

# Chapitre 3 Capteurs inertIELS

---

## Table des matières

|  |    |
|--|----|
| <b>Introduction .....</b>  | 33 |
| <b>I. Accéléromètre .....</b>                                    | 33 |
| 1. Définitions .....   | 33 |
| 2. Principe d'un ACCÉLÉROMÈTRE MEMS DE LA SÉRIE ADXL .....       | 33 |
| 3. Orientation .....   | 44 |
| 4. Mesure des angles à 3 axes cas Tangage, roulis et lacet ..... | 50 |
| 5. Déplacement rectiligne .....                                  | 52 |
| 6. Programmation .....   | 53 |
| <b>II. Gyroscope.....</b>  | 60 |
| 1. Définition .....  | 60 |
| 2. Principe Gyromètre MEMS .....                                 | 60 |
| 3. Détermination de l'orientation .....                          | 61 |
| 4. Programmation .....   | 62 |
| 5. Applications .....  | 65 |
| <b>III. Capteur inertiel.....</b>                                | 65 |
| 1. Définition .....  | 65 |
| 2. Fusion des capteurs .....                                     | 66 |
| 3. Programmation .....   | 70 |
| 4. Digital Motion Processor (DMP) .....                          | 72 |

# Introduction

Ce chapitre est composé en trois parties : les accéléromètres, les gyromètres et les capteurs inertIELS

## I.Accéléromètre

### 1. Définitions

- Accélération est une grandeur physique vectorielle utilisée pour représenter la modification de la vitesse d'un mouvement en fonction du temps
- Formule de l'accélération : correspond à la dérivée temporelle de la vitesse « v » :  $\vec{a} = \frac{d\vec{v}}{dt}$
- Unité : L'accélération est définie en m/s<sup>2</sup> (SI) et la majorité des documentations sur ces capteurs expriment l'accélération en " g " (= 9,80665 m/s<sup>2</sup>).
- 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  $\frac{100 \text{ km/h}}{5 \text{ s}} = 20 \left( \frac{\text{km}}{\text{h}} / \text{s} \right) = 5,6 \text{ m/s}^2 = 0,56g$
  - ✓ 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  $\frac{-30 \text{ km/h}}{0,1 \text{ s}} = -300 \left( \frac{\text{km}}{\text{h}} / \text{s} \right) = -83 \text{ m/s}^2 = -8,3g$

### 2. Principe d'un ACCÉLÉROMÈTRE MEMS DE LA SÉRIE ADXL

#### 2.1. Définition d'un accéléromètre

Les accéléromètres sont des capteurs mesurant des accélérations (m/s<sup>2</sup>). Ils peuvent être mono, bi ou triaxial. Nous pouvons relier cette grandeur physique à une force (Newton ou en g) grâce au principe fondamentale de la dynamique.

***Enoncé du principe fondamental de la dynamique*** L'accélération subie par ce corps dans un référentiel galiléen est proportionnelle à la résultante des forces qu'il subit, et inversement proportionnelle à sa masse  $m$ .       $\sum \vec{F}_t = m \vec{a}$

Parmi les accéléromètres on trouve ***Les accéléromètres de type MEMS***

#### 2.2. Définition d'un MEMS

MEMS est l'abréviation de 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.

### 2.3. Définition d'un accéléromètre MEMS

Il composé de deux peignes complémentaires conducteurs électriques en Silicium. L'un est fixe. L'autre est mobile, suspendu à une lamelle flexible, et constitue une masse sensible à l'accélération. Le peigne mobile peut se déplacer d'une vingtaine de nanomètre par inertie lors d'un changement vitesse.

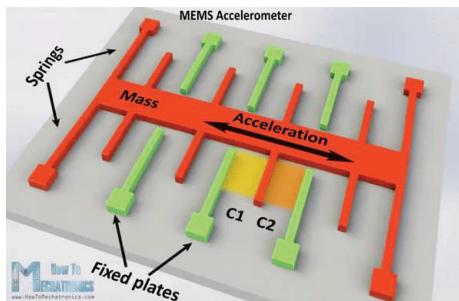


Figure 1

### 2.4. Miniaturisation des capteurs et richesse fonctionnelle

Cette partie traitera des séries d'accéléromètres ADXL. En 1980, Les premiers occupaient une carte d'environ  $50 \text{ cm}^2$ . Dès 1995 la partie active d'un ADXL50 (les applications d'airbag) était réduite à un carré de  $16 \text{ mm}^2$ . Le composant sur lequel le capteur est basée cette étude date de 2006 occupe seulement  $3 \text{ mm}^2$ , comme les composants plus récents. Selon leur étendue de mesures, ils sont qualifiés de Low-g (ADXL103) pour des accélérations inférieures à vingt fois l'accélération de la pesanteur et de high-g (ADXL78) au-delà. D'autre série ADIS exploite des systèmes tout numériques programmables plus sophistiqués, incluant du traitement et du contrôle, de l'interfaçage, des auto-tests, la gestion de la consommation, etc. On peut considérer ces capteurs comme « intelligents ». Dans la série ADXL, le capteur MEMS et son électronique de conditionnement sont intégrés dans le même composant.



Figure 2 Capteur de la série ADXL

### 2.5. Réalisation du capteur

Le capteur est réalisé à la surface d'une tranche de silicium (« wafer ») à l'aide de techniques spécifiques, tels la photolithographie et le micro-usinage de couches minces. Après réalisation, on observe par microscopie électronique que la surface du composant présente des micro-structures en silicium telles des micro-poutres, des microlamelles, etc. Les dimensions typiques des éléments de ces structures sont de 1 à 100  $\mu\text{m}$ .

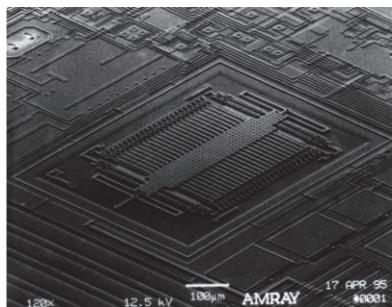


Figure 3 Vue générale du capteur MEMS d'un accéléromètre ADXL50 high-g un axe.

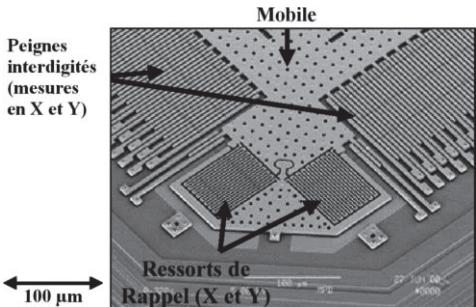


Figure 4 Détails d'un accéléromètre type ADXL202 low-g deux axes

Ces photographies montrent la « masse sismique » (beam) dont on étudie le mouvement et qui subit globalement les effets de l'accélération à mesurer. Celle-ci est reliée « au bâti » par des micro-poutres d'ancrage (anchor) pour un système un axe ou des micro-ressorts positionnés dans les coins pour un système deux axes. Ces parties flexibles gouvernent le mouvement du mobile.

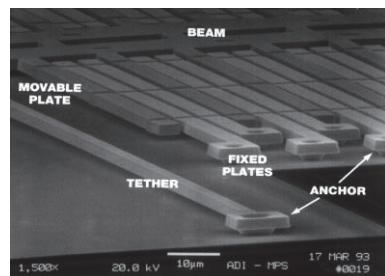


Figure 5 Détails du capteur MEMS d'un ADXL50 originel

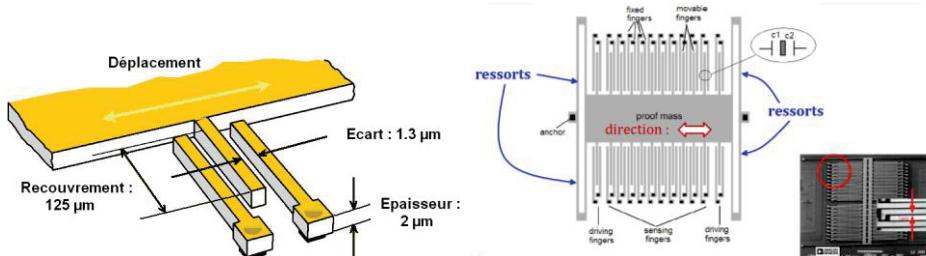


Figure 6 Description des lamelles associées à un doigt d'un capteur actuel.

Le suivi du déplacement est assuré par un capteur capacitif organisé autour de séries de lamelles. On qualifie d>IDT (pour InterDigital Transducer) cette structure particulière en forme de « peignes » digités interpénétrés. La multiplicité de ces lamelles améliore évidemment la sensibilité de la mesure.

## 2.6. Condensateur : géométrie et intérêt :

Un transducteur électrique est un condensateur dont au moins un des éléments (électrodes) est mobile, et donc réalise un lien direct du domaine mécanique (la coordonnée de la partie mobile) vers le domaine électrique (la valeur de la capacité).  $c = \epsilon \epsilon_0 \frac{s}{d}$

Avec  $\epsilon$  permittivité du vide,  $\epsilon_0$  est la constante diélectrique du matériau entre les électrodes,  $s$  est l'aire de recouvrement des électrodes et  $d$  est la distance entre les électrodes. Généralement, dans les MEMS,  $\epsilon \approx 1$ . Dans la formule de la capacité ci-dessus, il y a deux paramètres géométriques que l'on peut faire varier : l'aire de recouvrement et la distance entre les plans.

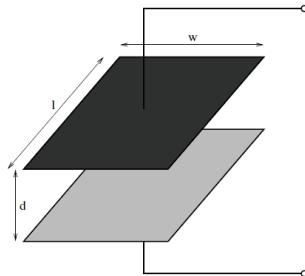


Figure 7 Condensateur plan

### Transducteur en mouvement vertical :

Soit un condensateur plan dont une électrode est fixée, l'autre est mobile et peut se déplacer uniquement selon l'axe perpendiculaire aux plans des électrodes, (Figure7). Dans ce cas, la distance entre les électrodes varie comme  $d-x$ , où  $d$  est la distance (gap) au repos.

La capacité s'exprime comme :  $c = \epsilon_0 \frac{s}{d-x}$

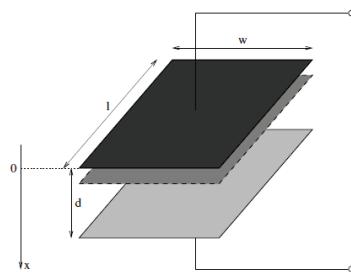


Figure 8 Condensateur en mouvement vertical

### Transducteur en mouvement latéral :

La variation se fait au niveau de S, en faisant glisser une électrode de sorte à ce qu'il reste dans son plan (figure 8). L'électrode a une forme rectangulaire, avec longueur  $l$  et largeur  $w$ , et si le mouvement de l'électrode mobile fait varier la largeur, on obtient :  $c = \epsilon_0 \frac{l(\omega - |x|)}{d}$

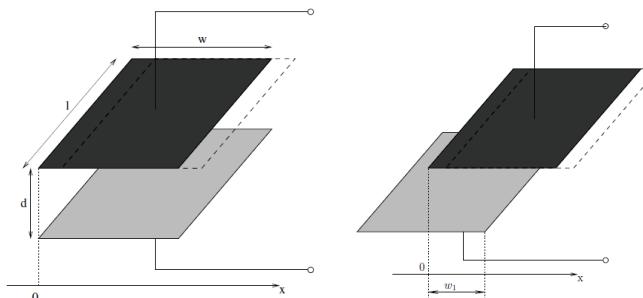


Figure 9

## 2.7. Transduction du domaine électrique vers le domaine mécanique : Étude mécanique du fonctionnement d'un capteur un axe

Le capteur différentiel est modélisé par un simple mobile de masse  $m$  posé sur un support horizontal et pouvant se déplacer le long de l'axe des  $x$ . Le mobile est relié au support par des systèmes ressorts/amortisseurs décrivant le comportement mécanique des éléments du MEMS

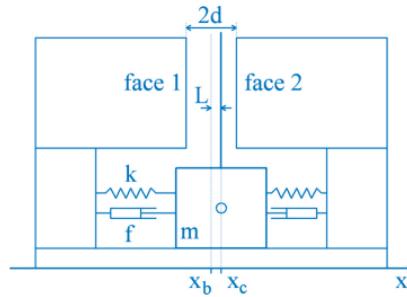


Figure 10 Modélisation pour le mouvement sur l'axe de  $x$  d'un élément mobile

on note  $k$  la raideur,  $f$  le coefficient de frottement,  $x_c$  la position du centre de masse du mobile en mouvement et  $x_b$  sa position au repos par rapport au support (c'est-à-dire en fait celle du support :  $x_b$  ne varie que si le support se déplace, c'est-à-dire si l'accéléromètre se déplace en bloc). le rôle de ce système est de permettre la mesure de cette accélération  $a(t)$ .

Dans ce modèle, il n'y a pas de frottement entre la partie horizontale du support et le bas de la partie mobile. Quand le « ressort 2 » est en compression (force de rappel  $T_2$ ), le « ressort 1 » est en extension (force de rappel  $T_1$ ) et réciproquement, ce avec proportionnalité à l'élongation  $L$ . Les forces de freinage  $F_1$  et  $F_2$  sont similaires dans les deux cas et proportionnelles à la vitesse relative du mobile par rapport au « bâti » (c'est-à-dire la dérivée temporelle de  $L$ ).

