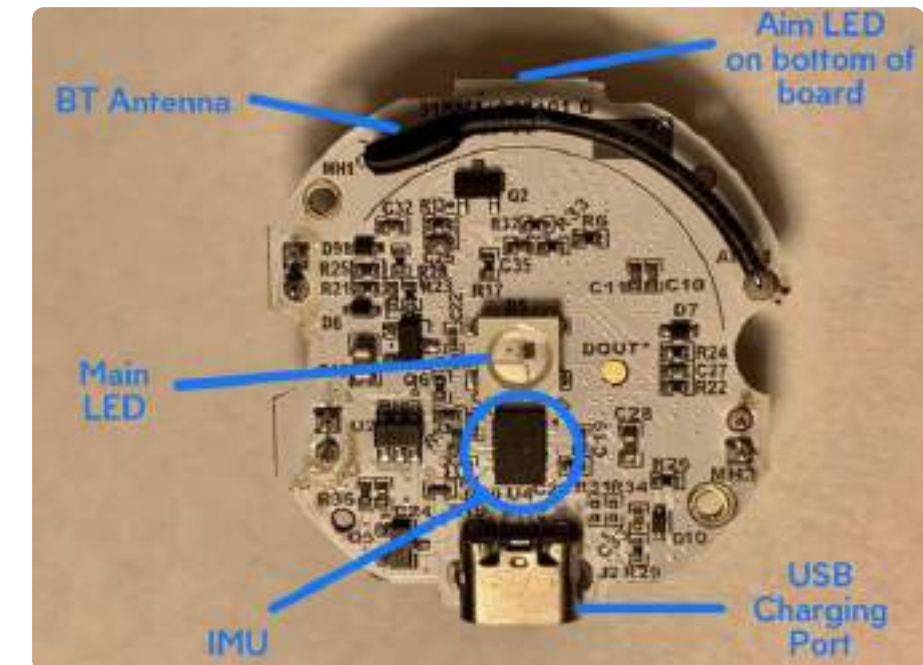


Figure 11.8

- **Le MEMS** est un petit dispositif qui combine des composants mécaniques, électriques et magnétiques sur une seule puce.

IV. Choix des capteurs

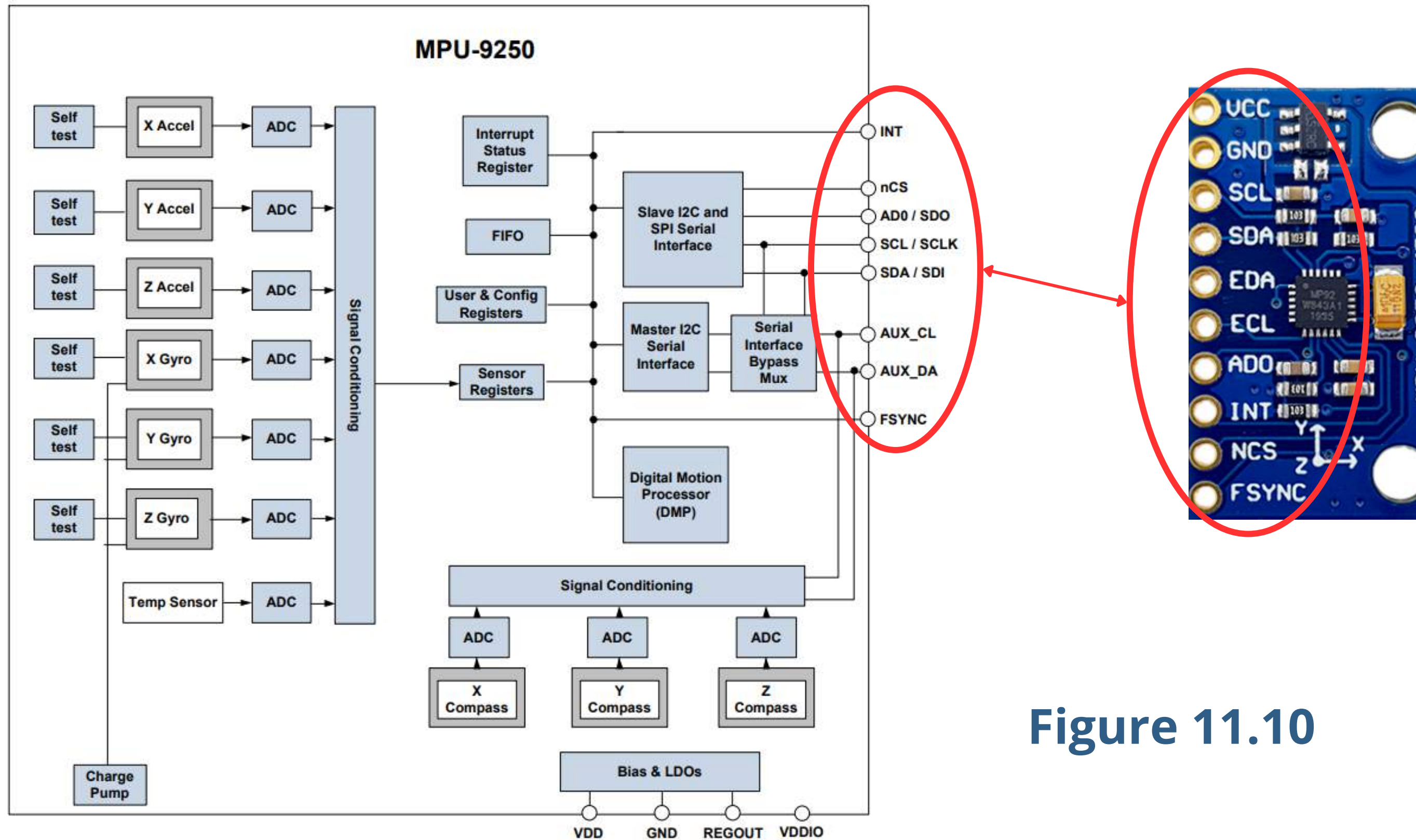
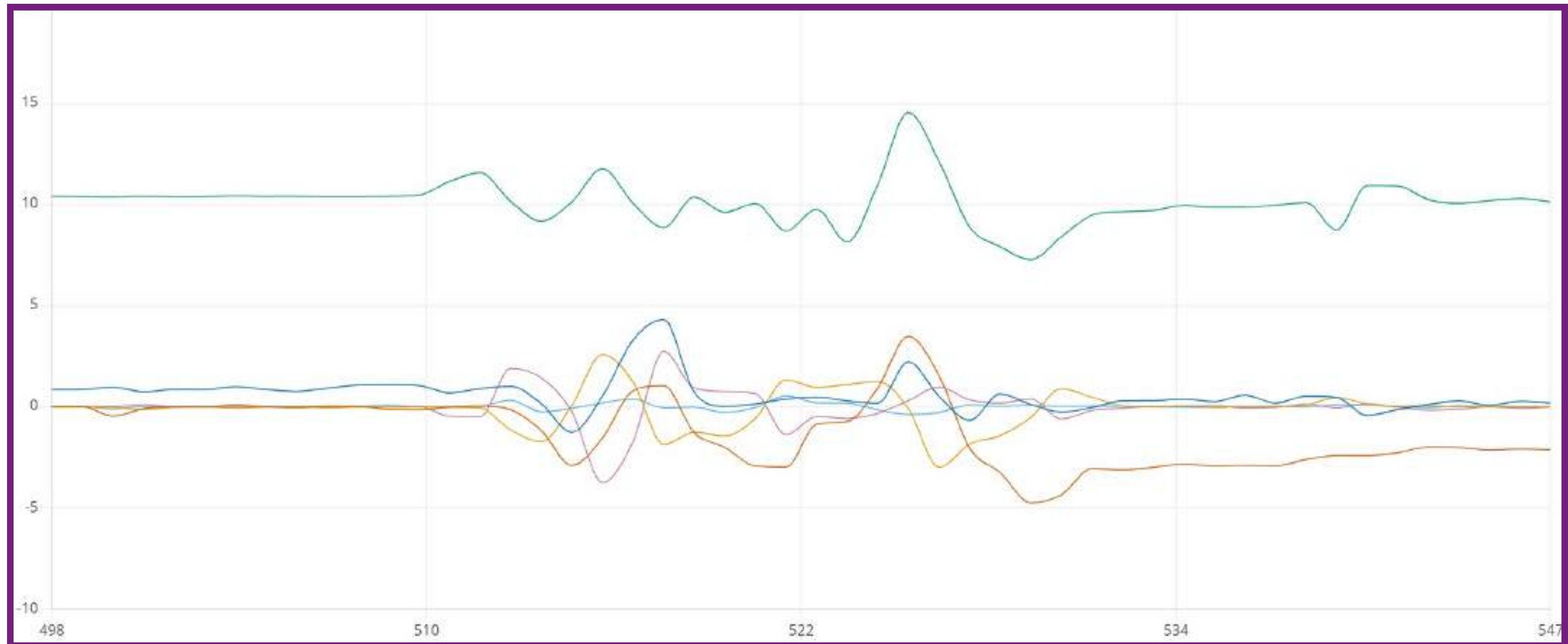


