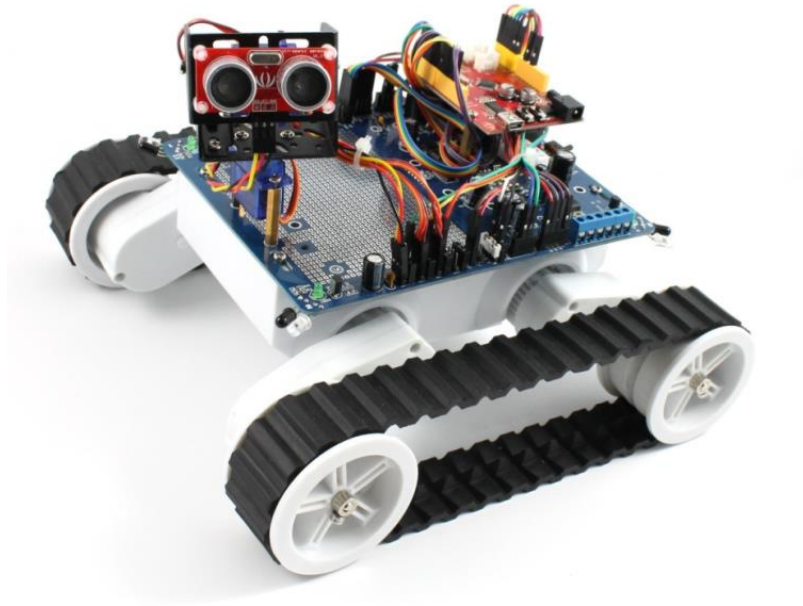


Eléments de robotique avec Arduino



Pascal MASSON

(pascal.masson@unice.fr)

*Version projection
Edition 2018-2019-V12*

Accéléromètre

Introduction

1. MEMS
2. Principe de fonctionnement
3. Mesure d'un angle avec 1 axe
4. Mesure d'un angle avec 2 axes
5. Tangage, roulis et lacet
6. Déplacement rectiligne
7. Accéléromètre GY-61
8. Accéléromètre GY-521

□ C'est quoi l'accélération ?

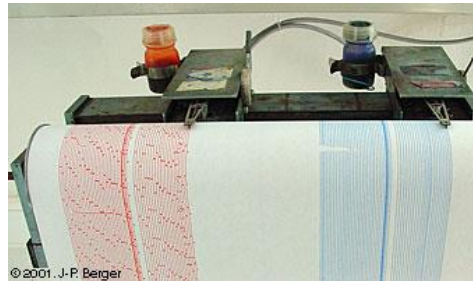
- L'accélération est une grandeur physique vectorielle utilisée pour représenter la modification de la vitesse d'un mouvement en fonction du temps
- L'accélération « a » correspond à la dérivée temporelle de la vitesse « v » :
$$\vec{a} = \frac{d\vec{v}}{dt}$$
- L'accélération est définie en m/s^2 (SI) et la majorité des documentations sur ces capteurs expriment l'accélération en " g " ($= 9,80665 \text{ m/s}^2$).
- L'accélération exerce une force qui dépend de la masse : $\vec{F} = m \cdot \vec{a}$
- Exemples :
 - ✓ Une voiture passe de 0 à 100 km/h en 5 s, elle a une accélération de $(100 \text{ km/h})/(5 \text{ s}) = 20 \text{ (km/h)/s} = 5,6 \text{ m/s}^2 = 0,56 \text{ g}$
 - ✓ lors d'un choc frontal, une voiture roulant à 30 km/h s'arrête en environ 0,1 s, ce qui représente une variation de vitesse de $(-30 \text{ km/h})/(0,1 \text{ s}) = -300 \text{ (km/h)/s} = -83 \text{ m/s}^2 = -8,3 \text{ g}$

❑ Qu'est qu'un accéléromètre et qu'est qu'on en fait ?

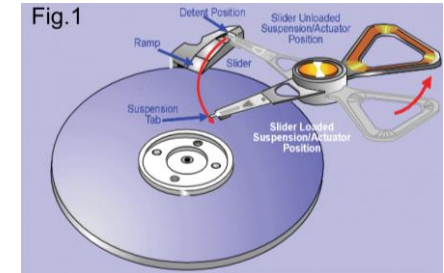
- C'est un capteur (MEMs, Microelectromechanical systems) qui, fixé à un mobile, permet de mesurer l'accélération de ce dernier suivant 1, 2 ou 3 axes.
- Il permet aussi de déterminer l'inclinaison d'un objet grâce à l'effet de la gravité terrestre.
- On les utilise pour :



Drone, robot, fusée



Séisme



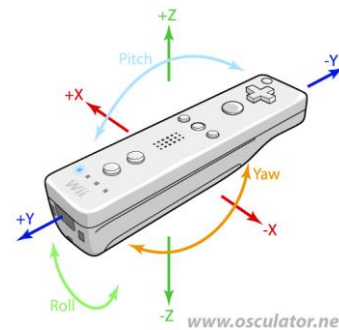
Disque dur



Smartphone



Stabilisateur



Wii



Airbags

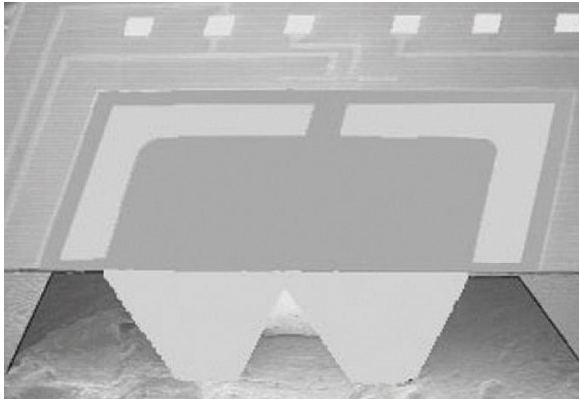
1.1. Définition

- MEMS pour Microelectromechanical systems, ou microsystème électromécanique en Français
- C'est un microsystème (taille micrométrique) qui comprend un ou plusieurs éléments mécaniques réalisant une fonction de capteur ou d'actionneur.

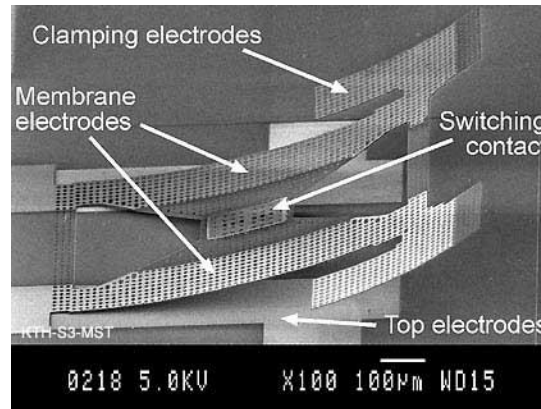
1.2. Origine

- Les MEMS ont été développés au début des années 1970 en tant que dérivés de la micro-électronique
- La première commercialisation remonte aux années 1980 avec des capteurs de pression sur silicium