L'équation du mouvement donne  $L = x_c - x_b$  et l'accélération est obtenue en appliquant le principe fondamental de la mécanique dans le référentiel galiléen du sol :

$$m\vec{a}_c = \sum \overrightarrow{F_{appliquées}}$$

En projection sur l'horizontale (axe  $x$ ) :

$$m \frac{d^2x_c}{dt^2} = -2k(x_c - x_b) - 2f \frac{d(x_c - x_b)}{dt}$$

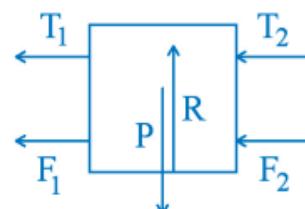
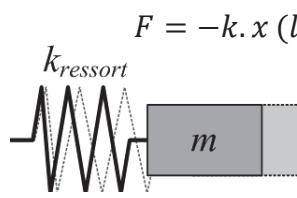


Figure 11

Avec  $x_c = (x_C - x_b) + x_b = L + x_b$

$$m \cdot \frac{d^2L}{dt^2} + m \cdot \frac{d^2x_b}{dt^2} = -2k \cdot L - 2f \frac{dL}{dt} \text{ Où } \frac{d^2x_b}{dt^2} = a(t)$$

$$m \cdot \frac{d^2L}{dt^2} = -2k \cdot L - 2f \frac{dL}{dt} - m \cdot a(t)$$

On obtient donc l'équation de mouvement en élongation :

$$\frac{d^2L}{dt^2} + \frac{2f}{m} \frac{dL}{dt} + \frac{2k}{m} \cdot L = -a(t)$$

L'analyse harmonique de la partie mécanique du capteur se fait en transformant  $\frac{d}{dt} \rightarrow j\omega$  :

$$-\omega^2 \bar{L} + j\omega \frac{2f}{m} \bar{L} + \frac{2k}{m} \cdot \bar{L} = -\bar{a} \Rightarrow \left( -\omega^2 + j\omega \frac{2f}{m} + \frac{2k}{m} \right) \cdot \bar{L} = -\bar{a}$$

$$\frac{\bar{L}}{\bar{a}} = -\frac{1}{-\omega^2 + j\omega \frac{2f}{m} + \frac{2k}{m}}$$

On pose la pulsation caractéristique  $\omega_0^2 = \frac{2k}{m}$  et le paramètre d'amortissement  $\mu = \frac{f}{\sqrt{2km}}$

$$\frac{\bar{L}}{\bar{a}} = -\frac{\frac{1}{\omega_0^2}}{-\frac{\omega^2}{\omega_0^2} + 2j\mu \frac{\omega}{\omega_0} + 1}$$

La fonction de transfert  $\frac{\bar{L}}{\bar{a}}$  est de type passe-bas du second ordre.

Mesurer le déplacement  $L$  des lamelles du capteur différentiel revient donc à mesurer l'accélération du support, pour  $\omega \ll \omega_0$ . On a, en régime établi :

$$L \approx -\frac{m}{2k} a(t)$$

Les feuilles de données des deux capteurs pris comme exemples indiquent une « résonance mécanique » à 55 khz pour ADXL103 et 24 khz pour ADXL78.

### ➤ Étude du conditionnement électronique du capteur

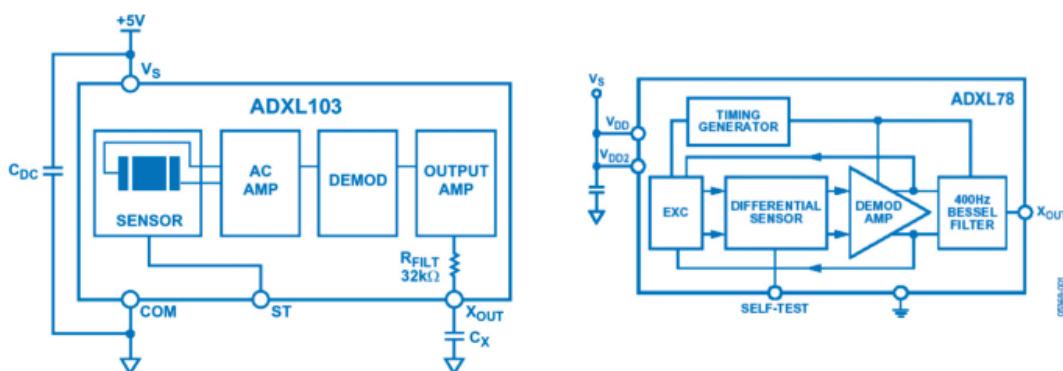


Figure 12 Schéma-bloc des accéléromètres ADXL103 et ADXL78.

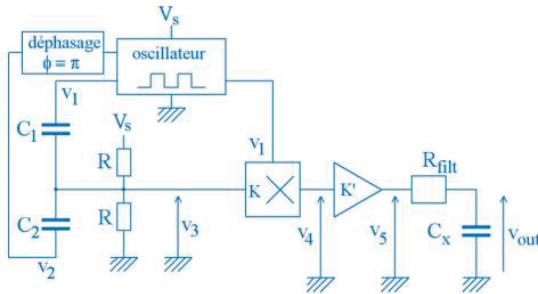


Figure 13 Schéma d'interprétation détaillé selon le fichier de données ADXL103.

Le composant est alimenté de façon asymétrique entre 0 V et + 5 V : les tensions intervenant dans le fonctionnement du circuit devront donc être positives.

#### - Transduction du déplacement par un pont capacitif en modulation

Le mobile est solidaire d'une lame pouvant donc se déplacer entre les faces 1 et 2. Les paires de faces en regard définissent deux condensateurs  $C_1$  et  $C_2$  de capacités dépendant de  $L$ . L'oscillateur génère un signal carré d'horloge à une fréquence  $f_h$ , représentée par la tension  $v_1(t)$ . D'origine numérique, il est positif. De niveau haut  $V_s$  et de niveau bas 0 V, il a une moyenne égale à  $\frac{V_s}{2}$ .  $v_1 = \frac{V_s}{2} + V_1 \sin(\omega t)$  et  $v_2 = \frac{V_s}{2} - V_1 \sin(\omega t)$

En effet plus loin, on verra que la dernière étape est un filtrage passe-bas qui élimine toutes les composantes de pulsation supérieure ou égale à  $\omega_0$ . Les autres harmoniques ne donneraient donc que des termes correctifs qui seraient éliminés dans le calcul.

Le pont formé par les deux capacités est ainsi alimenté par deux tensions dont les parties variables sont en opposition de phase.

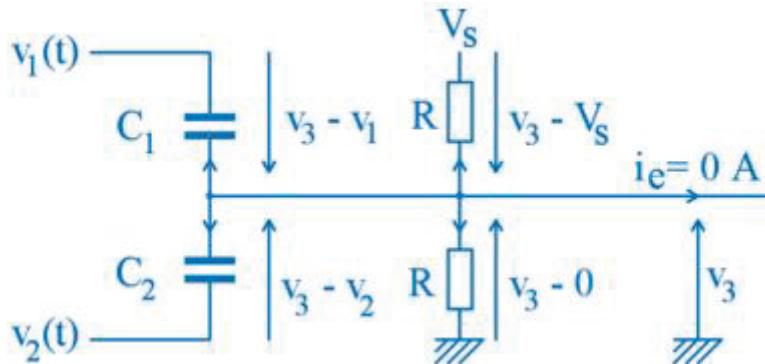


Figure 14

on suppose que le courant d'entrée du multiplicateur d'amplification  $K$  est négligeable du fait d'une forte impédance d'entrée, soit  $i_e = 0A$ .

Les deux résistances  $R$  forment un pont de polarisation et ont pour objet d'établir une tension de mode commun  $\frac{V_s}{2}$  nécessaire par l'emploi d'une alimentation asymétrique.

Afin de calculer la tension issue du pont, on applique la loi des nœuds sur les cinq branches :

$$c_1 \frac{d}{dt} (v_3 - v_1) + c_2 \frac{d}{dt} (v_3 - v_2) + \frac{v_3 - V_s}{R} + \frac{v_3 - 0}{R} = 0$$

en tenant compte de  $i_e = 0A$ . On remplace et on arrange :

$$(c_1 + c_2) \frac{d}{dt} (v_3) + \frac{2}{R} v_3 = c_1 \frac{d}{dt} \left( \frac{V_s}{2} + V_1 \sin(\omega t) \right) + c_2 \frac{d}{dt} \left( \frac{V_s}{2} - V_1 \sin(\omega t) \right) + \frac{V_s}{R}$$

$$(c_1 + c_2) \frac{d}{dt} (v_3) + \frac{2}{R} v_3 = (c_1 - c_2) V_1 \frac{d}{dt} (\sin(\omega t)) + \frac{V_s}{R}$$

$$\frac{d}{dt} (v_3) + \frac{2}{R(c_1 + c_2)} v_3 = \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \frac{d}{dt} (\sin(\omega t)) + \frac{V_s}{R(c_1 + c_2)}$$

En définissant la constante de temps  $\tau = \frac{R}{2}(c_1 + c_2)$

$$\frac{d}{dt} (v_3) + \frac{v_3}{\tau} = \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \frac{d}{dt} (\sin(\omega t)) + \frac{V_s}{2\tau}$$

La solution est de la forme :

$$v_3(t) = A \exp \left( -\frac{t}{\tau} \right) + B \sin(\omega t + \varphi) + \frac{V_s}{2}$$

Le premier terme est la réponse libre de l'équation avec second membre  $\frac{V_s}{2}$ . L'autre terme est une solution particulière. Le théorème de superposition permet de rechercher cette solution en distinguant chacun des termes forçant.

$$v'_3(t) = A \exp \left( -\frac{t}{\tau} \right) + \frac{V_s}{2}$$

La solution particulière harmonique s'obtient directement en calcul complexe :

$$\frac{d}{dt} (v_3) + \frac{v_3}{\tau} = \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \frac{d}{dt} (\sin(\omega t))$$

$$j\omega \bar{v}_3 + \frac{1}{\tau} \bar{v}_3 = \frac{(c_1 - c_2)}{(c_1 + c_2)} j\omega V_1 e^{j\omega t} \rightarrow \bar{v''}_3 = \frac{j\omega \tau}{1 + j\omega \tau} \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 e^{j\omega t}$$

Le temps de réaction pour un système de sécurité automobile est 5 ms. Il est typiquement estimé par un temps de montée à 5 %, soit pour un premier ordre ;  $t_{r5\%} = 3\tau \approx 5ms \rightarrow \tau \approx 1.5ms$

Les fréquences de modulation sont en centaine de khz (140 khz pour ADXL103 et 400 khz pour ADXL78). on constate donc que ou plus et donc que de façon générale  $\omega \gg \frac{1}{\tau}$ . Donc :

$$\left| \frac{j\omega \tau}{1 + j\omega \tau} \right| \rightarrow 1 \text{ et } \arg \left( \frac{j\omega \tau}{1 + j\omega \tau} \right) = \frac{\pi}{2} - \arctan(j\omega \tau) \rightarrow 0$$

$$v''_3(t) = \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \sin(\omega t)$$

$$v_3(t) = v'_3(t) + v''_3(t) = A \exp \left( -\frac{t}{\tau} \right) + \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \sin(\omega t) + \frac{V_s}{2}$$

La détermination de constante  $A$  n'a pas d'importance, car après au plus 5 ms, le régime transitoire est terminé car le terme exponentiel décroît rapidement vers 0. On peut obtenir :

$$v_3(t) = \frac{V_S}{2} + \frac{(c_1 - c_2)}{(c_1 + c_2)} V_1 \sin(\omega t)$$

### - Relation entre les capacités et le déplacement L

Au vu de la photographie des lamelles, on suppose que les condensateurs  $c_1$  et  $c_2$  peuvent être assimilés à des condensateurs plans :  $c \approx \frac{\epsilon S}{e}$  avec  $e_1 = d + L$  et  $e_2 = d - L$

$$\text{D'où } C_1 \approx \frac{\epsilon S}{d+L} \text{ et } C_2 \approx \frac{\epsilon S}{d-L} \text{ D'où } \frac{(c_1 - c_2)}{(c_1 + c_2)} = \frac{\left(\frac{\epsilon S}{d+L} - \frac{\epsilon S}{d-L}\right)}{\left(\frac{\epsilon S}{d+L} + \frac{\epsilon S}{d-L}\right)} = \frac{(d-L)-(d+L)}{(d-L)+(d+L)} = \frac{-2L}{2d} = \frac{-L}{d}$$

D'où l'expression de  $v_3$  en fonction du déplacement  $L$  :

$$v_3(t) = \frac{V_S}{2} - \frac{L}{d} V_1 \sin(\omega t)$$

### - Conditionnement du pont capacitif

La démodulation synchrone du signal issu du pont consiste en une multiplication par le signal modulant le pont capacitif, suivie d'un filtrage passe-bas adéquat.

Soit  $v_4$  issue de la multiplication de  $v_3$  par  $v_1$ , avec le facteur de multiplication  $K$  (en  $V^{-1}$ ) :

$$v_4(t) = K v_3 v_1 = K \left( \frac{V_S}{2} - \frac{L}{d} V_1 \sin(\omega t) \right) \times \left( \frac{V_S}{2} + V_1 \sin(\omega t) \right)$$

$$v_4(t) = K \left( \left( \frac{V_S}{2} \right)^2 - \frac{V_S L}{2d} V_1 \sin(\omega t) + \frac{V_S}{2} V_1 \sin(\omega t) - \frac{L}{d} V_1^2 \sin^2(\omega t) \right)$$

Or  $\sin^2(\omega t) = \frac{1}{2}(1 - \cos(2\omega t))$  fait apparaître un second terme constant

La tension  $v_4$  peut donc se mettre sous la forme d'un terme constant global et d'un terme oscillant composite. De plus, la sortie de l'accéléromètre doit être capable de fournir une intensité suffisante à sa charge. Il convient donc de placer en sortie un amplificateur « tampon » d'amplification  $K'$  pour élaborer le signal à filtrer. On détermine donc la tension :

$$v_5(t) = K' v_4(t)$$

$$v_5(t) = K K' \left( \left( \frac{V_S}{2} \right)^2 - \frac{L}{2d} V_1^2 \right) + K K' \left( \frac{V_S}{2} \left( 1 - \frac{L}{2d} \right) V_1 \sin(\omega t) + \frac{L}{2d} V_1^2 \cos(2\omega t) \right)$$

Pour rendre le signal constant, il reste encore à éliminer le terme oscillant par un filtrage passe-bas convenable. Le filtrage passe-bas de la démodulation synchrone est assuré par le couple  $R_{filt}$  et  $C_x$  qui forme une cellule du premier ordre dont la fréquence de coupure est

$$f_c = \frac{1}{2\pi R_{filt} C_x} = \frac{1}{2\pi \times 32 \times 10^3 \times 0.1 \times 10^{-6}} \approx 400Hz$$

$C_x = 12.5nF \rightarrow$  Soit un temps caractéristique  $R_{filt} C_x \approx 0.4ms$

Par exemple, pour ADXL103, le signal correspondant au terme en  $\omega$  a une fréquence  $f$  de 140 khz, c'est-à-dire fois  $\frac{140 \times 10^3}{400} = 350$  fois plus que la fréquence de coupure  $f_c$ .  $\log_{10}(350) = 2.54$  décades et  $2.54 \times 20dB/dec = 51dB$ .  $10^{\frac{51}{20}} = 355$ .

Le filtrage passe-bas élimine bien la partie variable du signal, dépendant de  $\omega t$  ; en fait, il est conçu pour extraire la moyenne de la tension  $v_5$ . Donc :  $v_{out} = \langle v_5 \rangle = KK' \left(\frac{V_s}{2}\right)^2 - KK' \frac{m}{4kd} V_1^2 a(t)$

Le système fonctionne en capteur d'accélération, car  $L \approx -\frac{m}{2k} a(t)$  D'où

$$v_{out} = \langle v_5 \rangle = KK' \left(\frac{V_s}{2}\right)^2 + KK' \frac{m}{4kd} V_1^2 a(t)$$

#### - Tension de sortie et données numériques

De façon synthétique, la tension de sortie se met sous la forme générale :

$$v_{out} = V_0 + S \times a$$

La tension de sortie de l'accéléromètre se présente sous une forme affine où  $V_0$  est la tension « de repos » lorsque l'accélération à mesurer est nulle et  $S$  est la sensibilité de l'accéléromètre en  $V.m^{-1}.s^2$ .  $S' = S/g$  est la sensibilité en « V/g » (avec  $g = 9.8m.s^{-2}$ )

#### Exemple d'accéléromètre Low-g : ADXL103

ADXL103 est un accéléromètre Low-g destiné, en sécurité automobile, à participer au contrôle dynamique du véhicule VDC, au programme de gestion de la stabilité ESP, au contrôle électronique du freinage ABS, etc. Les spécifications précisent (pour  $v_s = 5V$ ) :

- tension de repos  $V_0 = 2.5V$  [Zero g bias voltage : 2,5 V typ (2,4 à 2,6 V)] ;
- la sensibilité  $S'$  du capteur est de 1 V/g [Sensitivity : 1000 mV/g typ (960 à 1040)] ;
- L'étendue de mesure minimale est  $[-1,7 g ; +1,7 g]$  permise par la tension de sortie maximale 4,5 V [Measurement range :  $\pm 1,7 g$  min ; output swing high : 4,5 V typ (max 4,8 V)];
- Une non-linéarité de  $\pm 0,2\%$  de la pleine échelle, typiquement ;
- la bande passante est comprise entre 0 hz et  $f_c$  qui dépend du choix de  $C_x$ .

Pour une capacité de filtrage  $C_x = 12.5nF$ ,  $f_c = 400Hz$  ce modèle permet de calculer :

$$a = 0 \rightarrow v_{out}(0) = KK' \left(\frac{V_s}{2}\right)^2 = V_0 = 2.5V$$

$$a = 1.7g \rightarrow v_{out}(1.7) = KK' \left(\frac{V_s}{2}\right)^2 + KK' \frac{m}{4kd} V_1^2 \times 1.7 = 2.5 + 1 \times 1.7 = 4.2V \leq 4.5V$$

Ces résultats ne sont évidemment compatibles avec les spécifications de ADXL103 qu'aux conditions suivantes  $V_0 = 2.5V$  et  $S' = 1V/g$ .

#### Exemple d'accéléromètre High-g

Un accéléromètre de type high-g, destiné à la détection d'un accident frontal (car-crash) et au déclenchement de l'airbag, est ADXL78 (iMEMS). Ses caractéristiques: étendue de mesure  $\pm 70$  g, sensibilité 27 mV/g, consommation 1,3 mA pour 5 V, modulation à 400 khz, filtrage à 400 hz. Le signal issu d'un accident peut contenir des composantes de fortes amplitudes et fréquences élevées. Ces composantes contiennent peu d'informations utiles et sont réduites par le filtre de sortie. Ces saturations peuvent intervenir sans pour autant gêner la détection de l'événement accidentel. On peut noter que le fonctionnement reste linéaire jusqu'à environ huit fois la valeur de pleine échelle, soit quand même 560 g.

#### 2.8. Applications

Les accéléromètres sont utilisés pour enregistrer à la fois les accélérations statiques (la gravité) et dynamique (choc, mouvement).

- Automobile : pour le déclenchement des airbags, la gestion du freinage ABS et la correction de trajectoire ESP,
- Détection Séisme
- Disque dur : un accéléromètre peut aussi être utilisé pour capter si un objet est en chute libre qui implique un choc. Ça permet de protéger des objets fragiles (disque dur).
- La manette de jeu vidéo « Wiimote » de Nintendo : ils sont utilisés pour reproduire à l'écran les mouvements de tennis ou golf
- Smartphone : Une des applications des accéléromètres est le calcul d'inclinaison. Grâce à l'effet de la gravité, un accéléromètre informe sur l'orientation d'un objet par rapport à la terre. Sur un smart phone, ça vous permet de passer du mode portrait au mode paysage.



Figure 15 Schéma d'implantation d'un système de sécurité

Figure 16 Application

### 3. Orientation

#### 3.1. Mesure d'un angle avec 1 axe

- Le Capteur capacitif permet l'utilisation d'un condensateur plan dont la valeur dépend de la distance entre les armatures

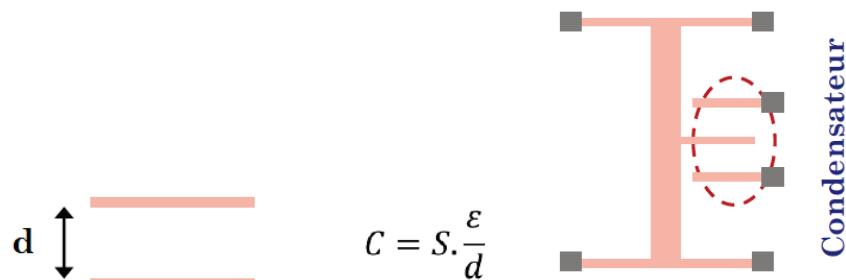


Figure 17

- 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

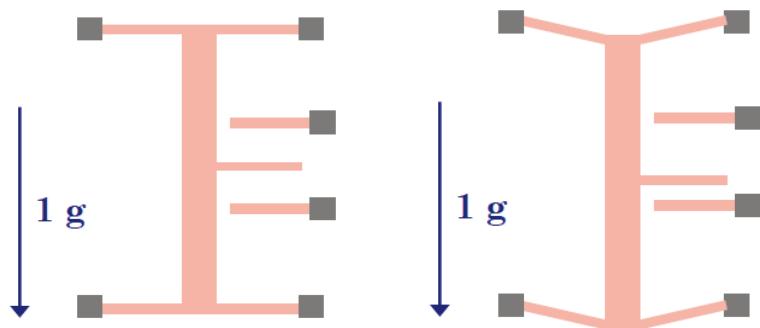


Figure 18

- 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

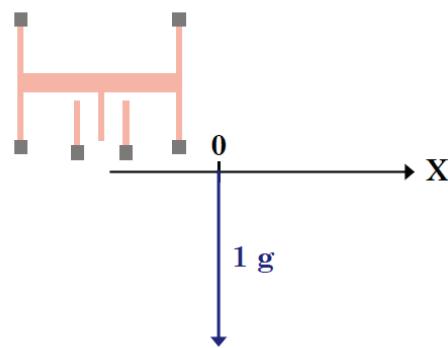


Figure 19

- Inclinaison

- On applique une inclinaison  $\theta$  de l'axe X
- La capacité de l'accéléromètre est modifiée

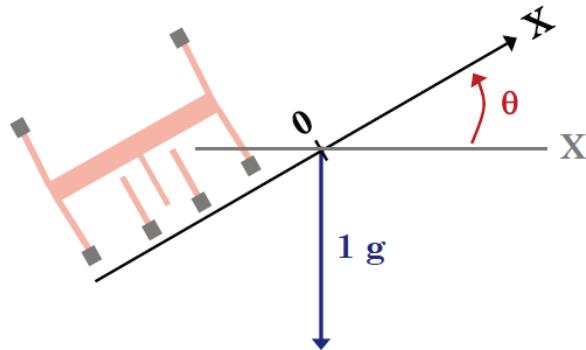


Figure 20

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

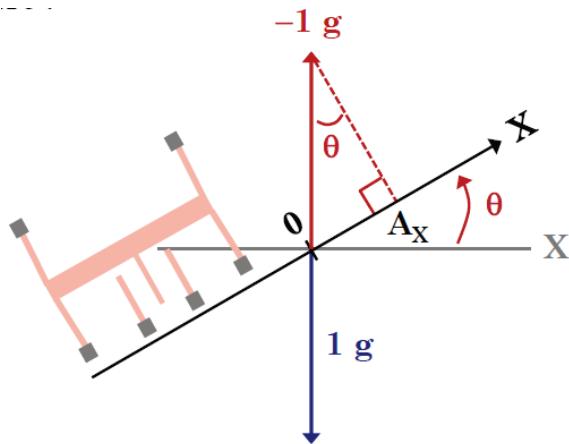


Figure 21

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

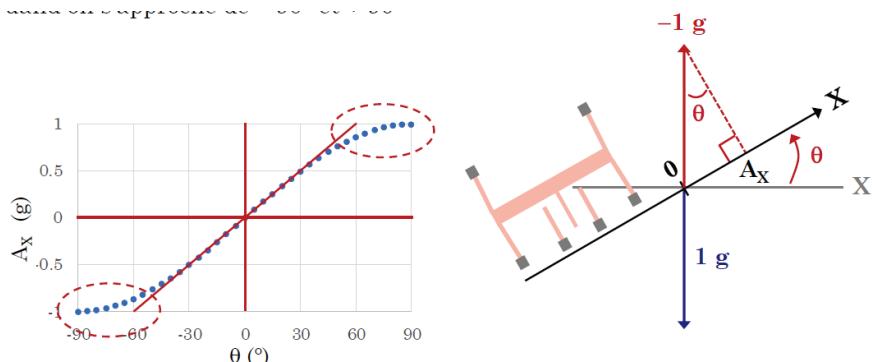


Figure 22

- On peut aussi remarquer que pour un angle faible, l'accélération dépend linéairement de l'angle  $\theta \approx \frac{A_x}{g}$

Exemple A  $45^\circ$ , l'accélération attendue sur l'axe  $A_x = \sin(45) * g = 0.707g$

- Problème d'angle La propriété du sinus induit qu'on ne peut pas différencier les angles  $\theta$

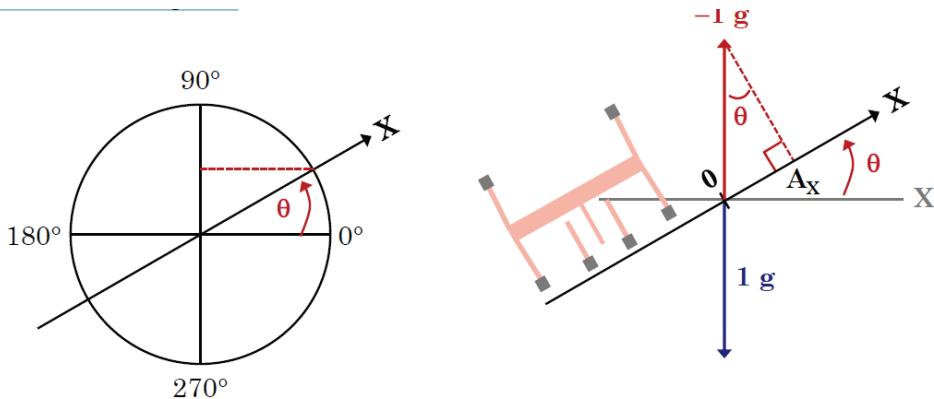


Figure 23

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

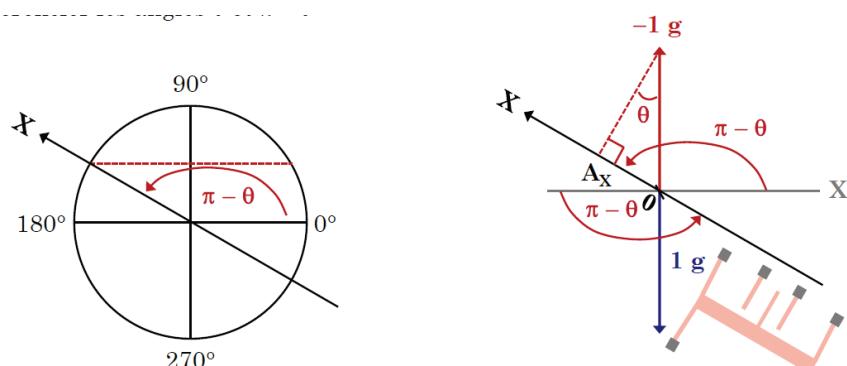


Figure 24

### 3.2. Mesure d'angle avec 2 axe

- Position à l'origine On considère 2 axes dont Y est perpendiculaire à la gravité

