**Procédure de vérification « PBAL » de la voiture autonome RGA**

**Matériels utilisés :**

* Câble micro HDMI mâle à HDMI mâle
* Clavier USB
* Souris USB
* Écran moniteur

**Programmes nécessaires :**

Situés dans lidar\_env/src/ :

* Code\_de\_base/test\_lidar.py
* Code\_upgrade/TESTDirectionVitesseEtc.py

**Mise en situation**

Dans le cadre du concours de robotique de Saclay, une voiture autonome a été développée pour la compétition de Cachan. Avant de valider son bon fonctionnement, une procédure d’essai est mise en place afin de tester son comportement en conditions réelles.

Afin de satisfaire le cahier des charges, cette vérification se fera en 4 étapes :

1. **Mise en place du matériel**
2. **Test du fonctionnement du Lidar**
3. **Test du fonctionnement de la direction et de la propulsion**
4. **Test d’autonomie**

**Étapes de vérification**

**Étape 1 : Analyse des fonctions à vérifier**

La voiture autonome repose sur trois fonctions principales :

* **Direction** : permet d’orienter le véhicule
* **Propulsion** : contrôle l’accélération et le freinage
* **Lidar** : détecte les obstacles et mesure les distances

Le **Raspberry Pi** est utilisé pour gérer ces fonctions via les programmes de test.

**Étape 2 : Mise en place du matériel**

1. Connecter le **Raspberry Pi** à l’écran via le câble micro HDMI.
2. Brancher le **clavier et la souris** en USB.
3. Alimenter le Raspberry Pi et attendre son démarrage.
4. Accéder au terminal et se rendre dans le dossier contenant les scripts de test :

cd lidar\_env/src/

1. Activer l’environnement virtuel, permettant un bon fonctionnement des dépendances :

source venv/bin/activate

**Étape 3 : Test du fonctionnement du LiDAR**

1. Exécuter le script de test du LiDAR :
2. python3 Code\_de\_base/test\_lidar.py
3. Vérifier l’affichage des données de distance dans le terminal.
4. Tester la détection d’un obstacle en plaçant un objet devant le LiDAR et s’assurer que la mesure varie.
5. **Critères de validation :**
   * Affichage correct des distances mesurées
   * Réactivité du capteur en présence d’un obstacle

**Étape 4 : Test du fonctionnement de la direction et de la propulsion**

**1. Lancer le script de test**

Dans le terminal, exécuter :

python3 Code\_upgrade/TESTDirectionVitesseEtc.py

**2. Vérification des différentes fonctions**

Une fois le script lancé, un menu interactif s'affiche avec plusieurs options :

**a) Tester la vitesse**

* Sélectionner l’option 1.
* Entrer une vitesse entre 0 et 2 m/s.
* Observer si la propulsion s'ajuste en conséquence.
* Vérifier la sortie [DEBUG] Vitesse demandée dans le terminal.

**b) Tester le contrôle de direction via le duty cycle**

* Sélectionner l’option 2.
* Entrer une valeur entre 0 et 100%.
* Vérifier que l’angle des roues change et que la sortie [TEST] Duty cycle direction réglé à X% s’affiche.

**c) Tester l’arrêt du moteur**

* Sélectionner l’option 3.
* Observer si le moteur s’arrête correctement et si la sortie [TEST] Moteur arrêté apparaît.

**d) Tester le mode oscillation (simulation de virages)**

* Sélectionner l’option 5.
* Observer si les roues oscillent de gauche à droite pendant 4 secondes.
* Vérifier que l’affichage [TEST] Début du mode oscillation... apparaît.
* Observer si l’angle varie selon une fonction sinusoïdale.

**e) Quitter proprement**

* Sélectionner l’option 4 pour arrêter le moteur et quitter le programme.

**3. Critères de validation**

* La vitesse change en fonction des valeurs saisies.
* La direction répond instantanément aux ajustements.
* Le mode oscillation fonctionne correctement.
* L'arrêt du moteur réinitialise bien les valeurs PWM.

**Étape 5 : Test d’autonomie**

1. Lancer le programme complet d’autonomie :
2. python3 Code\_upgrade/conduiteAutoR.py
3. Laisser la voiture naviguer dans un circuit prédéfini conforme au concours : https://ajuton-ens.github.io/CourseVoituresAutonomesSaclay/
4. Observer le comportement :
   * Évite-t-elle correctement les obstacles ?
   * Suit-elle une trajectoire cohérente ?
   * S’arrête-t-elle en cas de danger ?
5. **Critères de validation :**
   * Détection et réaction correcte aux obstacles
   * Navigation fluide et stable
   * Aucune panne ou arrêt imprévu

|  |  |  |
| --- | --- | --- |
| Etape | Conformité | Commentaires |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |