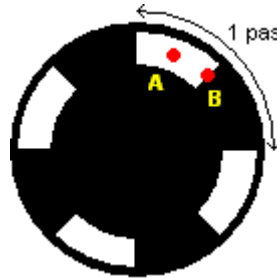


# Décodage des Capteurs en Quadrature

## 1. Principe

Un capteur en quadrature est constitué de deux capteurs (que l'on nommera ultérieurement A et B) et d'une série de fente dans une roue solidaire de la roue folle. Les deux capteurs sont séparés d'un quart de pas (voir schéma).



Sur le schéma ci-dessus le capteur est constitué de 4 pas et 16 fronts alors que les autres auront 128 pas et 512 fronts.

Un capteur envoie le signal 1 lorsqu'il y a une fente (la lumière passe) et 0 sinon.

## 2. Décodage du sens dans lequel tourne la roue folle

Suivant le sens on aura une suite différente de signaux :

- le capteur tourne dans le sens des aiguilles d'une montre :  
A 00110  
B 01100
- dans le sens inverse des aiguilles d'une montre :  
A 01100  
B 00110

Il nous suffit donc de comparer l'actuelle valeur de A avec l'ancienne valeur de B :

A	0	0	1	1	0
B	0	1	1	0	0
sens horaire					
A	0	1	1	0	0
B	0	0	1	1	0
sens inverse					

Soit « a » la valeur que nous renvoie le capteur A, et « a\_old » l'ancienne valeur de A (avant le dernier front). De même on a « b » et « b\_old » les valeurs associées au capteur B.

D'où :

- en sens horaire :  $(a == b\_old) = \text{true}$  (a et b sont égaux).
- en sens inverse :  $(a == b\_old) = \text{false}$  (a et b sont différents).

On définira le sens de la manière suivante :

```
bool sens;  
sens = (a == b_old);
```

### 3. Décodage de la distance parcourue par une roue folle

A chaque front on incrémente (ou décrémente) une variable de la manière suivante :

```
short i; // variable à incrémenter
if(a!=b) // if(sens)
    i++;
else
    i--;
```

Un front est détecté lorsque :  $(a!=a\_old) \parallel (b!=b\_old)$ ,

Une variable indiquant le nombre de pas que la roue aura effectuée sera en sortie 8 bits de l'AVR ATtiny2313.

### 4. Discrétisation du temps et fréquence

On considère les hypothèses suivantes :

- rayon de la roue folle :  $r = 20$  mm
- vitesse max du robot :  $v = 1$  m/s

Calcul de la fréquence minimale des mesures par les capteurs :

Périmètre de la roue folle :  $p = 2\pi \cdot 20 = 126$  mm  
Nombre de tours par seconde :  $w = 1000/126 = 8,0$  tours/sec  
Nombre de fronts par seconde :  $n = 8,0 \cdot 512 = 4100$  fronts/sec  $\equiv 4,1$  kHz

Or pour éviter les emmerdes on prendra  $4,1 \cdot 2 = 8,2$  kHz. Disons plutôt 10 kHz.

Donc les capteurs A et B feront des mesures à une fréquence de **10 kHz**.

La variable indiquant le nombre de pas (en sortie de l'AVR ATtiny) étant codé sur 8 bits (type short) on a :  $-128 \leq i \leq 127$ . Or la fréquence de mesure étant 10 kHz, il risque d'y avoir un dépassement des 8 bits sur cette variable si on ne la récupère pas à une fréquence  $f$  telle que :

Nombre de pas par seconde :  $m = w \cdot 128 = 1000$  pas/sec  
Donc on définit  $f$  par :  $f = m/128 = 8$  Hz

Pour éviter les emmerde on prendra  $f = 2 \cdot 8 = 16 = 20$  Hz.

Donc l'AVR ATmega32 interrogera l'ATtiny à une fréquence de 20 Hz pour connaître le nombre de pas qu'a effectué la roue folle.