Figure 11.10



Réalisation des expériences sur MPU6050 en vue d'explorer ses fonctionnalités et d'analyser ses performances



V. Experience

- Translation suivant l'axe \overrightarrow{OX}
- Translation suivant l'axe \overrightarrow{OY}
- Translation suivant l'axe \overrightarrow{OZ}

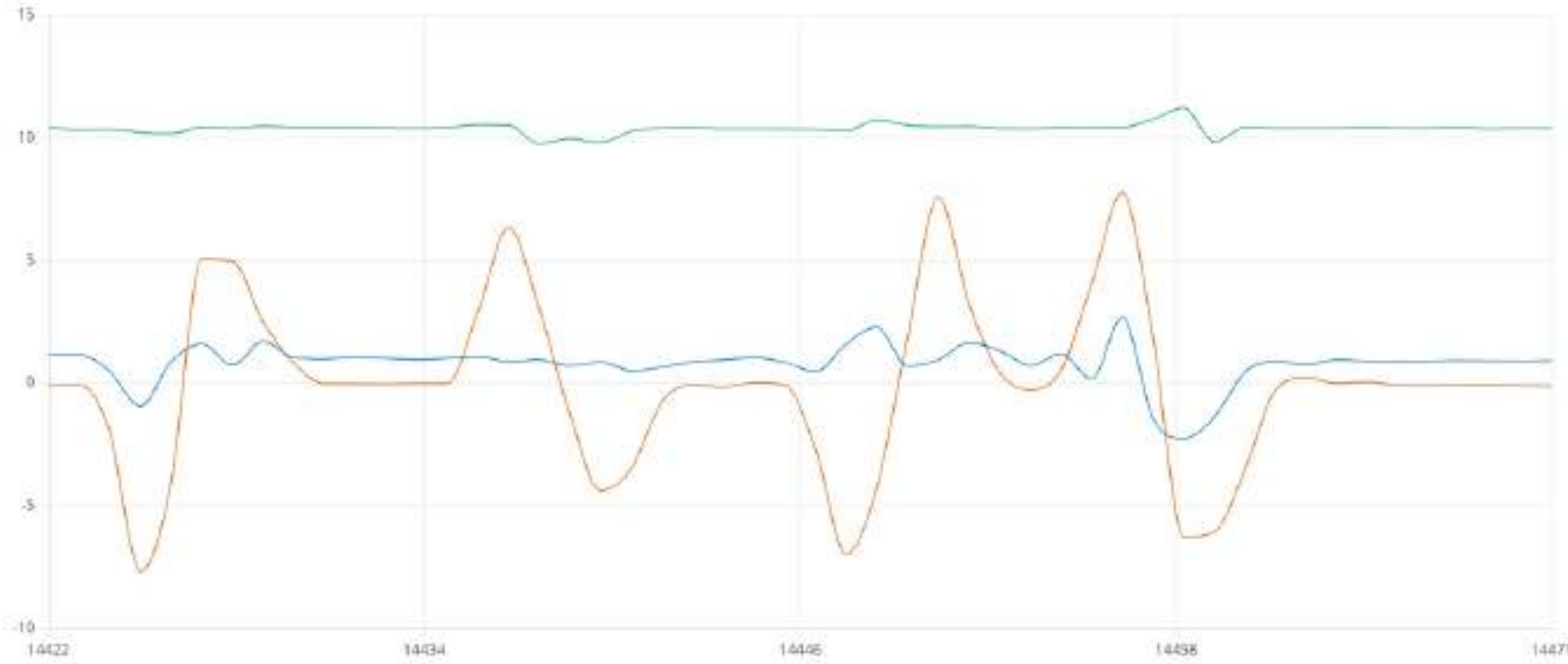


Figure 12.1

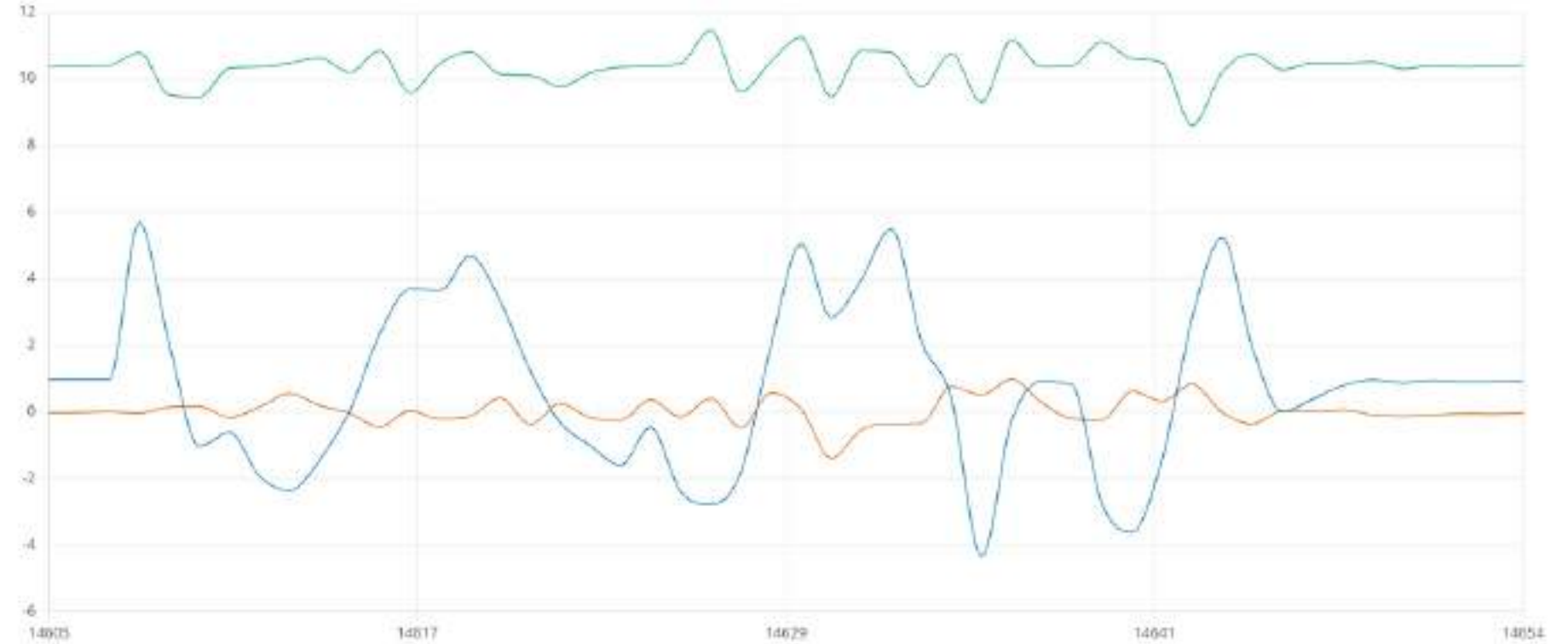


Figure 12.2

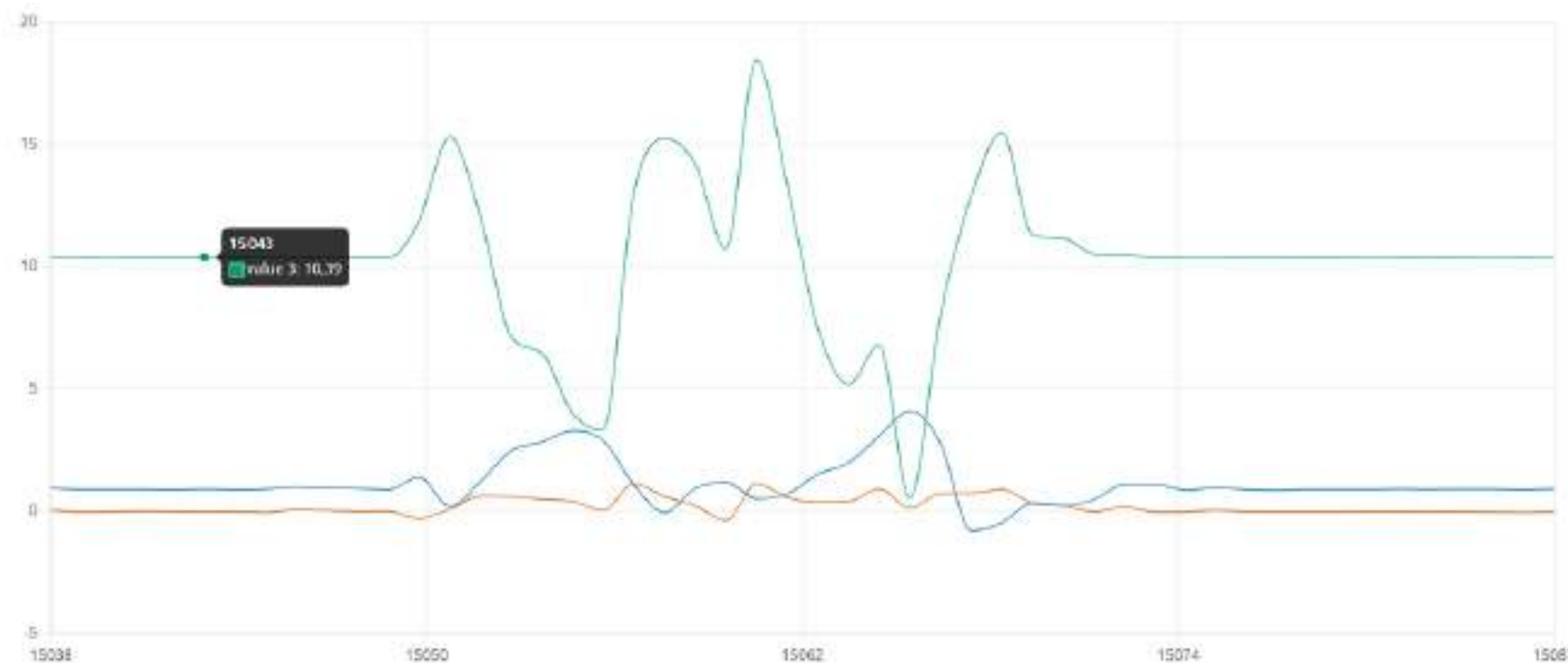


Figure 12.3

V. Experience

- Rotation suivant l'axe \vec{OX}
- Rotation suivant l'axe \vec{OY}
- Rotation suivant l'axe \vec{OZ}