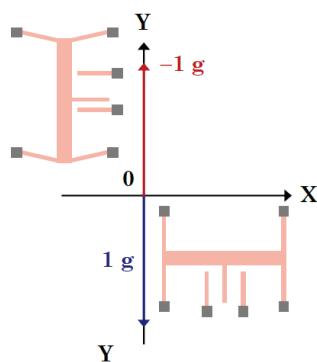


Figure 25

- On applique une inclinaison  $\theta$ 
  - 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  $A_y = g \cdot \cos(\theta)$

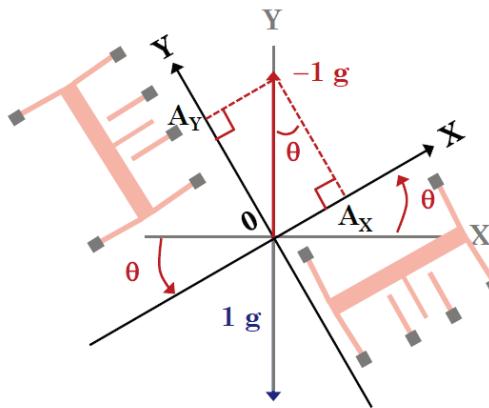


Figure 26

- Détermination de l'angle
  - Le signe de  $A_x$  et  $A_y$  permet de déterminer dans quel quadrant se trouve  $\theta$

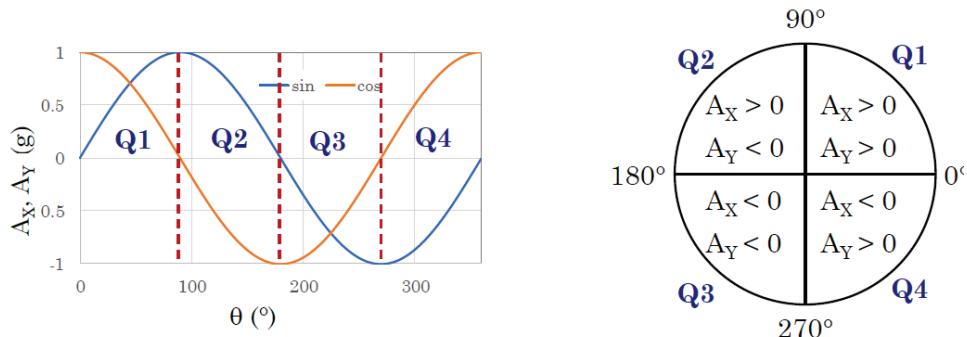


Figure 27

- L'utilisation des données sur les axes permet de déterminer l'angle en utilisant la tangente :  $\theta = \tan^{-1} \left( \frac{A_x}{A_y} \right)$

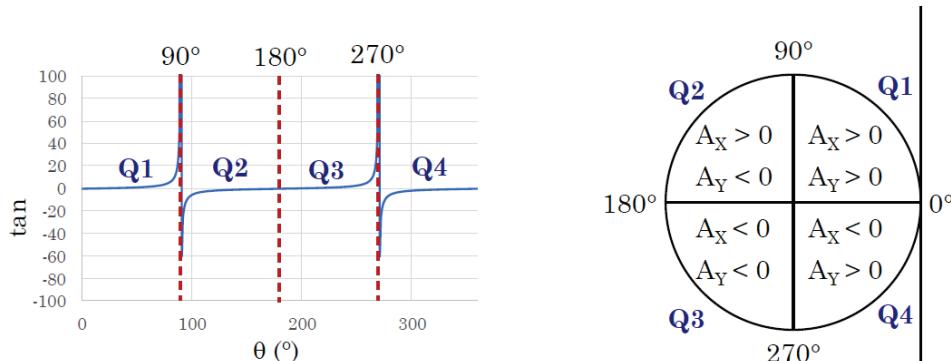
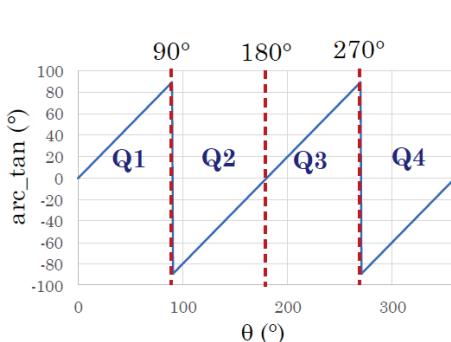


Figure 28

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



- $Q_1 : \theta = \tan^{-1} \left( \frac{A_x}{A_y} \right)$
- $Q_2 : \theta = \tan^{-1} \left( \frac{A_x}{A_y} \right) + 180$
- $Q_3 : \theta = \tan^{-1} \left( \frac{A_x}{A_y} \right) + 180$
- $Q_4 : \theta = \tan^{-1} \left( \frac{A_x}{A_y} \right) + 360 = -\tan^{-1} \left( \frac{A_x}{A_y} \right)$

C'est pour cela on utilise la fonction  $\text{atan2}(y, x) = \begin{cases} \varphi \cdot \text{sgn}(y) & x > 0 \\ \frac{\pi}{2} \cdot \text{sgn}(y) & x = 0 \\ (\pi - \varphi) \cdot \text{sgn}(y) & x < 0 \end{cases}$

$\varphi$  est l'angle compris entre 0 et  $\frac{\pi}{2}$  avec : $\tan(\varphi) = \left| \frac{y}{x} \right|$

Atan2 est semblable à atan, excepté qu'elle retourne des valeurs dans la l'intervalle  $(-\pi, \pi)$ , alors que atan retourne des valeurs sur  $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$ .

### 3.3. Accéléromètre à 3 axes

Un accéléromètre sert à mesurer une accélération, mais il est possible de s'en servir pour mesurer l'inclinaison d'un objet immobile. L'accéléromètre mesure la composante de la force de gravité selon 3 axes:

- l'axe des x (orienté selon le sens de la longueur)
- l'axe des y (orienté selon le sens de la largeur)
- l'axe des z (orienté perpendiculairement à la surface).

AXES OF ACCELERATION SENSITIVITY

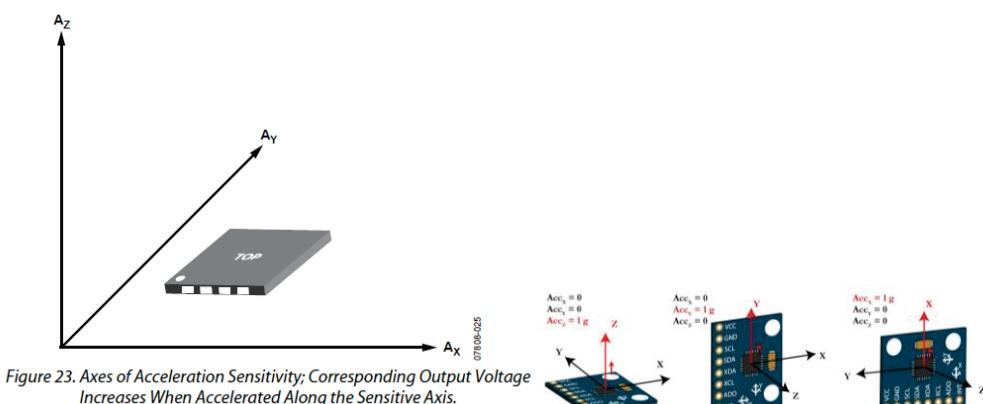


Figure 29

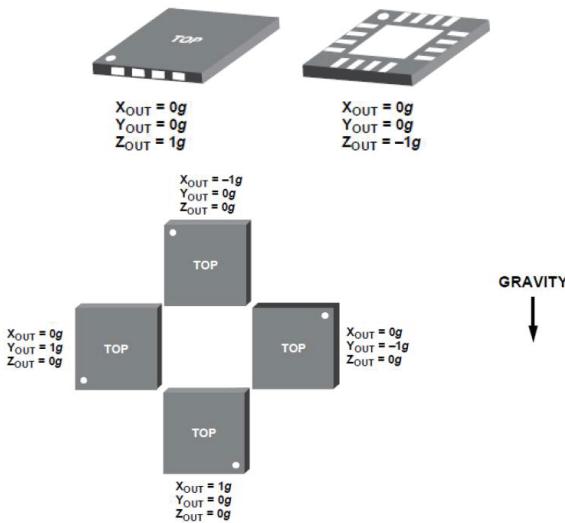


Figure 30

Ainsi, pour une tension d'alimentation Vs de 3 V et un zéro à Vs/2:

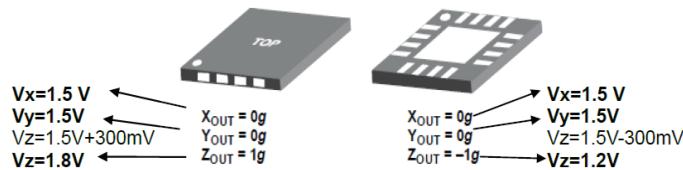


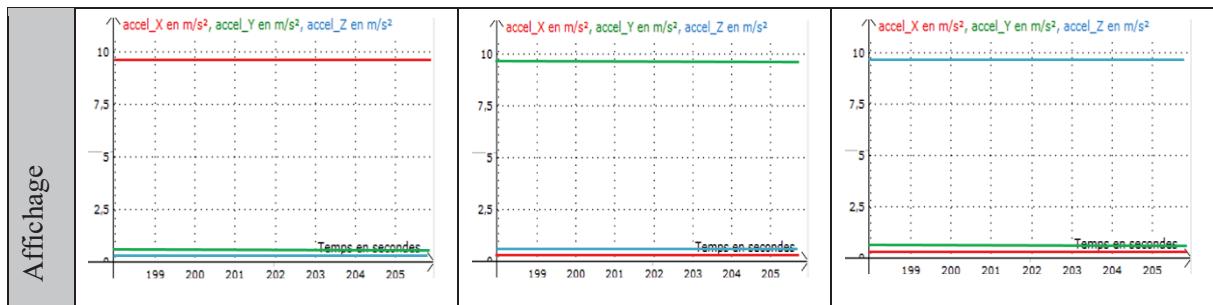
Figure 31

En fonction de la position, les sorties X, Y, Z seront:

| Stationary position | Accelerometer (signed integer) |       |       |
|---------------------|--------------------------------|-------|-------|
|                     | $A_x$                          | $A_y$ | $A_z$ |
| $Z_b$ down          | 0                              | 0     | +1 g  |
| $Z_b$ up            | 0                              | 0     | -1 g  |
| $Y_b$ down          | 0                              | +1 g  | 0     |
| $Y_b$ up            | 0                              | -1 g  | 0     |
| $X_b$ down          | +1 g                           | 0     | 0     |
| $X_b$ up            | -1 g                           | 0     | 0     |

Exemple Smartphone

|  |            |   |   |
|--|------------|---|---|
| Position   |            |   |   |
| Smartphone à l'horizontale, posé sur sa tranche. |            | Smartphone à la verticale, posé sur sa tranche. | Smartphone en position couchée sur une table. |
| (1g, 0, 0)                                       | (0, 1g, 0) | (0, 0, 1g)                                      | (0, 0, 1g)                                    |



## 4. Mesure des angles à 3 axes cas Tangage, roulis et lacet

### 4.1. Plan de référence

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

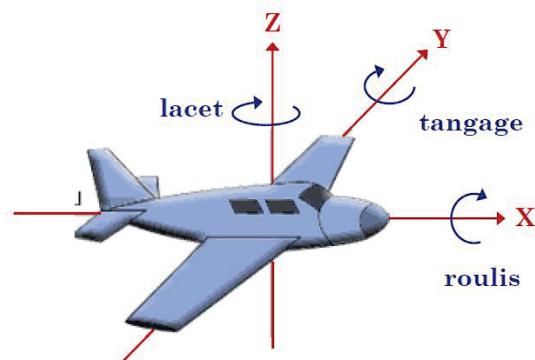


Figure 32

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

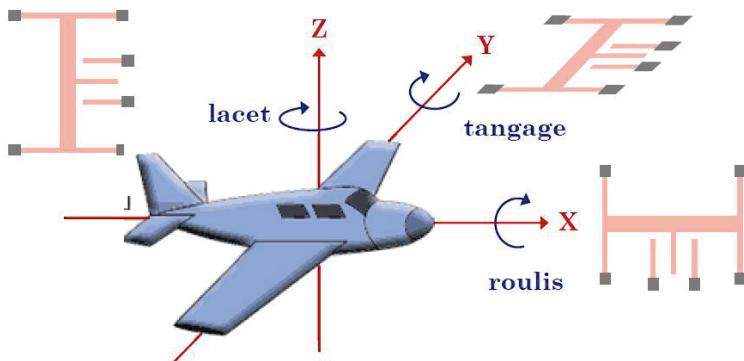


Figure 33

- Les axes sont indiqués sur les modules:

