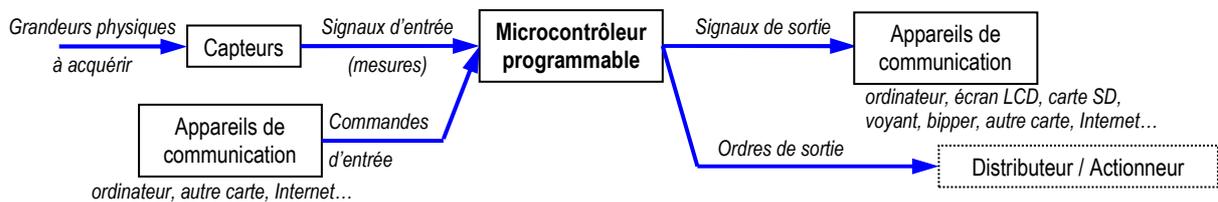


CHAÎNE D'INFORMATION – cours de synthèse

Acquérir (capteurs), Traiter (microcontrôleurs) et Communiquer (signaux)



Exemple de modélisation de la chaîne d'information d'un système mécatronique

Sommaire

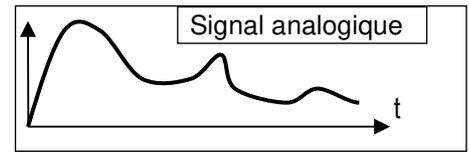
- 1) Nature et codage des signaux2**
 - 1.1) Natures des signaux.....2
 - 1.2) Conversions (CAN et CNA).....2
 - 1.3) Codage3
 - 1.4) Signaux particuliers5
- 2) Cartes programmables.....6**
- 3) Capteurs et mesures : caractéristiques générales.....7**
 - 3.1) Erreur7
 - 3.2) Incertitude7
 - 3.3) Précision : justesse et fidélité8
 - 3.4) Étendue de mesure8
 - 3.5) Résolution.....8
 - 3.6) Sensibilité8
 - 3.7) Influence de perturbations.....8
- 4) Capteurs : les différents types9**
 - 4.1) Détecteurs (capteurs tout ou rien) et leurs différentes technologies9
 - 4.2) Bouton poussoir et résistance de pull-down ou pull-up9
 - 4.3) Codeur absolu de position..... 10
 - 4.4) Codeur incrémental de position 10
 - 4.5) Potentiomètre (analogique) 11
 - 4.6) Capteurs résistifs et pont diviseur de tension..... 11
 - 4.7) Capteur d'effort par jauges d'extensométrie..... 12
 - 4.8) Accéléromètre, gyromètre, magnétomètre... (centrale inertielle) 12
 - 4.9) Autres exemples de capteurs 12

1) Nature et codage des signaux

1.1) Natures des signaux

Analogique : pouvant prendre n'importe quelle valeur.

Exemples : température, position, vitesse, accélération, effort, tension, intensité, résistance, inductance, capacité, luminosité...



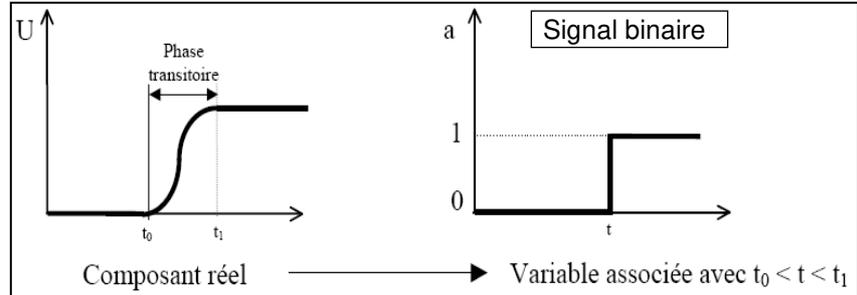
Binaire, ou Logique, ou Booléen, ou « Tout ou rien » (TOR) :

Ne pouvant prendre qu'une valeur parmi deux possibles : {0 ; 1}.

C'est une modélisation du comportement réel, comme indiqué ci-contre.

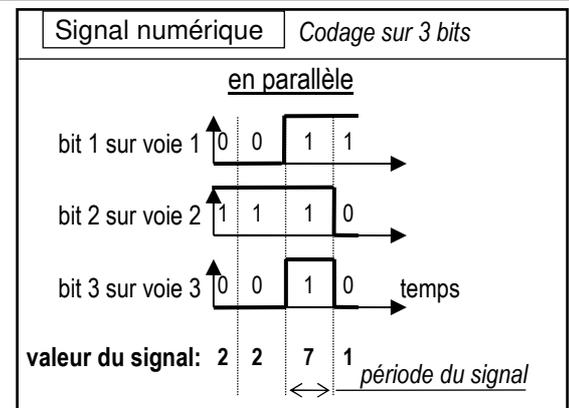
En électronique, une unité binaire est appelée **bit**, tandis qu'un groupement de 8 bits est appelé **octet** (byte en anglais).

Exemples : {absent ; présent}, {éteint ; allumé}, {False ; True}...



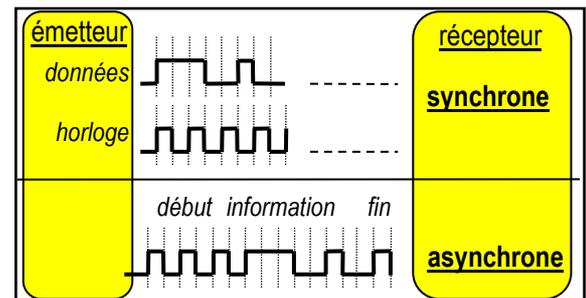
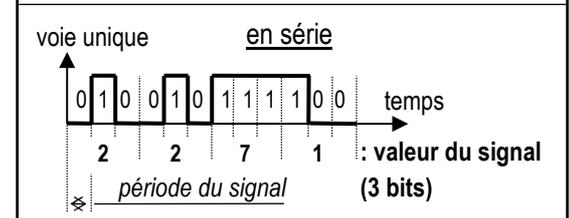
Numérique (Digital) : pouvant prendre une valeur parmi un nombre fini de possibilités. Un signal numérique est constitué d'un ou plusieurs signaux binaires associés, soit :

- en parallèle : transmission simultanée des bits, mais nécessite autant de voies que de bits (peu utilisé) ;
- en série : transmission dans laquelle les éléments d'information se succèdent, les uns après les autres, sur une seule voie – nécessite une fréquence synchronisée entre l'émetteur et le récepteur. Exemples : USB, I²C, Ethernet...



