

Presentation du Projet Metier

Mise en oeuvre et stabilisation d'un bras de drone quadrirotor

Ahmed Amine NOUABI

Mamane Bello Abdoul Hakim

Encadre par: Mr. TALEB

Jury: Mr. TALEB Mr.SAADI Mr. LAGRIOUI

6 Fevrier 2025





Table of Contents

1 Introduction

► Introduction

► Conception du systeme

► Conception Logiciel



Objectifs du projet.

1 Introduction

- Stabiliser un bras rotative.
- Utilisation d'un capteur MPU6050 et un algorithme de calcul d'angle.
- Mise en oeuvre d'un contrôleur PID pour commander les moteurs.



Dynamique du systeme.

1 Introduction

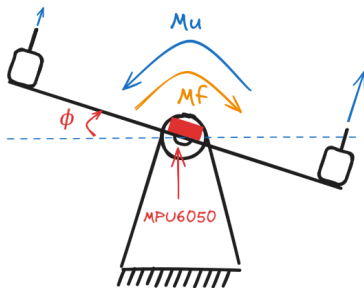


Figure: Dynamique du systeme.

Presentation du projet.

1 Introduction

La Dynamique du systeme est modelisée comme suit:

Dynamique du systeme

$$J_{\Delta} \ddot{\phi} + f \dot{\phi} = M_u = u \frac{L}{2}$$

$$G(s) = \frac{\Phi(s)}{U(s) \frac{L}{2}} = \frac{1}{J_{\Delta} s^2 + f s}$$

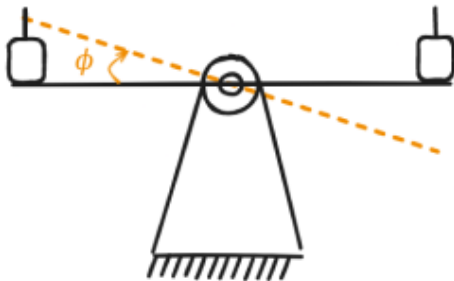




Table of Contents

2 Conception du systeme

► Introduction

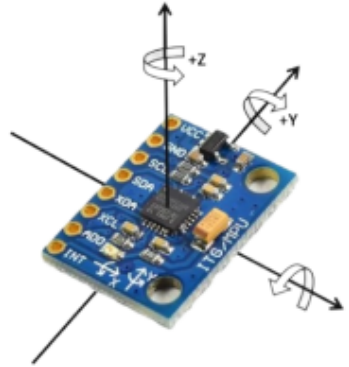
► Conception du systeme

► Conception Logiciel

Calcul d'angle

2 Conception du systeme

- Le MPU6050 est un capteur qui combine un accéléromètre et un gyroscope.
- L'accéléromètre mesure l'accélération linéaire tandis que le gyroscope mesure la vitesse angulaire.





Calcul d'angle par gyroscope

2 Conception du système

Modele du gyroscope

$$\omega = \frac{d\phi}{dt} + b_g(t) + n_g(t)$$

Calcul d'angle par gyroscope

$$\hat{\phi}_{gyro}(n) = \sum_{i=0}^n \omega(i) * T$$

- T est le temps d'échantillonnage $0.004s$. ($f_e = 250Hz$)



Calcul d'angle par accéléromètre

2 Conception du système

Modele de l'accéléromètre

$$\vec{a} = \frac{dv}{dt} - R * \vec{g} + b_a(t) + n_a(t)$$

Calcul d'angle par accéléromètre

$$\vec{a} = -R * \vec{g}$$

$$a_x = g \sin(\phi)$$

$$\hat{\phi}_{accelo}(n) = \arcsin\left(\frac{a_x(n)}{g}\right)$$

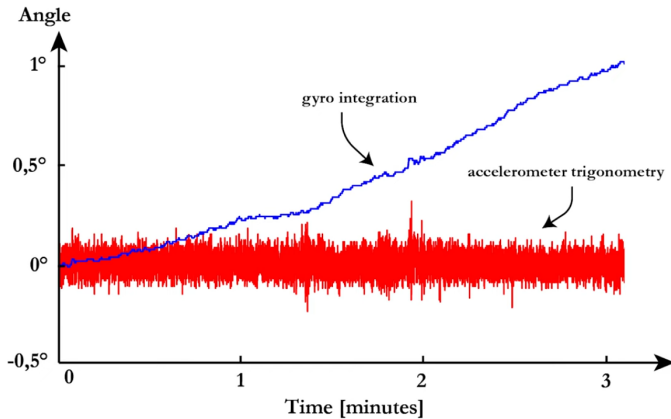


Figure: Limitations Des capteurs

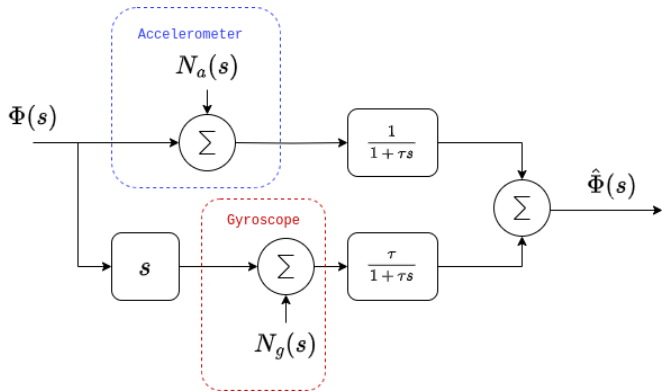


Figure: Chaîne de retour.



