

Introduction:

Les systèmes autonomes tels que les drones ou les robots mobiles équilibrés occupent une place croissante dans les applications industrielles et sociétales. L'un des principes sur lequel ces systèmes reposent est la capacité à stabiliser en temps réel des systèmes naturellement instables, en exploitant des capteurs inertiels et des algorithmes de régulation.

Durant ma deuxième année à l'ECE, j'ai participé au projet DrawBot, qui consistait à construire un robot à partir de la plateforme électronique fournie par l'école. J'ai ainsi réalisé la mise en place matérielle (soudure, câblage, montage des moteurs, des roues et du microcontrôleur ESP32) puis programmé le système afin qu'il puisse tracer différentes formes, telles qu'une rosace, une rose des vents orientée vers le Nord magnétique ou encore un escalier à angles droits.

Souhaitant aller au-delà de ce cadre académique et perfectionner mes compétences en commande PID, en exploitation de capteurs inertiels et en traitement de données par filtrage, je me suis lancé le défi de réutiliser la structure matérielle du DrawBot pour concevoir un GyroBot : un robot équilibriste capable de se stabiliser en continu autour de sa position d'équilibre.

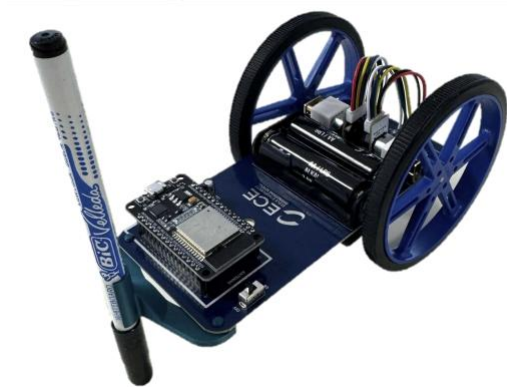


Photo du DrawBot



Photo du GyroBot

Matériel utilisé :

Microcontrôleur - NodeMCU ESP32 double cœur cadencé à 240 MHz, intégrant 512 kB de SRAM et 4 MB de mémoire Flash. Il assure à la fois l'acquisition des capteurs, le calcul de la régulation (PID) et le pilotage des moteurs.

Centrale inertielle - LSM6DS3 (IMU 6 axes) Capteur intégrant un accéléromètre triaxial et un gyroscope triaxial :

1. L'accéléromètre fournit une estimation fiable de l'angle sur le long terme, mais bruitée sur le court terme.
2. Le gyroscope offre une grande précision instantanée mais présente une dérive cumulée au fil du temps.

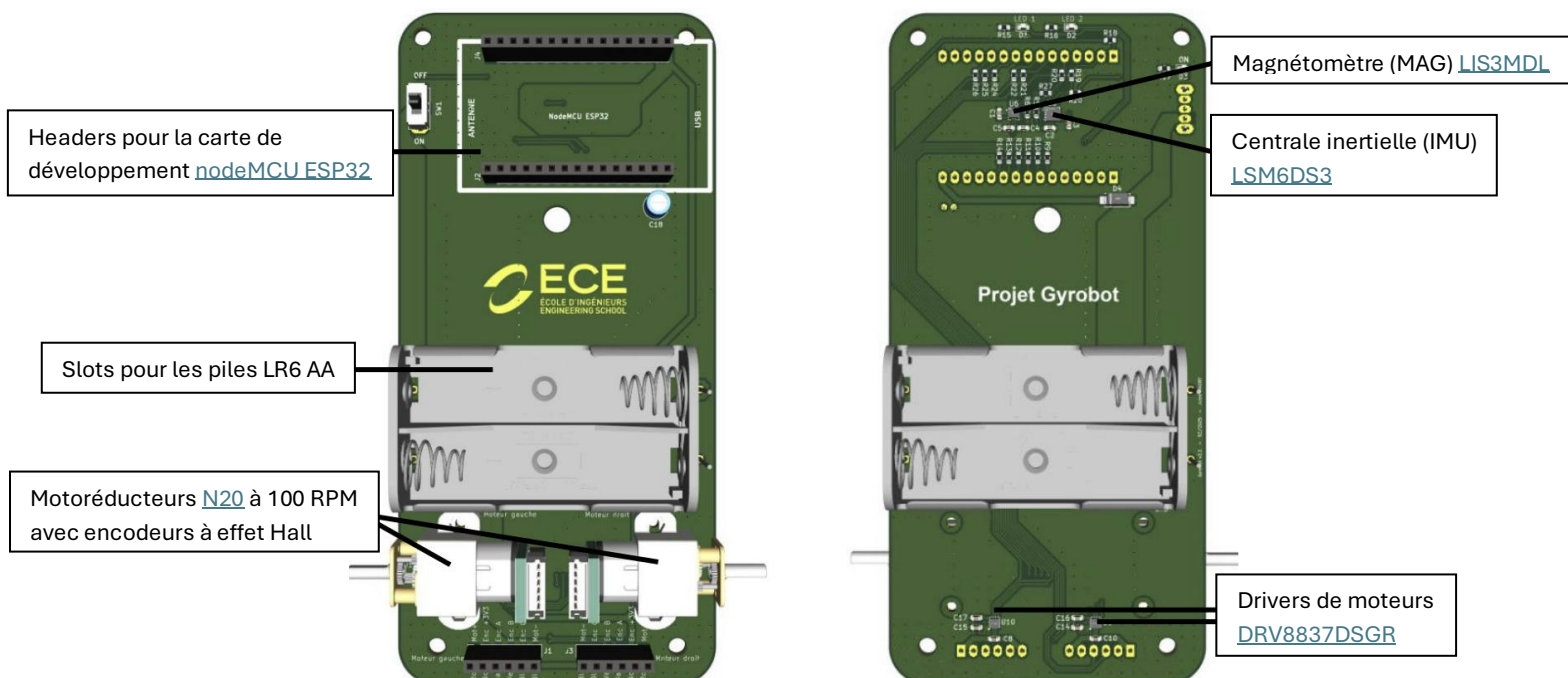
La combinaison des deux via un filtre complémentaire permet d'obtenir un angle robuste et réactif, indispensable pour l'équilibre du robot.

