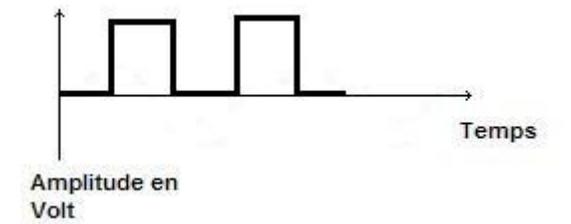
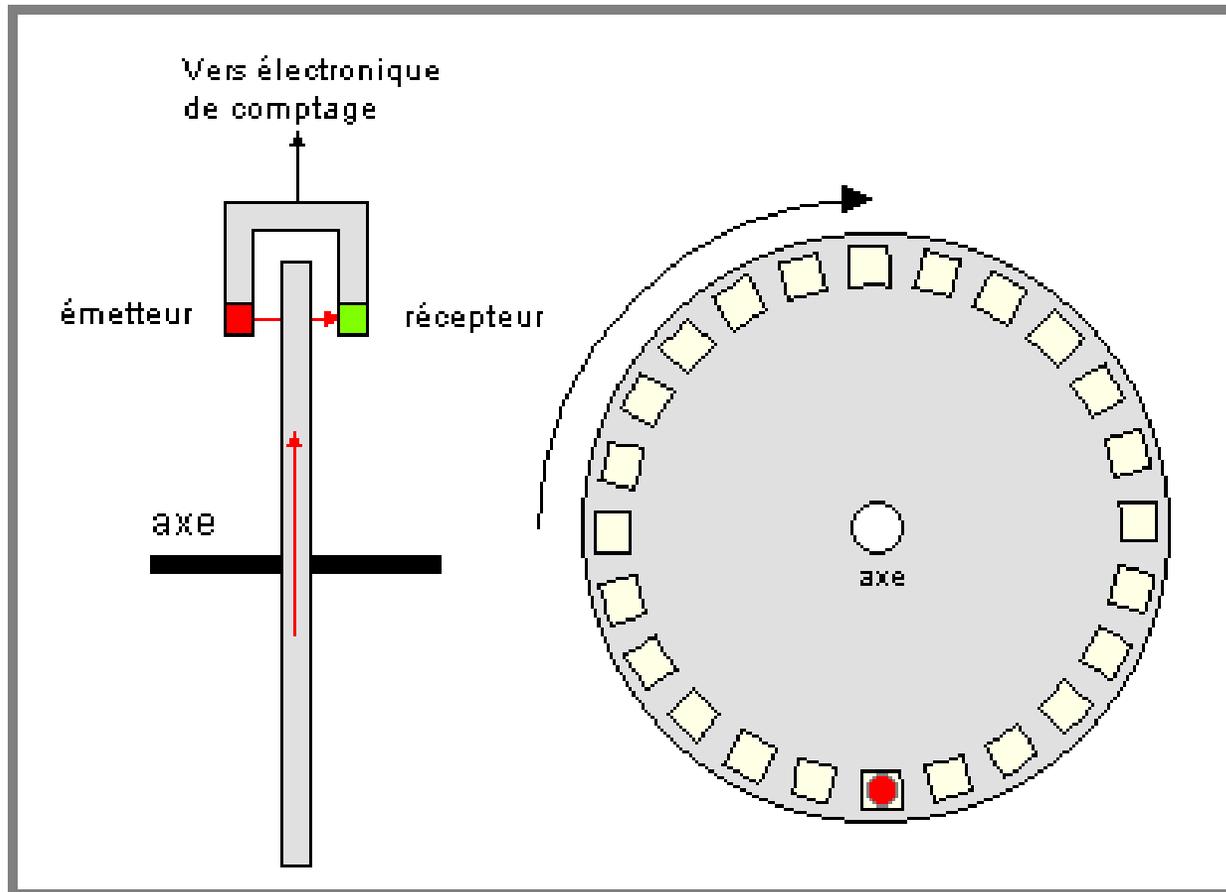


