Muhammad Ahmed Nazir Shaikh

Durham, NC • (919)-236-1638 • ms1242@duke.edu • LinkedIn • GitHub • Portfolio

Summary

Robotics systems-integration engineer (MS, Duke, May 2026) with a track record of standing up robot setups and keeping them running: brought a manipulator from unboxing to reliable operation (ROS 2, regression checks holding failure under 2%), built a multimodal robot on Jetson whose camera-audio-ML pipeline sustains ~87% at 30 Hz, and wrote real-time control and I2C sensor drivers from scratch on a microcontroller (C++, 50 Hz). Hands-on with ROS 2, Python, and Linux daily at the systems level; debugs across the stack, oscilloscopes and logic analyzers at the hardware end, logs and telemetry through drivers and applications, timing and frame-drop bugs in ML pipelines; kernel-level driver work: ready to learn, stated plainly. Builds data pipelines and makes evaluation repeatable: benchmark-style pass/fail criteria, regression suites, CI, and monitoring.

Professional Experience

24 and Up, Robotics Systems Engineer May 2026 – Present

Raleigh, NC (Hybrid)

- Integrate robotic systems across sensors, drivers, compute, and software, standing them up and keeping them running.
- Build data pipelines and tooling (Python, CI/CD) with monitoring and alerting, evaluation made repeatable and reliable.
- Debug across the stack, from hardware signals to application and pipeline bugs, to verified closure.

Duke University, Junior R&D Engineer (Garage Lab) Jun 2025 – May 2026

Pratt School of Engineering, Durham, NC

- Worked hands-on with electromechanical systems daily: build, instrument, test, and troubleshoot.
- Built instrumented test setups, brought up electronics (soldering, schematics, oscilloscopes), and validated against requirements.

S.R.M. Institute of Science and Technology, Research Assistant (Tribology) Apr 2022 – Sep 2023

Chennai, India

- Ran controlled characterization campaigns (pin-on-disk wear, EIS, FTIR) with design of experiments and statistical analysis.
- Quantified a 30% corrosion-resistance improvement and 25% wear reduction against defined limits.
- Co-authored 2+ peer-reviewed publications documenting methods and results.

Robot Bring-Up, Pipelines & Evaluation Projects

Kinova Gen3 Lite Manipulator, Stood Up and Kept Running (GitHub) Aug 2025 – Dec 2025

- Brought a vendor manipulator from unboxing to reliable operation: drivers, ROS 2 stack, planning (MoveIt 2), and task behavior on real hardware (Python/C++, Linux).
- Built the evaluation myself, instrumented commanded-versus-actual logging with pass/fail criteria, and root-caused a 40% failure rate to a sim-to-real divergence, driven under 2% at ~4 mm.

NeuroBot, Sensor Stack to ML Pipeline on Jetson (GitHub) Jan 2025 – May 2026

- Stood up a robot sensor stack (camera, audio) feeding an ML and LLM pipeline on embedded Linux, sustaining ~87% accuracy at 30 Hz.
- Debugged ML pipeline bugs from logs: cross-module timing, dropped frames, and integration faults.

Robot-to-Data Pipeline & Automation (Python, CI/CD) 2025 – 2026

- Built and operate a data pipeline running unattended: scheduled collection across 50+ sources, filtering, structured storage, and reporting.
- Made it reliable: health monitoring, fault detection, and alerting built in, improved from its own run data.

CRUTCH, Real-Time Control and Sensor Drivers from Scratch (ESP32, C++) (GitHub) Aug 2024 – Dec 2024

- Wrote the real-time control loop (50 Hz) and I2C sensor drivers myself, with telemetry over WiFi/TCP, validated across a 50-cycle protocol (600 N to 140 N).
- Root-caused a cross-stack failure, vibration saturating a sensor and corrupting control, with oscilloscope and logic analyzer, fixed and re-verified.

Education

Duke University, M.S., Mechanical Engineering & Materials Science May 2026

GPA 3.74/4.0, Duke Competitive Research Grant

S.R.M. Institute of Science and Technology, B.Tech, Mechanical Engineering May 2023

CGPA 8.18/10, Chennai, India

Technical Skills

Systems Integration: Robots, drivers, sensors, and real-time control integrated on real hardware; MCU-level sensor drivers written from scratch; kernel-level driver work: ready to learn

ROS 2, Python & Linux: ROS 2 and MoveIt 2 on hardware, Python (primary), C++, bash, Linux daily at the systems level, Git

Debug Across the Stack: Oscilloscopes and logic analyzers at the hardware end, telemetry and log analysis through drivers and applications, ML pipeline timing and frame-drop bugs

Pipelines & Evaluation: Data pipelines with structured storage and reporting, benchmark-style pass/fail criteria, regression suites, CI/CD, monitoring and alerting, repeatable evaluation

Platforms: Kinova manipulator (same class as Franka), Jetson embedded Linux, ESP32 real-time firmware, cameras, IMUs, load cells

Research-Lab Mindset: Robot as a means to an end: instrumented experiments, 2+ peer-reviewed publications, clear documentation