

Guide Tests et Contrôle SO-ARM 101

Phase 4 : Test et contrôle manuel des robots

Service Ecoles Médias (SEM)

⌚ Prérequis

- Phase 1 complétée (LeRobot installé)
- Phase 2 complétée (Servos configurés avec IDs 1-6)
- Phase 3 complétée (Calibration effectuée)
- Scripts SEM installés depuis GitHub
- Environnement `1erobot` activé

🎯 Objectif des tests

Pourquoi tester ? Les scripts de contrôle permettent de vérifier que tous les servos fonctionnent correctement et respectent les limites de calibration. C'est l'étape finale avant la téléopération.

Les tests permettent de :

- **Vérifier le bon fonctionnement** de chaque servo
- **Valider les limites** définies lors de la calibration
- **S'habituer au contrôle** du robot
- **Identifier les problèmes** avant la téléopération

→ Étape 1 : Préparation

Activation de l'environnement

```
# Activer l'environnement conda  
conda activate lerobot  
  
# Se placer dans le dossier des scripts  
cd ~/lerobot/SO-ARM-101/scripts  
  
# Vérifier les scripts disponibles  
ls SEM_so101_control_*.py
```

Vous devriez voir :

- `SEM_so101_control_follower.py` - Pour tester le Follower
- `SEM_so101_control_leader.py` - Pour tester le Leader

Configuration matérielle

Élément	Vérification
Adaptateur USB	Branché sur le port USB du robot à tester
Alimentation	Active (LED allumée)
Espace de travail	Dégagé pour permettre les mouvements
Calibration	Fichier de calibration présent (Phase 3 complétée)

⚠️ Important : Un seul robot à la fois ! Branchez uniquement l'adaptateur USB du robot que vous voulez tester.

Étape 2 : Lancement du contrôle

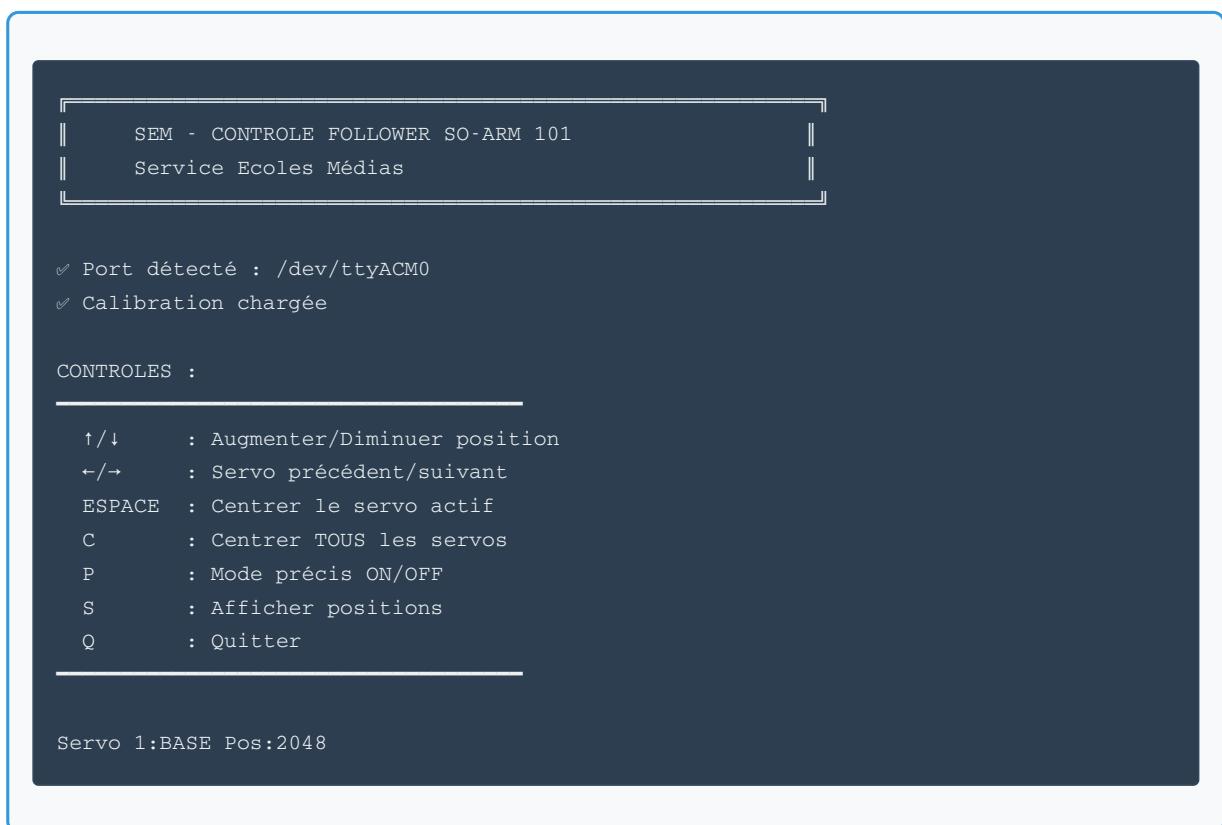
Pour tester le FOLLOWER

```
python SEM_so101_control_follower.py
```

Pour tester le LEADER

```
python SEM_so101_control_leader.py
```

Écran d'accueil



 **Note :** Au démarrage, tous les servos sont automatiquement centrés à leur position de calibration.

III Étape 3 : Commandes de contrôle

Contrôles de base



Augmente la position du servo actif



Diminue la position du servo actif



Sélectionne le servo précédent



Sélectionne le servo suivant

Commandes spéciales



Centre le servo actif à sa position de calibration



Centre TOUS les servos simultanément



Active/désactive le mode précis (pas de 10 au lieu de 50)