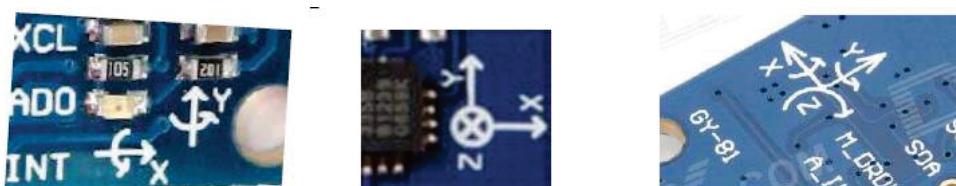


Figure 34

### - Plan de référence

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

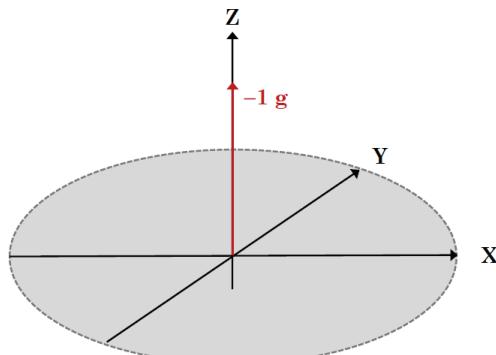


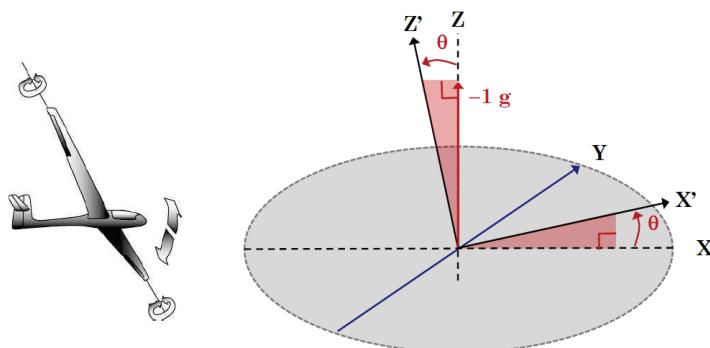
Figure 35

### 4.2. Tangage

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

#### - Angle de tangage: $\theta$

- 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( $\theta$ )

Figure 36 Angle tangage  $\theta$ 

### 4.3. Roulis

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

#### - Angle de roulis: $\phi$

- 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( $\phi$ )

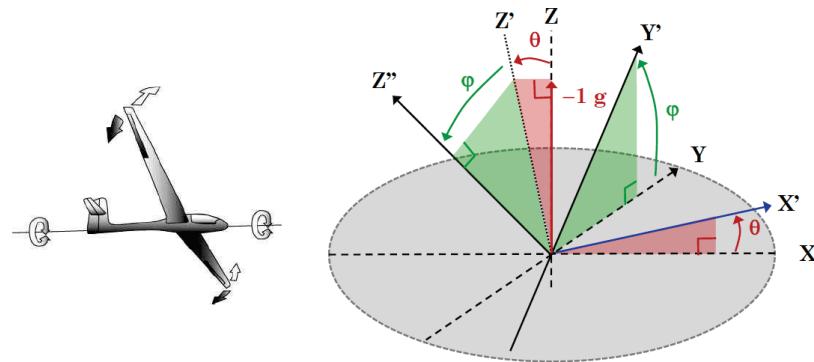


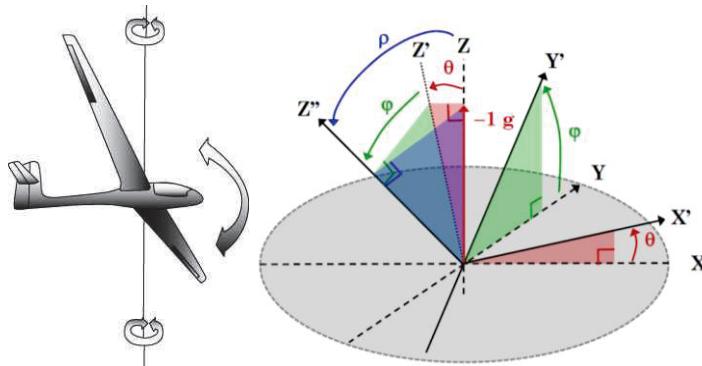
Figure 37

Angle de roulis:  $\varphi$ 

#### 4.4. Le lacet

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

- **Angle  $\rho$  :**  $\rho$  est l'angle entre la gravité et l'axe  $Z''$

Figure 38 angle lacet  $\rho$ 

#### Projection de la gravité sur les axes

**Expressions des 3 angles** D'habiles calculs permettent de trouver:

$$\theta = \arctan \left( \frac{A_x}{\sqrt{A_y^2 + A_z^2}} \right)$$

$$\varphi = \arctan \left( \frac{A_y}{\sqrt{A_x^2 + A_z^2}} \right)$$

$$\rho = \arctan \left( \frac{\sqrt{A_x^2 + A_y^2}}{A_z} \right)$$

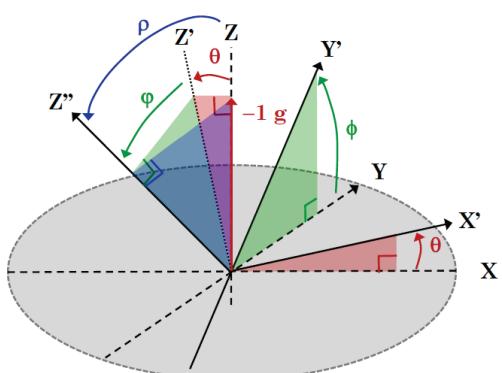


Figure 39 trois angles tangage, roulis et lacet

### 5. Déplacement rectiligne

#### 5.1. 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  $x(t) = \int v(t)dt = \int \int a(t)dt$

## 5.2. En échantillonné

- à  $t_{n=0} = 0$  :
  - $n = 0$  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  $a = a_0$
- à  $t_{n=1} = \Delta t_n$  :
  - 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  $x_n = v_n \cdot \Delta t_n + x_{n-1} = a_n \cdot \Delta t_n^2 + v_{n-1} \cdot \Delta t_n + x_{n-1}$

## 5.3. En pratique

- Le bruit de mesure entraîne une imprécision très importante sur la 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 ajoutent des sauts d'accélérations ce qui altère d'autant plus l'estimation de la position

# 6. Programmation

## 6.1. Accéléromètre ADXL335

- Les accéléromètres analogiques fournissent des niveaux de tension dans une plage prédéfinie qui doit être converti en une valeur numérique en utilisant un module CAN.
- L'accéléromètre GY 61 est constitué du module ADXL 335 qui est un triple accéléromètre analogique c'est à dire qu'il donne une tension continue pour chaque axe

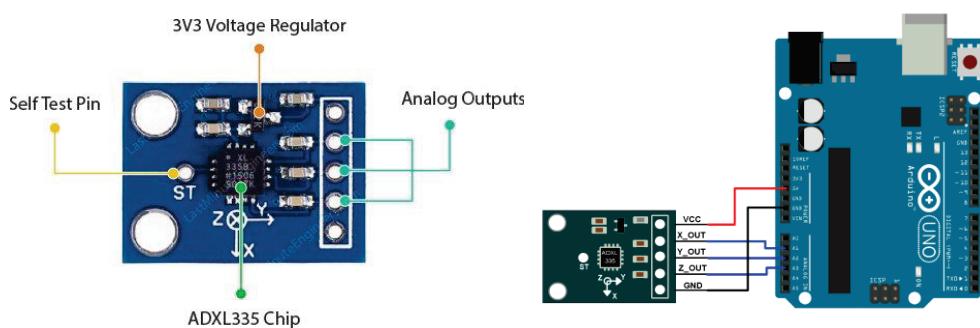


Figure 40 ADXL335

## 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
- 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
- Chaque axe peut mesurer une accélération de  $\pm 3 \text{ g}$

| Parameter   | Conditions          | Min     | Typ        | Max  | Unit                                 |
|---|---------------------|---------|------------|------|--------------------------------------|
| SENSOR INPUT  |                     |         |            |      |                                      |
| Measurement Range   | Each axis           | $\pm 3$ | $\pm 3.6$  |      | $g$                                  |
| Nonlinearity  | % of full scale     |         | $\pm 0.3$  |      | %                                    |
| Package Alignment Error   |                     |         | $\pm 1$    |      | Degrees                              |
| Interaxis Alignment Error                                       |                     |         | $\pm 0.1$  |      | Degrees                              |
| Cross-Axis Sensitivity <sup>1</sup>                             |                     |         | $\pm 1$    |      | %                                    |
| SENSITIVITY (RATIOMETRIC) <sup>2</sup>                          | Each axis           |         |            |      |                                      |
| Sensitivity at $X_{\text{OUT}}, Y_{\text{OUT}}, Z_{\text{OUT}}$ | $V_S = 3 \text{ V}$ | 270     | 300        | 330  | $\text{mV/g}$                        |
| Sensitivity Change Due to Temperature <sup>3</sup>              | $V_S = 3 \text{ V}$ |         | $\pm 0.01$ |      | $\%/\text{ }^{\circ}\text{C}$        |
| ZERO $g$ BIAS LEVEL (RATIOMETRIC)                               |                     |         |            |      |                                      |
| 0 $g$ Voltage at $X_{\text{OUT}}, Y_{\text{OUT}}$               | $V_S = 3 \text{ V}$ | 1.35    | 1.5        | 1.65 | $\text{V}$                           |
| 0 $g$ Voltage at $Z_{\text{OUT}}$                               | $V_S = 3 \text{ V}$ | 1.2     | 1.5        | 1.8  | $\text{V}$                           |
| 0 $g$ Offset vs. Temperature                                    |                     |         | $\pm 1$    |      | $\text{mg}/\text{ }^{\circ}\text{C}$ |

**TraitemenT des données** Soit  $A_{mx}, A_{my}$ , et  $A_{mz}$  les valeurs analogiques mesurées.

$$A_{mx} = \frac{\left(A_{mx} * \frac{V_{ref}}{2^n} - V_0\right)}{\text{sensibilité}}, A_{my} = \frac{\left(A_{my} * \frac{V_{ref}}{2^n} - V_0\right)}{\text{sensibilité}}, A_{mz} = \frac{\left(A_{mz} * \frac{V_{ref}}{2^n} - V_0\right)}{\text{sensibilité}}$$

Programme :

```
#include <math.h>
const int x_out = A1; /* connect x_out of module to A1 of UNO board */
const int y_out = A2; /* connect y_out of module to A2 of UNO board */
const int z_out = A3; /* connect z_out of module to A3 of UNO board */

void setup() {
  Serial.begin(9600);
}

void loop() {
  int x_adc_value, y_adc_value, z_adc_value;
  double x_g_value, y_g_value, z_g_value;
  double roll, pitch, yaw;
  x_adc_value = analogRead(x_out); /* Digital value of voltage on x_out pin */
  y_adc_value = analogRead(y_out); /* Digital value of voltage on y_out pin */
  z_adc_value = analogRead(z_out); /* Digital value of voltage on z_out pin */
  Serial.print("x = ");
  Serial.print(x_adc_value);
  Serial.print("\t\t");
  Serial.print("y = ");
  Serial.print(y_adc_value);
  Serial.print("\t\t");
}
```

```

Serial.print("z = ");
Serial.print(z_adc_value);
Serial.print("\t\t");
//delay(100);
x_g_value = ( ((double)(x_adc_value * 5)/1024) - 1.65 ) / 0.330; /* Acceleration in x-direction
in g units */
y_g_value = ( ((double)(y_adc_value * 5)/1024) - 1.65 ) / 0.330; /* Acceleration in y-direction
in g units */
z_g_value = ( ((double)(z_adc_value * 5)/1024) - 1.80 ) / 0.330; /* Acceleration in z-direction
in g units */

Serial.print("Ax = ");
Serial.print(x_g_value);
Serial.print("\t");
Serial.print("Ay = ");
Serial.print(y_g_value);
Serial.print("\n\n");
delay(1000);
}

```

## 6.2. Accéléromètre GY-521

- **Présentation**

- L'accéléromètre 3 axes GY-521 est basé sur le MEMS MPU-6050 qui contient aussi un gyroscope 3axes
- Les mesures des capteurs sont obtenues via le bus I2C
- 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: $\pm 2, \pm 4, \pm 8, \pm 16g$
- L'adresse I2C du MPU-6050 est 0x68 lorsque AD0=0V et elle devient 0x69 lorsque AD0=5V
- Les données et les informations sur le fonctionnement sont contenues dans des registres de 8bits



Figure 41 GY-521

- **Exemple de montage**

- Le montage est simple puis qu'il n'utilise que l'alimentation +5V 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)

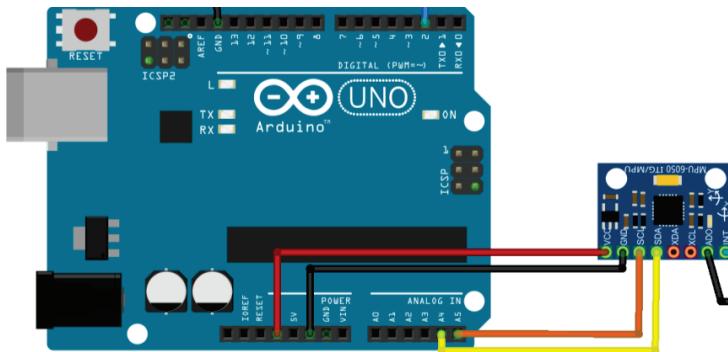


Figure 42 Câblage avec la carte arduino

## • 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: `constint MPU_addr=0x68;`

### - 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()
```

### - Lecture des données

- On définit les variables où seront stockées les 3 accélérations