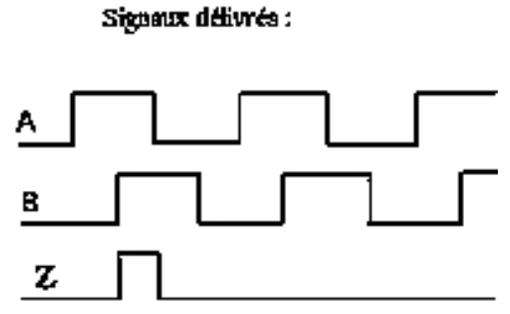
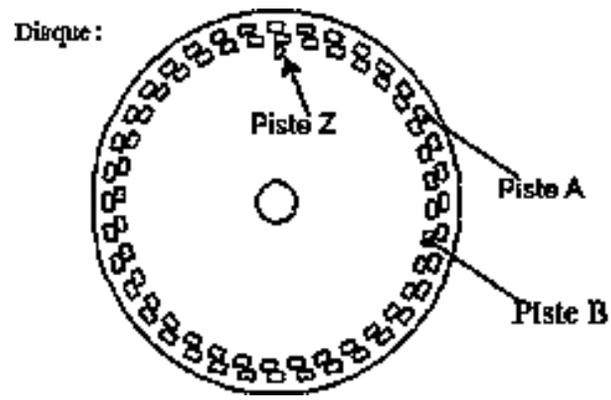
# Principe et démonstration du fonctionnement des codeurs optiques appliqués au contrôle du pointage d'un instrument (projet PushTo PLM)

---

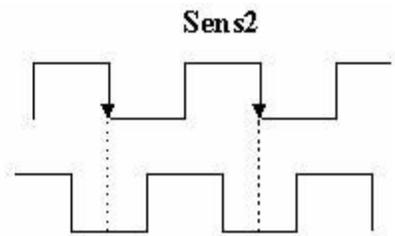
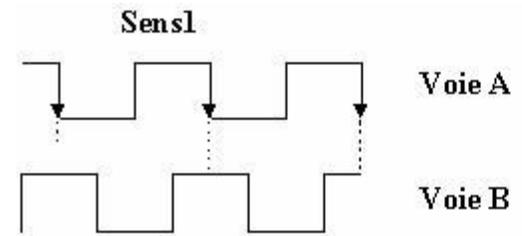




Crédit <https://rco.fr.nf/index.php/2016/07/06/codeurs-et-robotique/>



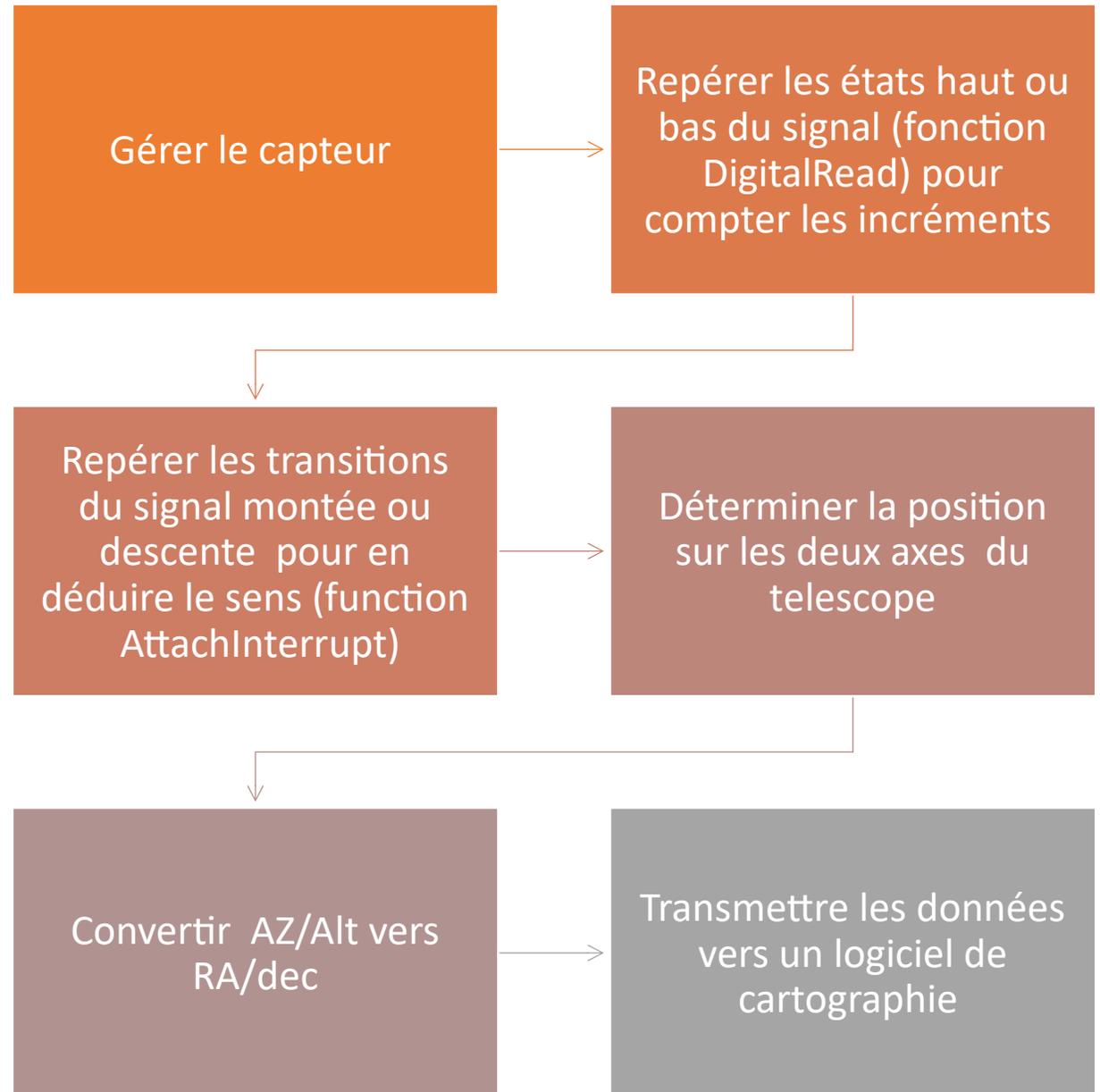
En fait deux pistes A et B décalées de 25% (déphasage 90°)



