Johnson&Johnson
MedTech

Diagnostic Report

Report Title	Diagnostic Report: rotate_joint Command Latency
Name	Aida Aryafar
Date	November 14, 2025
Ticket ID	#2437

Executive Summary

- **Purpose:** To diagnose and identify the root cause of the reported response time delay in the rotate_joint command for the RBA-2201 surgical arm.
- **Key Findings: The rotate_joint command showed an initial response time of 0.155s, significantly exceeding the expected 0.10s. The root cause was identified as a non essential time.sleep (0.15) line in the control code, which artificially created a bottleneck. A simulated optimization demonstrated a 20% improvement, and the final recommendation is the complete removal of this line to restore real time performance.**

Issue Description

- **Problem Statement:** Delayed response in the surgical robotic arm's rotation function (rotate_joint), impacting real time adjustments.
- **Symptoms and Impact:**Noticeable delay when rotating the arm to adjust positioning during procedures. This poses a potential risk to surgical accuracy in high stakes scenarios
- **Client Information:**
  - **Hospital Name:** Mercy General Hospital
  - **Location:** Rockville, Maryland
  - **Reported By:** Dr. Emily Chen, Senior Surgeon

Johnson&Johnson
MedTech

Diagnostic Process

- **Initial Observations:** Initial tests showed response times of: move_arm (0.105s), rotate_joint (0.155s), and adjust_grip (0.055s). The rotate_joint command was the clear outlier and failed to meet performance expectations.
- **Commands Tested:** move_arm, rotate_joint, adjust_grip
- **Hypothesis:** The rotate_joint command exhibits a significant delay (approx. 0.15s) compared to other commands. Analysis of the check_response_time function reveals a hard-coded time.sleep(0.15) statement specifically for this command. This artificial delay is the root cause of the reported latency.
- **Tools and Techniques Used:** Python (time, numpy libraries), Control System Diagnostic Notebook (Jupyter), and iterative testing

Findings and Analysis

Response Time Data:

Command	Expected Response Time	Initial Response Time	Optimized Response Time
move_arm	0.10 seconds	0.105	0.084
rotate_joint	0.10 seconds	0.155	0.12
adjust_grip	0.09 seconds	0.055	0.044

- **Analysis of Findings:** The rotate_joint command was the only one failing to meet its expected performance benchmark (0.155s observed vs. 0.10s expected). This aligns perfectly with the hypothesis that the time.sleep(0.15) line is the sole cause of the latency, as the delay (0.155s) almost exactly matches the sleep time (0.15s)

Optimizations and Solutions

- **Code modifications:** A simulated optimization (optimized_command function) was applied, which reduced the response time by a fixed 20%. The recommended modification is the complete removal of the time.sleep(0.15) line from the check_response_time function.

- **Impact of Optimizations:** The simulated 20% improvement (to 0.12s) confirmed the command is responsive to optimization. The recommended fix (full removal) is expected to reduce the response time to be near-instantaneous (< 0.01s), fully resolving the ticket.

Conclusion

- **Summary of Findings:** A high priority latency issue in the rotate_joint command was diagnosed. The root cause was not an algorithmic error but a hard coded time.sleep(0.15) line in the diagnostic code, likely left over from previous testing.
- **Overall Impact:** Removing this line will restore real-time responsiveness to the 'rotate_joint' command. This eliminates the delay and mitigates the risk to surgical accuracy at Mercy General Hospital
- **Next Steps:**
  - Immediate Fix: Remove the time.sleep(0.15) line from the check_response_time function to eliminate the identified artificial delay.
  - Validation: Re-run the initial diagnostic test (from Step 3) after the fix to confirm the rotate_joint response time is now minimal and in line with other commands.
  - Review: Conduct a broader code review to identify any other non-essential sleep commands or inefficient loops that could impact real-world performance.

.

Johnson&Johnson
MedTech