Les signaux numériques en série peuvent être transmis de façon, soit :

- synchrone : L'émetteur envoie un signal d'horloge sur une ligne et l'information sur une autre. Le récepteur reçoit de façon continue (même lorsque aucun bit n'est transmis) les informations au rythme où l'émetteur les envoie. Ex: I²C...
- asynchrone : L'émetteur émet le signal d'horloge et l'information sur la même ligne, la trame de transmission de l'information peut démarrer à n'importe quel moment (par exemple : appui sur une touche...) et elle doit être assez longue, car elle contiendra le signal de début d'émission (permettant de déterminer la fréquence du signal, et de préciser le début de l'information), puis l'information, et enfin un signal de fin d'émission. Ex: Ethernet...



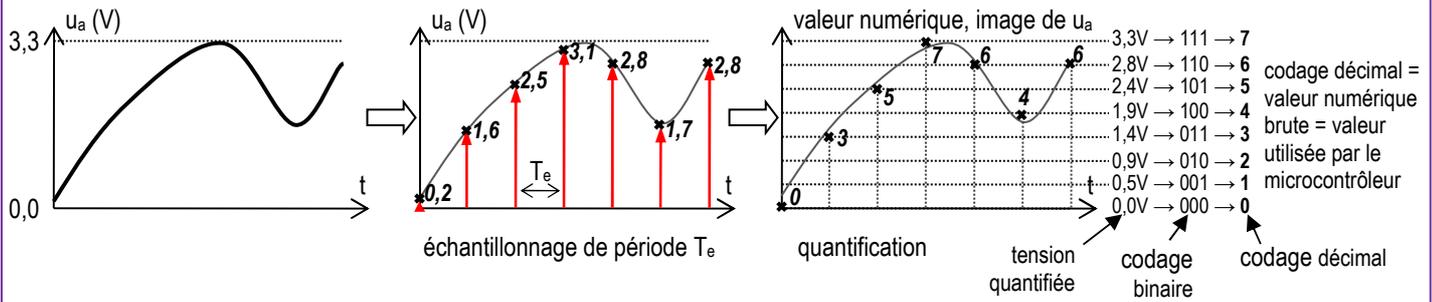
1.2) Conversions (CAN et CNA)

Conversion Analogique-Numérique (CAN) :

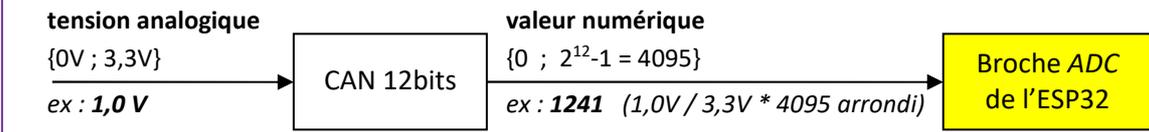
Elle peut être divisée en trois étapes :

- échantillonnage temporel : relève des valeurs à une période fixe ;
- quantification : découpage d'une plage de valeurs continues en 2^n valeurs possibles (n : nombre de bits du CAN) dont les valeurs numériques brutes varient de 0 à $2^n - 1$;
Exemple d'un CAN 3 bits : il peut coder $2^3 = 8$ valeurs possibles, nombres entiers de 0 à 7.
- codage : souvent en code binaire naturel ou en son équivalent décimal : valeur numérique brute entière comprise entre 0 et $2^n - 1$, parfois nommée *incrément* ou *points*... – cf. ci-dessous.

Exemple de la **conversion analogique-numérique** sur 3 bits d'une tension pouvant varier entre 0V et +3,3V :



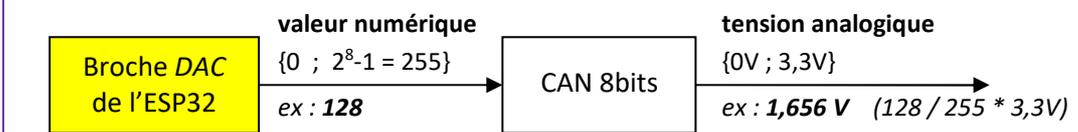
Exemple du **CAN 12 bits** sur l'entrée analogique ADC (entre 0V et +3,3V) d'une carte ESP32 :



Conversion Numérique-Analogique (CNA) :

Réalise l'opération inverse du CAN :

Exemple du **CNA 8 bits** sur la sortie analogique DAC (entre 0V et +3,3V) d'une carte ESP32 :



1.3) Codage

Nécessité du codage et du décodage :

Les circuits logiques (en particulier les microprocesseurs) ne peuvent traiter que des informations **numériques**. Ainsi, tout problème, avant d'être traité par ordinateur, doit être transcrit sous forme numérique : cette opération s'appelle le **codage**.

Le résultat du calcul est donné sous forme numérique par le calculateur. Il faut alors le retranscrire dans le langage original, exploitable par l'humain : cette opération s'appelle le **décodage**.

Exemple :



Bases de numération :

Dans une base, un nombre est représenté par la juxtaposition de symboles appelés digits pris parmi les éléments de la base B considérée.

Les bases les plus couramment utilisées sont :

- la base 10 (décimale) contenant 10 symboles : 0, 1, 2, ..., 9
- la base 2 (binaire) contenant 2 symboles : 0, 1
- la base 16 (hexadécimale) comptant 16 symboles : 0, ..., 9, A, ..., F

Exemples :

- (10)₁₀ (ou **0d10**, lire « dix ») représente le nombre 10 en base 10 ;
- (10)₂ (ou **%10** ou **0b10**, lire « un, zéro ») représente le nombre 2 écrit en base 2 ;
- (10)₁₆ (ou **\$10** ou **0x10**) représente le nombre 16 écrit en base 16.

Passage d'une base X vers la base décimale :

Un nombre (abcd)_x (avec a,b,c,d quatre digits) dans une base X correspond au nombre suivant en base décimale : **(abcd)_x = a.X³ + b.X² + c.X¹ + d.X⁰**

Base décimale B ₁₀	Base binaire B ₂	Base hexadécimale B ₁₆
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

On parle de système de numérotation pondéré. On appelle poids le rang de chaque digit (exemple : « a » digit de poids le plus fort).

Exemples : **Binaire vers décimal** : $(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (11)_{10}$

Hexadécimal vers décimal : $(A3B)_{16} = 10 \times 16^2 + 3 \times 16^1 + 11 \times 16^0 = (1619)_{10}$

Passage d'une base décimale vers une base X :

On utilise la méthode des divisions successives.