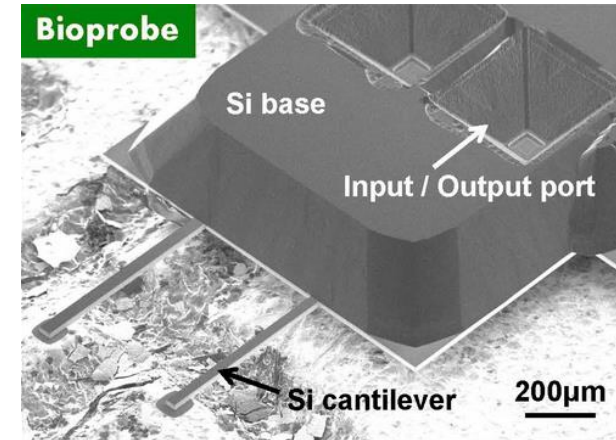
1.3. Exemples



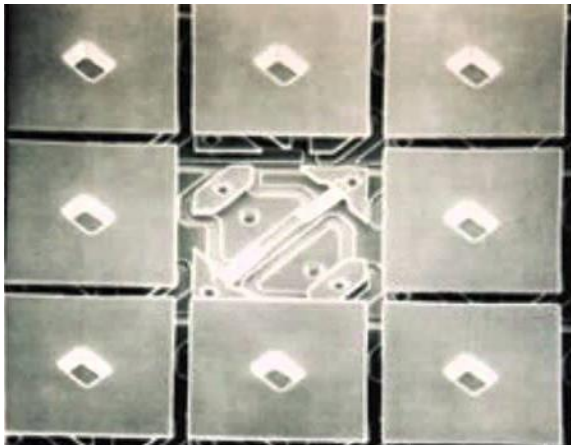
Pression



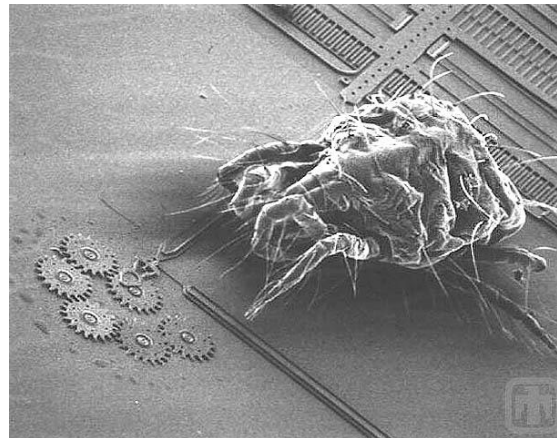
Interrupteur



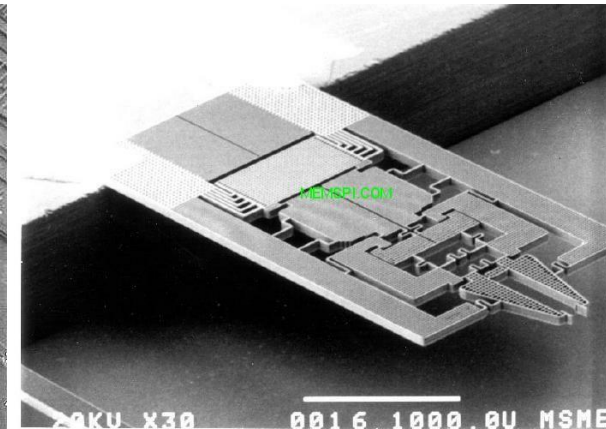
Topologie de surface



**Miroirs pour
vidéo projecteur**



Moteur



Pince

1.4. MEMS market

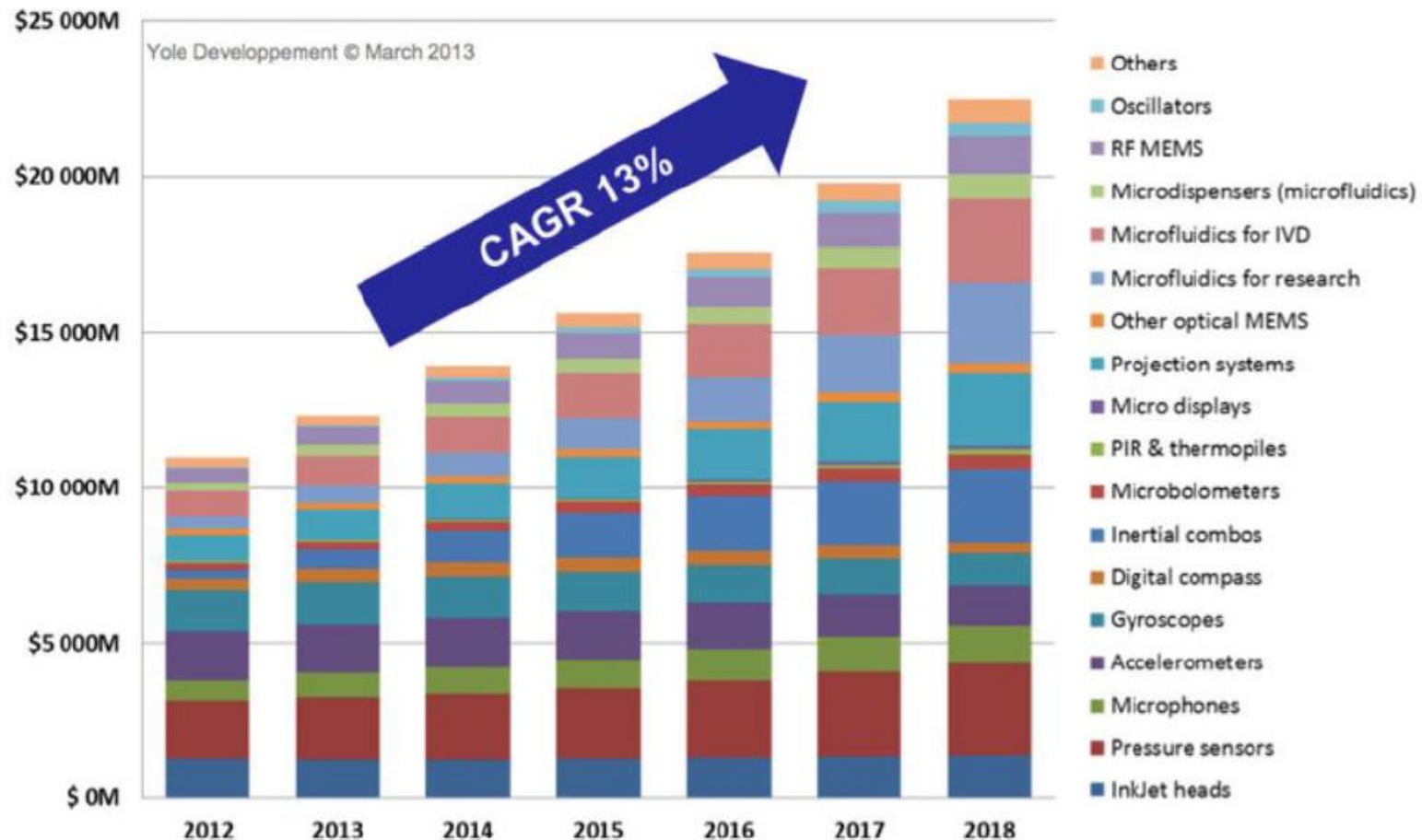
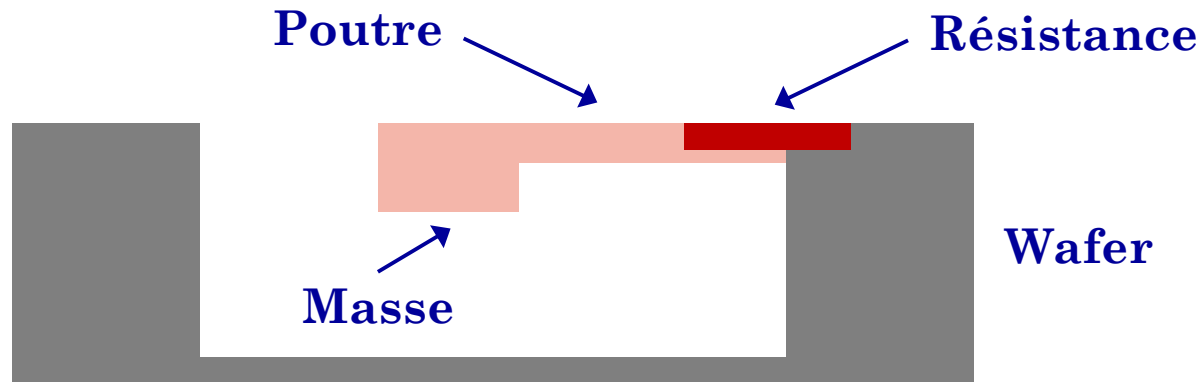


Figure 2: MEMS Sensor Market forecast
(Source: Yole Développement, April 2015)

2.1. Poutre et contraintes

□ Présentation du MEMs



- Une résistance est intégrée à la poutre et change de valeur en fonction de la torsion.

2.1. Poutre et contraintes

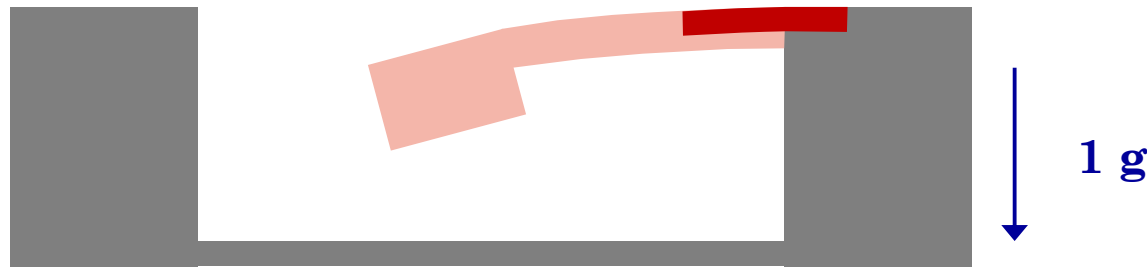
□ Poutre dans « l'espace »



- Aucune force s'exerce sur la poutre

2.1. Poutre et contraintes

□ Poutre sur terre



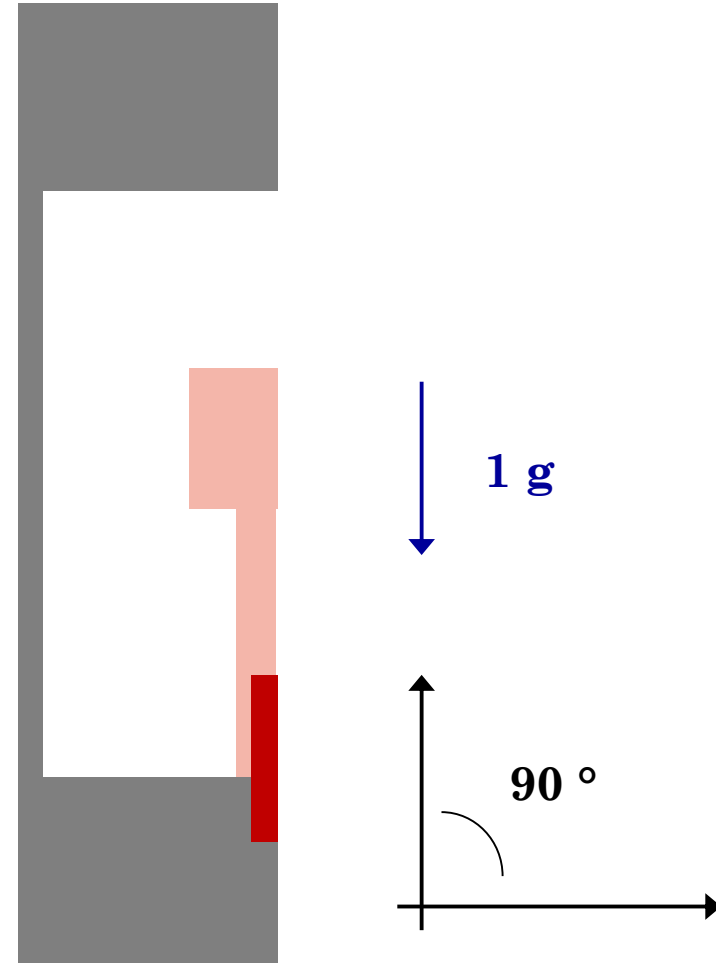
- L'attraction terrestre fléchit la poutre et modifie la valeur de la résistance
- La force exercée par la terre sur la poutre s'écrit : $F = m.a$ avec dans notre cas $a = 1\text{ g} = 9.8\text{ ms}^{-2}$ qui correspond à l'accélération



2.1. Poutre et contraintes

□ Poutre sur terre

- Si on tourne le MEMs de 90° , alors la poutre ne fléchit plus
- Il est donc possible de connaître l'inclinaison du MEMs par rapport à la terre en mesurant la valeur de la résistance ou plutôt la tension à ses bornes



2.2. Capteur capacitif

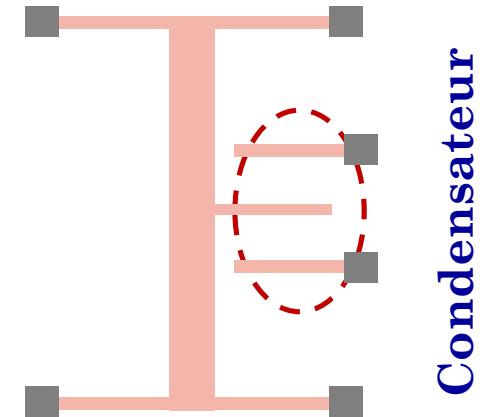
□ Principe

- Plutôt que d'utiliser une poutre, les industriels préfèrent l'utilisation d'un condensateur plan dont la valeur dépend de la distance entre les armatures



$$C = S \cdot \frac{\epsilon}{d}$$

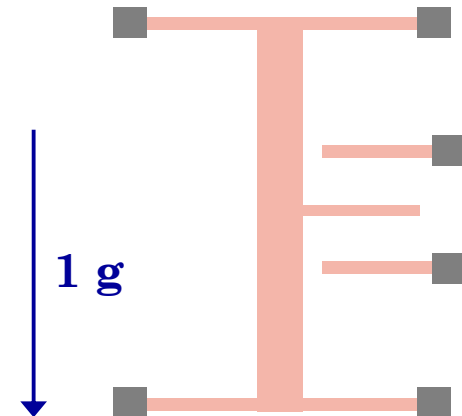
- La valeur du condensateur fixe la fréquence d'oscillation d'un oscillateur électrique



2.2. Capteur capacitif

□ Principe

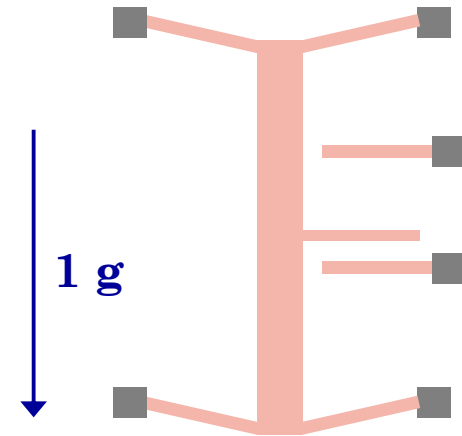
- L'attraction terrestre modifie la forme de la capacité et donc sa valeur
- La fréquence d'oscillation est modifiée ce qui permet de déterminer l'accélération



2.2. Capteur capacitif

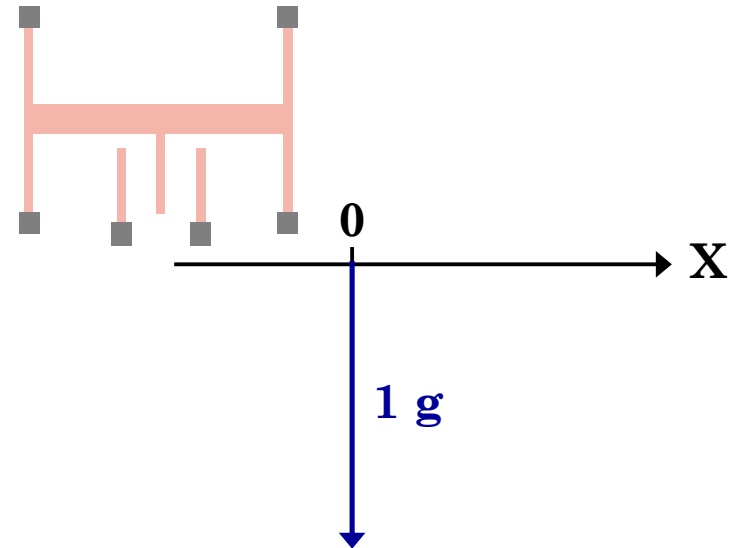
□ Principe

- L'attraction terrestre modifie la forme de la capacité et donc sa valeur
- La fréquence d'oscillation est modifiée ce qui permet de déterminer l'accélération