Figure 12.4

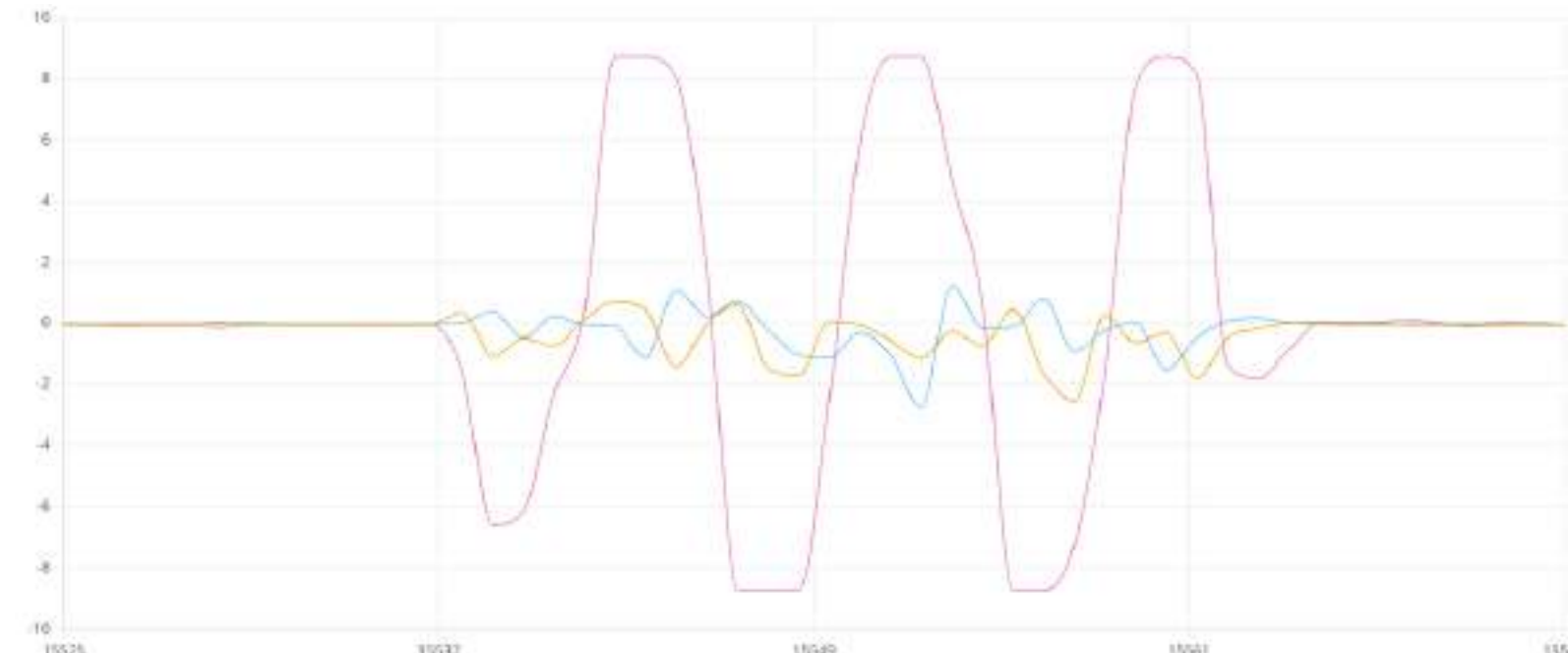


Figure 12.5

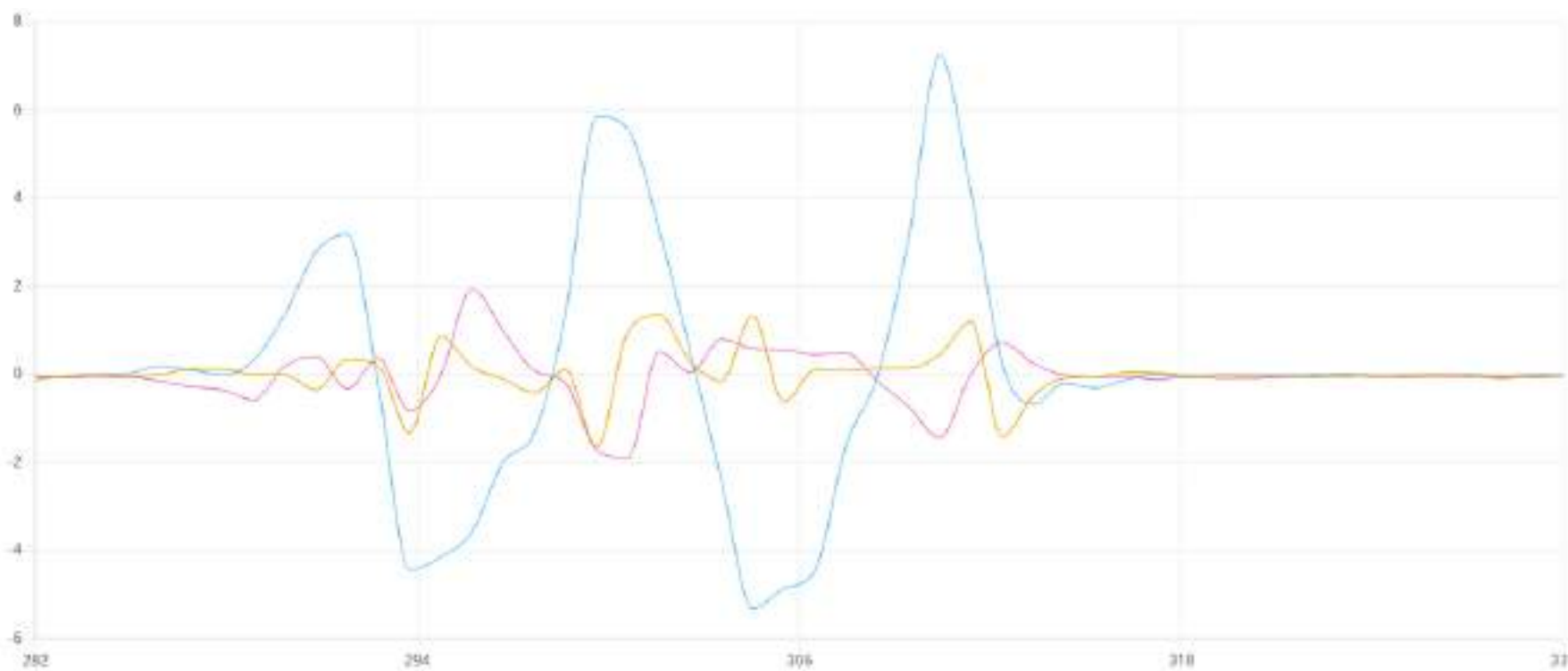


Figure 12.7

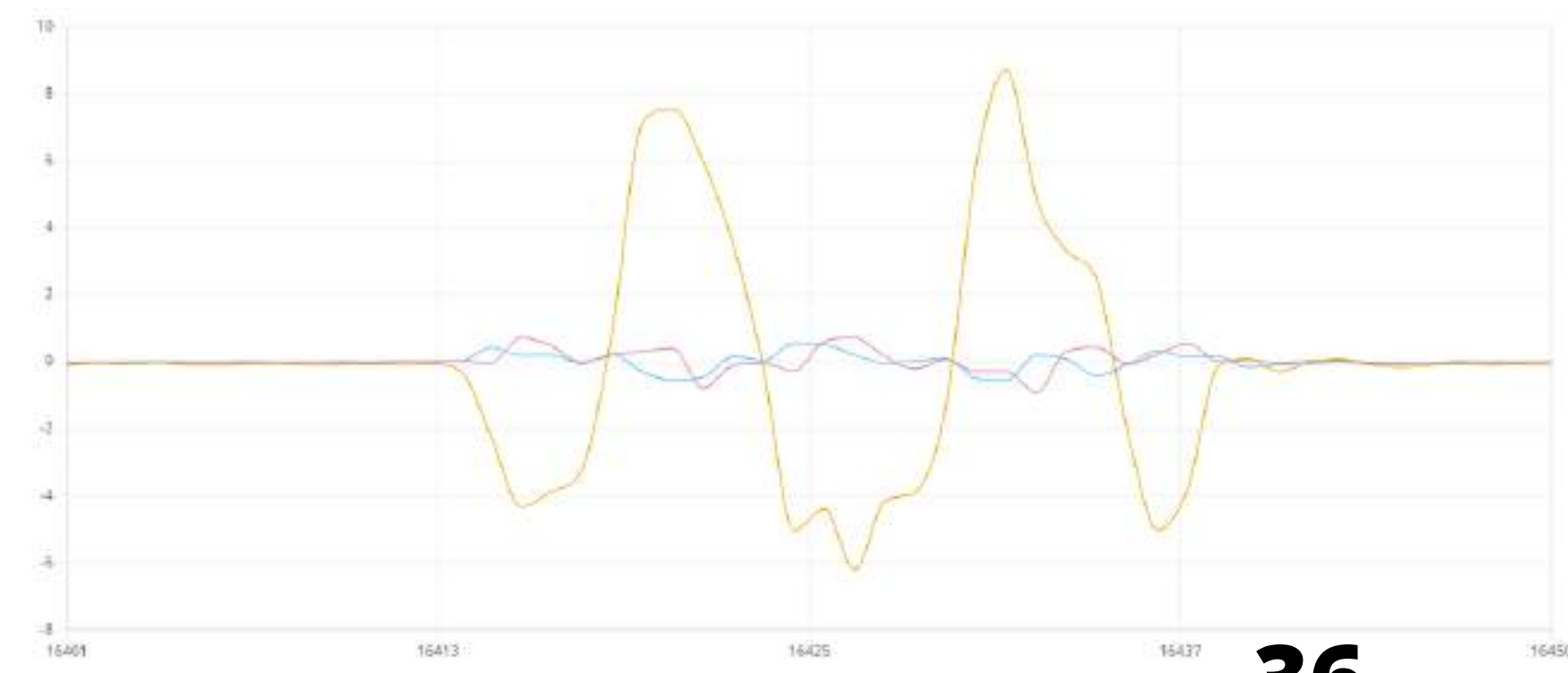


Figure 12.6

V. Experience

En mesurant l'angle de rotation :

- Angle autour de l'axe \overrightarrow{OX}
- Angle filtré autour de l'axe \overrightarrow{OX}
- Angle autour de l'axe \overrightarrow{OY}
- Angle filtré autour de l'axe \overrightarrow{OY}