Soit un nombre N défini sur une base X et comprenant n digits. Son écriture en base 10 est la suivante :

$$(N)_X = n_p \cdot X^p + n_{p-1} \cdot X^{p-1} + \dots + n_1 \cdot X^1 + n_0 \cdot X^0$$

Effectuer le changement de base revient à déterminer $n_0, n_1, \dots, n_{p-1}, n_p$

En mettant X en facteur il vient :

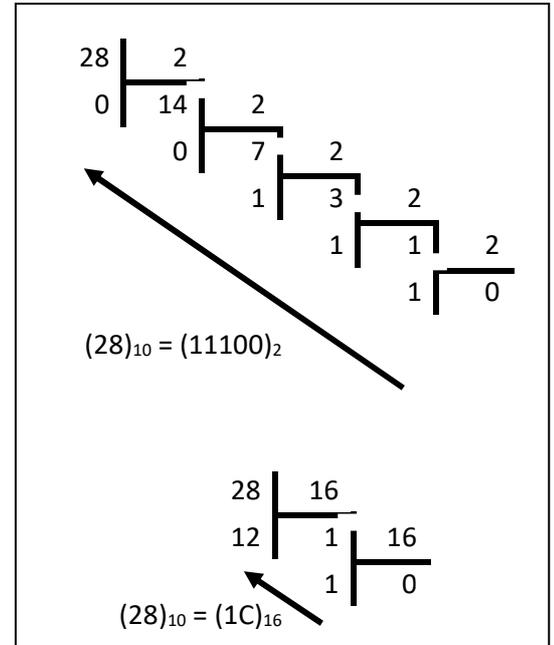
$$(N)_X = X \cdot (n_p \cdot X^{p-1} + n_{p-1} \cdot X^{p-2} + \dots + n_1 \cdot X^0) + n_0$$

qui est de la forme : $(N)_X = X \cdot (\text{quotient}) + \text{reste}$

Ceci permet de déterminer n_0 en prenant le reste de la division euclidienne de $(N)_X$ par X . En reprenant le quotient précédent et en factorisant de nouveau par X on trouve un nouveau reste n_1 et ainsi de suite jusqu'à n_p .

Exemples : Pour écrire 28 en binaire (cf. ci-contre), il faut lui faire des divisions successives par 2. Le sens de lecture, et donc d'écriture, se fait selon la flèche. 28 s'écrit donc en binaire « 0b11100 ».

Pour écrire 28 en hexadécimal (cf. ci-contre), il faut lui faire des divisions successives par 16. Ainsi $28 = 0x1C$.



Les codes :

Il est parfois utile de regrouper un ensemble de valeurs binaires suivant une autre organisation qu'un système de nombres. Ces autres organisations sont appelées des **codes**. Il en existe un très grand nombre et chacun peut en créer selon un besoin spécifique.

Les qualités requises pour un code sont principalement :

- la taille du codage (nombre de bits nécessaires) ;
- la fiabilité de lecture ;
- la simplicité de manipulation.

Exemple : Code ASCII (American Standard Code For Information Interchange) : Il permet de coder 128 caractères sur 8 bits (un octet). La lettre A est codée 65 en décimal, 01000001 en binaire, 41 en hexadécimal. C'est la norme la plus répandue de codage des caractères.

On distingue un code pondéré pour lequel chaque chiffre possède un poids (ex : le **code binaire naturel**, qui est identique à la base binaire) d'un code non pondéré qui nécessite une table de correspondance pour décoder un nombre (ex : ASCII, code Gray).

Code binaire réfléchi, ou code Gray

Ce code non pondéré est un arrangement du système binaire. Le passage d'un nombre à son suivant se fait **en changeant l'état d'un seul bit**. Il est très utilisé pour décrire des automatismes (un changement d'état d'un composant correspond à un bit qui change), en particulier dans les codeurs de position absolue.

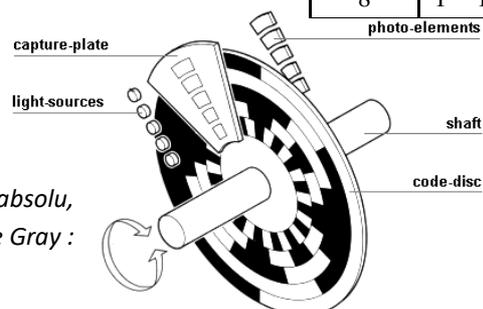
Pour obtenir le code Gray, il faut faire toutes les deux (2^1), quatre (2^2), huit (2^3),... lignes, une symétrie en changeant la valeur du bit de gauche.

Décimal	Binaire réfléchi
0	0 0 0 0
1	0 0 0 1
2	0 0 1 1
3	0 0 1 0
4	0 1 1 0
5	0 1 1 1
6	0 1 0 1
7	0 1 0 0
8	1 1 0 0

Avantage : Il apporte une garantie d'interprétation avec une erreur maximale d'une incrémentation.

L'inconvénient de ce code non pondéré est qu'il est nécessaire de stocker la table de correspondance pour décoder un nombre.

Codeur de position rotatif absolu, en codage Gray :



Code décimal codé binaire (DCB)

C'est un code pondéré basé sur le code binaire naturel, mais qui est adapté à la représentation des nombres en base 10. La propriété du code DCB est d'associer quatre bits différents à chaque puissance de 10. Ainsi $(1664)_{(10)} = (0001.0110.0110.0100)_{(DCB)}$. Il est utilisé pour les afficheurs 7 segments.



1.4) Signaux particuliers

MLI (Modulation de largeur d'impulsion), ou PWM en anglais (Pulse Width Modulation)

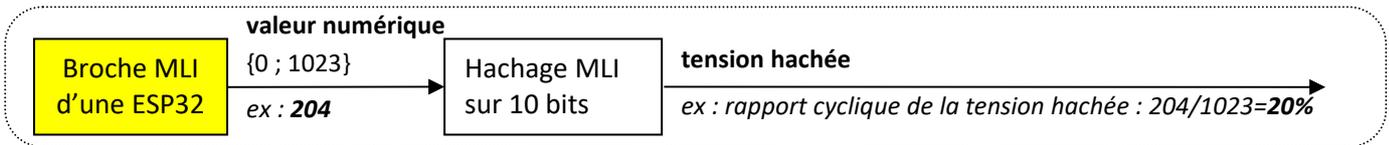
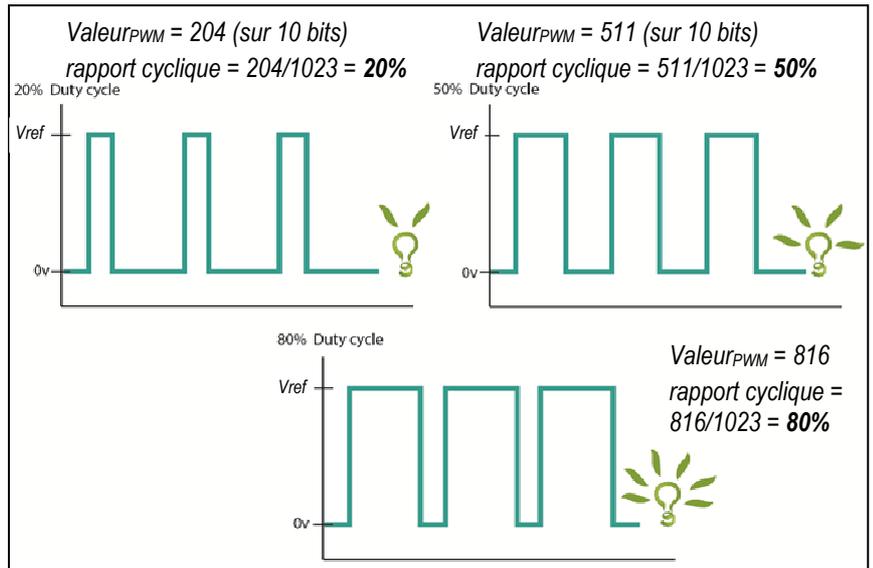
La MLI est une technique permettant de créer des signaux **pseudo analogiques** à l'aide de circuits à fonctionnement tout ou rien.