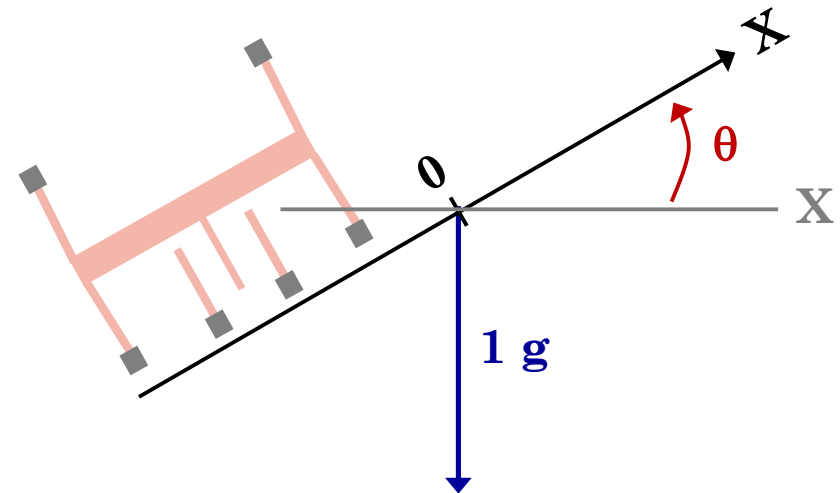
3.1. Position à l'origine

- On considère que l'axe X est tangent à la terre et donc que l'attraction terrestre n'affecte pas l'accéléromètre



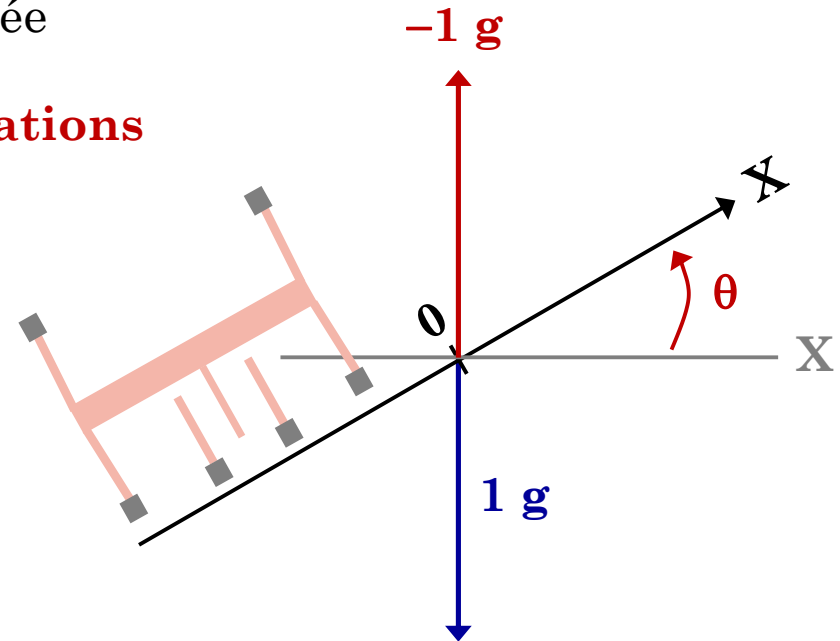
3.2. Inclinaison

- On applique une inclinaison θ de l'axe X
- La capacité de l'accéléromètre est modifiée



3.2. Inclinaison

- On applique une inclinaison θ de l'axe X
- La capacité de l'accéléromètre est modifiée
- **L'accéléromètre renvoie les informations d'une accélération égale à -1 g**



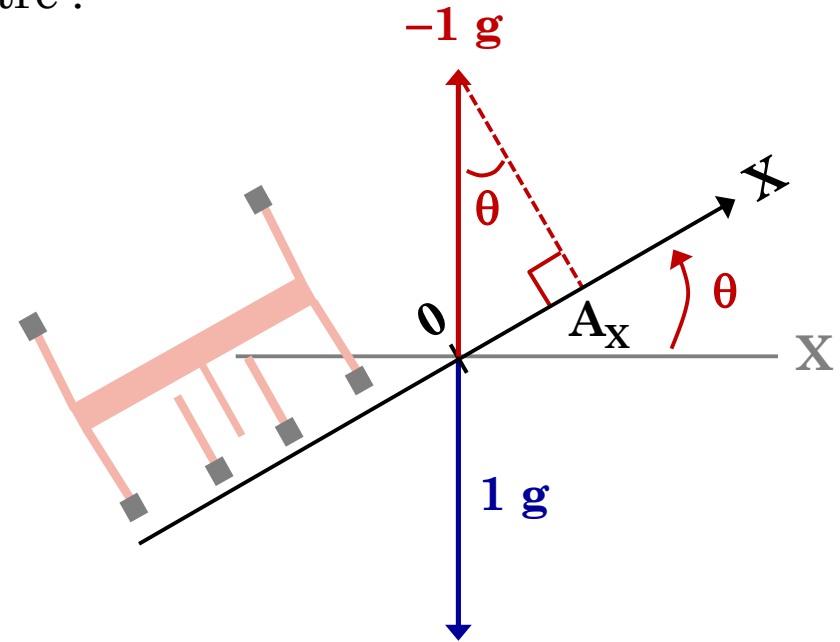
3.2. Inclinaison

- La projection de l'accélération sur l'axe X donne la valeur renvoyée par l'accéléromètre :

$$A_x = g \cdot \sin(\theta)$$

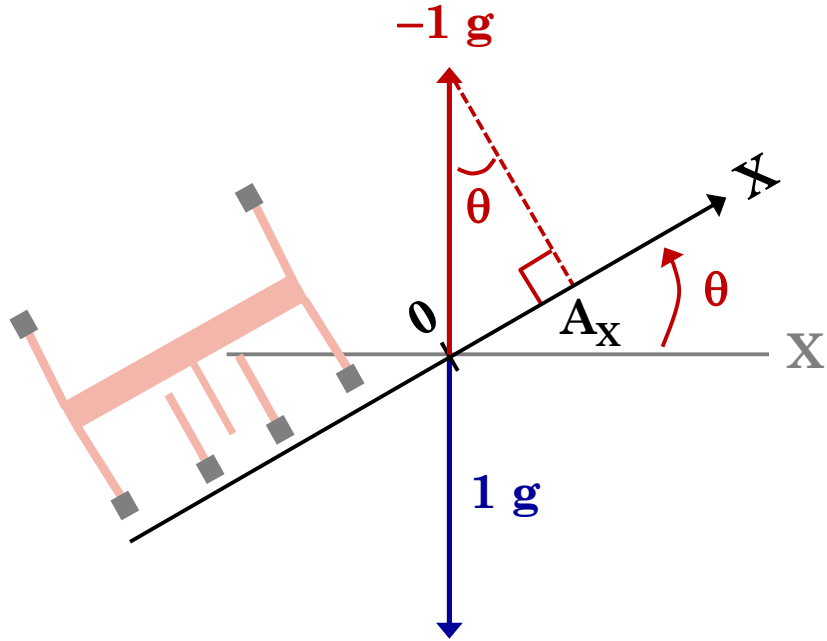
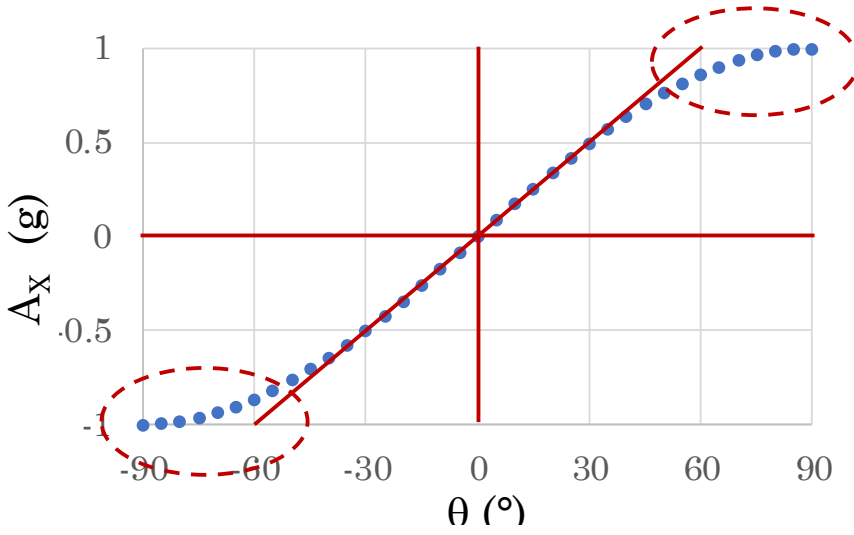
- On trouve la valeur de l'angle avec :

$$\theta = \sin^{-1}\left(\frac{A_x}{g}\right)$$



3.3. Problème d'angle

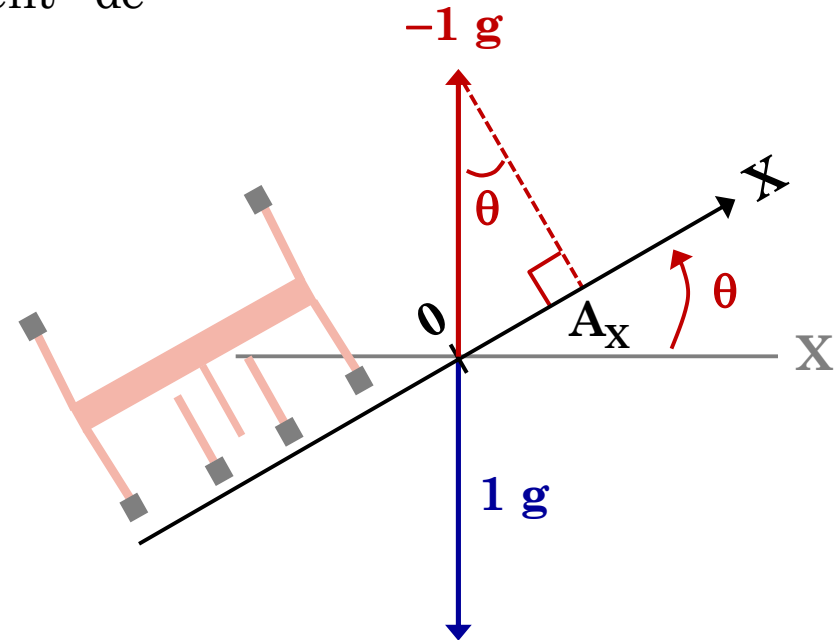
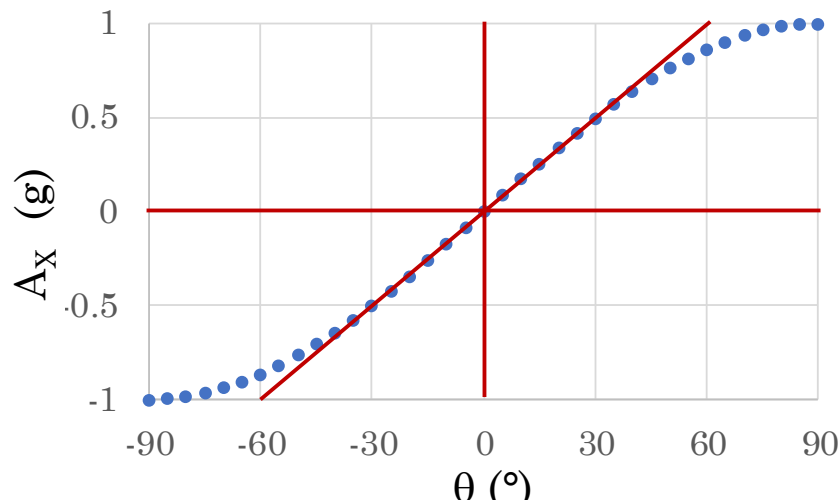
- On voit que A_x varie peu (perte de sensibilité) quand on s'approche de -90° et $+90^\circ$