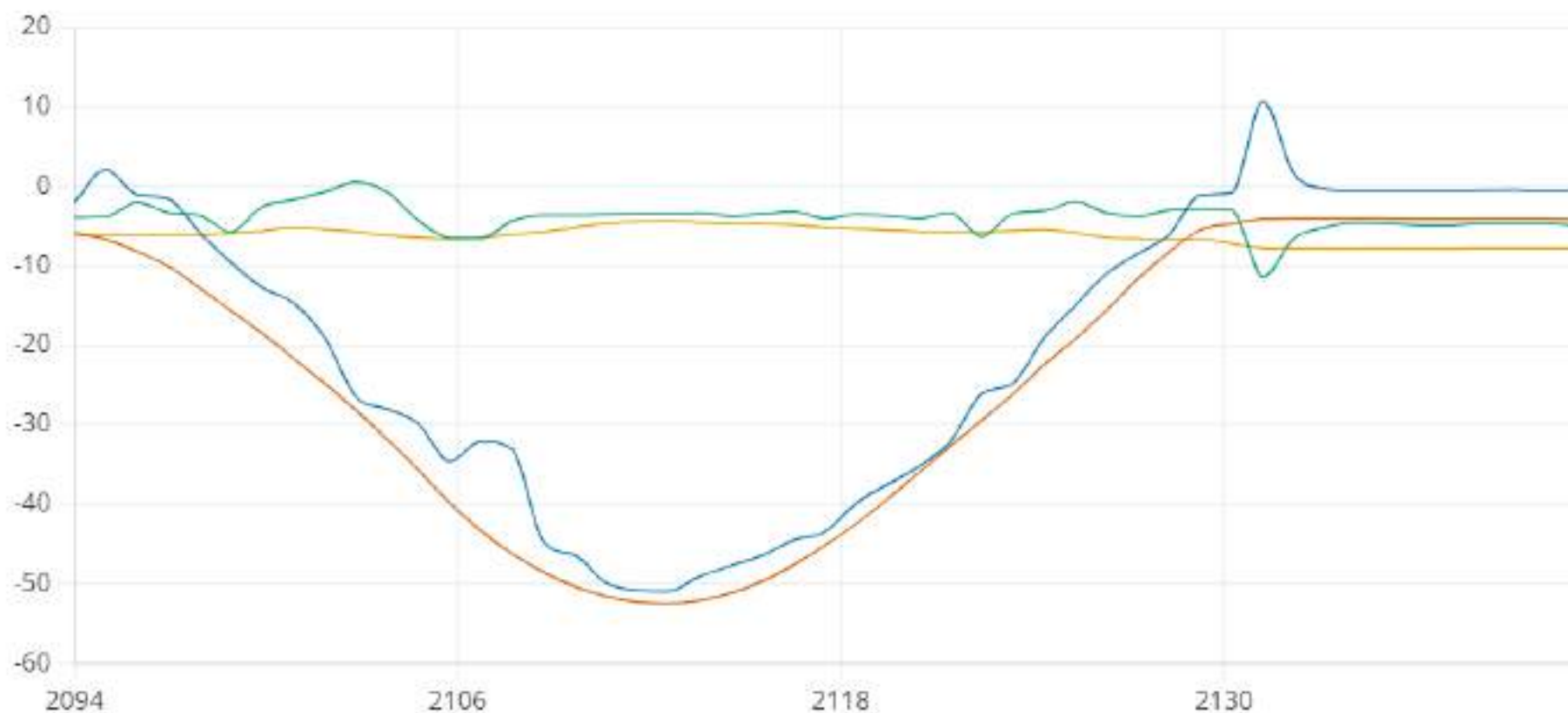


Figure 12.9