Les sorties MLI peuvent émettre très rapidement un signal haché valant $0V$ ou V_{ref} , cf. schéma ci-contre.

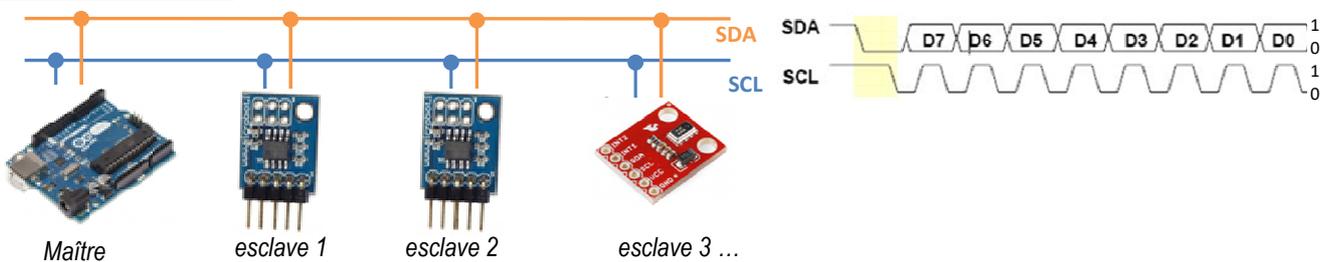
Le récepteur branché à cette broche (LED, moteur...) fonctionne comme s'il recevait la tension moyenne, à condition que son temps de réponse soit supérieur à la période de hachage.

La valeur de la tension moyenne est proportionnelle au **rapport cyclique** choisi.

Par exemple (ci-dessous), les cartes ESP32 ont une MLI sur 10 bits, donc une valeur numérique pwm pouvant varier entre 0 et 1023 ($2^{10}-1$). Afin d'envoyer un rapport cyclique de 20%, il faudra donc donner la valeur pwm de 204 ($1023*0,2$) à la broche.



Communication I²C



La communication I²C fonctionne globalement ainsi (signal numérique en série sur SDA, synchronisé par SCL) :

- L'horloge SCL (= *Serial Clock*) permet de cadencer les instants où les informations sont à prendre en compte : à chaque front descendant de SCL le bit suivant est écrit, et à chaque front montant le bit actuel est lu.
- Le maître envoie sur la ligne SDA (= *Serial Data*) l'adresse du composant avec qui il souhaite communiquer, chacun des esclaves ayant une adresse fixe et distincte.
- L'esclave qui reconnaît son adresse répond à son tour par un signal de confirmation (acquiescement) sur SDA, puis le maître continue la procédure de communication (écriture/lecture) sur SDA.

Inconvénients : consommation d'énergie ; lenteur relative ; longueur maximale du bus : 2m ; un seul maître (qui démarre la communication).

Avantages : seulement 2 voies dédiées (en plus de la masse et de V_{ref}) pour 128 appareils connectables (car l'adresse I²C est codée sur 7 bits et $2^7 = 128$).

Communication SPI (très proche du I2C) :

- inconvenient du SPI : autant de broches nécessaires pour les lignes dédiées vers chacun de ses esclaves ;
- avantages du SPI : plus rapide et souple que l'I2C.

Communication en réseau (avec en particulier le protocole TCP/IP) : cf. cours de synthèse sur le TP éponyme.

3) Capteurs et mesures : caractéristiques générales

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable appelée mesure (souvent une tension électrique analogique ou une valeur numérique).

3.1) Erreur

Lors de l'utilisation d'un capteur dans un protocole expérimental, on distinguera deux sources d'erreurs :

- **l'erreur de mesure** décrit l'écart entre la valeur affichée et la valeur « correcte », elle n'est due qu'au capteur ;
- **l'erreur de méthode** est due à un protocole expérimental défectueux, une manipulation imprécise, ou encore une mauvaise exploitation des mesures, et n'est pas due au capteur lui-même.

Les **erreurs systématiques** sont celles qui n'ont aucun caractère aléatoire. La mesure dans des conditions identiques donne toujours la même erreur par rapport à la valeur vraie.

Elles sont dues à : l'étalonnage (erreur sur l'offset ou le gain), ou la non-linéarité du capteur, cf. trois schémas ci-contre.

Les **erreurs aléatoires** sont celles qui varient à chaque mesure supposée identique.

Elles sont dues : au bruit du capteur, aux erreurs de lecture (aiguille, résolution), aux perturbations non détectables (courants électriques parasites...).

Pour les évaluer, on fait appel à des méthodes statistiques : valeurs moyennes, écart-type... sur un grand nombre de mesures. cf. **fidélité**.

3.2) Incertitude

En pratique une erreur ne peut qu'être estimée, et on préférera étudier une incertitude : écart évalué statistiquement par rapport à la valeur vraie.

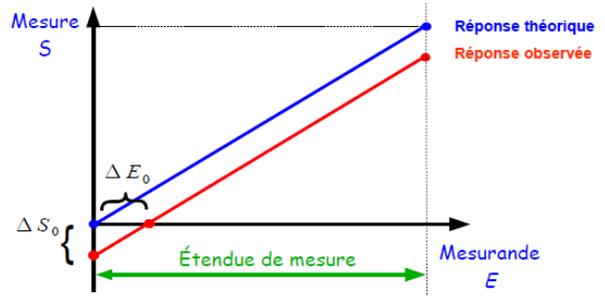
On appelle incertitude de mesure ΔX , la limite supérieure de la valeur absolue de l'écart entre la valeur mesurée et la valeur exacte de la mesurande notée $\Delta X = \sup |X_{\text{mesuré}} - X_{\text{exact}}|$.

l'incertitude (absolue) ΔX a la même unité que la grandeur mesurée ;

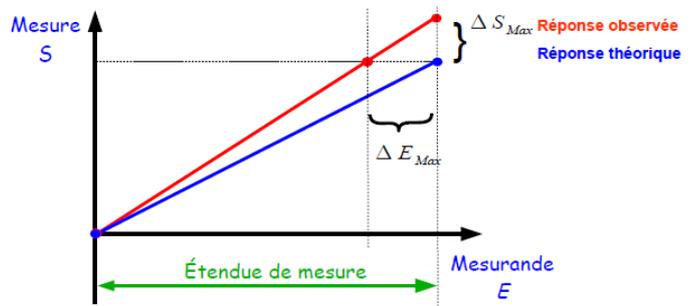
l'incertitude relative $\Delta X/X$ s'exprime en %.