3.3. Problème d'angle

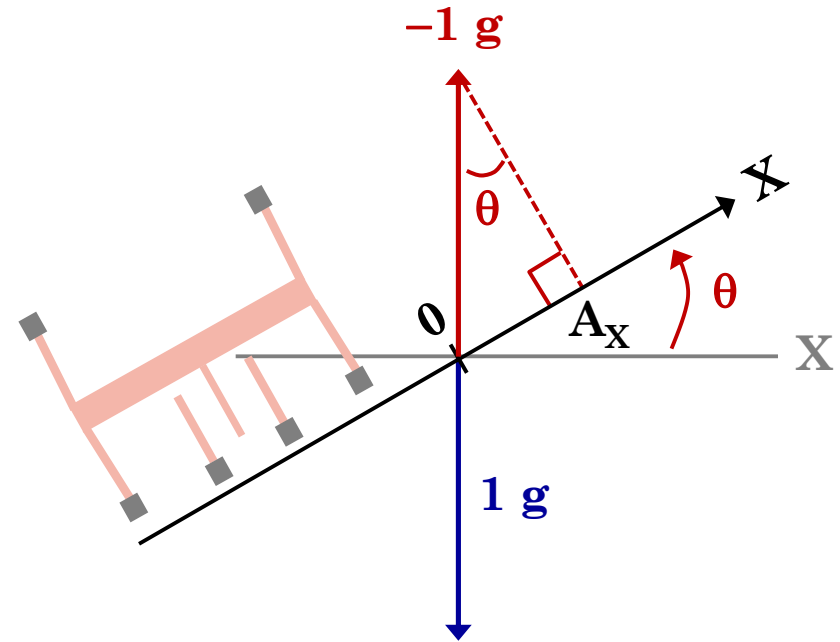
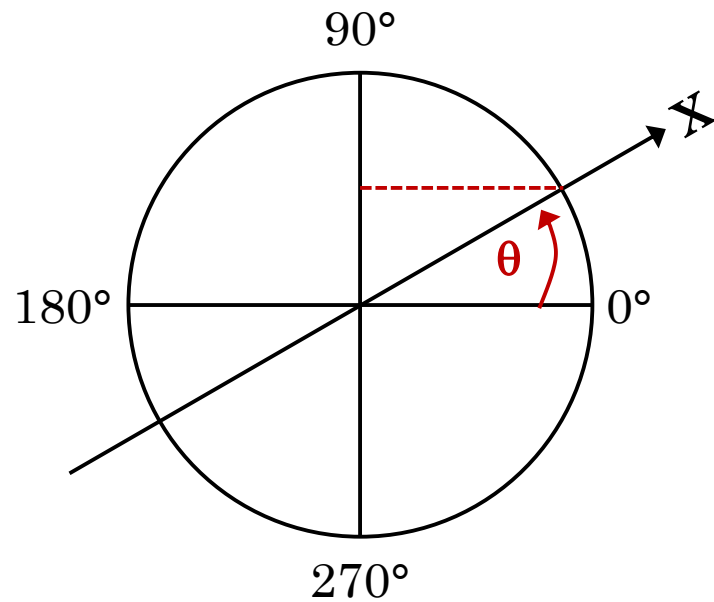
- On peut aussi remarquer que pour un angle faible, l'accélération dépend linéairement de l'angle

$$\theta \approx \frac{A_x}{g}$$



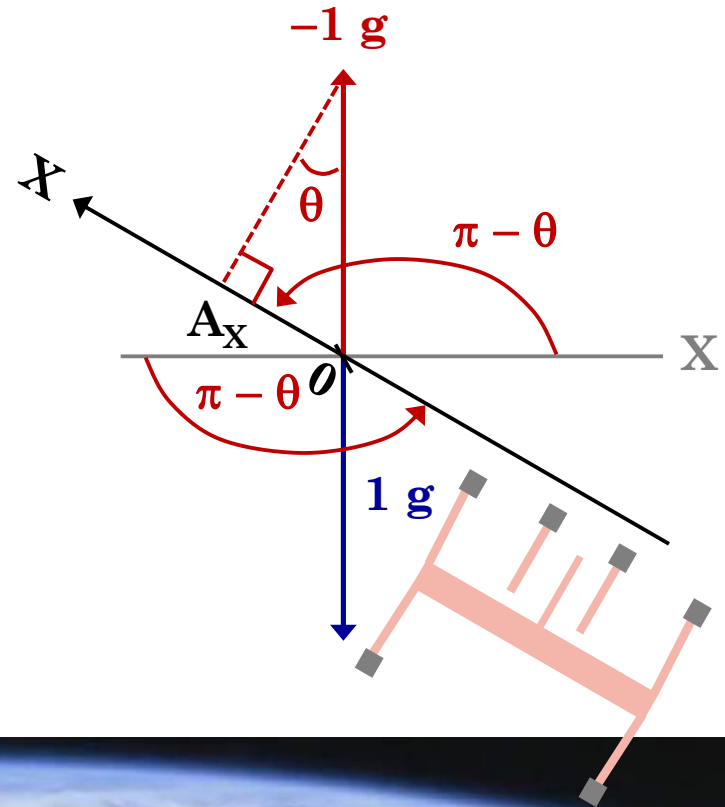
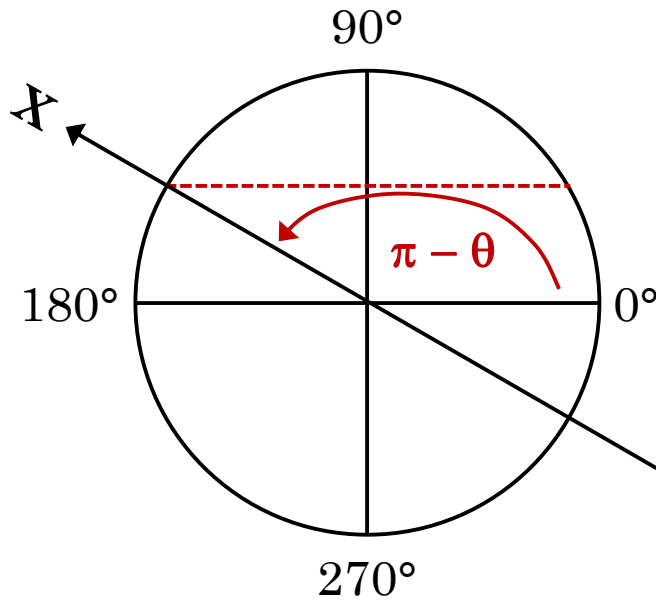
3.3. Problème d'angle

- La propriété du sinus induit qu'on ne peut pas différencier les angles θ



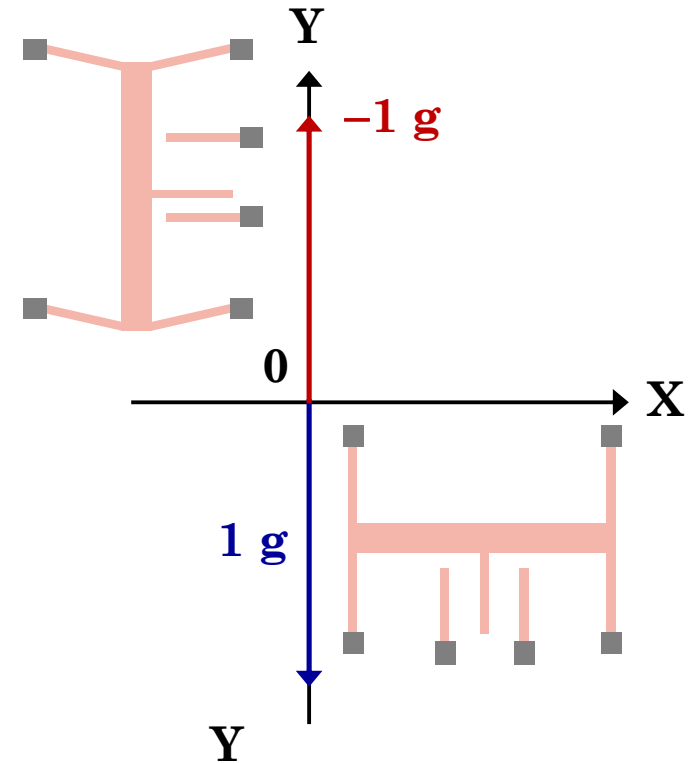
3.3. Problème d'angle

- La propriété du sinus induit qu'on ne peut pas différencier les angles θ et $\pi - \theta$



4.1. Position à l'origine

- On considère 2 axes dont Y est perpendiculaire à la gravité



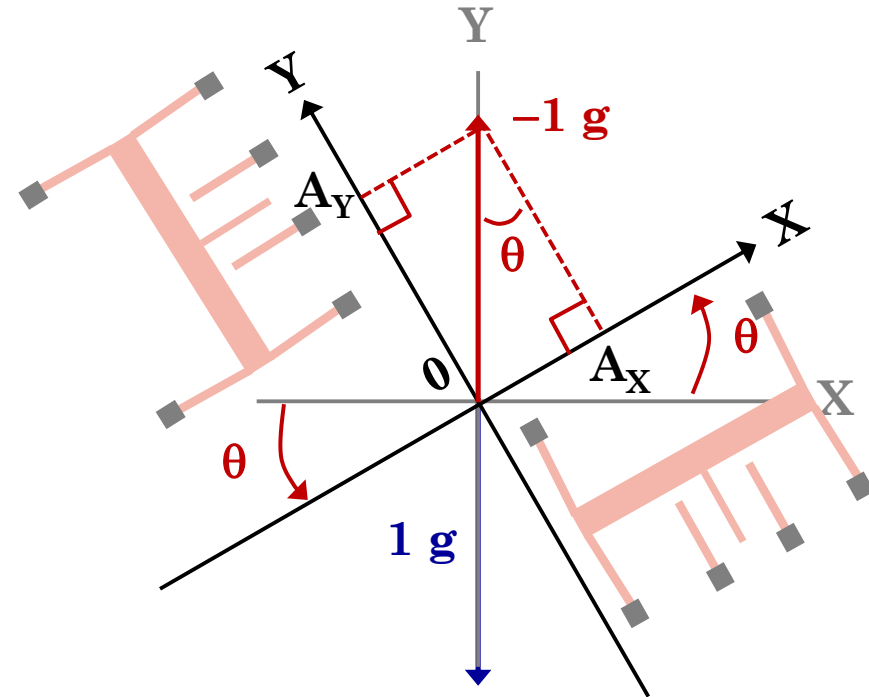
4.2. Inclinaison

- On applique une inclinaison θ
- La projection de l'accélération sur l'axe X donne :

$$A_X = g \cdot \sin(\theta)$$

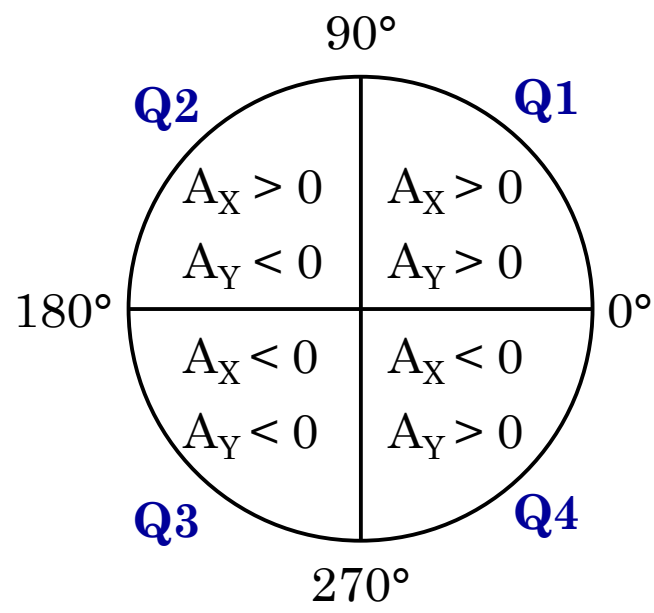
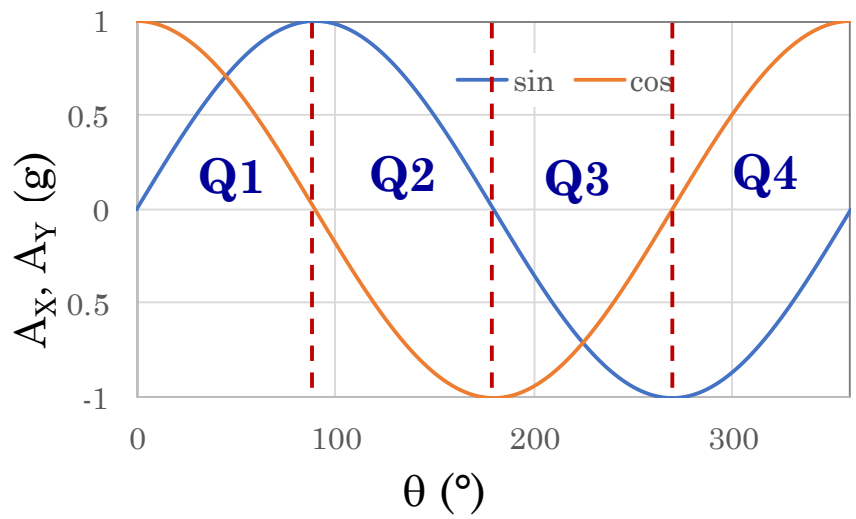
- La projection de l'accélération sur l'axe Y donne :

