



### Sommaire

- [1 Objectifs de la vidéo](#)
- [2 Projet connexe](#)
  - [2.1 Commande d'un servomoteur avec Arduino et liaison IR](#)
- [3 Forme d'onde du signal PWM](#)
- [4 Le programme Arduino](#)
  - [4.1 Fonction de mise à jour de la vitesse du moteur](#)
  - [4.2 Fonction de lecture de données brutes du capteur](#)
  - [4.3 Le programme complet](#)

## Objectifs de la vidéo

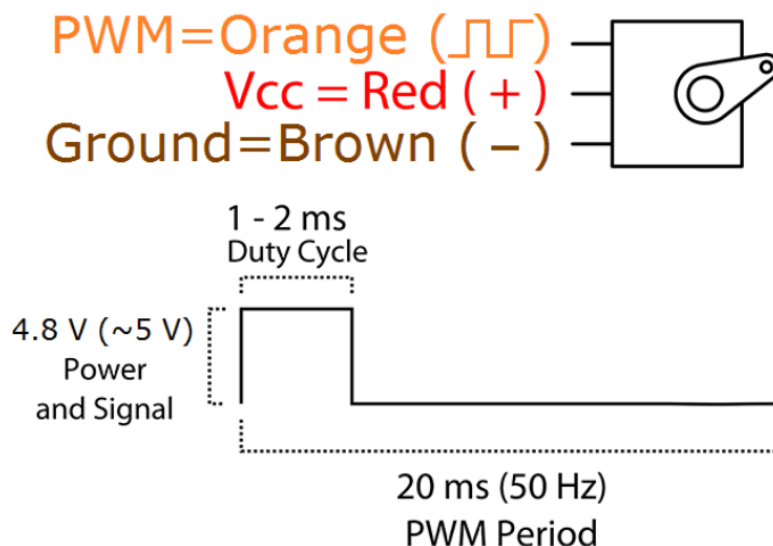
1. Savoir varier la vitesse d'un moteur brushless avec Arduino
2. Savoir le forme d'onde du signal de la commande
3. Savoir la différence entre la commande d'un servomoteur et un moteur brushless
4. Savoir varier la vitesse d'un moteur brushless en fonction de l'accélération
5. Savoir lire les données du capteur MPU-6050
6. Se familiariser avec les équations mathématiques avec Arduino
7. Et plus encore.

## Projet connexe

- [Commande d'un servomoteur avec Arduino et liaison IR](#)



## Forme d'onde du signal PWM



## Le programme Arduino

### Fonction de mise à jour de la vitesse du moteur

```
void SetSpeedMotor(double *PitchRollYaw, byte Vitesse_0, double Gain)
{
  // Calcul de la partie variable de la vitesse
  double P=PitchRollYaw[0];
  double R=PitchRollYaw[1];
  double Y=PitchRollYaw[2];
  double Vitesse=Vitesse_0+Gain*(P+R); // Dépendance R&P: -180<R+P<180
  //double Vitesse=Vitesse_0+Gain*(R); // Dépendance R: -90<R<90
  //double Vitesse=Vitesse_0+Gain*(P); // Dépendance P: -90<P<90
}
```



## Drone | Arduino #7: Comment varié la vitesse d'un moteur brushless en fonction de l'inclinaison du drone ?

```
// Mise à jour de la vitesse du moteur
M1.write(Vitesse); // 0-180
}
```

## Fonction de lecture de données brutes du capteur

```
void ReadGY521_0( long *GyAccTemp)
{
  long  GATCorr_0[NumData]={0,0,0,0,0,0,0};
  ReadGY521(GyAccTemp,GATCorr_0);
}
```

## Le programme complet

```
#include<Wire.h>
#include <math.h>
#include <Servo.h>

#define NumData      7
#define pi           3.1415926535897932384626433832795
#define MPU          0x68
#define PinM1        9

// Pinout
/*
VCC -> 3.3 V / 5 V (préférable)
GND -> GND
SCL -> A5
SDA -> A4
XDA -> NC (non connecté)
XCL -> NC
```