```
floatAcc_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]  |

- 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)
```

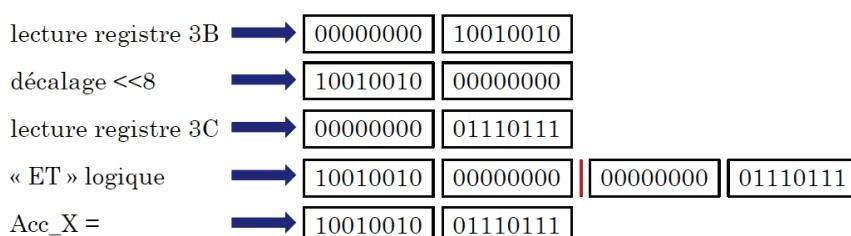
- On indique alors à l'accéléromètre que l'on va lire 6 registres donc 6 octets (à partir du registre 0x3B)

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

- 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



### Exemple de valeur stockée dans l'accéléromètre

| valeur d'accélération sur axe X | ACCEL_XOUT[15..8] | ACCEL_XOUT[7..0] |
|---------------------------------|-------------------|------------------|
| 0.15g                           | 0x09              | 0x99             |
| 1g                              | 0x40              | 0x00             |

### - 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 :  $\pm 2, \pm 4, \pm 8, \pm 16g$
- 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

#### 4.5 Register 28 – Accelerometer Configuration ACCEL\_CONFIG

Type: Read/Write

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

**Description:** 0 0 0 1 0 0 0 0

This register is used to trigger accelerometer self test and configure the accelerometer full scale range. This register also configures the Digital High Pass Filter (DHPF).

AFS\_SEL selects the full scale range of the accelerometer outputs according to the following table.

68);

| AFS_SEL | Full Scale Range |
|---------|------------------|
| 0       | $\pm 2g$         |
| 1       | $\pm 4g$         |
| 2       | $\pm 8g$         |
| 3       | $\pm 16g$        |

$$2 = 1 \times 2^1 + 0 \times 2^0$$

$$\pm 2g = B00000000$$

$$\pm 4g = B00001000$$

$$\pm 8g = B00010000$$

$$\pm 16g = B00011000$$

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

$$\pm 2g = 16384/g$$

$$\pm 4g = 8192/g$$

$$\pm 8g = 4096/g$$

$$\pm 16g = 2048/g$$

- Voici le code pour travailler dans la gamm  $\pm 2g$

```
Wire.beginTransmission(0x68)
Wire.write (0x1c)
Wire.write (0x00)
Wire.endsTransmission () ;
```

- Convertir les mesures en valeur physiques : Pour finir, on peut prendre en compte la sensibilité pour obtenir la valeur en g sur les 3 axes. Par exemple pour la gamme  $\pm 2g$

| AFS_SEL | Full Scale Range | LSB Sensitivity |
|---------|------------------|-----------------|
| 0       | $\pm 2g$         | 16384 LSB/g     |
| 1       | $\pm 4g$         | 8192 LSB/g      |
| 2       | $\pm 8g$         | 4096 LSB/g      |
| 3       | $\pm 16g$        | 2048 LSB/g      |

$$\text{AccX} = (\text{Float})\text{Acc\_x} / 16384;$$

$$\text{AccY} = (\text{Float})\text{Acc\_y} / 16384;$$

$$\text{AccZ} = (\text{Float})\text{Acc\_z} / 16384;$$

### - 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);
```

- Affichage des axes de l'accélération

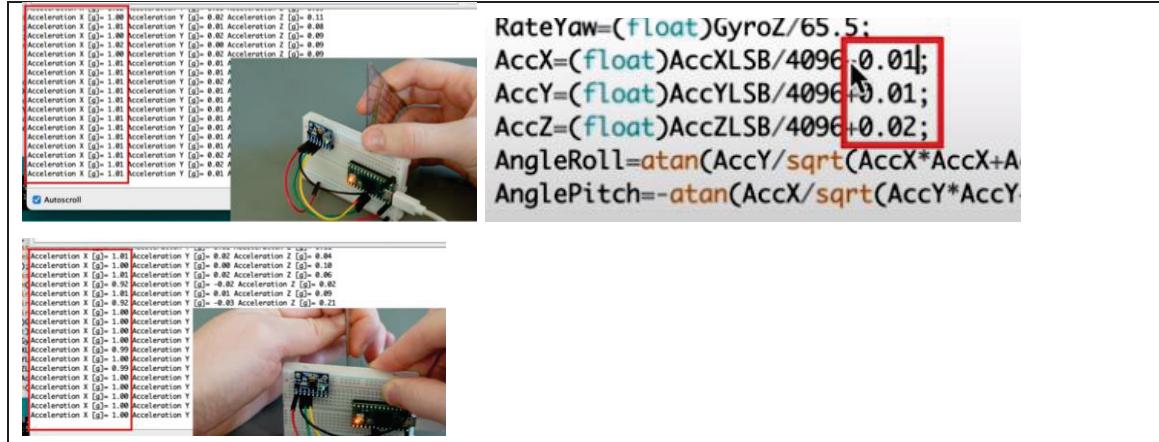
```
42 void setup() {
43     Serial.begin(57600);
44     pinMode(13, OUTPUT);
45     digitalWrite(13, HIGH);
46     Wire.setClock(400000);
47     Wire.begin();
48     delay(250);
49     Wire.beginTransmission(0x68);
50     Wire.write(0x6B);
51     Wire.write(0x00);
52     Wire.endTransmission();
53 }

54 void loop() {
55     gyro_signals();
56     Serial.print("Acceleration X [g]= ");
57     Serial.print(AccX);
58     Serial.print(" Acceleration Y [g]= ");
59     Serial.print(AccY);
60     Serial.print(" Acceleration Z [g]= ");
61     Serial.println(AccZ);
62     delay(50);
63 }
```

Communication with the gyroscope and calibration (project 4 and 5)

Print the accelerometer values

- Calibration des axes l'accéléromètre



## II.Gyroscope

### 1. Définition

Un gyromètre mesure la vitesse angulaire du capteur selon un axe de rotation en mesurant la force de Coriolis qui lui est appliquée. Lorsque l'on effectue un déplacement dans un référentiel en rotation (Figure ci-dessous), on subit une force inertielle, comprenant deux composantes : la force d'inertie d'entrainement et la force d'inertie de Coriolis.

Cette dernière s'exerce dans une direction parallèle au sens de rotation : dans le sens de rotation si l'on se rapproche du centre de rotation, dans le sens inverse de la rotation si l'on s'écarte du centre de rotation

$$\vec{F}_c = -2m\vec{\Omega} \wedge \vec{V}_R$$

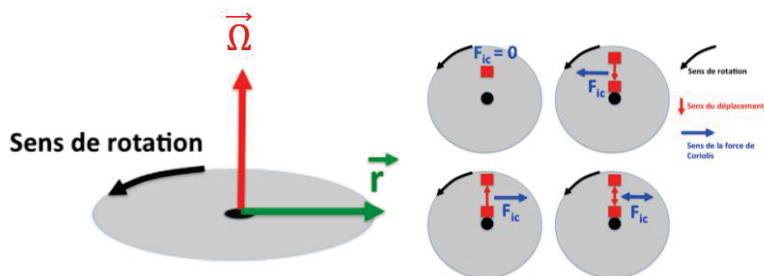


Figure 43 Référentiel en

Figure 44 Variation de la force de Coriolis en fonction du

### 2. Principe Gyromètre MEMS

Si on met une masse en mouvement selon un axe perpendiculaire à l'axe de rotation, on peut observer une variation de cette force (Figure 44). En mettant une masse en oscillation, on peut observer l'écart de cette force entre un point proche du centre et un point éloigné du centre. Cet écart est nul s'il n'y a pas de rotation, et est d'autant plus grand que la vitesse de rotation est grande. En pratique, les gyromètres utilisent un principe de détection proche de celui des accéléromètres. Une masse est mise en oscillation à très haute fréquence, un système en peigne mesure le déplacement quand la masse est en position basse, c'est à dire proche du centre de rotation, et le déplacement quand la masse est en position haute, c'est à dire loin du centre de rotation (Figure ci-dessous). La différence de déplacement entre ces deux positions nous donne directement la vitesse angulaire

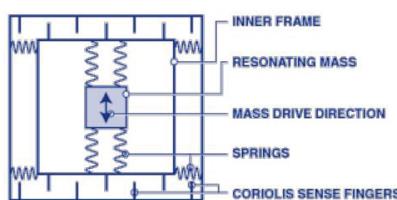


Figure 45 Schéma de fonctionnement d'un

Trois gyromètres dirigés selon les trois axes de rotation permettent donc de mesurer la vitesse de rotation d'un objet. L'intégration de cette information à partir d'une position de base fournit donc une position angulaire, c'est à dire orientation dans l'espace. Cette orientation est décrite traditionnellement par les angles de Tait-Bryan, ou angles d'Euler, utilisés en navigation (Figure ci-dessous) : lacet, tangage et roulis (yaw, pitch et roll).

### Modèle mathématique du gyroscope MEMS

Le modèle à paramètres localisés de la conception du gyroscope MEMS est illustré à la figure 46. Tels que  $m_d$  : La masse d'entraînement (drive mass),  $m_s$  (sens frame) et  $k_y, k_x$  : le ressort mécanique

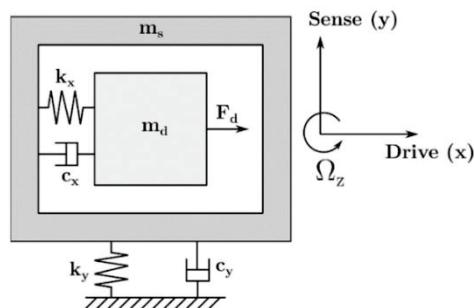


Figure 46

Les équations de mouvement pour la conception du gyroscope MEMS peuvent être écrites comme suit :

$$m_d \frac{d^2x}{dt^2} + c_x \frac{dx}{dt} + k_x x = F_d$$

$$(m_d + m_s) \frac{d^2y}{dt^2} + c_y \frac{dy}{dt} + k_y y = -2 \cdot m_d \Omega_z \frac{dx}{dt}$$

Les valeurs des capacités sont comme suit :

$$C_{1t}(y) = \frac{\varepsilon_0 S_t}{(y_{0t} - y)} \text{ et } C_{2t}(y) = \frac{\varepsilon_0 S_t}{(y_{0t} + y)}$$

### 3. Détermination de l'orientation

$$\underbrace{\text{Angle}_{pitch}}_{[\circ]} = \int_0^{kT_s} \underbrace{\Omega_{pitch}}_{[^\circ/S]} dt$$

$$\text{angle}_{pitch}(k) = \text{angle}_{pitch}(k-1) + T_s \cdot \Omega_{Gy}(k)$$

$k$  nombre d'itérations,  $T_s$  temps d'itération

## 4. Programmation

```
#include<Wire.h>

Float RateRoll, RatePitch, RateYaw; //declaration des variables

void gyro_signals (void) {           //start I2C communication with a gyro
    Wire.beginTransmission(0x68)
```

### 4.34 Register 117 – Who Am I WHO\_AM\_I

Type: Read Only

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3          | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|------|------|------|------|---------------|------|------|------|
| 75             | 117                | -    | -    | -    | -    | WHO_AM_I[6:1] | -    | -    | -    |

#### Description:

This register is used to verify the identity of the device. The contents of WHO\_AM\_I are the upper 6 bits of the MPU-60X0's 7-bit I<sup>2</sup>C address. The least significant bit of the MPU-60X0's I<sup>2</sup>C address is determined by the value of the ADO pin. The value of the ADO pin is not reflected in this register.

The default value of the register is 0x68.

Bits 0 and 7 are reserved. (Hard coded to 0)

#### Parameters:

WHO\_AM\_I Contains the 6-bit I<sup>2</sup>C address of the MPU-60X0.

The Power-On-Reset value of Bit6:Bit1 is 110 100.

```
Wire.write(0x1A);           // switch on the low pass filter

Wire.write(0x05);

Wire.endTransmission() ;
```

### 4.3 Register 26 – Configuration CONFIG

Type: Read/Write

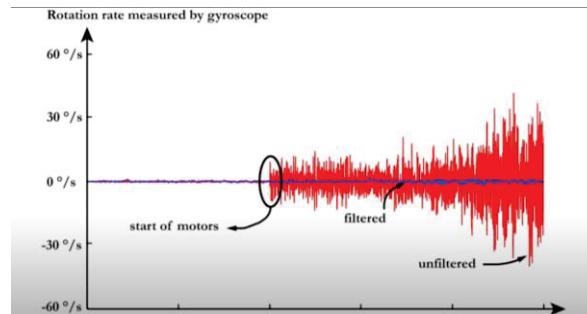
| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5              | Bit4          | Bit3 | Bit2 | Bit1 | Bit0          |
|----------------|--------------------|------|------|-------------------|---------------|------|------|------|---------------|
| 1A             | 26                 | -    | -    | EXT_SYNC_SET[2:0] | DLPF_CFG[2:0] | -    | -    | -    | DLPF_CFG[2:0] |

#### Description:

This register configures the external Frame Synchronization (FSYNC) pin sampling and the Digital Low Pass Filter (DLPF) setting for both the gyroscopes and accelerometers.