$$A_Y = g \cdot \cos(\theta)$$



4.3. Détermination de l'angle

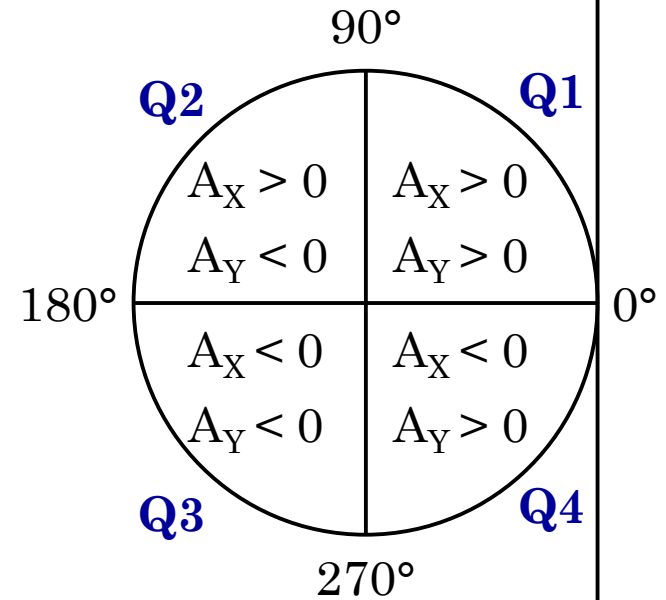
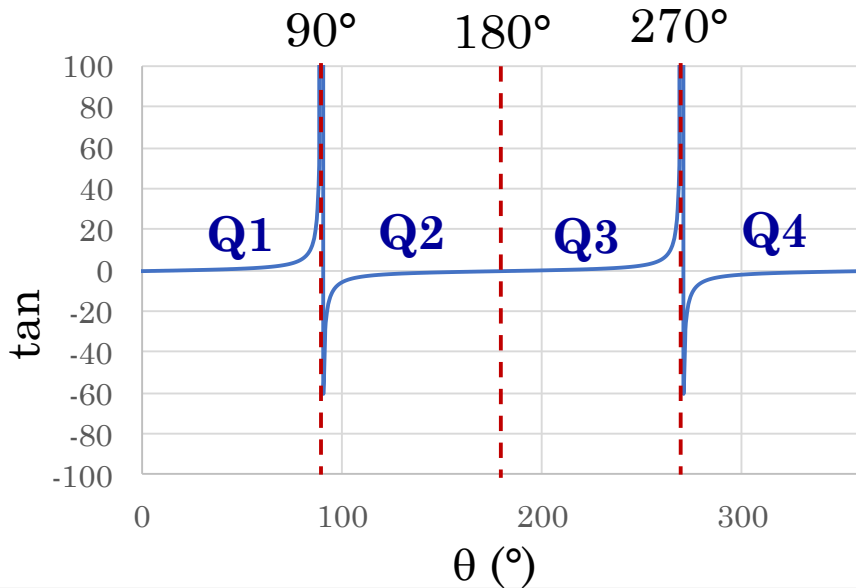
- Le signe de A_X et A_Y permet de déterminer dans quel quadrant se trouve θ



4.3. Détermination de l'angle

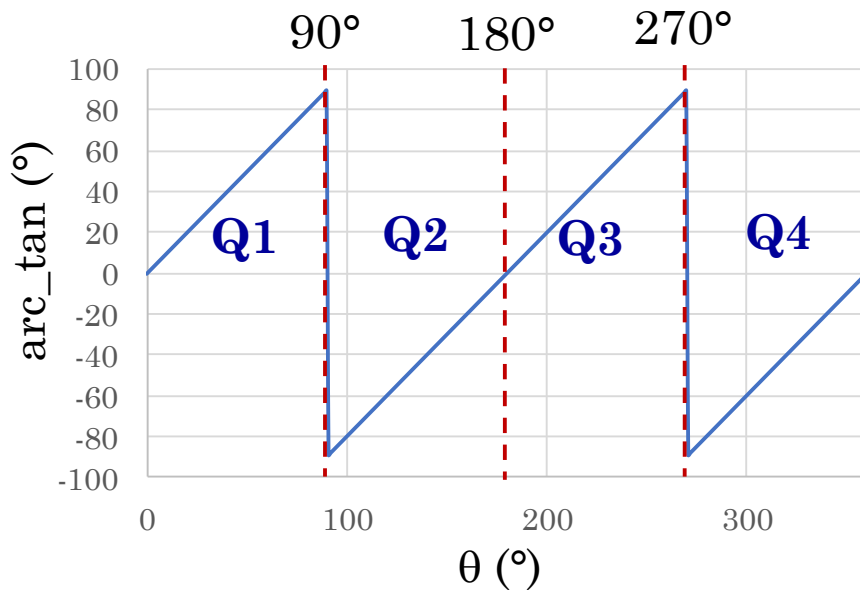
- L'utilisation des données sur les axes permet de déterminer l'angle en utilisant la tangente :

$$\theta = \tan^{-1} \left(\frac{A_X}{A_Y} \right)$$



4.3. Détermination de l'angle

- La visualisation de la fonction arc-tangente permet de trouver l'angle en fonction du cadran :

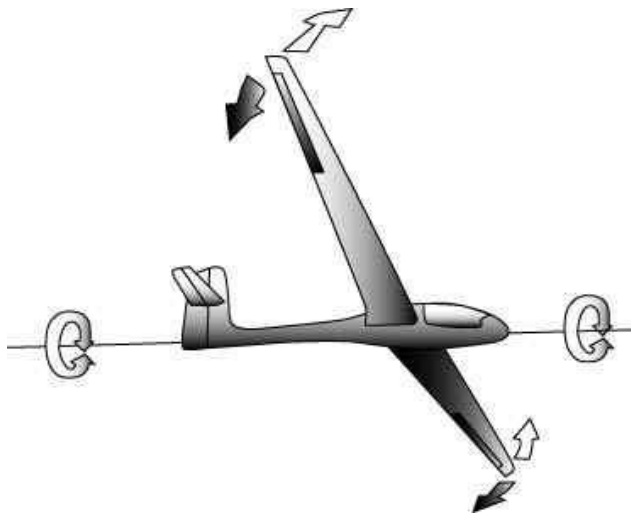


- Q1 : $\theta = \tan^{-1}\left(\frac{A_X}{A_Y}\right)$
- Q2 : $\theta = \tan^{-1}\left(\frac{A_X}{A_Y}\right) + 180$
- Q3 : $\theta = \tan^{-1}\left(\frac{A_X}{A_Y}\right) + 180$
- Q4 : $\theta = \tan^{-1}\left(\frac{A_X}{A_Y}\right) + 360$

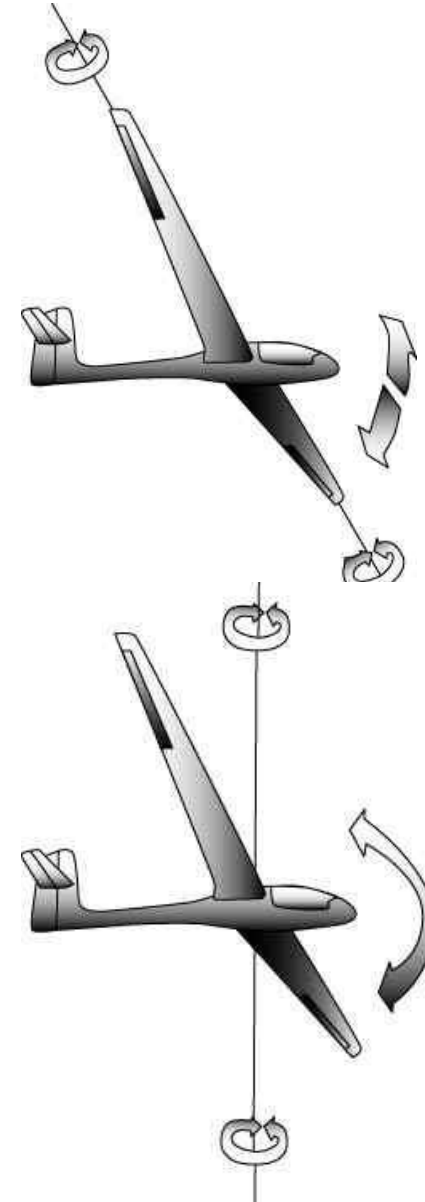


5.1. Définition (wikipedia)

- Le tangage (pitch) est un mouvement de rotation autour de l'axe transversal d'un objet en mouvement.



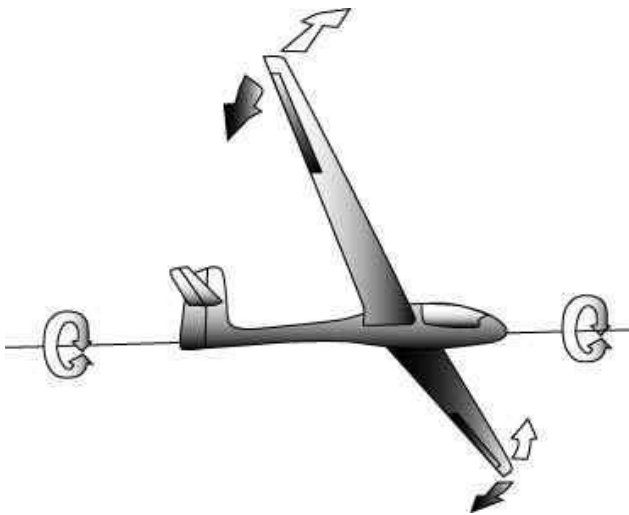
- Le roulis (roll) est un mouvement de rotation d'un mobile autour de son axe longitudinal (axe de roulis)



- Le lacet (yaw) est le mouvement de rotation horizontal d'un mobile autour d'un axe vertical

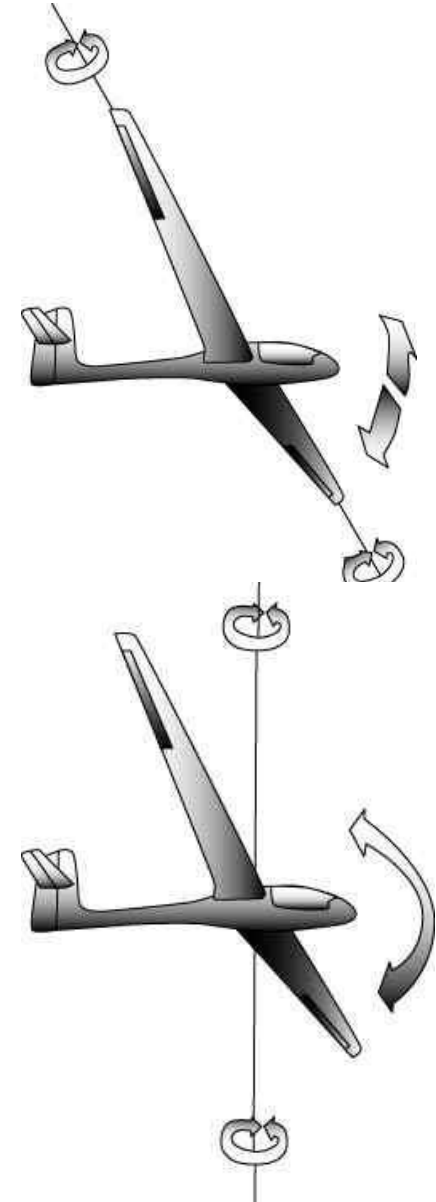
5.1. Définition (wikipedia)

- Le tangage (pitch) est un mouvement de rotation autour de l'axe transversal d'un objet en mouvement.