Fusion des resultats

2 Conception du systeme

- Pour pallier l'erreur cumulee du gyroscope et au bruits de l'accelerometre , on combine les résultats filtrés des deux.

Fusion des resultats

$$\hat{\phi}(n) = \alpha * \hat{\phi}_{gyro}(n) + (1 - \alpha) * \hat{\phi}_{accelo}(n)$$

Alpha

$$\alpha = e^{-\omega_c T_e}$$

$$f_c = \frac{-\ln(\alpha)}{2\pi T_e}$$



Commode des moteurs

2 Conception du système

- on utilise un ESC (Electronic Speed Controller) pour commander le moteur.
- ESC est commandé par un signal de période $T = T_{on} + T_{off}$.

Plage de commande

$$1000\mu s \leq T_{on} \leq 2000\mu s$$



Contrôleur PID

2 Conception du système

Equation du contrôleur PID

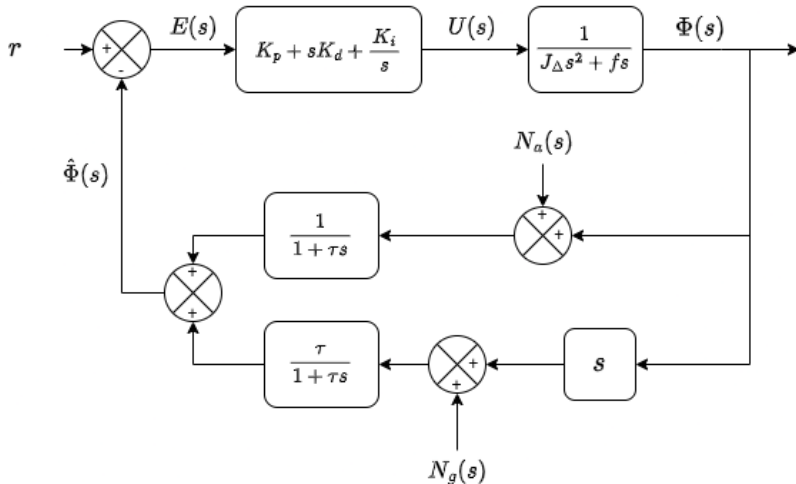
$$u(t) = K_p * e(t) + K_i * \int e(t)dt + K_d * \frac{de(t)}{dt}$$

$$C(s) = K_p + \frac{K_i}{s} + K_d * s \quad (1)$$

- Pour utiliser une commande symétrique. $\gamma = 1500 \pm u(t)$
- γ valeur de la commande du moteur.
- On limite $u(t)$ entre -400 et 400 .

Schema du systeme

2 Conception du systeme





Modele CAD 3D

2 Conception du systeme

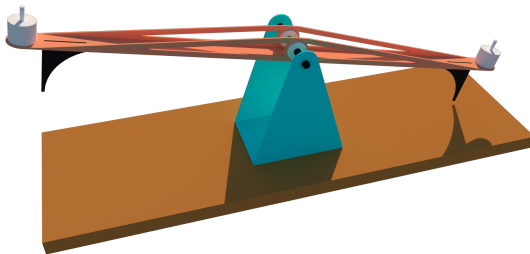


Figure: Modele mecanique 3D.

Realisation finale

2 Conception du systeme



Figure: Modele Final.



Table of Contents

3 Conception Logiciel

► Introduction

► Conception du systeme

► Conception Logiciel



Conception Logiciel

3 Conception Logiciel

- Pour faciliter la maintenance et l'évolution du code, on a utilisé une approche orientée objet (OOP).
- On utilise la librairie `Wire` pour la communication I2C avec le capteur MPU6050.
- On utilise la librairie `ESC` pour la commande des moteurs.

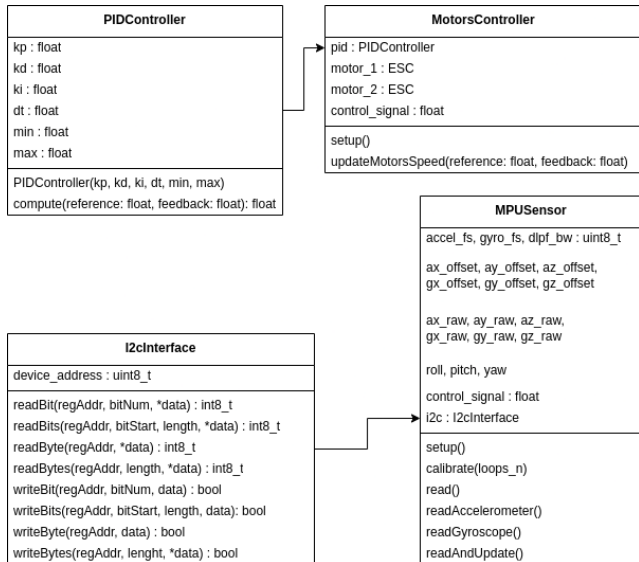


Figure: UML des Composants Logiciel.



Code Arduino - Setup

3 Conception Logiciel

```
MPUSensor mpu;  
MotorsController motors;  
  
double deltaTime_micros = 0.004 / 1000000.0;  
float pitch_ref = 0.0f;  
  
void setup() {  
    mpu.setup();  
    motors.setup();  
}
```



Code Arduino - Loop

3 Conception Logiciel

```
void loop() {  
    current_micros = micros();  
  
    mpu.readAndUpdate();  
    motors.update(pitch_ref, mpu.pitch);  
  
    // Pour assurer une boucle de controle de 4ms.  
    while (micros() - current_micros < deltaTime_micros);  
}
```



Merci pour votre attention!