The DLPF is configured by DLPF\_CFG. The accelerometer and gyroscope are filtered according to the value of DLPF\_CFG as shown in the table below.

| DLPF_CFG | Accelerometer (F <sub>s</sub> = 1kHz) |            |                | Gyroscope  |          |  |
|----------|---------------------------------------|------------|----------------|------------|----------|--|
|          | Bandwidth (Hz)                        | Delay (ms) | Bandwidth (Hz) | Delay (ms) | Fs (kHz) |  |
| 0        | 260                                   | 0          | 256            | 0.98       | 8        |  |
| 1        | 184                                   | 2.0        | 188            | 1.9        | 1        |  |
| 2        | 94                                    | 3.0        | 98             | 2.8        | 1        |  |
| 3        | 44                                    | 4.9        | 42             | 4.8        | 1        |  |
| 4        | 21                                    | 8.5        | 20             | 8.3        | 1        |  |
| 5        | 10                                    | 13.8       | 10             | 13.4       | 1        |  |
| 6        | 5                                     | 19.0       | 5              | 18.6       | 1        |  |
| 7        | RESERVED                              | RESERVED   | RESERVED       | RESERVED   | 8        |  |



```
Wire.beginTransmission(0x68)

Wire.write(0x1B);           // set the sensitivity scale factor

Wire.write(0x8);

Wire.endTransmission() ;
```

#### 4.4 Register 27 – Gyroscope Configuration GYRO\_CONFIG

Type: Read/Write

| Register (Hex) | Register (Decimal) | Bit7  | Bit6  | Bit5  | Bit4        | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|-------|-------|-------|-------------|------|------|------|------|
| 1B             | 27                 | XG_ST | YG_ST | ZG_ST | FS_SEL[1:0] | -    | -    | -    | -    |

**Description:**

This register is used to trigger gyroscope self-test and configure the gyroscopes' full scale range.

Switch on the low

#### 4.4 Register 27 – Gyroscope Configuration GYRO\_CONFIG

Type: Read/Write

| Register (Hex) | Register (Decimal) | Bit7  | Bit6  | Bit5  | Bit4        | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|-------|-------|-------|-------------|------|------|------|------|
| 1B             | 27                 | XG_ST | YG_ST | ZG_ST | FS_SEL[1:0] | 0    | 0    | 0    | 1    |

**Description:**

This register is used to trigger gyroscope self-test and configure the gyroscopes' full scale range.

Each 16-bit gyroscope measurement has a full scale defined in FS\_SEL (Register 27). For each full scale setting, the gyroscopes' sensitivity per LSB in GYRO\_xOUT is shown in the table below:

| FS_SEL | Full Scale Range          | LSB Sensitivity             |
|--------|---------------------------|-----------------------------|
| 0      | $\pm 250^\circ/\text{s}$  | 131 LSB/ $^\circ/\text{s}$  |
| 1      | $\pm 500^\circ/\text{s}$  | 65.5 LSB/ $^\circ/\text{s}$ |
| 2      | $\pm 1000^\circ/\text{s}$ | 32.8 LSB/ $^\circ/\text{s}$ |
| 3      | $\pm 2000^\circ/\text{s}$ | 16.4 LSB/ $^\circ/\text{s}$ |

$$0x2^7 + 0x2^6 + 0x2^5 + 0x2^4 + 1x2^3 + 0x2^2 + 0x2^1 + 0x2^0 = 8$$

**Wire.beginTransmission(0x68)**

```
Wire.write(0x43); // Acess registers storing gyro measurements
Wire.endTransmission();
Wire.requestFrom(0x68,6)
```

#### 4.20 Registers 67 to 72 – Gyroscope Measurements

GYRO\_XOUT\_H, GYRO\_XOUT\_L, GYRO\_YOUT\_H, GYRO\_YOUT\_L, GYRO\_ZOUT\_L

Type: Read Only

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0            |
|----------------|--------------------|------|------|------|------|------|------|------|-----------------|
| 43             | 67                 |      |      |      |      |      |      |      | GYRO_XOUT[15:8] |
| 44             | 68                 |      |      |      |      |      |      |      | GYRO_XOUT[7:0]  |
| 45             | 69                 |      |      |      |      |      |      |      | GYRO_YOUT[15:8] |
| 46             | 70                 |      |      |      |      |      |      |      | GYRO_YOUT[7:0]  |
| 47             | 71                 |      |      |      |      |      |      |      | GYRO_ZOUT[15:8] |
| 48             | 72                 |      |      |      |      |      |      |      | GYRO_ZOUT[7:0]  |

**Parameters:**

GYRO\_XOUT 16-bit 2's complement value.

Stores the most recent X axis gyroscope measurement.

GYRO\_YOUT 16-bit 2's complement value.

Stores the most recent Y axis gyroscope measurement.

GYRO\_ZOUT 16-bit 2's complement value.

Stores the most recent Z axis gyroscope measurement.

```
int16_t GyroX = (Wire.read() << 8 | Wire.read()); //read the gyro measurements around the Xaxis
```

```
int16_t GyroY = (Wire.read() << 8 | Wire.read());
```

```
int16_t GyroZ = (Wire.read() << 8 | Wire.read());
```

```
RateRoll, = (Float)GyroX /65.5; //convert the measurement units to °/s
```

```
RatePitch, = (Float)GyroY /65.5;
```

RateYaw, = (**Float**)GyroZ /65.5;

Each 16-bit gyroscope measurement has a full scale defined in **FS\_SEL** (Register 27). For each full scale setting, the gyroscopes' sensitivity per LSB in **GYRO\_xOUT** is shown in the table below:

| <b>FS_SEL</b> | <b>Full Scale Range</b> | <b>LSB Sensitivity</b> |
|---------------|-------------------------|------------------------|
| 0             | ± 250 °/s               | 131 LSB/°/s            |
| 1             | ± 500 °/s               | 65.5 LSB/°/s           |
| 2             | ± 1000 °/s              | 32.8 LSB/°/s           |
| 3             | ± 2000 °/s              | 16.4 LSB/°/s           |

```
23 void setup() {
24   Serial.begin(57600);
25   pinMode(13, OUTPUT);
26   digitalWrite(13, HIGH);
27   Wire.setClock(400000); Set the clock speed
28   Wire.begin();
29   delay(250);
```

The MPU-6000 family is comprised of two parts, the MPU-6000 and **MPU-6050**. These parts are identical to each other with two exceptions. The **MPU-6050** supports I<sup>C</sup> communications at up to 400kHz and has a VLOGIC pin that defines its interface voltage levels; the **MPU-6000** supports SPI at up to 20MHz in addition to I<sup>C</sup>, and has a single supply pin, VDD, which is both the device's logic reference supply and the analog supply for the part.

```
Wire.beginTransmission(0x68);
Wire.write(0x6B);
Wire.write(0x00); //start the gyro in power mode
Wire.endTransmission() ;
}
```

#### 4.30 Register 107 – Power Management 1 PWR\_MGMT\_1

Type: Read/Write

| Register (Hex) | Register (Decimal) | Bit7         | Bit6  | Bit5  | Bit4 | Bit3     | Bit2        | Bit1 | Bit0 |
|----------------|--------------------|--------------|-------|-------|------|----------|-------------|------|------|
| 68             | 107                | DEVICE_RESET | SLEEP | CYCLE | -    | TEMP_DIS | CLKSEL[2:0] |      |      |

##### Description:

This register allows the user to configure the power mode and clock source. It also provides a bit for resetting the entire device, and a bit for disabling the temperature sensor.

By setting **SLEEP** to 1, the MPU-60X0 can be put into low power sleep mode. When **CYCLE** is set to 1 while **SLEEP** is disabled, the MPU-60X0 will be put into Cycle Mode. In Cycle Mode, the device cycles between sleep mode and waking up to take a single sample of data from accelerometer at a rate determined by **LP\_WAKE\_CTRL** (register 108). To configure the wake frequency, use **LP\_WAKE\_CTRL** within the Power Management 2 register (Register 108).

```
Void loop() {
Gyro_signals();
Serial.print("Roll rate [°/s]=");
Serial.print(rateRall);
Serial.print("Pitch rate [°/s]=");
Serial.print(ratePitch);
Serial.print("Yaw rate [°/s]=");
Serial.print(rateYaw);
}
```

## 5. Applications



## III. Capteur inertiel

### 1. Définition

On parle de **centrale inertIELLE** (ou IMU = “Inertial Measurement Unit”) lorsqu'on cherche à mesurer l'ensemble des 6 accélérations (linéaires et angulaires). Bien que l'accélération linéaire soit définie en m/s<sup>2</sup> (SI), la majorité des documentations sur ces capteurs exprime en « g » (environ 9,81 m/s<sup>2</sup>) l'accélération (comme celle causée par la gravitation terrestre).

- Typiquement: combinaison d'un accéléromètre et d'un gyroscope (mais l'inclusion d'un magnétomètre est possible aussi)
- Estimation de 6 DDL (« 6 axes »)
  - Position :  $x, y, z$
  - Orientation : angles de lacet, tangage, roulis  $\rho, \theta, \varphi$

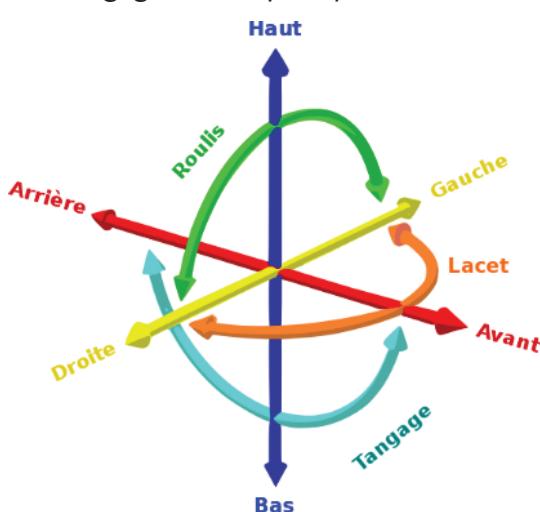


Figure 47 6 axes d'un IMU

La référence du capteur que nous allons utiliser est **GY-521** ; il est architecturé autour d'un **MPU 6050** (capteur intelligent) qui est composé de deux capteurs et d'un processeur :

Le MPU-6050 est une puce couramment utilisée qui combine un gyroscope MEMS et un accéléromètre MEMS et utilise un bus I2C standard pour la transmission de données et un processeur « Digital Motion Processor » (DMP) pouvant stocker et restituer des données.

En général c'est un magnétomètre qui est ajouté pour essayer de repérer le Nord magnétique. Ces 9 axes sont en général suffisants pour réaliser, en combinant les mesures de ces trois capteurs, une centrale à inertie. Mais ces trois capteurs ont des défauts rendant leur utilisation délicate. Nous allons nous contenter de six axes dans la suite du cours dont l'objectif peut se résumer dans la figure 48 ci-dessus : réaliser à l'aide d'un traitement des données (fusion des données).

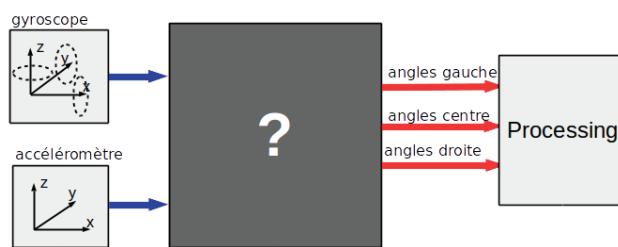


Figure 48

## 2. Fusion des capteurs

### 2.1. La notion d'attitude

L'attitude en robotique (et en astronautique) désigne la direction des axes de la pièce mobile du robot ; généralement caractérisée par trois angles (roulis, tangage et cap ou lacet).

### 2.2. Principe

Il est important de noter que les résultats de l'accéléromètre fournissent des angles d'orientation précis tant que la gravité est la seule force agissant sur le capteur. Cependant, lors du déplacement et de la rotation du capteur, nous lui appliquons des forces, ce qui fait fluctuer les mesures. Le résultat net est que les données de l'accéléromètre ont tendance à être très bruyantes, avec des perturbations brèves mais importantes. Si ceux-ci peuvent être moyennés, l'accéléromètre fournit des résultats précis sur des échelles de temps plus longues que les perturbations.

Le calcul de l'orientation à partir du capteur du gyroscope est différent, car le gyroscope mesure la vitesse angulaire, *et non l'orientation angulaire* elle-même. Pour calculer l'orientation, nous devons d'abord initialiser la position du capteur, puis mesurer la vitesse angulaire ( $\Omega$ ) autour des axes X, Y et Z à des intervalles mesurés ( $\Delta t$ ). Alors  $\Omega \times \Delta t = \text{changement d'angle}$ . Le nouvel angle d'orientation sera l'angle d'origine plus ce changement. Le problème avec cette approche est que nous *intégrons* – en additionnant de nombreux petits intervalles calculés – pour trouver l'orientation. L'addition répétée d'incrément de  $\Omega \times \Delta t$  entraînera de petites

erreurs systématiques amplifiées au fil du temps. C'est la cause de la dérive gyroscopique, et sur de longues échelles de temps, les données du gyroscope deviendront de plus en plus imprécises.

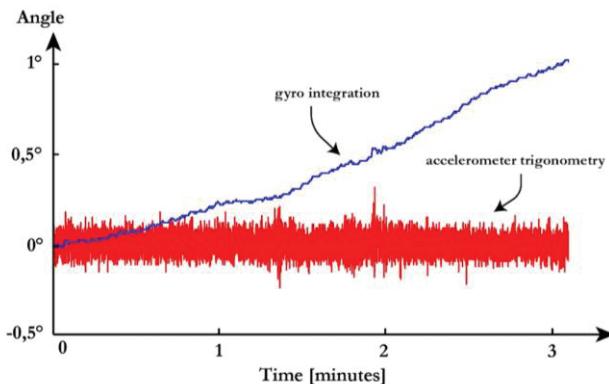


Figure 49

Les données de l'accéléromètre et du gyroscope sont sujettes à des erreurs systématiques (figure 49). L'accéléromètre fournit des données précises sur le long terme, mais est bruyant à court terme. Le gyroscope fournit des données précises sur le changement d'orientation à court terme, mais l'intégration nécessaire fait dériver les résultats sur des échelles de temps plus longues. La solution à ces problèmes est de fusionner les données de l'accéléromètre et du gyroscope de manière à ce que les erreurs s'annulent.