Affiche les positions actuelles de tous les servos



Quitte le programme et libère les servos

Mode précis

Mode normal : Pas de mouvement = 50 unités

Mode précis : Pas de mouvement = 10 unités (pour ajustements fins)

L'indicateur **[PRECIS]** apparaît quand le mode est actif :

```
Servo 3:COUDE Pos:2048 [PRECIS]
```

❸ Étape 4 : Tests systématiques

Test 1 : Vérification individuelle de chaque servo

1. Utilisez → pour sélectionner chaque servo (1 à 6)
2. Pour chaque servo :
 - Testez le mouvement vers le MAX avec ↑
 - Testez le mouvement vers le MIN avec ↓
 - Recentrez avec ESPACE
3. Vérifiez que le servo s'arrête aux limites de calibration

Test 2 : Centrage global

1. Bougez plusieurs servos de leur position centrale
2. Appuyez sur C
3. Vérifiez que tous les servos reviennent au centre simultanément

Test 3 : Mode précis

1. Sélectionnez un servo sensible (ex: servo 6 - Pince)
2. Activez le mode précis avec P
3. Testez les mouvements fins avec les flèches
4. Désactivez avec P pour revenir au mode normal

Test 4 : Affichage des positions

Appuyez sur S pour voir toutes les positions :

```
POSITIONS ACTUELLES:  
=====  
Servo 1 (BASE      ): 2048  
Servo 2 (EPAULE    ): 2304  
Servo 3 (COUDE     ): 2048  
Servo 4 (POIGNET-F ): 2176  
Servo 5 (POIGNET-R ): 2048  
Servo 6 (PINCE     ): 2560  
=====
```

⌚ Étape 5 : Tests de coordination

Mouvement de préhension (Follower)

1. **Servo 2 (Épaule)** : Lever le bras
2. **Servo 3 (Coude)** : Plier pour approcher
3. **Servo 4 (Poignet)** : Ajuster l'angle
4. **Servo 6 (Pince)** : Ouvrir puis fermer

Rotation complète (Leader ou Follower)

1. **Servo 1 (Base)** : Rotation gauche maximum
2. **Servo 5 (Poignet rotation)** : Rotation opposée
3. Recentrer avec  C
4. Répéter dans l'autre sens

Position de repos

Pour mettre le robot en position de repos sécurisée :

1. Appuyez sur  C pour centrer tous les servos
2. Puis  Q pour quitter et libérer les moteurs

✓ Tests réussis si :

- Tous les servos répondent aux commandes
- Les limites de calibration sont respectées
- Le centrage fonctionne pour chaque servo
- Pas de bruits anormaux ou de résistance

🔧 Dépannage

Problème	Cause possible	Solution
Script ne démarre pas	Port USB non détecté	Vérifier branchement et permissions : <code>sudo chmod 666 /dev/ttyACM*</code>
"Pas de calibration"	Phase 3 non complétée	Lancer d'abord <code>SEM_so101_calibrate.py</code>
Servo ne bouge pas	Alimentation coupée	Vérifier alimentation (LED)
Mouvement saccadé	Pas trop grand	Activer mode précis avec <code>P</code>
Servo force en butée	Calibration incorrecte	Refaire calibration Phase 3
Flèches ne fonctionnent pas	Terminal incompatible	Utiliser un terminal standard Linux
Position affichée incorrecte	Décalage mécanique	Recentrer avec <code>ESPACE</code>

💡 Conseils d'utilisation

- Commencez lentement** : Testez d'abord en mode normal avant le mode précis
- Surveillez les limites** : Le script empêche de dépasser les valeurs calibrées
- Libérez après usage** : Toujours quitter avec `Q` pour libérer les servos
- Un robot à la fois** : Ne branchez qu'un seul adaptateur USB
- Position de sécurité** : Utilisez `C` en cas de doute

III Comprendre l'affichage

Ligne d'état

```
Servo 3:COUDE Pos:2048 [PRECIS]
```

Élément	Signification
Servo 3	Numéro du servo actif (1-6)
COUDE	Nom du servo
Pos:2048	Position actuelle (0-4095)
[PRECIS]	Mode précis activé (optionnel)

Correspondance servos

ID	Nom	Fonction	Mouvement
1	BASE	Rotation horizontale	Gauche ↔ Droite
2	ÉPAULE	Lever le bras	Haut ↔ Bas
3	COUDE	Plier l'avant-bras	Plié ↔ Tendu
4	POIGNET-F	Flexion du poignet	Haut ↔ Bas
5	POIGNET-R	Rotation du poignet	Gauche ↔ Droite
6	PINCE/POIGNÉE	Préhension	Ouvert ↔ Fermé

❶ Commandes de référence rapide

```
# Installation (si pas fait) cd ~/lerobot git clone https://github.com/yankosem/SO-ARM-101.git # Lancement test Follower conda activate lerobot cd ~/lerobot/SO-ARM-101/scripts python SEM_so101_control_follower.py # Lancement test Leader python SEM_so101_control_leader.py # Contrôles clavier ↑↓↔ : Navigation et mouvement ESPACE : Centre servo actif C : Centre tous les servos P : Mode précis S : Afficher positions Q : Quitter
```

❷ Notes finales

⌚ **Validation complète :** Si tous les tests passent avec succès, votre robot est prêt pour :

- La téléopération (Leader contrôle Follower)
- L'enregistrement de trajectoires
- L'apprentissage par démonstration

✓ **Phase 4 terminée quand :**

- Les 6 servos du Leader répondent correctement
- Les 6 servos du Follower répondent correctement
- Les limites de calibration sont respectées
- Vous maîtrisez les commandes de contrôle