

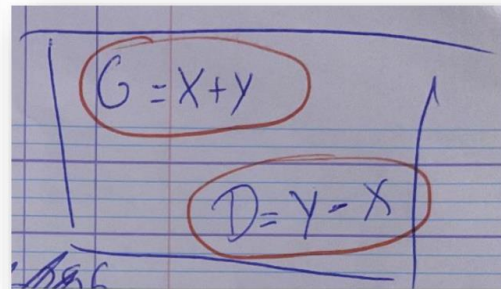
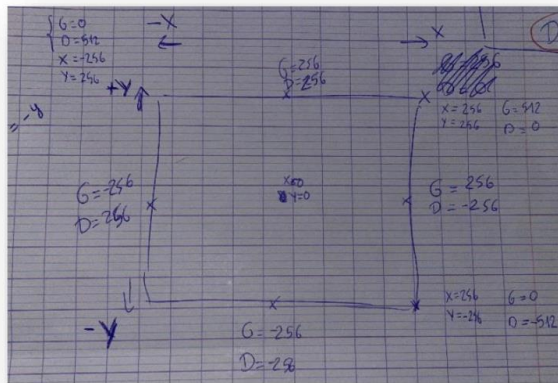


### Objectifs de la séance :

- Contrôler le Robot à l'aide d'un PAD (dans toutes les directions) et non de flèches

### Réalisations :

- La semaine dernière j'ai terminé toute la partie théorique et cherché comment coder ce PAD :



- Or il y avait des problèmes pour le changement de signe de X à droite, la stratégie était compliquée. J'ai donc renoncé à mon idée de changer l'origine afin de traiter que des valeurs positives

### Détails de l'écriture du code ANNEXE 1 :

- Je commence par reprendre la partie du code qui récupérait les commandes Bluetooth de l'ancien code.  
Sur les nouvelles cartes des moteurs, il y a 2 pins qui permettent de contrôler le sens et la vitesse des moteurs. Je nomme le premier pin 'dir' pour direction et 'pwn' pour vitesse. Et cela de chaque côté, pour rappel les moteurs sont en parallèles.

On nomme speedLeft et speedRight les vitesses appliqués aux 2 moteurs de chaque côté. Ensuite selon le signe de pwm on applique la direction souhaité.

### Résultat :

- Le code fonctionne mais la vitesse ne monte pas progressivement ce qui provoque des à-coups et rend le robot impossible à piloter.

```
// ANNEXE 1 :

#include <SoftwareSerial.h>

SoftwareSerial BT(4, 2); // RX, TX

#define dir_g 8
#define pwm_g 6
#define dir_d 10
#define pwm_d 11

int padX = 0, padY = 0;
char a;

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

  pinMode(dir_g, OUTPUT);
  pinMode(pwm_g, OUTPUT);
  pinMode(dir_d, OUTPUT);
  pinMode(pwm_d, OUTPUT);
}

void loop() {

  if (BT.available()) {

    a=BT.read();
    //Serial.write(char(a));

    if ( a=='X') {
      padX=BT.parseInt();
      Serial.print("Axe X :");
      Serial.println (padX);
    }
    if ( a=='Y') {
      padY=BT.parseInt();
      Serial.print("  Axe Y :");
      Serial.println(padY);
    }

    // Calculer les vitesses des moteurs en fonction de la position du pad
    int speedLeft = (padY - 255) + (padX - 255);
    int speedRight = (padY - 255) - (padX - 255);

    // Limiter les vitesses des moteurs entre -255 et 255
    speedLeft = constrain(speedLeft, -255, 255);
    speedRight = constrain(speedRight, -255, 255);

    Serial.print("Gauche : ");
    Serial.print(speedLeft);
    Serial.print("  Droite : ");
    Serial.println(speedRight);

    // Contrôler les moteurs en fonction des vitesses calculées
    if (speedLeft > 0) {
```

```
    digitalWrite(dir_g, HIGH);
    analogWrite(pwm_g, speedLeft);
  } else {
    digitalWrite(dir_g, LOW);
    analogWrite(pwm_g, -speedLeft);
  }

  if (speedRight > 0) {
    digitalWrite(dir_d, HIGH);
    analogWrite(pwm_d, speedRight);
  } else {
    digitalWrite(dir_d, LOW);
    analogWrite(pwm_d, -speedRight);
  }
}
}
```