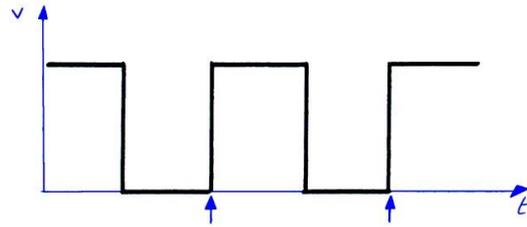
Cela va permettre de repérer le sens de rotation



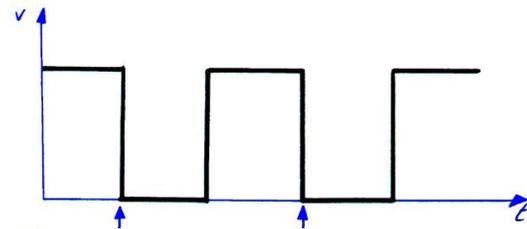
En quoi une  
carte Arduino  
va-t-elle nous  
être utile ?



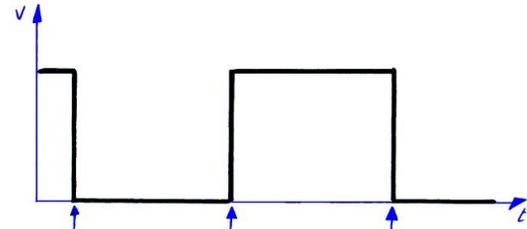
Les interruptions (ARDUINO par ex) permettent de surveiller des changements de signaux pendant qu'un programme continue de fonctionner



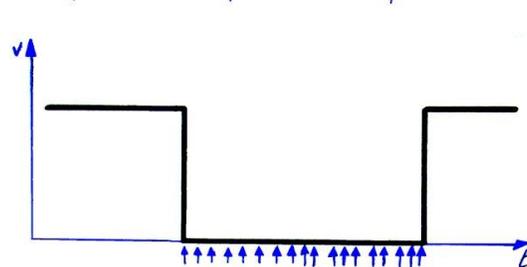
Détecter une montée de signal : Rising



Détecter une descente de signal : Falling



Détecter un changement bas vers haut ou haut vers bas: Change



Scruter en continu un état bas: Low (peu utilisé)

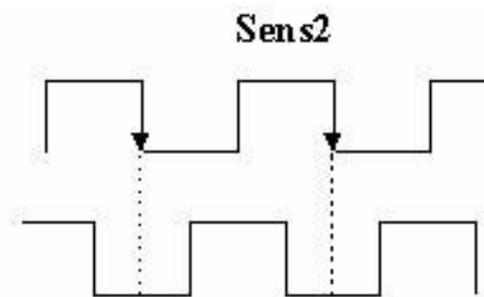
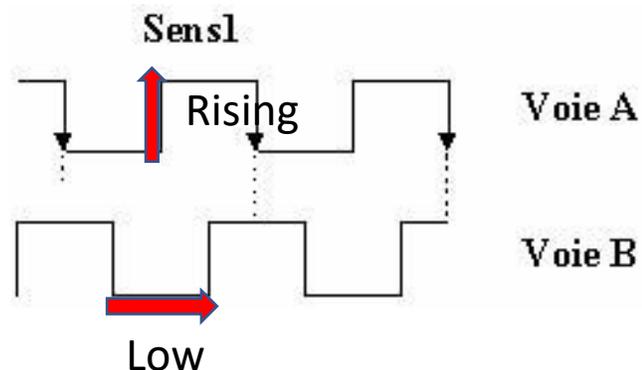
Sur une carte UNO deux broches (2,3) seulement peuvent surveiller des interruptions (0 et 1)