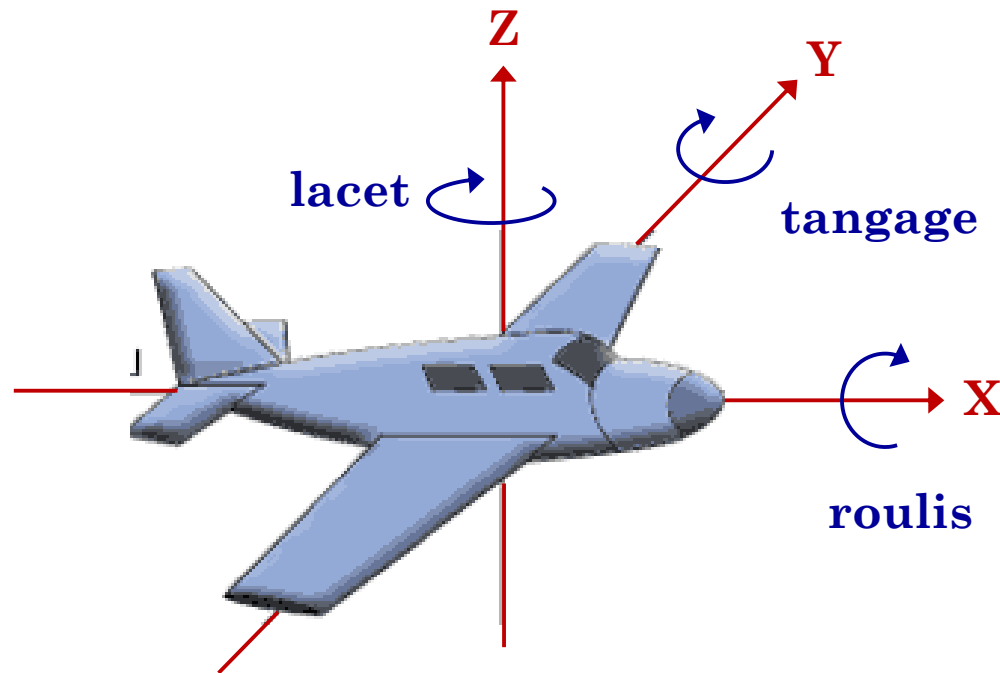
- Le roulis (roll) est un mouvement de rotation d'un mobile autour de son axe longitudinal (axe de roulis)

- Le lacet (yaw) est le mouvement de rotation horizontal d'un mobile autour d'un axe vertical



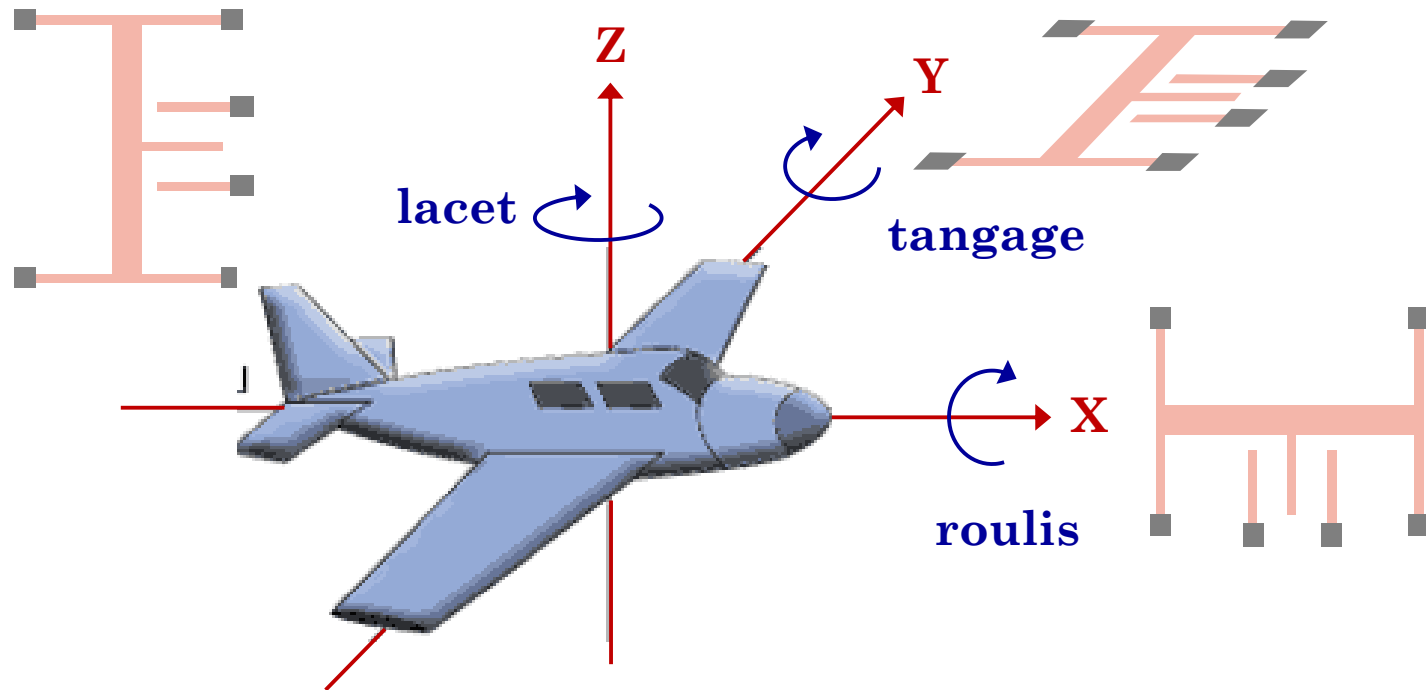
5.1. Définition (wikipedia)

- On définit à présent un repère orthonormé X, Y, Z

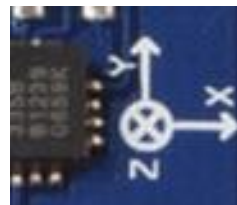


5.2. Agencement des accéléromètres

- Un accéléromètre est positionné dans chacun des plans X, Y et Z



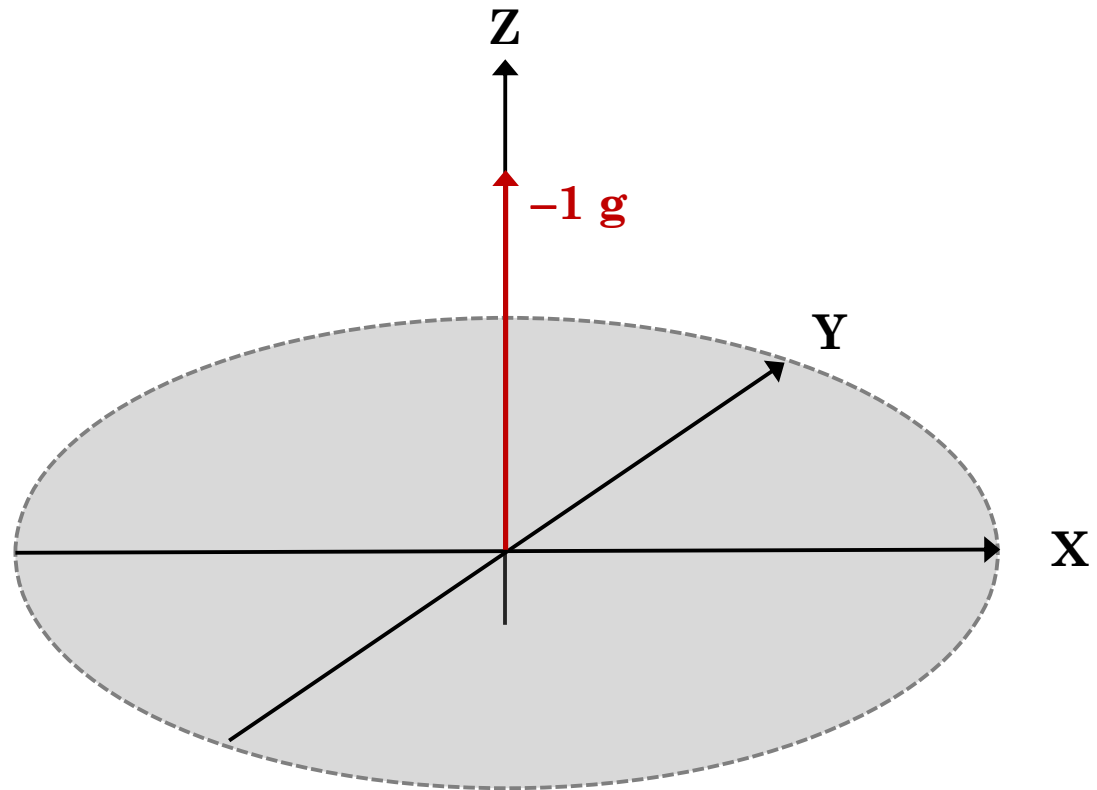
- Les axes sont indiqués sur les modules :



5.3. Les 3 angles

□ Plan de référence

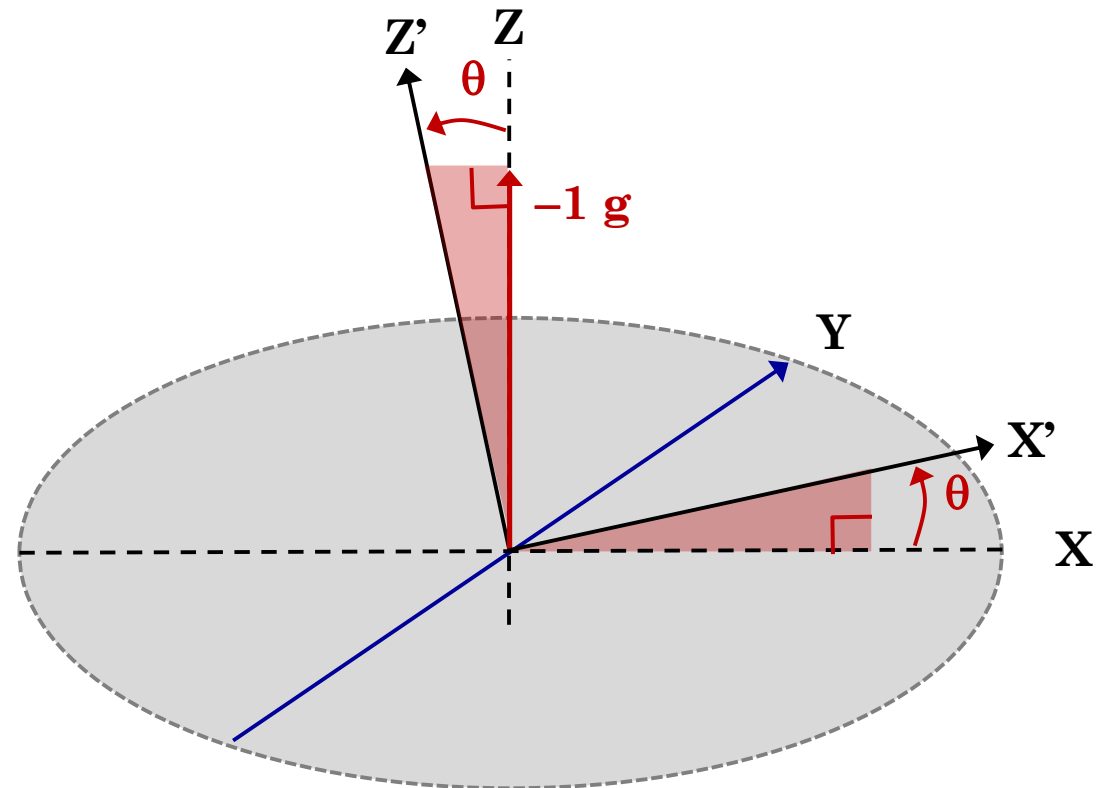
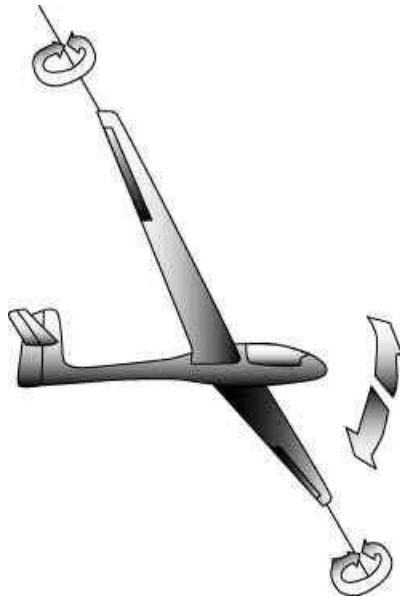
- L'accéléromètre en Z' est le seul à détecter la gravité
- Le cercle gris du plan XY nous servira de référence