Motorisation - Motoréducteurs N20 avec encodeurs intégrés, deux moteurs à courant continu de type N20, chacun équipé d'un encodeur à effet Hall permettant de mesurer la rotation des roues (utile pour un asservissement précis pour le faire se déplacer, même si non indispensable dans ce projet). Les réducteurs limitent la vitesse à environ 100 RPM, ce qui augmente le couple disponible et rend le déplacement du robot possible malgré la faible puissance nominale. Cette contrainte impose cependant une marge d'erreur très fine pour la commande.

Ces motoréducteurs étant issus d'un ancien projet, un jeu mécanique dû à l'usure est présent dans les roues, introduisant des perturbations supplémentaires. Celles-ci doivent être compensées par l'asservissement pour garantir un mouvement stable.

Alimentation – 2 packs de piles LR6 (AA). L'alimentation est répartie de manière symétrique de part et d'autre de la carte, ce qui favorise la répartition des masses et limite le déséquilibre initial du châssis. La tension délivrée ($\approx 6\text{ V}$) est régulée pour alimenter les moteurs et l'électronique embarquée.

La géométrie du robot étant initialement prévue pour un projet DrawBot, n'est pas adaptée à un robot équilibriste ce qui augmente la difficulté de la stabilisation.



Méthodologie :

1 – Récupération et traitement des données de capteurs

Pour déterminer l'angle du GyroBot, deux capteurs de l'IMU ont été exploités :

- Le gyroscope : l'angle est obtenu par l'intégration temporelle de la vitesse angulaire. Cependant, ce capteur présente un offset qui provoque une dérive progressive de l'angle au cours du temps : plus l'intégration se prolonge, plus l'erreur s'accumule.
- L'accéléromètre : l'angle estimé à partir de la répartition de la gravité sur ses axes. Cette mesure est fiable sur le long terme, mais fortement bruitée, avec des fluctuations pouvant atteindre $\pm 5^\circ$ même lorsque que le système est statique.

Ainsi :

- Le gyroscope est précis à court terme mais instable à long terme (dérive).
- L'accéléromètre est un peu exploitable à court terme (bruit), mais précis à long terme.

Solution adoptée : un filtre complémentaire a été implémenté. Celui-ci combine :

- La précision instantanée du gyroscope,
- La stabilité à long terme de l'accéléromètre.

Résultat : un angle très précis, avec une variation réduite à $\pm 0.02^\circ$ à l'équilibre statique.

Une analyse rapide a confirmé que l'angle de roulis (roll, axe X) était le degré de liberté pertinent pour l'équilibrage, celui-ci devenant la variable à asservir."

