

# HRIDYA SATISH PISHARADY

📍 Irvine, CA, 92617 📞 (949) 689-5688 📩 hridyasatishpisharadyok@gmail.com

🔗 <https://github.com/hridyasatish> 💬 <https://www.linkedin.com/in/hridya-satish/>

## Education

<b>University of California, Irvine, California, United States</b> <i>Masters in Embedded and Cyber Physical Systems</i>	<b>Sep 2024 – Dec 2025</b>
Coursework: Embedded Systems Modelling, Wireless Sensors and Actuators, Real-Time Operating Systems(RTOS), IoT, Machine Learning	GPA: 3.9/4.0
<b>SRM Institute of Science and Technology, Chennai, India</b> <i>Bachelors in Electronics and Communication with Specialization in Instrumentation</i>	<b>Jun 2019 – Jun 2023</b>
Coursework: Digital Signal Processing, Analog and Digital Electrical circuits, Signals and Systems, Control Systems, Circuit designing	GPA: 3.72/4.0

## Technical Skills

- Programming Skills:** C, C++, Linux, Bash, Python, Data Structures, Algorithms, OOPS, Git/GitHub  
**Embedded Systems:** ESP32, ESP8266, STM32 (ARM Cortex-M0), Arduino, Raspberry Pi, Distributed Systems  
**Testing Equipment:** Function Generators, Oscilloscopes, Logic Analyzers, Debuggers, Soldering, Digital Multimeters  
**Hardware Design:** Eagle, Simulink, LTSpice, KiCAD, MATLAB, Circuit Debugging, PCBA Schematics/Layout  
**Communication Protocols:** I2C, UART, UDP, Ethernet, CAN, USB, SPI, TCP/IP, Bluetooth, BLE, Wi-Fi, SDIO, Cellular

## Experience

<b>Canvas Construction Inc</b> <i>Electrical Engineering Intern</i> <a href="#">[GitHub]</a>	<b>Jun 2025 – Sep 2025</b>
	<i>San Francisco, USA</i>

• Designed industrial-grade **ESP32-based CAN nodes** for drywall painting robots, managing the full hardware prototyping from **schematic capture** and **PCB layout** in KiCad to **BoM optimization** and manual assembly using **SMT/THT** soldering.

• Created a robust **3.3V power supply** using **buck converters** to step down 24V inputs for the ESP32, implemented digital input protection with **MOSFETs** and analog signal conditioning with **op-amps** ensuring **EMI resilience** and **signal integrity**.

• Executed **CAN bus debugging and diagnostics** for **TJA1051T transceivers** using **PCAN tools**, and validated **signal integrity** and **EMC compliance** using **LTspice**, **oscilloscopes**, and waveform-based root cause/risk analysis.

## Synaptics Incorporated

<i>Embedded Systems and IOT Apprentice</i>	<b>Mar 2025 – Present</b>
	<i>Irvine, CA, USA</i>

• Built an embedded navigation assistant on the **Synaptics Astra AI board** running **Yocto Linux**, enabling real-time processing, structured GPIO mapping and efficient resource utilization on low-power edge hardware tailored for visually impaired users.

• Integrated **HC-SR04 ultrasonic** sensors with haptic motor feedback driven by a **ULN2003 Darlington IC** and implemented a fall detection feature using MPU6050 which triggered a buzzer alert and logged sensor data to **CSV files** using Python.

• Used a Logitech webcam with a Roboflow-trained model converted through Synaptics **SyNAP** offline optimization. Modeled navigation state transitions in **MATLAB Stateflow** for reliable and predictable system behavior.

## GradeUp

<i>Embedded Systems Developer</i>	<b>Dec 2022 – Apr 2023</b>
	<i>Chennai, India</i>

• Developed real-time embedded firmware in **C** for **ESP8266** MCU-based home automation using bare-metal programming with hardware register access to optimize interrupt handling and built a web-based user interface to monitor sensor states over Wi-Fi.

• Designed a motion-sensing algorithm using **PIR sensors** and implemented interrupt-driven logic for automated lighting control.

## Projects

<b>Light Swarm: Master–Slave Communication over Wi-Fi   IoT Networking, Data Visualization</b> <a href="#">[GitHub]</a>	
• Developed firmware in <b>C</b> for <b>ESP8266</b> using Arduino IDE to read <b>photoresistor</b> values, control <b>PWM</b> to implement master selection logic, and broadcast sensor data via <b>UDP multicast</b> for distributed master–slave communication.	

• Built a Python-based Raspberry Pi controller to log transitions into **CSV files** for real-time network monitoring.

• Visualized sensor data with **Matplotlib** in Python, demonstrating embedded networking concepts in a distributed **Wi-Fi mesh**.

## Line-Following Robot | Arduino, PID Control, PCB Design

<a href="#">[GitHub]</a>	
• Implemented embedded firmware in <b>Arduino</b> for a line-following robot, integrating <b>IR sensors</b> for surface detection and an <b>SPI-based IMU</b> for motion tracking, designed the <b>PCB in Eagle</b> , and ensured seamless hardware–software integration.	

• Implemented actuator control using **PWM** and **H-Bridge** motor drivers with **UART** command interface for precise motor drive.

• Tuned the **PID controller** for robust control stability under actuator wear, battery voltage sag, and reflective surface variability, validating **sensor reliability** through repeated-cycle tests.

## Canny Edge Detector | C++, Real-Time Systems, Multithreading, DSP

<a href="#">[GitHub]</a>	
• Developed an <b>Edge Detection Algorithm</b> in <b>C++</b> with multithreading and <b>PREEMPT-RT Linux</b> patches, integrating a <b>Pi Camera</b> for real-time edge detection and achieving <b>98% accuracy</b> across 30 test images.	

• Optimized performance to **30 FPS** using **DSP**-based noise filtering and utilized **CPU affinity** on Raspberry Pi's **4-core** setup to isolate real time **RT** and non-real time **NRT** tasks, improving system throughput and reducing jitter under load .

## Patent

<b>Sensor Based Physics Forceps for Tooth Extraction (Patent No. 542943)</b> <a href="#">[GitHub]</a>	<b>Sep 2022 – Jul 2024</b>
• Modified the <b>Physics Forceps</b> using an <b>ESP32</b> and <b>FSR pressure sensor</b> to monitor force levels during dental procedures.	

• Created a custom wearable wristband by integrating an **MPU6050** via **I2C**, providing real-time angle feedback to clinicians.

• Demonstrated embedded system integration and signal acquisition, delivering real-time alerts via **Wi-Fi (Blynk App)** and audio feedback for unsafe force detection, ensuring **95%** of force applications remained within safe limits during clinical testing.