5.3. Les 3 angles

□ Angle de tangage : θ

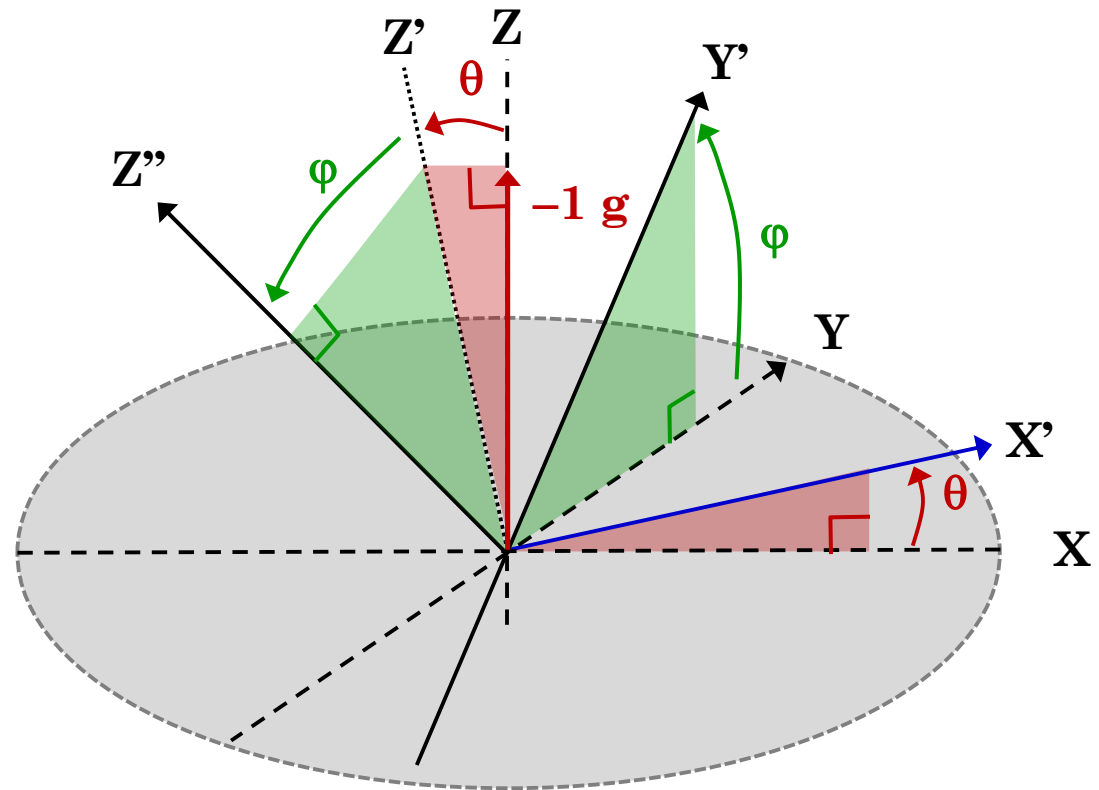
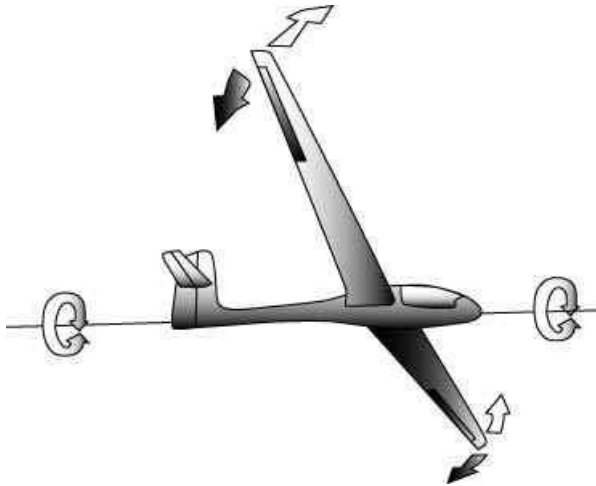
- On fixe l'axe Y et on effectue une rotation du plan XZ
- L'angle entre l'axe X' et le plan XY est appelé angle de tangage (θ)



5.3. Les 3 angles

□ Angle de roulis : φ

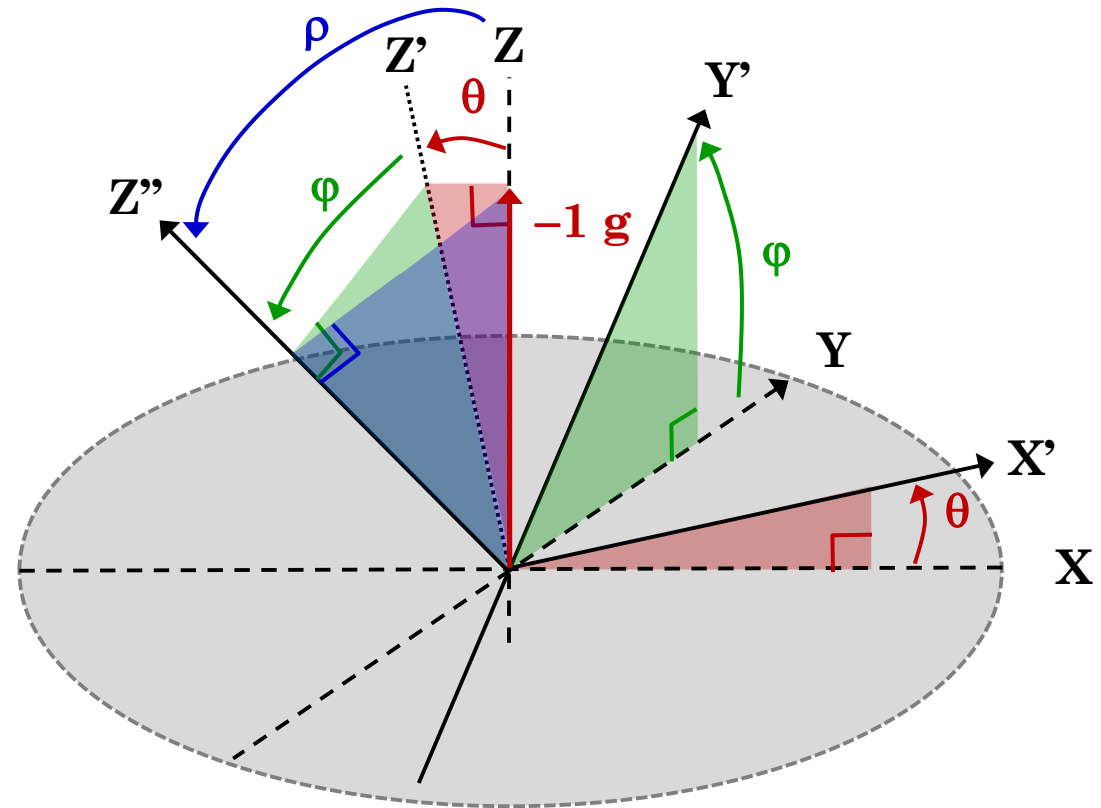
- On fixe l'axe X' et on effectue une rotation du plan $Y'Z'$
- L'angle entre l'axe Y' et le plan XY initial, est appelé angle de roulis (φ)



5.3. Les 3 angles

□ Angle ρ

- ρ est l'angle entre la gravité et l'axe Z''



5.4. Projection de la gravité sur les axes

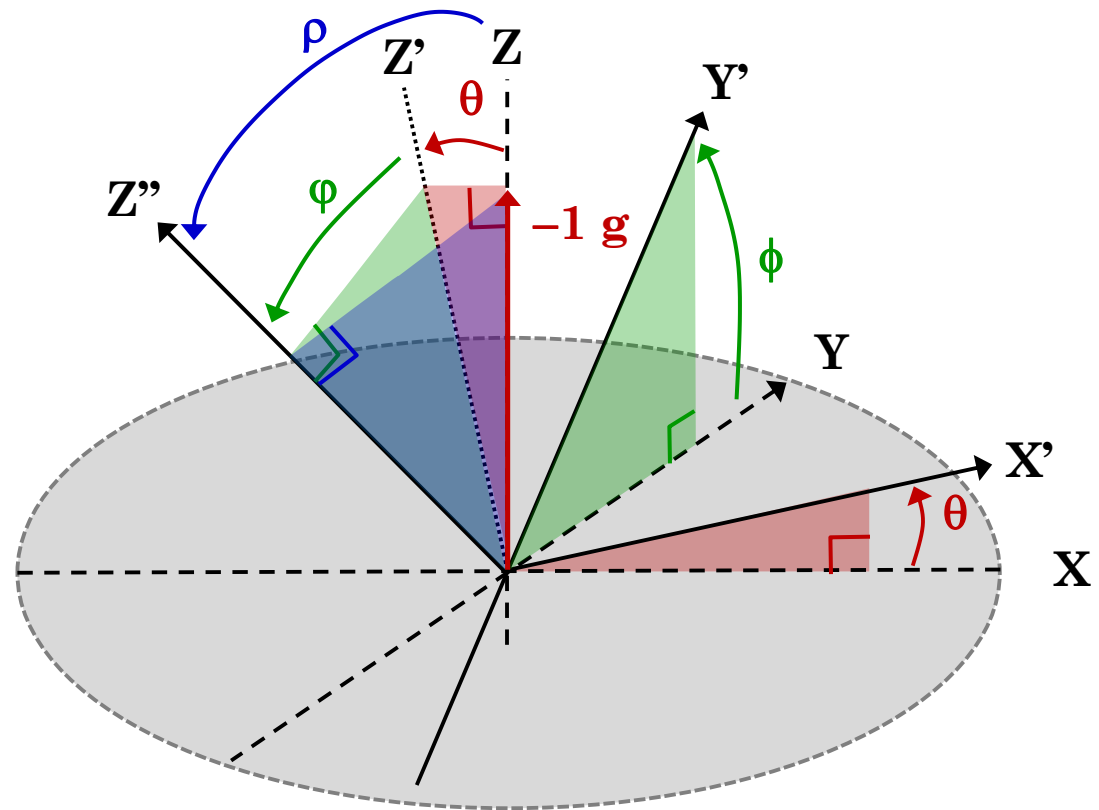
□ Expressions des 3 angles

- D'habiles calculs permettent de trouver :

$$\theta = \arctan\left(\frac{A_X}{\sqrt{A_Y^2 + A_Z^2}}\right)$$

$$\varphi = \arctan\left(\frac{A_Y}{\sqrt{A_X^2 + A_Z^2}}\right)$$

$$\rho = \arctan\left(\frac{\sqrt{A_X^2 + A_Y^2}}{A_Z}\right)$$



6.1. Rappel des équations

- Nous nous plaçons ici dans une seule dimension, X.
- L'accélération s'écrit : $a(t) = \frac{dv(t)}{dt}$
- Ce qui donne pour la vitesse : $v(t) = \int a(t).dt$
- On trouve au final la position du mobile : $x(t) = \int v(t).dt = \iint a(t).dt$

6.2. En échantillonné

- A $t_{n=0} = 0$:
 - ✓ $n (= 0)$ est un entier qui correspond au numéro de l'échantillon
 - ✓ La position du mobile est $x = x_0$
 - ✓ La vitesse du mobile est $v = v_0$
 - ✓ L'accélération du mobile est $a = a_0$

- A $t_{n=1} = \Delta t_n$:
 - ✓ L'accélération du mobile est $a_{n=1}$
 - ✓ La vitesse du mobile est $v_n = a_n \cdot \Delta t_n + v_{n-1}$
 - ✓ La position du mobile est $x_n = v_n \cdot \Delta t_n + x_{n-1} = a_n \cdot \Delta t_n^2 + v_{n-1} \cdot \Delta t_n + x_{n-1}$

6.3. En pratique

- Le bruit de mesure entraine une imprécision très importante sur le vitesse et encore plus sur la position puisqu'elle résulte d'une deuxième intégrale
- De plus les chocs plus ou moins importants lors du déplacement ajoute des sauts d'accélérations ce qui altère d'autant plus l'estimation de la position