Pour formaliser ce principe de fusion, l'angle estimé est calculé grâce au filtre complémentaire suivant :

$$\theta(t) = \alpha \cdot (\theta(t - \omega_{gyro}(t)\Delta t) + (1 - \alpha) \cdot \theta_{acc}(t))$$

α : poids du gyroscope

ω_{gyro} : vitesse angulaire calculée par le gyroscope

θ_{acc} : angle calculé par l'accéléromètre

2 – Détermination de l'angle d'équilibre

Problème :

La structure mécanique du robot n'étant pas parfaitement adaptée à un robot équilibriste, la position d'équilibre ne correspond pas à un angle de 90° (verticale idéale). Il a donc été nécessaire de réaliser une étape spécifique pour identifier expérimentalement l'angle réel d'équilibre.

Méthodologie :

L'outil Teleplot intégré à Visual Studio Code a été utilisé pour visualiser en temps réel l'angle mesuré par l'IMU. Le robot a ensuite été stabilisé manuellement à plusieurs reprises, jusqu'à identifier la position où il parvenait à se maintenir le plus longtemps possible sans intervention extérieure.

Résultat :

L'angle d'équilibre du robot a été déterminé expérimentalement à 98,3°, valeur retenue comme consigne de référence pour l'asservissement.



Représentation graphique montrant la recherche de la position d'équilibre du GyroBot manuellement

3 – Génération du signal PWM pour la commande des moteurs

Problème :

Pour contrôler la vitesse de rotation des roues, il est nécessaire d'appliquer une tension variable aux moteurs. Or, l'ESP32 ne dispose pas de sorties analogiques, uniquement de sorties numériques (HIGH/LOW). Sans solution adaptée, le microcontrôleur ne pourrait donc fournir que deux états possibles aux moteurs (arrêt ou pleine vitesse), rendant impossible une régulation fine de la vitesse.

Solution :

L'ESP32 intègre 16 canaux PWM (Pulse Width Modulation). Cette technique consiste à faire osciller

rapidement une sortie numérique à une fréquence donnée (ici 1 kHz), tout en modulant le rapport cyclique (rapport HIGH/LOW). La valeur moyenne de ce signal est perçue comme une tension intermédiaire par les moteurs, ce qui permet de simuler une sortie analogique et ainsi contrôler précisément leur vitesse.

Résultat :

L'utilisation du PWM rend possible la modulation continue de la vitesse des roues à partir des sorties numériques de l'ESP32.

Ce principe peut se résumer mathématiquement par la relation :

$$V_m = \frac{t_{high}}{T_{cycle}} \cdot V_{alim}$$

4 – Asservissement par correcteur PID

Problème :

L'objectif principal du projet est de stabiliser le robot autour de son angle d'équilibre. Pour cela, un asservissement en boucle fermée est nécessaire. Cependant, la modélisation complète du système (fonction de transfert) n'était pas disponible, ce qui a rendu impossible un réglage théorique du correcteur PID.

Méthodologie du PID :

Le correcteur PID appliqué pour stabiliser le GyroBot se traduit par l'équation suivante :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

avec $e(t) = \theta_{ref} - \theta(t)$

Le réglage du PID a été effectué empiriquement, par essais successifs sur les coefficients K_p , K_i et K_d :

1. Réglage du correcteur proportionnel (P) :

Le terme K_p transforme directement l'erreur d'angle en commande moteur. Il a été ajusté en premier (avec $K_i=K_d=0$). Une valeur adéquate permet au robot de se redresser, mais provoque encore des oscillations.

2. Réglage du correcteur dérivé (D) :

Le terme K_d agit comme un amortisseur en limitant les variations rapides de l'angle. En l'ajustant (avec $K_i=0$), les oscillations ont été réduites, ce qui a permis d'obtenir un robot capable de s'équilibrer sur une courte durée.

3. Réglage du correcteur intégral (I) :

Enfin, K_i a été introduit afin de corriger l'erreur résiduelle qui subsistait à long terme et empêchait l'équilibre permanent. Ce terme permet de compenser les petits biais (mécaniques ou capteurs) et d'assurer la stabilité continue du robot.

Résultat :

Grâce à cette procédure de réglage incrémentale, le robot parvient à maintenir sa position d'équilibre de manière autonome et durable, avec une réponse stable et sans dérive temporelle significative.