Cela constitue une limitation pour notre projet : des cartes plus performantes comme la carte DUE sont capables de gérer des interruptions sur toutes les broches.



Interrupt 0 > voie A pin 2

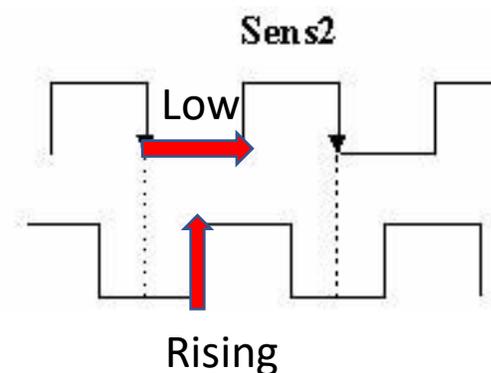
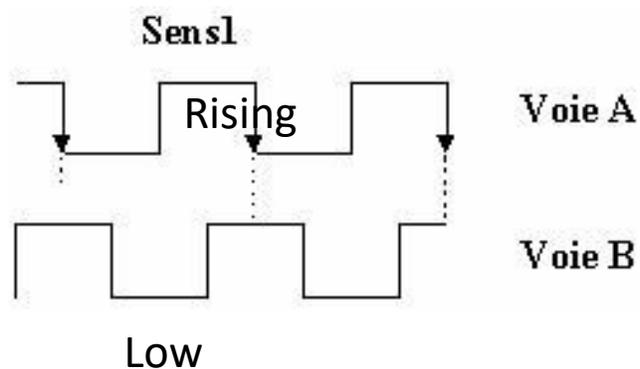
DigitalRead pin 3 Voie B



Je regarde si il y a un front montant (Rising) sur la voie A (fonction attachInterrupt 0)  
Je lit le niveau de B : si il est low : **je suis dans le sens 1** et j'incrémente mon compteur

DigitalRead pin 2 Voie A

Interrupt 1 > voie B pin 3



Je lit le niveau de A : si il est "low "

Je regarde si il y a un front montant (Rising) sur la voie B (fonction attachInterrupt 1) **je suis alors dans le sens 2**  
et je décrémente mon compteur

# Démo

Branchement du codeur sur un Arduino uno

Rouge + 5V

Noir GND

Blanc pin 2 voie A

Vert pin3 voie B

Programme d'affichage sur le moniteur

Prochaines étapes : carte Arduino DUE +  
adaptation signaux 3.3V

volatile unsigned int temp, counter = 0; //This variable will increase or decrease depending on the rotation of encoder

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

  pinMode(2, INPUT_PULLUP); // internal pullup input pin 2
  pinMode(3, INPUT_PULLUP); // internal pullup input pin 3
  //Setting up interrupt
  //A rising pulse from encodenren activated ai0(). AttachInterrupt 0 is DigitalPin nr 2 on moust Arduino.
  attachInterrupt(0, ai0, RISING);

  //B rising pulse from encodenren activated ai1(). AttachInterrupt 1 is DigitalPin nr 3 on moust Arduino.
  attachInterrupt(1, ai1, RISING);
}

void loop() {
  // Send the value of counter
  if( counter != temp ){
    Serial.println (counter);
    temp = counter;
  }
}

void ai0() {
  // ai0 is activated if DigitalPin nr 2 is going from LOW to HIGH
  // Check pin 3 to determine the direction
  if(digitalRead(3)==LOW) {
    counter++;
  }else{
    counter--;
  }
}

void ai1() {
  // ai0 is activated if DigitalPin nr 3 is going from LOW to HIGH
  // Check with pin 2 to determine the direction
  if(digitalRead(2)==LOW) {
    counter--;
  }else{
    counter++;
  }
}
```

- Les entrées de la carte DUE doivent être limitées à 3,3V (et non 5v)
- Utilisation d'un pont diviseur de tension et réglage des voies avec un multimètre