## Drone | Arduino #7: Comment varié la vitesse d'un moteur brushless en fonction de l'inclinaison du drone ?

```
ADO -> NC
INT -> NC
*/

Servo M1;
long GyAccTemp[NumData];
long GATCorr[NumData]={0,0,0,0,0,0,0};
double PitchRoll[3];

void setup()
{
  // Init module GY-512
  Wire.begin();
  Wire.beginTransmission(MPU);
  Wire.write(0x6B);
  Wire.write(0);
  Wire.endTransmission(true);
  // Init port série
  Serial.begin(19200);

  // Affectation du pin au servomoteur
  M1.attach(PinM1);
}
void loop()
{
  // Lecture des données: Acce + Temp + Gyro
  //ReadGY521(GyAccTemp,GATCorr); // Lecture avec correction
  ReadGY521_0(GyAccTemp); // Lecture des données bruts

  // Calcul du Pitch / Roll / Yaw atan(x/sqrt(y^2+z^2)
  ComputeAngle(GyAccTemp, PitchRoll); //+-90
  // Mise à jour de la vitesse du moteur
  SetSpeedMotor(PitchRoll,90, 0.5); // 0(0%) - 180(100%)
}

void SetSpeedMotor(double *PitchRollYaw, byte Vitesse_0, double Gain)
{
  // Calcul de la partie variable de la vitesse
  double P=PitchRollYaw[0];
  double R=PitchRollYaw[1];
  double Y=PitchRollYaw[2];
}
```



## Drone | Arduino #7: Comment varié la vitesse d'un moteur brushless en fonction de l'inclinaison du drone ?

```
double Vitesse=Vitesse_0+Gain*(P+R); // Dépendance R&P: -180<R+P<180
//double Vitesse=Vitesse_0+Gain*(R); // Dépendance R: -90<R<90
//double Vitesse=Vitesse_0+Gain*(P); // Dépendance P: -90<P<90

// Mise à jour de la vitesse du moteur
M1.write(Vitesse); // 0-180
}

// Lecture des données des capteurs avec Corrections
void ReadGY521( long *GyAccTemp, long *GATCorrr)
{
  // Init du module GY-521
  Wire.beginTransmission(MPU);
  Wire.write(0x3B);
  Wire.endTransmission(false);
  Wire.requestFrom(MPU,14,true);

  // Lecture des données (3 axes accéléromètre + température + 3 axes gyroscope
  for (long i=0;i<NumData;i++)
  {
    if(i!=3)
    {
      GyAccTemp[i]=(Wire.read()<<8|Wire.read()) - GATCorrr[i];
    }
    else
    {
      GyAccTemp[i]=(Wire.read()<<8|Wire.read()) - GATCorrr[i];
      GyAccTemp[i] = GyAccTemp[i]/340 + 36.53;
    }
  }
}

// Lecture des données des capteurs sans Corrections
void ReadGY521_0( long *GyAccTemp)
{
  long GATCorr_0[NumData]={0,0,0,0,0,0,0};
  ReadGY521(GyAccTemp,GATCorr_0);
}

//Conversion des données accéléromètre en pitch/roll/yaw
void ComputeAngle(long *GyAccTemp, double *PitchRol)
```



## Drone | Arduino #7: Comment varié la vitesse d'un moteur brushless en fonction de l'inclinaison du drone ?

```
{
  double x,y,z, X, Y, Z;
  x= (double)GyAccTemp[0];
  y= (double)GyAccTemp[1];
  z= (double)GyAccTemp[2];

  X=sqrt((y*y) + (z*z));
  Y=sqrt((x*x) + (z*z));
  Z=sqrt((x*x) + (y*y));
  PitchRol[0] = atan(x/X); // pitch
  PitchRol[1] = atan(y/Y); // roll
  PitchRol[2] = atan(z/Z); // yaw
  //Conversion Radian en degré
  PitchRol[0] = PitchRol[0] * (180.0/pi);
  PitchRol[1] = PitchRol[1] * (180.0/pi) ;
  PitchRol[2] = PitchRol[2] * (180.0/pi) ;
}
```

[Accueil Drone avec Arduino](#)