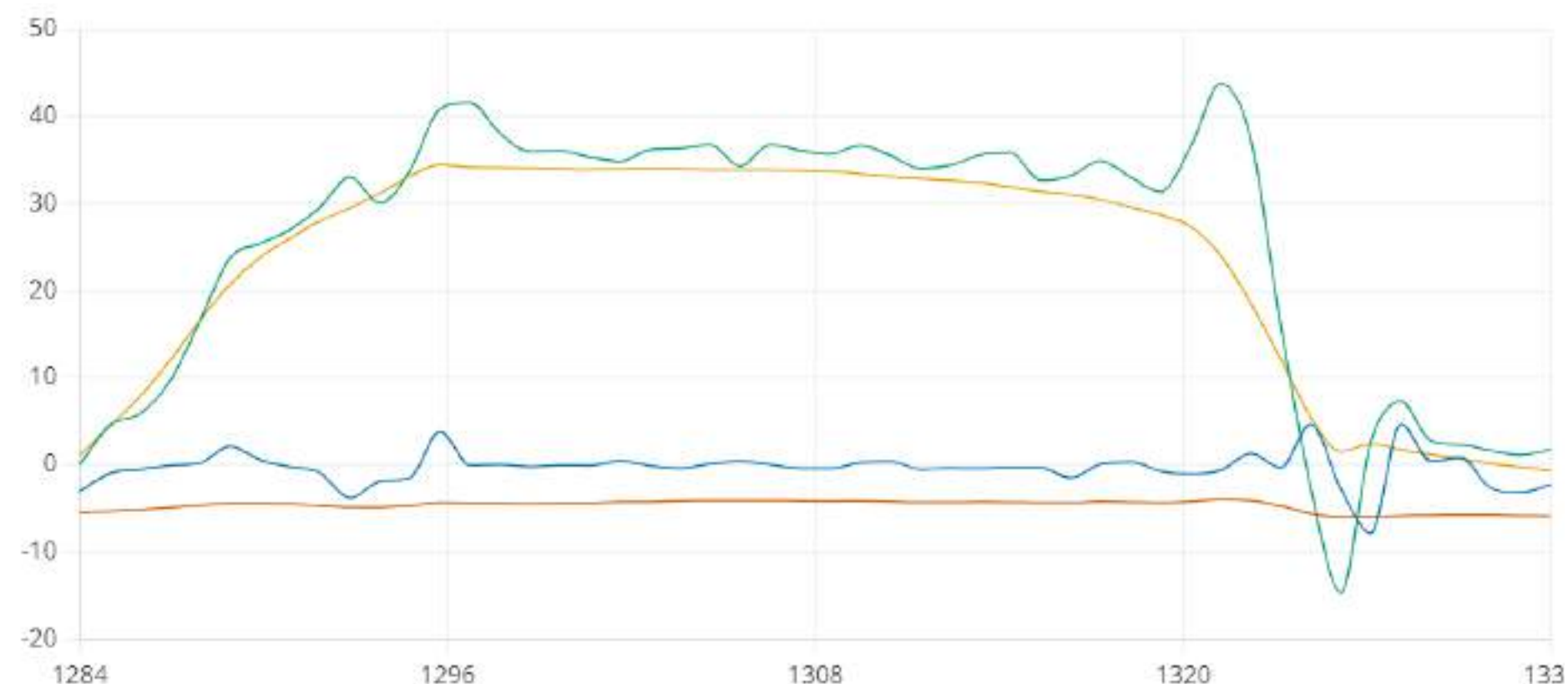
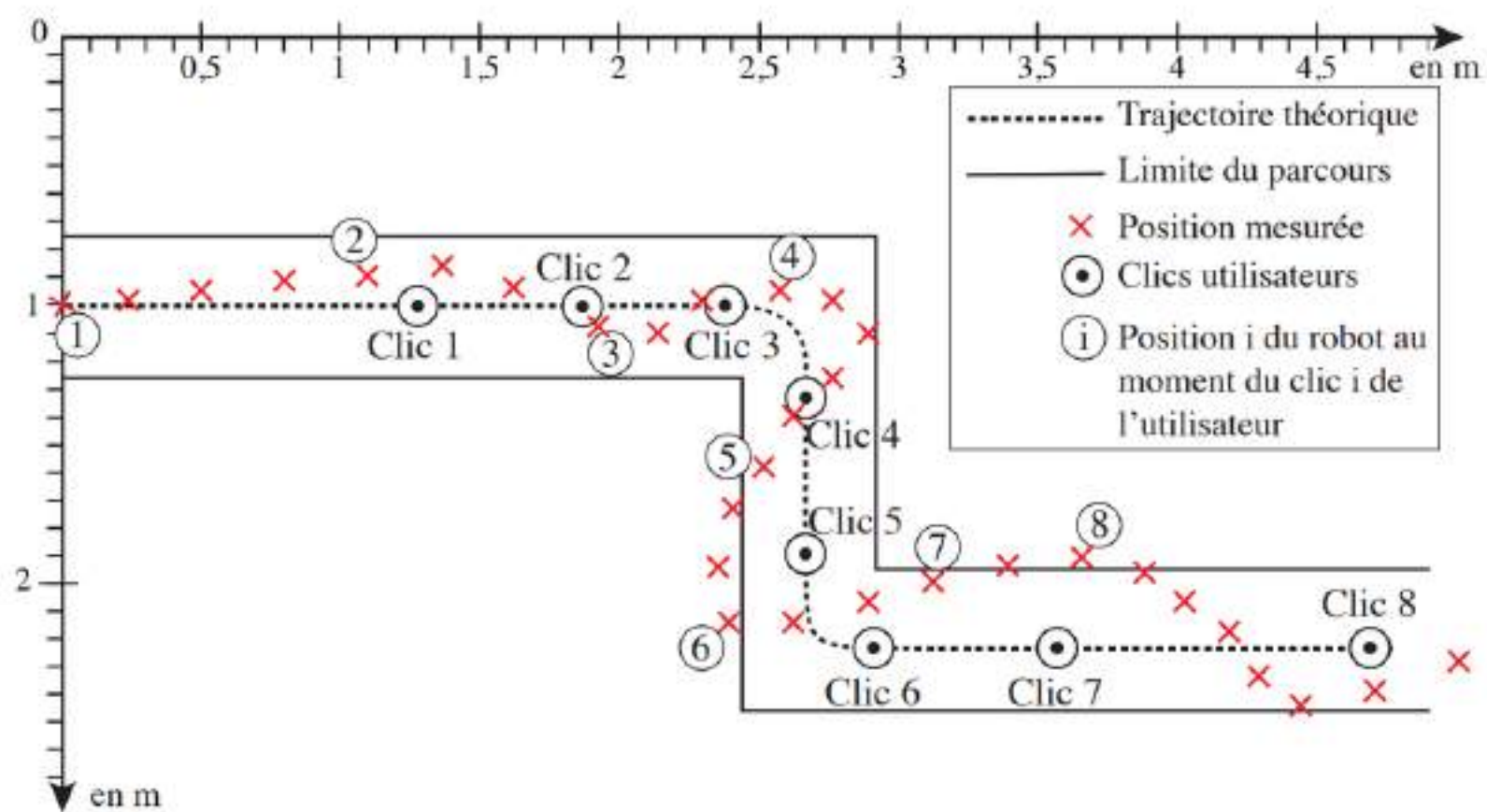