Les incertitudes doivent apparaître sur **l'affichage d'une courbe**, sous forme de barres en chaque point de mesure, une horizontale pour l'incertitude sur l'abscisse, et une verticale pour l'incertitude sur l'ordonnée (cf. ci-contre).

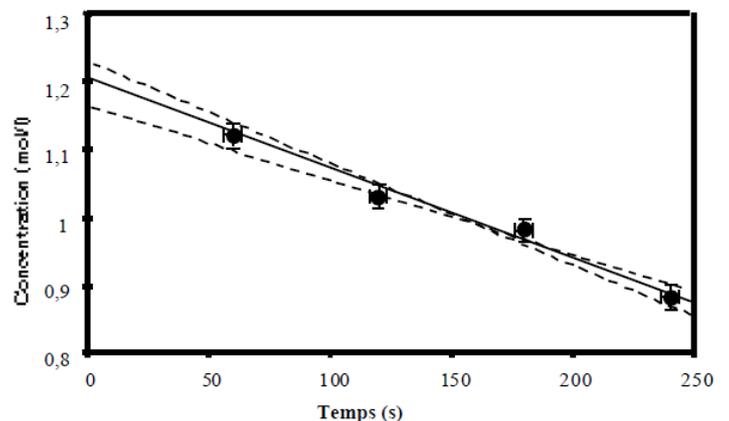
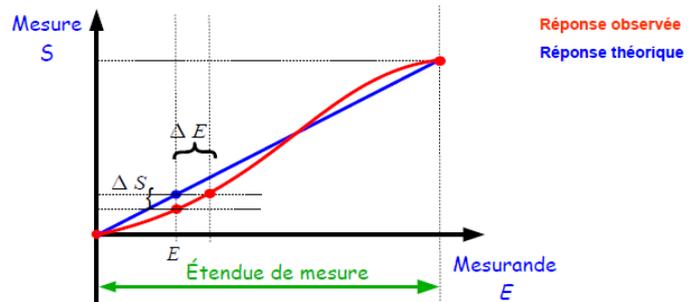
- **L'erreur d'offset** ou **décalage** est la différence entre la valeur « vraie » de la mesure et celle obtenue à partir de la réponse du capteur pour la **borne inférieure** de l'étendue de mesure



- **L'erreur de gain** est l'erreur de pente de la courbe caractéristique du capteur; elle est visible essentiellement pour la **borne supérieure** de l'étendue de mesure.

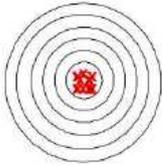


- **L'erreur de linéarité** est l'erreur entre la courbe caractéristique du capteur et la droite théorique de réponse.



3.3) Précision : justesse et fidélité

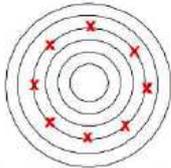
Un capteur est précis s'il est juste et fidèle.



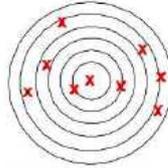
Juste et fidèle → précis



Fidèle, non juste



Juste, non fidèle



ni fidèle, ni juste

Justesse

Aptitude à donner des indications égales à la grandeur mesurée, les erreurs de fidélité n'étant pas prises en considération. Un capteur est juste si l'écart entre la moyenne des résultats et la valeur « vraie » est faible, même si l'écart-type est grand.

Fidélité

Aptitude à donner, pour une même valeur de la grandeur mesurée, des indications voisines entre elles. Un capteur est fidèle si l'écart-type sur les réponses à une même valeur de la grandeur mesurée est faible.

3.4) Étendue de mesure

L'étendue de mesure est la zone nominale d'emploi : zone dans laquelle les caractéristiques du capteur correspondent aux spécifications de fonctionnement normal.

3.5) Résolution

La résolution est la plus petite variation de la mesure qu'il est possible d'observer.

Pour un appareil numérique, la résolution est l'étendue de mesure divisée par le nombre de points de la mesure moins un. Exemple : une position captée entre 0 et 100mm, puis codée par un CAN-10-bits aura une résolution de « $100\text{mm} / (2^{10}-1) = 100/1023 = 0,0977\text{mm}$ »

3.6) Sensibilité

En régime établi (statique) du capteur, la sensibilité en un point de mesure est le rapport entre la variation de sortie sur la variation d'entrée autour de ce point de mesure. C'est donc la pente en un point de mesure de la courbe statique $\text{Sortie} = f(\text{Entrée})$. C'est aussi sa fonction de transfert linéarisée autour de ce point de mesure.

$$S = \frac{\Delta \text{Sortie}}{\Delta \text{Entrée}}$$

Un capteur est dit **linéaire** si sa sensibilité est constante sur l'étendue de mesure.

3.7) Influence de perturbations

Certaines perturbations peuvent faire varier les caractéristiques d'un capteur, comme la **température** d'utilisation, la **pression** ambiante, l'**humidité**, des éventuelles **poussières**... C'est pour cela que souvent les fabricants fournissent des conditions d'utilisation à respecter pour que les caractéristiques données soient fiables.

De plus, la plupart des capteurs nécessitent d'être **étalonnés** (calibrés) avant chaque utilisation, afin de régler leur justesse selon les conditions ambiantes.

4) Capteurs : les différents types

4.1) Détecteurs de présence (tout ou rien) et capteurs de proximité (TOR ou analogique)

Mesure via une carte programmable :

3 fils : il faut leur fournir la masse ($GND = 0V$) et une tension ($V_{ref} = 3,3V$ ou $5V$ en général), puis de lire la seule voie de sortie (voie de détection) en la reliant à une broche d'entrée digitale d'une carte programmable :

- Lorsque la voie est à $0V$ (signal booléen 0 , *low*, ou *False*), le détecteur ne détecte pas la grandeur physique,
- Tandis que lorsque la voie est à V_{ref} (signal booléen 1 , *high* ou *True*) la grandeur physique est détectée.

Si la voie lue possède une tension intermédiaire (entre 0 et V_{ref}), cela permet d'en déduire la distance : le capteur est alors un capteur de proximité analogique.

