

Fonctionnement détaillé

1. Connexion Wi-Fi

Le Pico W se connecte automatiquement au réseau Wi-Fi grâce à la bibliothèque **network**. Si la connexion échoue au bout d'un certain temps, un message d'erreur est affiché.

2. Synchronisation de l'heure

- Le programme tente d'abord de récupérer l'heure UTC via le protocole **NTP** (`ntptime`).
- Si NTP n'est pas disponible, il utilise une **API HTTP** (worldtimeapi.org) pour obtenir l'heure exacte.
- L'heure est ensuite enregistrée dans la **horloge interne RTC** du Pico.

3. Calcul de l'angle du servo

- Le programme associe chaque **heure de 12 heures** à un **angle entre 0° et 180°**.
 - 12 h correspond à 0°
 - 6 h correspond à 90°
 - 3 h correspond à 45°, etc.
- L'angle est calculé en continu selon la formule :

$$\text{angle} = 15^\circ \times (\text{heure} + \frac{\text{minutes}}{60})$$

Ce qui permet un mouvement fluide de l'aiguille.

4. Pilotage du servo

- Le servo-moteur (branché sur la broche **GP20**) est commandé par la bibliothèque `servo.py`.
- Le programme envoie régulièrement la nouvelle position au servo en fonction de l'heure locale.

5. Mise à jour périodique

- La position est actualisée toutes les `SAFE_UPDATE_S` secondes (par défaut : 1 seconde).
- Cette fréquence peut être modifiée facilement pour ralentir ou accélérer la mise à jour.

6. Bonus : ajout du bouton poussoir

Un **bouton poussoir** (broche **16**) permet de **changer de fuseau horaire** à la volée.

Chaque appui :

Change le fuseau parmi une liste prédéfinie (ex. UTC, UTC+1, UTC-5, UTC+3).

Affiche immédiatement l'heure correspondant au nouveau fuseau.

Repositionne le servo pour indiquer cette heure actualisée.

Un système d'**interruption matérielle (IRQ)** et un **anti-rebond logiciel** garantissent une détection fiable de chaque appui, même si la boucle principale est en cours d'exécution. Ainsi, l'utilisateur peut passer d'un fuseau horaire à un autre **instantanément**, sans redémarrer le programme.

