

Servo and Controller Reliability Testing - Gen2 Camera ITS

Author: suketuparekh@

Date: Feb 24, 2026

Overview

This document outlines the method of performing Reliability testing for Gen2 Camera ITS rigs. The purpose of these tests is to qualify critical parts of the Rig and eliminate infant mortality risk for the rigs by cycling all the moving parts within the rig and reporting on potential issues with the quality of parts or assembly. Reliability testing as outlined will take less than 5 minutes of execution time.

Hardware Description

To execute this test, the manufacturing rig must meet the following specifications:

- **Controller:** The Custom controller PCB is being used as per Design (USB-to-Serial adapter using the CH340 chipset)
- **Servo:** Lynxmotion Smart Servo (LSS) HT1 connected to the Controller board.
- **Baud Rate:** 115200
- **Power:** External power supply capable of sustaining peak current for the specific LSS model used - 12V - 10A

Automated Test Sequence

The provided script automates the following two-phase stress test. Each phase performs 50 full cycles (100 movements per phase).

Phase	Motion Range	Speed/Interval	Purpose
Initialization	Move to 0°	2sec Delay	Establishing a neutral baseline.

Phase1: Macro Move	$-90^{\circ} \leftrightarrow +90^{\circ}$	2sec Interval	Gear train travel and potentiometer consistency and PCB stress test.
Phase2: Micro Move	$-15^{\circ} \leftrightarrow +15^{\circ}$	0.25sec Interval	High-speed direction reversals and motor thermal load and PCB stress reliability.

Note: The script uses a specific "Dither" test (Phase 2) with a very short **0.25s** interval. This is designed to test the responsiveness of the LSS bus and the motor's ability to handle rapid PWM switching. Ensure the servo is not obstructed during this phase, as the rapid direction changes generate the highest mechanical stress.

Execution Instructions for Operators

Environment Setup: Ensure the Python environment has `pyserial` (`pip install pyserial`) and the `lss` library (included with ZIP file) installed.

- **Connection:** Connect the PCB controller to the PC and the LSS Servo to the controller and Power as per nominal configuration.
- **Launch:** Unzip `ServoReliability_Gen2ITS.zip`. Once done, navigate to the folder in the terminal and run `python Gen2_ITS_Motor_Cycling.py`.
- **Monitoring:** The script will output "Current Position" in real-time. The operator can verify that the physical movement roughly matches the degrees printed in the terminal.
- **Completion:** Upon completion, the script will generate a Unique Validation Code (e.g., `4R24EM. . .`). This code can be recorded against the Rig assembled for future use.

Automated Reporting

The script automatically generates a timestamped log file in the `./Reports/` directory.

- **File Name Format:** `TestReport_[Timestamp].txt`
- **Log Data:** Includes port detection, connection status, per-cycle position telemetry, and the final validation hash.

Pass/Fail Criteria

While the script checks for software connectivity, the technician must fail the unit if any of the following occur during the 100 cycles:

- **Script Hangs:** If the "**Bus closed**" message does not appear or the script retries connection indefinitely.
- **Position Deviation:** If the "**Current Position**" reported by the servo deviates significantly from the commanded target (indicating a slipping gear or internal encoder error).

Note: *It is expected for the Servo to not reach the target position in the given time due to higher moment of inertia of the mounting plate - this is usually OK. The cycles can show vastly different positions on each cycle - this is NOT OK as it can point at internal mechanical failures in the Servo.*

- **Hardware Alert:** Any "**Controller Not Connected**" error messages in the terminal.

Troubleshooting: Connectivity & Drivers

If the script returns the message "**Controller Not Connected!**", follow these steps to resolve the hardware interface:

- **USB Connections:** Ensure that the USB-C cable used to connect the Computer to the PCB Controller is seated properly on both ends and is working fine.
- **Driver Requirement:** The rig must have the WCH CH341/CH340 serial driver installed. This is usually installed by default for Windows/Mac/Linux systems.
- **Linux/Ubuntu:** Ensure the user has permissions to the dialout group: `sudo usermod -a -G dialout $USER`.

Common Script Errors

Error Message	Likely Cause	Solution
Found CH340 at: None	Hardware not recognized.	Check USB cable; ensure the Vendor ID (0x1A86) matches the hardware.
Error connecting: [Errno 13]	The port is busy.	Close any other Serial Monitors or IDEs using the COM port.
Bus closed. (immediately)	LSS Library failure.	Ensure <code>lss.py</code> and <code>lss_const.py</code> are in the



		same directory as the test script.
--	--	------------------------------------

Operational Safety Notes

- **Cable Strain:** Because Phase 2 executes 50 cycles at a 0.25s interval, ensure the servo wiring has enough slack to prevent tension during rapid oscillations.
- **Emergency Stop:** To terminate the test manually, use `Ctrl+C` in the terminal. The finally block in the script will attempt to close the LSS Bus safely before exiting.
- **Log Preservation:** Do not rename the `./Reports/` folder, as the script relies on this relative path to save the validation data.