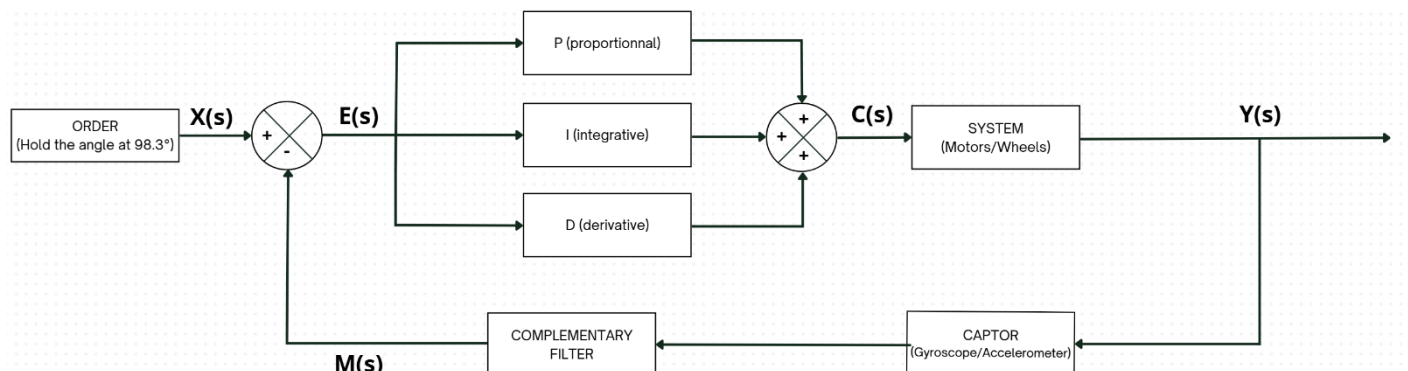


Schéma-bloc représentant le fonctionnement de l'asservissement de mon GyroBot en boucle fermée



Représentation graphique de la procédure d'équilibre du GyroBot montrant la réaction de l'ensemble des correcteurs (PID) et de l'angle par rapport à la consigne

Perspective d'amélioration :

Améliorations mécaniques

- Concevoir un châssis dédié au robot équilibriste, avec une géométrie symétrique et une répartition des masses équilibrée (batteries et électronique positionnées de façon miroir) pour abaisser le centre de gravité et réduire les bruits parasites.
- Réduire le jeu mécanique sur les axes (ajustement des paliers, rondelles de calage, choix de roulements adaptés).
- Augmenter la puissance des motoréducteurs avec une meilleure alimentation et améliorer l'adhérence/qualité des roues pour une réponse plus vive.

Optimisation du contrôle

- Établir un modèle différentiel simplifié (petits angles) et utiliser Scilab pour estimer plus finement les gains K_p , K_i , K_d avant tests réels.
- Implémenter un filtre de Kalman pour une estimation d'angle plus robuste qu'un filtre complémentaire pur, notamment face au bruit et à la dérive.

Fonctionnalités avancées

- Intégrer une interface Bluetooth/Wi-Fi pour monitorer l'angle et ajuster les gains PID en temps réel (télémétrie et tuning distant).

Conclusion :

Ce projet a démontré la transformation réussie d'une plateforme initialement conçue comme un DrawBot en un GyroBot auto-équilibré, capable de se stabiliser autour de son point d'équilibre. En exploitant les capteurs inertiels, en appliquant un filtre complémentaire pour obtenir une estimation précise de l'angle, et en réglant un correcteur PID, le robot a pu maintenir son équilibre malgré des contraintes mécaniques telles que la puissance limitée des moteurs ou le jeu structurel.

- Traitement de données capteurs et fusion, via la mise en œuvre d'un filtre complémentaire pour estimer l'angle d'inclinaison,
- Programmation temps réel et pilotage matériel, notamment par la manipulation des signaux PWM pour contrôler précisément la vitesse des moteurs,
- Démarche expérimentale itérative, grâce à l'utilisation d'outils comme Teleplot pour identifier et valider le comportement du système."
- Mise en œuvre de la théorie du contrôle, avec le réglage empirique d'un correcteur PID.

Le GyroBot a ainsi constitué un véritable laboratoire pratique pour expérimenter la stabilisation d'un système naturellement instable, et a permis de poser des bases solides pour des projets robotiques plus

avancés. Dans la continuité, les développements futurs porteront sur une conception mécanique plus adaptée, l'utilisation de filtres plus avancés, ainsi que l'intégration de fonctionnalités de monitoring et de pilotage à distance.

En définitive, ce projet représente une étape essentielle dans mon parcours en robotique, et prépare directement la réalisation de systèmes plus complexes tels que les drones autonomes, où la stabilisation multi-axes, la fusion capteurs et la planification de trajectoire jouent un rôle central.

DOC EXTERNE :

Vidéo fonctionnelle : <https://youtu.be/EcyiQm1V0ns> 