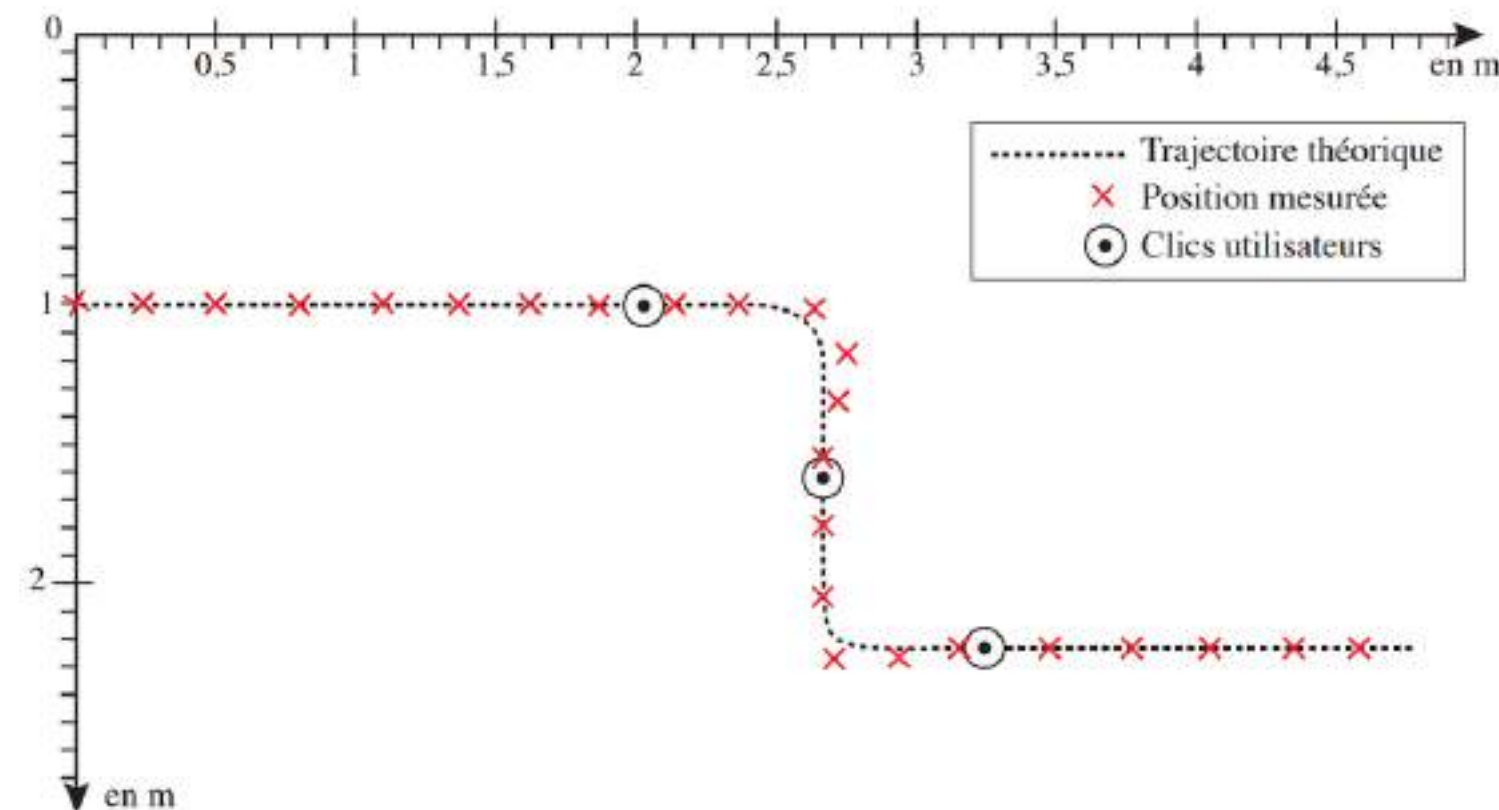
Figure 12.8

V. Experience

- En conséquent :



Sans le filtre



Avec le filtre

Figure 12.10

ANNEXES

```
1  #include <Adafruit_MPU6050.h>
2  #include <Adafruit_Sensor.h>
3  #include <Wire.h>
4
5  Adafruit_MPU6050 mpu;
6
7  void setup(void) {
8      Serial.begin(115200);
9
10     if (!mpu.begin()) {
11         Serial.println("Failed to find MPU6050 chip");
12         while (1) {
13             delay(10);
14         }
15     }
16
17     mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
18
19     mpu.setGyroRange(MPU6050_RANGE_500_DEG);
20
21     mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
22
23     delay(100);
24 }
25
```

```
26 void loop() {
27     sensors_event_t a, g, temp;
28     mpu.getEvent(&a, &g, &temp);
29
30     Serial.print(a.acceleration.x);
31     Serial.print(",");
32     Serial.print(a.acceleration.y);
33     Serial.print(",");
34     Serial.print(a.acceleration.z);
35     Serial.print(", ");
36     Serial.print(g.gyro.x);
37     Serial.print(",");
38     Serial.print(g.gyro.y);
39     Serial.print(",");
40     Serial.print(g.gyro.z);
41     Serial.println("");
42
43     delay(100);
44 }
```

ANNEXES

```
1 #include "I2Cdev.h"
2 #include "MPU6050.h"
3 #include "Wire.h"
4
5 MPU6050 sensor;
6
7 int16_t ax, ay, az;
8 int16_t gx, gy, gz;
9
10 float ax_g, ay_g, az_g;
11 float gx_dps, gy_dps, gz_dps;
12
13 unsigned long tiempo_prev;
14 float dt;
15 float ang_x = 0.0;
16 float ang_y = 0.0;
17 float ang_x_prev = 0.0;
18 float ang_y_prev = 0.0;
19
20 const float alpha = 0.98;
21
22 void setup() {
23     Serial.begin(57600);
24     Wire.begin();
25     sensor.initialize();
26
27     if (sensor.testConnection()) {
28         Serial.println("Le capteur fonctionne correctement");
29     } else {
30         Serial.println("Erreur lors de l'initialisation du capteur");
31     }
32
33     tiempo_prev = millis();
34 }
35
```

```
36 void loop() {
37     sensor.getAcceleration(&ax, &ay, &az);
38     sensor.getRotation(&gx, &gy, &gz);
39
40     ax_g = ax / 16384.0;
41     ay_g = ay / 16384.0;
42     az_g = az / 16384.0;
43
44     gx_dps = gx / 131.0;
45     gy_dps = gy / 131.0;
46     gz_dps = gz / 131.0;
47
48     unsigned long currentTime = millis();
49     dt = (currentTime - tiempo_prev) / 1000.0;
50     tiempo_prev = currentTime;
51
52     float accel_ang_x = atan2(ay_g, sqrt(ax_g * ax_g + az_g * az_g)) * (180.0 / PI);
53     float accel_ang_y = atan2(-ax_g, sqrt(ay_g * ay_g + az_g * az_g)) * (180.0 / PI);
54
55     ang_x = alpha * (ang_x_prev + gx_dps * dt) + (1 - alpha) * accel_ang_x;
56     ang_y = alpha * (ang_y_prev + gy_dps * dt) + (1 - alpha) * accel_ang_y;
57
58     ang_x_prev = ang_x;
59     ang_y_prev = ang_y;
60
61     // Afficher les angles
62     Serial.print("Rotation en X: ");
63     Serial.print(ang_x);
64     Serial.print("\tRotation en Y: ");
65     Serial.println(ang_y);
66
67     delay(10);
68 }
```


ANNEXES

PARAMETER	CONDITIONS	MIN	TYP	MAX	Units	Notes
SUPPLY VOLTAGES						
VDD		2.4	2.5	3.6	V	
VDDIO		1.71	1.8	VDD	V	
SUPPLY CURRENTS						
Normal Mode	9-axis (no DMP), 1 kHz gyro ODR, 4 kHz accel ODR, 8 Hz mag. repetition rate		3.7		mA	
	6-axis (accel + gyro, no DMP), 1 kHz gyro ODR, 4 kHz accel ODR		3.4		mA	
	3-axis Gyroscope only (no DMP), 1 kHz ODR		3.2		mA	
	6-axis (accel + magnetometer, no DMP), 4 kHz accel ODR, mag. repetition rate = 8 Hz		730		μA	
	3-Axis Accelerometer, 4kHz ODR (no DMP)		450		μA	
	3-axis Magnetometer only (no DMP), 8 Hz repetition rate		280		μA	
Accelerometer Low Power Mode (DMP, Gyroscope, Magnetometer disabled)	0.98 Hz update rate		8.4		μA	1
	31.25 Hz update rate		19.8		μA	1
Full Chip Idle Mode Supply Current			8		μA	
TEMPERATURE RANGE						
Specified Temperature Range	Performance parameters are not applicable beyond Specified Temperature Range	-40		+85	°C	



**Merci pour votre
attention**