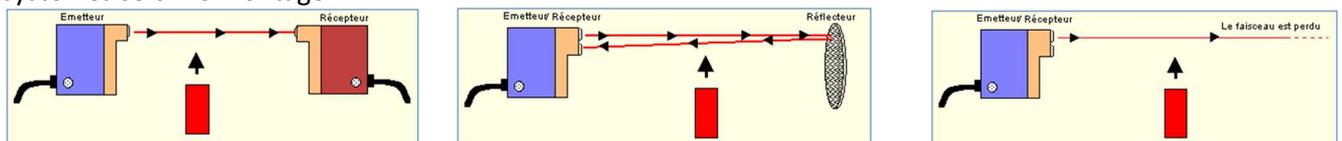


Détecteur par contact (interrupteur électromécanique, *TOR*) : permet de détecter des objets par fermeture d'un circuit électrique : cf. « bouton poussoir » ci-dessous.

Capteur inductif : permet de détecter sans contact des objets métalliques à une distance de 0 à 60 mm, par production d'un champ magnétique oscillant (via une inductance et un condensateur). *TOR* ou *analogique*.

Capteur capacitif : permet de détecter sans contact des objets quelconques à proximité, par création d'un champ électrostatique et détection des modifications de ce champ. *TOR* ou *analogique*.

Capteur photoélectrique (TOR ou analogique) : permet de détecter un objet pouvant être très éloigné. Il est constitué d'un émetteur (E) de lumière (souvent infrarouge *IR*) et d'un récepteur (R). Il existe plusieurs systèmes selon le montage :



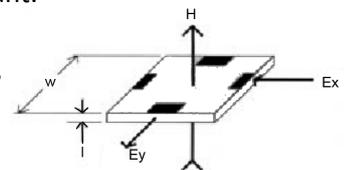
barrage (E et R alignés)

réflexion sur un réflecteur ou réflexion sur l'objet (E et R ensembles)

Capteur par ultrason : même principe que le photoélectrique mais avec un son. *TOR* ou *analogique*.

Capteur magnétique : un aimant qui se déplace est détecté par une bobine électroaimant.

Capteur à effet Hall : le capteur est alimenté et parcouru par un courant le long de la direction x. Lorsqu'un aimant se trouve à proximité, un champ magnétique H est alors appliqué perpendiculairement à la surface du capteur, et cela va générer une tension le long de la direction y (qu'il suffit de lire pour détecter la proximité d'un aimant). La tension générée dépend de la proximité et de l'intensité du champ magnétique, il est donc possible d'utiliser l'effet Hall dans des capteurs analogiques de proximité (*TOR* ou *analogique*).



4.2) Bouton poussoir et résistance de pull-down ou pull-up

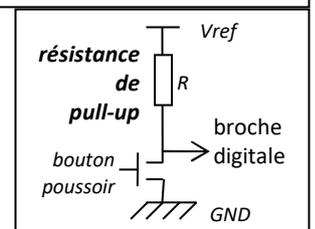
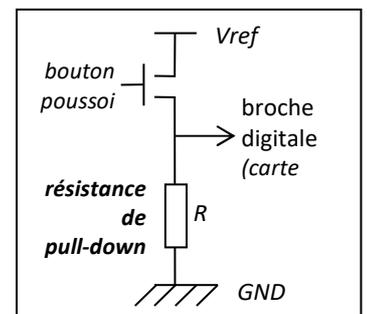
Le principe d'un **bouton poussoir** est :

- lorsqu'on appuie, l'interrupteur du bouton poussoir se ferme ;
- lorsque l'on relâche, l'interrupteur s'ouvre. Ou parfois inversement.

Un bouton poussoir a besoin d'une résistance de pull-down (ou de pull-up), et il est aussi préférable de lui adjoindre un condensateur pour éviter l'effet rebond.

La résistance de pull-down est nécessaire pour relier la broche digitale à la masse GND lorsque le bouton poussoir est libre (interrupteur ouvert), sinon son comportement serait aléatoire (sa tension pourrait rester à V_{ref} , ou autre...).

En revanche, si l'on souhaite que la valeur par défaut lue par la broche digitale soit V_{ref} (booléen à 1), alors il faudra monter le bouton poussoir avec une **résistance de pull-up**.



4.3) Codeur absolu de position (numérique)

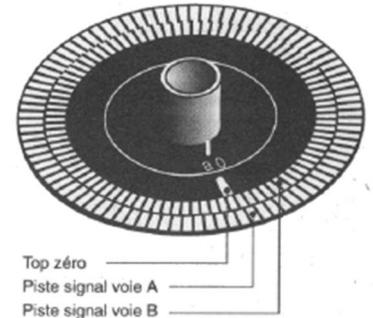
Il permet de connaître la position (angulaire ou linéaire, selon le codeur) à chaque instant. Nécessite autant de fils que de voies (monté en parallèle). Souvent codé en **binaire réfléchi (code Gray)** car cela évite les erreurs de lecture lors d'un changement de position, cf. page 4 (codage).



4.4) Codeur incrémental de position (numérique)

Il permet de connaître la **variation de position** (angulaire ou linéaire, selon le codeur) au cours du temps, et donc d'en déduire éventuellement la vitesse.

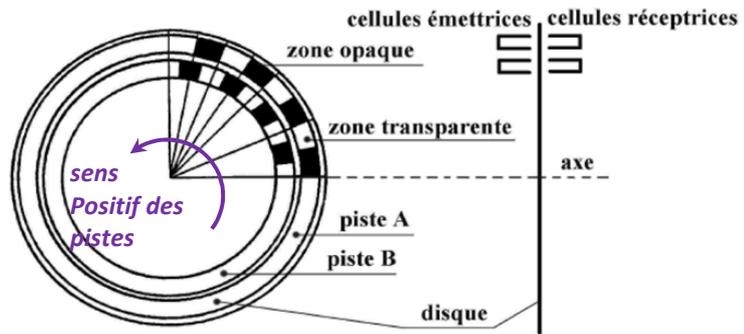
Si l'on souhaite en déduire la position absolue, il faudra alors **compter** le nombre de « variations » (incrément) depuis une position de référence, qui nécessite donc soit un détecteur de présence en plus, soit une voie supplémentaire sur ce même codeur pour la remise à zéro (cf. schéma ci-contre).