### 2.3. Méthodes de fusion de capteur

Prenons l'exemple de mesure d'angle d'orientation, plusieurs méthodes permettent de fusionner entre les capteurs accéléromètre et gyroscope

#### A partir d'un accéléromètre

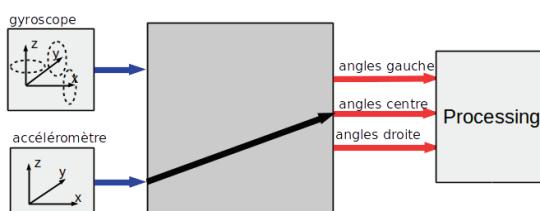


Figure 50

Avantages:

- Facile à coder.

Inconvénients:

- Bruyant.
- L'axe X lira toute accélération horizontale comme un changement d'angle.

### Filtre passe-bas

Une amélioration possible consiste à réaliser un filtre passe-bas pour filtrer les données de l'accélération. En effet, on remarque certainement noter la "nervosité" de la réponse de l'accéléromètre et un filtrage moyenNe s'impose donc de lui-même.

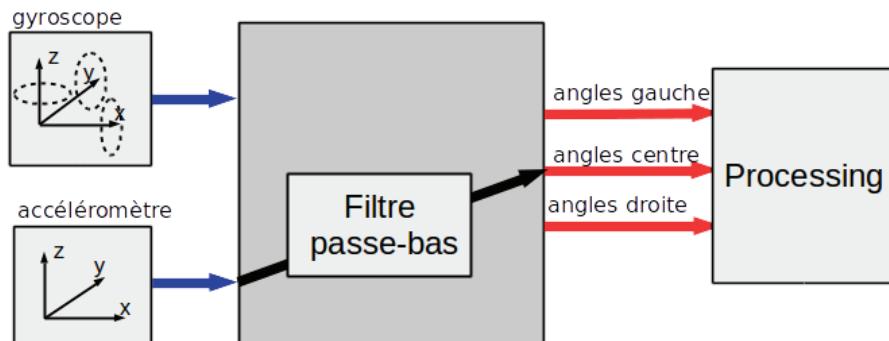


Figure 51

$$\text{angle}(k) = 0.75 * \text{angle}(k - 1) + 0.25 * \text{angle de l'accéléromètre}(k)$$

Avantages:

- Facile à coder.

Inconvénients:

- La mesure d'angle sera décalée en raison de la moyenne. Plus on filtre, plus il y en aura un décalage. Le décalage est généralement mauvais pour la stabilité.

### Intégration numérique

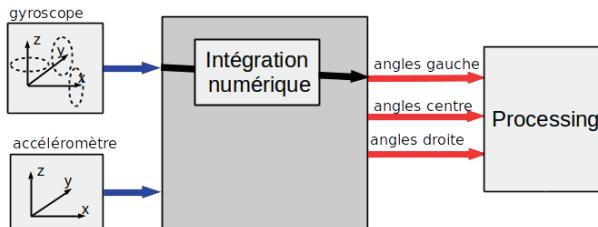


Figure 52

$$\text{angle}(k) = \text{angle}(k - 1) + \text{vitesse angulaire du gyroscope}(k) * dt$$

Nécessite de connaître l'intervalle de temps entre les mises à jour,  $dt$

Avantages:

- Un seul capteur à lire.
- Rapide, le décalage n'est pas un problème.
- Non soumis aux accélérations horizontales.
- Facile à coder.

Inconvénients:

- Le gyroscope ne lit pas parfaitement le zéro à l'arrêt, le petit taux continuera à s'ajouter à l'angle jusqu'à ce qu'il soit éloigné de l'angle réel. En effet, un des plus important est l'intégration de la moyenne (qui n'est jamais nulle) et qui finit par saturer la sortie. Il faudra développer la phase de calibration. Un moyen est qu'au départ du programme le GY-521 soit au repos et que l'on effectue une dizaine de mesures sur tous les axes pour en faire une moyenne. Cette moyenne sera considérée comme un décalage à soustraire.

En utilisant la discréttisation par approximation arrière :  $p = \frac{1-z^{-1}}{T}$  où  $T$  est la période d'échantillonnage.

L'équation de l'angle d'inclinaison dans le domaine de Laplace :  $\theta_{Gy}(p) = \frac{1}{p} \Omega_{Gy}(p)$

$\theta_{Gy}(z) = \frac{T}{1-z^{-1}} \Omega_{Gy}(z) \leftrightarrow \theta_{Gy}(z)(1 - z^{-1}) = T\Omega_{Gy}(z) \leftrightarrow \theta_{Gy}(z) = z^{-1}\theta_{Gy}(z) + T\Omega_{Gy}(z)$   
 $z^{-1}$  représente un retard d'une période

D'où l'équation discrète de l'angle d'inclinaison  $\theta_{Gy}(k) = \theta_{Gy}(k-1) + T\Omega_{Gy}(k)$

## Filtre de Kalman

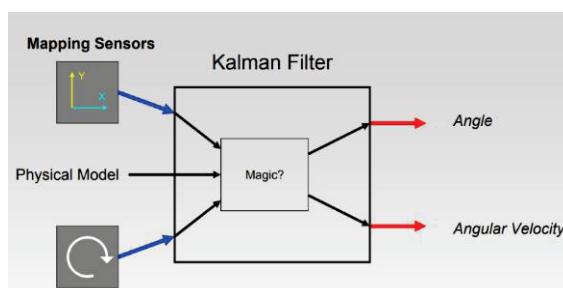


Figure 53

Le filtre de Kalman est tout indiqué pour la fusion des données. Son approche de manière théorique est difficile.

### Avantages:

- Filtre théoriquement idéal pour combiner des capteurs bruyants pour obtenir des estimations claires et précises.
- Tient compte des propriétés physiques du système (masse, inertie, etc.).

### Les inconvénients:

- Mathématiquement complexe, nécessitant quelques connaissances de l'algèbre linéaire. Aussi, différentes formules pour différentes situations.

- difficile à coder.
- le temps de processeur.

### Filtre complémentaire

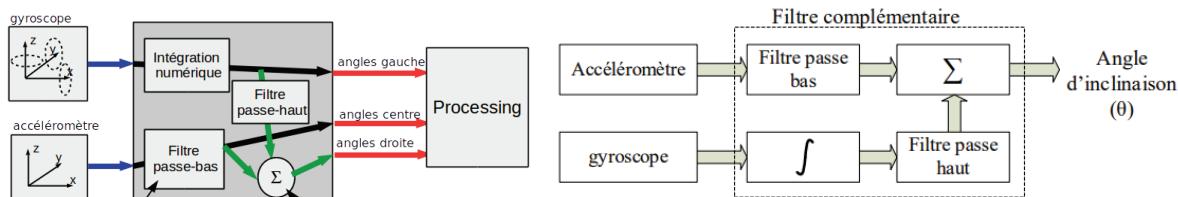


Figure 54

Le filtre numérique passe bas intégré n'élimine pas les bruits de basse fréquence engendrés par le gyroscope. Pour ce type d'application, on a souvent recours à utiliser le filtre de Kalman ou Le filtre complémentaire dont le schéma bloc est donné par la figure 54, consiste à combiner après filtrage les données issues de l'accéléromètre et du gyroscope pour obtenir une meilleure estimation de l'angle d'inclinaison.

Le passage en temps discret donne :  $\theta_{Gy}(k) = \beta (\theta_{Gy}(k-1) + T\Omega_{Gy}(k)) + (1 - \beta)\theta_{ACC}(k)$

avec  $\beta = \frac{\tau}{\tau+T}$ . La formule du filtre complémentaire peut être décomposée comme :

- angle = (angle + gyro \* dt) est une intégration
- La multiplication de l'intégration par 0.98 agit comme un filtre passe-haut
- Le terme ajouté à l'intégration agit comme un filtre passe-bas des données de l'accéléromètre

#### Avantages:

- Peut aider à corriger le bruit.
- Estimations rapides de l'angle, beaucoup moins de décalage que le filtre passe-bas seul.
- Peu d'utilisation des ressources de processeur.

#### Les inconvénients:

- Un peu plus de théorie à comprendre que les filtres simples, mais rien de tel que le filtre de Kalman

### 3. Programmation

#### 3.1. A partir d'un accéléromètre

```

1.      accel_angle_x = atan2(ax,sqrt(az*az+ay*ay)) *180/PI;
2.      accel_angle_y = atan2(ay,sqrt(az*az+ax*ax)) *180/PI;
3.      accel_angle_z = atan2(sqrt(ay*ay+ax*ax),az) *180/PI;

```

Problème : l'accéléromètre est sensible à la moindre vibration. Ainsi, apparaissent des mesures "parasites".

### 3.2. Filtre pass bas

```

1. float accel_angle_x=0.0,accel_angle_y=0.0,accel_angle_z=0.0;
2. accel_angle_x = 0.75*accel_angle_x+0.25*atan2(ax,sqrt(az*az+ay*ay));
3. accel_angle_y = 0.75*accel_angle_y+0.25*atan2(ay,sqrt(az*az+ax*ax));
4. accel_angle_z = 0.75*accel_angle_z+0.25*atan2(sqrt(ay*ay+ax*ax),az);

```

### 3.3. Intégration numérique

Dans cette partie, nous allons déterminer les angles d'inclinaison autour de l'axe Y et de l'axe X à partir du gyroscope. Le gyroscope nous donne la vitesse angulaire en rad/s. Pour le cas de l'inclinaison par rapport à l'axe Y, la vitesse de rotation autour de l'axe Y est :  $G_y = \frac{dangle}{dt}$  d'où  $dangle = G_y dt$  donc  $angle("actuel") = angle(" précédent") + G_y \cdot dt$ . Selon la datasheet, on voit que la sensibilité est de 131 et , le dt est fixé à 10ms=0,01s. Par conséquent, l'équation finale de l'angle est donc :  $angle = angle + G_y \cdot 0.01/131$

```

5. angle=angle+float(gy)*0.01/131; // 131 cf doc MPU6050 et 0.01=dt en s

```

Quand les relevés sont décalés par rapport à la réalité. Il convient alors d'estimer ce décalage pour le compenser. Pour faire ça, on va simplement prendre un lot de mesures et **calculer leur valeur moyenne**. Ça nous donnera l'offset moyen à appliquer sur chaque axe.

Exemple de code :

```

1. void calibrateMpu6050()
2. {
3.     int max_samples = 2000;
4.
5.     for (int i = 0; i < max_samples; i++) {
6.         readSensor();
7.
8.         gyro_offset[X] += gyro_raw[X];
9.         gyro_offset[Y] += gyro_raw[Y];
10.        gyro_offset[Z] += gyro_raw[Z];
11.
12.        // Just wait a bit before next loop
13.        delay(3);
14.    }
15.
16.    // Calculate average offsets
17.    gyro_offset[X] /= max_samples;
18.    gyro_offset[Y] /= max_samples;
19.    gyro_offset[Z] /= max_samples;
20. }

```

On obtient ainsi les angles Pitch et Roll

```

1. angle[PITCH] += (gyro_raw[X] - gyro_offset[X]) / (Frec * 131);
2. angle[ROLL]   += (gyro_raw[Y] - gyro_offset[Y]) / (Frec * 131);

```

*Malheureusement, au fur et à mesure, le gyroscope dérive et n'est donc pas fiable.*

### 3.4. Filtre complémentaire

On a vu précédemment que le gyroscope dérive et l'accéléromètre présente des mesures brusques "parasites". On va donc combiner les 2 en se débarrassant des variations "lentes" du gyroscope (filtre passe haut) et des variations "rapides" de l'accéléromètre (filtre passe-bas).

Pour cela, on utilise un "filtre complémentaire". Ainsi, l'équation de l'angle va être :

$$\text{angle} = 0.98 \times (\text{angle} + G_y \cdot 0.01/131) + 0.02 \times (\text{angle de l'accéléromètre en } ^\circ)$$

On peut affiner les coefficients 0,98 et 0,02 du moment que leur somme reste égale à 1

Exemple de code :

```
// avec filtrage pass bas
    accel_angle_x =
0.75*accel_angle_x+0.25*atan2(ax,sqrt(az*az+ay*ay));
    accel_angle_y =
0.75*accel_angle_y+0.25*atan2(ay,sqrt(az*az+ax*ax));
    accel_angle_z =
0.75*accel_angle_z+0.25*atan2(sqrt(ay*ay+ax*ax),az);

// calibration du temps
t_now = millis();
dT = (float)(t_now - last_read_time)/1000;
last_read_time = t_now;

// intégration récursive
unfiltered_gyro_angle_x += ((float)(GyX - base_x_gyro))*dT/FS_SEL;
unfiltered_gyro_angle_y += ((float)(GyY - base_y_gyro))*dT/FS_SEL;
unfiltered_gyro_angle_z += ((float)(GyZ - base_z_gyro))*dT/FS_SEL;
// balanced filter
filtered_gyro_angle_x = 0.98 * (filtered_gyro_angle_x + ((float)(GyX - base_x_gyro))*dT/FS_SEL) + 0.02 * accel_angle_x;
    filtered_gyro_angle_y = 0.98 * (filtered_gyro_angle_y + ((float)(GyY - base_y_gyro))*dT/FS_SEL) + 0.02 * accel_angle_y;
    filtered_gyro_angle_z = 0.98 * (filtered_gyro_angle_z + ((float)(GyZ - base_z_gyro))*dT/FS_SEL) + 0.02 * accel_angle_z;
```

## 4. DigitLal Motion Processor (DMP)

Ce qui fait la richesse de ce capteur intelligent MPU6050 c'est qu'il est associé avec un **DMP**. Il est donc possible d'utiliser ce processeur en lieu et place du filtre complémentaire (déjà rencontré) ou du filtre de Kalman.