7.1. Présentation

- L'accéléromètre GY-61 est constitué du module ADXL335 qui est un triple accéléromètre analogique c'est-à-dire qu'il donne une tension continue pour chaque axe
- Chaque axe peut mesurer une accélération de $\pm 3g$
- Le coût du GY-61 est d'environ 4 €



Small, Low Power, 3-Axis $\pm 3g$
Accelerometer
ADXL335

7.2. Datasheet

- La tension à 0 g est « typiquement » de 1.5 V mais peut aller de 1.35 V à 1.65 V pour les axes X et Y
- De même la sensibilité va de 270 à 330 mV/g
- Il est donc nécessaire de faire un étalonnage de l'accéléromètre en mesurant les valeurs de tension pour des angles connus

Parameter	Conditions	Min	Typ	Max	Unit
SENSOR INPUT	Each axis				
Measurement Range		±3	±3.6		g
Nonlinearity	% of full scale		±0.3		%
Package Alignment Error			±1		Degrees
Interaxis Alignment Error			±0.1		Degrees
Cross-Axis Sensitivity ¹			±1		%
SENSITIVITY (RATIOMETRIC) ²	Each axis				
Sensitivity at X _{OUT} , Y _{OUT} , Z _{OUT}	V _S = 3 V	270	300	330	mV/g
Sensitivity Change Due to Temperature ³	V _S = 3 V		±0.01		%/°C
ZERO g BIAS LEVEL (RATIOMETRIC)					
0 g Voltage at X _{OUT} , Y _{OUT}	V _S = 3 V	1.35	1.5	1.65	V
0 g Voltage at Z _{OUT}	V _S = 3 V	1.2	1.5	1.8	V
0 g Offset vs. Temperature			±1		mg/°C

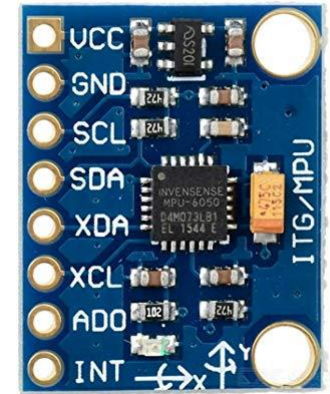
7.3. Exemple de résultats

- La tension à 0 g est « typiquement » de 1.5 V mais peut aller de 1.35 V à 1.65 V pour les axes X et Y

EN COURS D'ECRITURE

8.1. Présentation

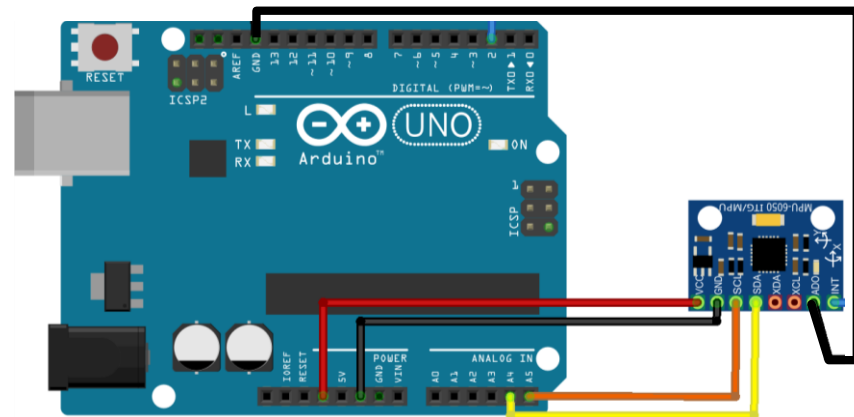
- L'accéléromètre 3 axes GY-521 est basé sur le MEMS MPU-6050 qui contient aussi un gyroscope 3 axes
- Le signal des capteurs est numérisé par un processeur de mouvement
- Il contient aussi un capteur de température pour corriger les dérives
- On peut aussi régler la plage de mesures : ± 2 , ± 4 , ± 8 , ± 16 g
- Les mesures des capteurs sont obtenues via le bus I²C
- L'adresse I²C du MPU-6050 est 0x68 lorsque AD0 = 0 V et elle devient 0x69 lorsque AD0 = 5 V
- Les données et les informations sur le fonctionnement sont contenues dans des registres de 8 bits



8.2. Mise en œuvre

□ Exemple de montage

- Le montage est simple puisqu'il n'utilise que l'alimentation + 5 V de l'arduino et les connections SCL et SDA du bus I²C
- SDA pour Serial DAta (I/O n° A5 de l'arduino)
- SCL pour Serial CLock (I/O n° A4 de l'arduino)



8.2. Mise en œuvre

❑ Déclaration de l'accéléromètre

- Il faut commencer par indiquer à la carte que l'on communique en I²C en intégrant la librairie « wire »

```
#include <Wire.h>
```

- Cela rend inutilisables les I/O A4 et A5 pour lire des valeurs analogiques
- L'adresse I²C du MPU-6050 est 0x68 (à changer s'il y a plusieurs accéléromètres dans le projet) et on déclare une constante en conséquence :

```
const int MPU_addr = 0x68;
```

8.2. Mise en œuvre

□ Réveil du MPU-6050

- Avant tout démarrage de communication I²C, il faut commencer par :

```
Wire.begin();
```

- On indique alors que l'on va communiquer avec l'accéléromètre

```
Wire.beginTransmission(MPU_addr);
```

- Par défaut, le MPU-6050 est en veille et on le réveille en écrivant 0 dans le registre 0x6B

```
Wire.write(0x6B);
```

```
Wire.write(0);
```

- Finalement on arrête la communication I²C pour libérer le BUS

```
Wire.endTransmission();
```

8.2. Mise en œuvre

❑ Lecture des données

- On définit les variables où seront stockées les 3 accélérations

```
float Acc_X, Acc_Y, Acc_Z;
```

- La datasheet du circuit MPU-6050 donne l'adresse des registres où sont stockées les données.
- Chaque valeur est stockée dans 2 registres un pour les bits de poids faibles et un autre pour les poids forts. Par exemple 3C et 3B respectivement pour l'accélération suivant l'axe X

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3B	59	ACCEL_XOUT[15:8]							
3C	60	ACCEL_XOUT[7:0]							
3D	61	ACCEL_YOUT[15:8]							
3E	62	ACCEL_YOUT[7:0]							
3F	63	ACCEL_ZOUT[15:8]							
40	64	ACCEL_ZOUT[7:0]							

8.2. Mise en œuvre

❑ Lecture des données

- On demande l'accès au premier registre de la liste :

```
Wire.beginTransmission(0x68);
```

```
Wire.write(0x3B);
```

- À noter que le protocole I²C propose une condition particulière nommée RESTART pour signifier le début d'une nouvelle trame dès la fin de la trame précédente. On ajoute cette ligne :

```
Wire.endTransmission(false);
```

- On indique alors à l'accéléromètre que l'on va lire 6 registres donc 6 octets (à partir du registre 0x3B)

```
Wire.requestFrom(0x68,6,true);
```

8.2. Mise en œuvre

□ Lecture des données

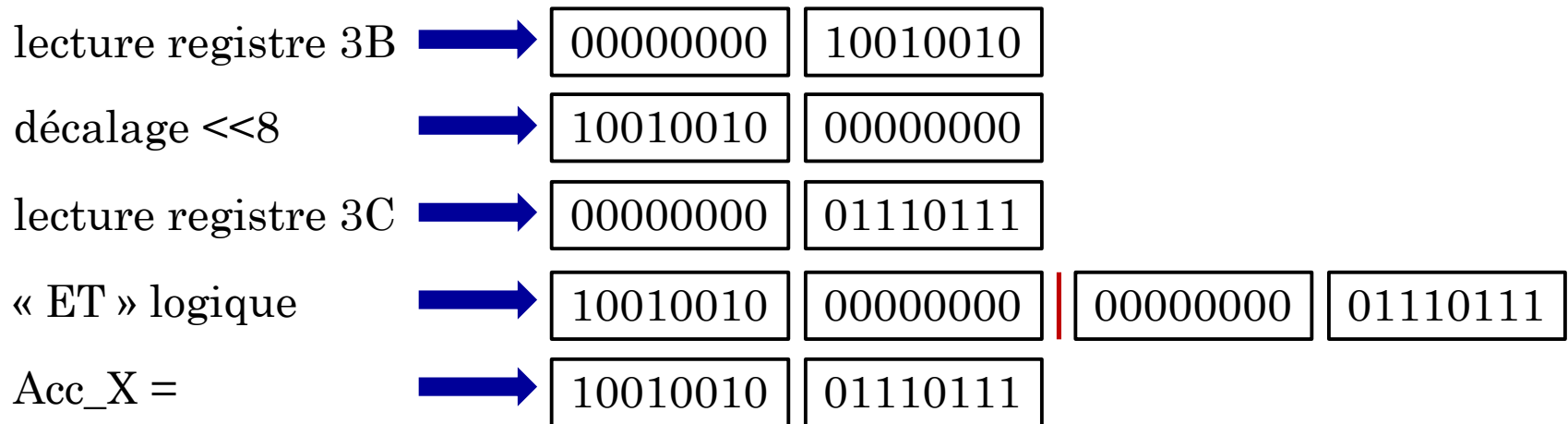
- Le chargement des données se fait comme suit :

```
Acc_X=(Wire.read()<<8 | Wire.read());
```

```
Acc_Y=(Wire.read()<<8 | Wire.read());
```

```
Acc_Z=(Wire.read()<<8 | Wire.read());
```

- Voici l'explication pour la première ligne :



8.2. Mise en œuvre

□ Réglage de la gamme de mesure (et de la sensibilité)

- La définition de la gamme de mesure se trouve dans le registre 0x1C et il existe 4 gammes ± 2 , ± 4 , ± 8 et ± 16 g
- La datasheet nous indique que ce sont les bits 3 et 4 qu'il faut modifier pour changer la gamme. Les bits 0 à 2 ne sont pas utilisés et les bits 5 à 7 informent s'il y a un self test

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1C	28	XA_ST	YA_ST	ZA_ST	AFS_SEL[1:0]		-		

- Pour les 4 gammes il faut envoyer le mot binaire :

$$\pm 2 \text{ g} = \text{B00000000}$$

$$\pm 4 \text{ g} = \text{B00001000}$$

$$\pm 8 \text{ g} = \text{B00010000}$$

$$\pm 16 \text{ g} = \text{B00011000}$$

8.2. Mise en œuvre

□ Réglage de la gamme de mesure (et de la sensibilité)

- Le changement de la gamme de mesure modifie la sensibilité (par g) qui a pour valeurs :

$$\pm 2 \text{ g} = 16384/\text{g}$$

$$\pm 4 \text{ g} = 8192/\text{g}$$

$$\pm 8 \text{ g} = 4096/\text{g}$$

$$\pm 16 \text{ g} = 2048/\text{g}$$

- Voici le code pour travailler dans la gamme $\pm 2 \text{ g}$:

```
Wire.beginTransmission(0x68);
```

```
Wire.write(0x1c);
```

```
Wire.write(B00000000);
```

```
Wire.endTransmission();
```

8.2. Mise en œuvre

□ Réglage de la gamme de mesure (et de la sensibilité)

- Pour finir, on peut prendre en compte la sensibilité pour obtenir la valeur en g sur les 3 axes. Par exemple pour la gamme ± 2 g

```
Acc_X=(Wire.read()<<8 | Wire.read())/16384;
```

```
Acc_Y=(Wire.read()<<8 | Wire.read())/16384;
```

```
Acc_Z=(Wire.read()<<8 | Wire.read())/16384;
```

8.2. Mise en œuvre

□ Détermination des angles

- Les angles suivant les axes X et Y s'écrivent :

`angle_x = (atan((Acc_X) / sqrt(pow((Acc_Y),2) + pow((Acc_Z),2))) * rad_deg);`

`angle_y = (atan((Acc_Y) / sqrt(pow((Acc_X),2) + pow((Acc_Z),2))) * rad_deg);`

- où « `atan` » correspond à l'arc tangente, « `sqrt` » à la racine carrée et « `pow` » au calcul d'une puissance
- « `rad_deg` » correspond à la conversion des radians en degrés

8.2. Mise en œuvre

□ Détermination des angles

- Les angles suivant les axes X et Y s'écrivent :

EN COURS D'ECRITURE