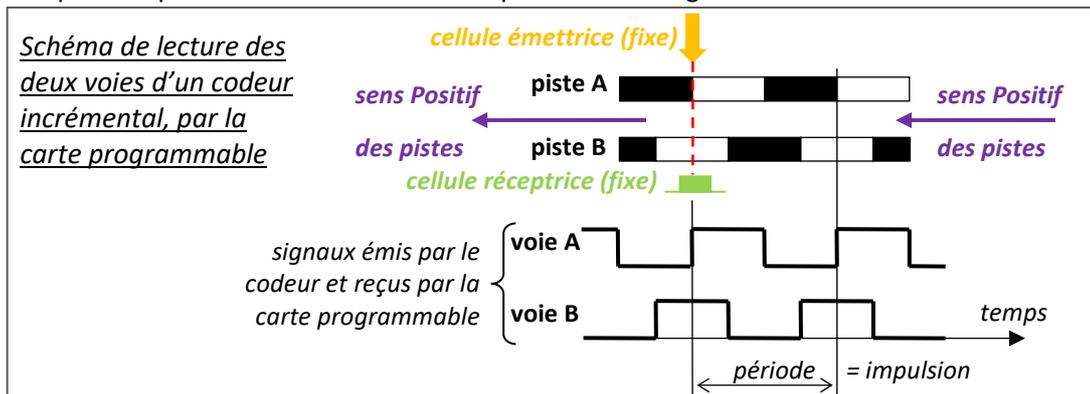
Deux technologies sont très utilisées, mais le principe reste le même pour le microcontrôleur : optique (par LED) ou à effet hall (par champs magnétiques).

Principe de fonctionnement (optique) : Ce sont des capteurs de position angulaire comportant :

- ✓ Un **disque optique** mobile avec 2 pistes (A et B) comportant chacune une succession de parties opaques et transparentes.
- ✓ Deux **cellules fixes**, pour chaque piste : une cellule émettrice de lumière d'un côté et une réceptrice de l'autre.



Chaque passage d'une zone transparente à une autre est détecté par les cellules réceptrices. Les 2 pistes sont décalées d'un quart de période et la rotation du disque donne les signaux suivants :



Les codeurs incrémentaux à 2 voies permettent 3 niveaux de précision d'exploitation :

- Utilisation des fronts montants de la voie A seule : permet de connaître le sens de rotation grâce à une lecture de la voie B en même temps, mais peut générer des erreurs autour d'une position fixe, donc à n'utiliser que pour déterminer la vitesse du moteur ;
- Utilisation des fronts montants et descendants de la voie A seule : permet de connaître le sens de rotation grâce à une lecture de la voie B en même temps, cf. ci-dessous ;
- Utilisation des fronts montants et descendants des voies A et B : le plus précis car la résolution est multipliée par 2, mais aussi le plus gourmand en temps de calcul.

Algorithme de lecture de la position :

Exemple pour une détection des fronts montants et descendants de la voie A seule.

Afin de détecter le sens de rotation, deux voies de codage sont nécessaires au minimum.

Voici l'algorithme minimal permettant de connaître la variation de position au cours du temps. Cet algorithme doit s'effectuer à **chaque front montant ou descendant de la voie A**, et il faudra donc créer une interruption sur la voie A afin que cet algorithme s'exécute immédiatement lorsqu'un changement de la voie A est détecté.

```
def Incrementer(pin):
    if voieA.value() == voieB.value():
        increments += 1
    else:
        increments -= 1
```

Détails sur les paramètres :

- *voieA.value()* renvoie la valeur actuelle de la voie A ; (idem pour la voie B) ;
- *increments* est un nombre entier comptant le nombre de variations de la voie A autour du sens de déplacement positif.

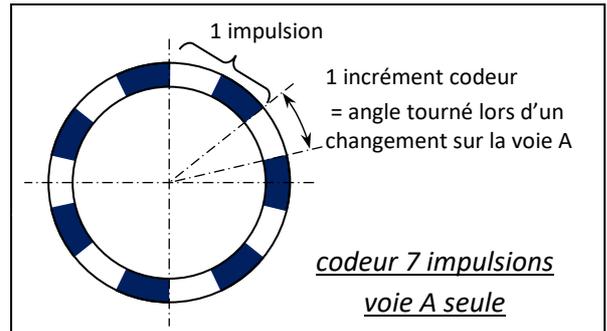
Afin de connaître la **position angulaire** (en degré), il faut connaître la position initiale (Position_Init) et la valeur angulaire d'un **incrément**, c'est-à-dire la résolution de ce codeur : angle dont le codeur doit tourner pour détecter un incrément suivant (un changement suivant de la voie A).

Exemple illustré ci-contre d'un codeur avec 7 impulsions : un

incrément vaut $1/14^e$ de tour, soit $360^\circ/14 = 25,714^\circ$

Et ainsi on complète l'algorithme avec :

$$Position = Position_Init + increments * 25.714$$



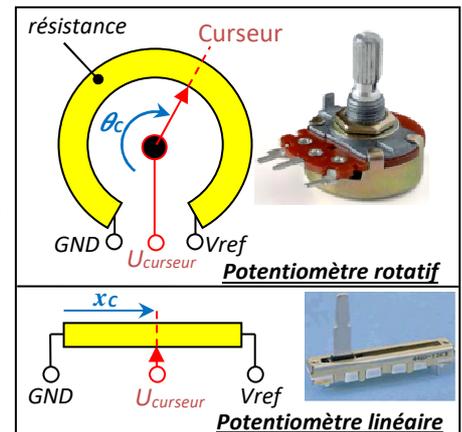
4.5) Potentiomètre (analogique)

Le potentiomètre est une résistance variable à trois bornes, dont une est reliée à un curseur se déplaçant sur une piste résistive terminée par les deux autres bornes (reliées à la masse GND et à Vref).

En connectant la borne du curseur à une entrée analogique d'un microcontrôleur, on peut lire la tension analogique $U_{curseur}$ qui est proportionnelle à la position du curseur.

Exemple : potentiomètre rotatif de 0° à 285° . Signal numérique lu par le microcontrôleur (CAN-12-bits) = 2457 $\Rightarrow 2457/4095=60\%=0,6$

$$\Rightarrow U_{curseur} = 0,6 * V_{ref} \Rightarrow Position \text{ du curseur } \theta_c = 0,6 * 285^\circ = 171^\circ$$

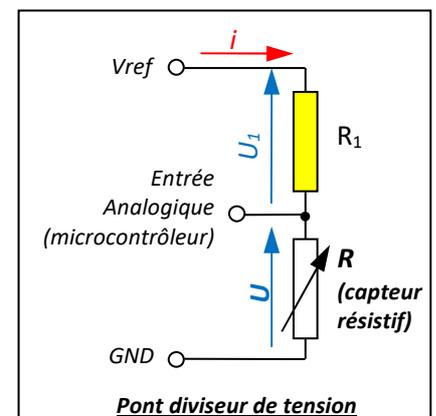


4.6) Capteurs résistifs et pont diviseur de tension (analogiques)

Les **capteurs résistifs** ont une résistance **R** qui varie en fonction de la grandeur mesurée. Exemples de capteurs résistifs :

- *photo-résistifs* : capteurs de luminosité ;
- *thermo-résistifs* : capteurs de température ;
- *élongo-résistifs* : capteurs de déformation ;
- *piézo-résistifs* : capteurs de pression ou de force ;
- *magnéto-résistifs* : capteurs de proximité magnétique...

Ils nécessitent généralement un montage en **pont diviseur de tension**, dont voici le schéma ci-contre. L'entrée analogique permet à un microcontrôleur de lire la valeur de la tension **U** (entre 0V et V_{ref}) aux bornes du capteur.



Grâce à la loi des mailles, on en déduit facilement la valeur de la résistance : $R = R_1 * \frac{U}{(V_{ref}-U)}$ dont on peut déduire la valeur de la grandeur à mesurer grâce à sensibilité du capteur (connue selon le capteur choisi).

4.7) Capteur d'effort par jauges d'extensométrie (analogique)

Un capteur d'effort par jauges d'extensométrie montées en **pont de wheatstone** fonctionne suivant le principe suivant : l'effort appliqué déforme une poutre sur laquelle sont collées des **résistances qui sont variables selon leur allongement**. Le pont de wheatstone (cf. schéma ci-contre), alimenté par une tension continue V_{ref} (3.3v ou 5v) fournit alors une tension (entre les fils S+ et S-) proportionnelle à la déformation et donc à l'effort appliqué, mais qui est très faible (**quelques mV**) et pouvant être **positive ou négative**. Cette tension ne peut pas être lue directement par une carte programmable, il existe deux technologies principales pour la lire, toutes deux nécessitant un calibrage.

Technique 1 - Amplification analogique (et décalage) du signal :

La faible tension est décalée et amplifiée pour pouvoir se trouver dans la plage de lecture d'une entrée analogique d'un microcontrôleur **{0V ; +Vref}**.

Avantage : rapidité : comme le signal est analogique, seule compte la rapidité de lecture de la carte programmable.

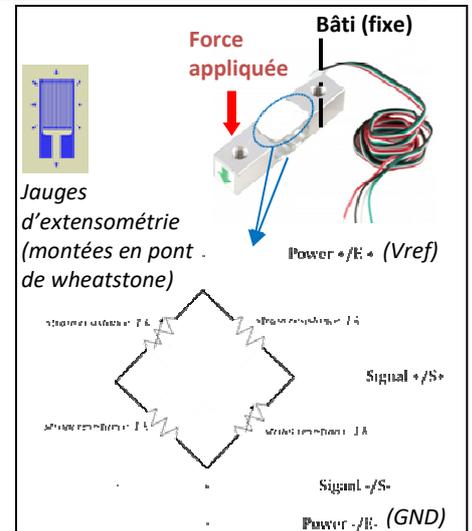
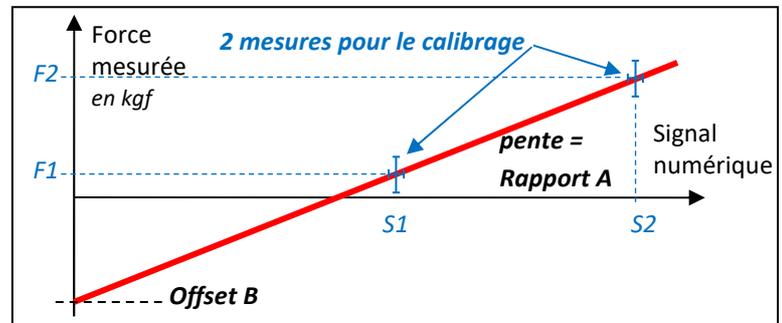
Technique 2 – Conversion analogique-numérique du signal : La faible tension est convertie en signal numérique par un *CAN-n-bits*. Avantage : moins cher, souvent très précis (dépend de n). Inconvénient : plus lent car le signal numérique est envoyé en série et il faut attendre la fin de l'envoi d'un signal pour en démarrer un nouveau.

Calibrage (ou étalonnage) :

La sensibilité d'un capteur est donnée de façon imprécise par le fabricant, et comme elle varie en fonction de la température, il conviendra de calibrer chaque capteur avant utilisation, c'est-à-dire déterminer les valeurs du **Rapport A** et de l'**Offset B** tels que :

$$\text{Force_mesurée} = A * \text{Signal} + B$$

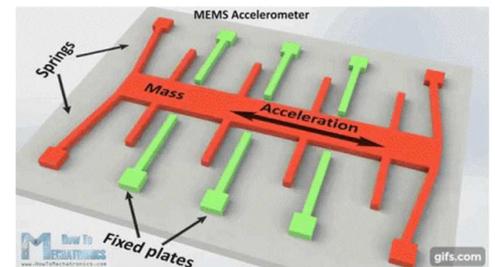
Pour cela il suffira d'appliquer deux forces connues différentes $F1$ et $F2$ sur le capteur (plus elles seront éloignées, plus précis sera le calibrage) et de relever le signal à chaque mesure, respectivement notés $S1$ et $S2$.



4.8) Accéléromètre, gyromètre, magnétomètre (centrale inertielle, ou AHRS)

accéléromètre : capteurs d'accélération linéaire, par déplacement d'une petite masse suspendue. Il peut être utilisé :

- pour détecter des **chocs** (airbag...) ou des **chutes libres**...
- pour déterminer l'**inclinaison** par rapport à la verticale, si l'accélération du système est très faible devant g .
- pour déterminer la **vitesse** et la **position** au cours du temps (en intégrant l'accélération), pour un système en translation pure, et des durées courtes (car l'erreur devient vite importante, c'est la dérive).



gyromètre : **tachymètre** (capteur de vitesse) angulaire, pouvant utiliser différents principes dont la rotation d'une masse. A ne pas confondre avec un **gyroscope** : capteur de position angulaire, qui peut parfois être déterminée avec le même appareil.

magnétomètre : capteur de champs magnétique (3 axes).

Une **centrale inertielle** ou « *Inertial Measurement Unit* » (**IMU**), ou encore **AHRS** (*Attitude and Heading Reference System*) est un ensemble de capteurs permettant de déterminer l'orientation et la position relative d'un système dans l'espace. Par exemple l'association d'un accéléromètre 3 axes avec un gyromètre 3 axes (et éventuellement un magnétomètre 3 axes pour être bien plus précis), permet d'obtenir les 3 angles d'inclinaison d'un système en 3D.

4.9) Autres exemples de capteurs

Génératrice tachymétrique : petite machine CC générant une tension proportionnelle à la vitesse de son rotor.

Pressostat ou manomètre : capteur de pression.

APN (appareil photo numérique) ou caméra.

Capteur de température (analogique) : thermocouple, capteur de température à résistance, ou thermistance.

Capteurs de tension, d'intensité (souvent par effet Hall, cf. page 9), etc.