

Shrey Patel

US Permanent Resident | (470)-654-8185 | spatel3207@gatech.edu | LinkedIn: [shrey-patel25](#) | GitHub: [shreyp3](#)

EDUCATION

Georgia Institute of Technology

Bachelor of Science in Computer Engineering

Atlanta, GA

Expected May, 2028

- Relevant Coursework: Data Structures and Algorithms, Microcomputer Fundamentals, VHDL Design, Signals & Systems, Circuits

EXPERIENCE

Mercer University School of Engineering

Undergraduate Researcher

Nov. 2024 – Jan. 2026

Macon, GA

- Operated and maintained the hardware research lab's **oscilloscopes**, **bench power supplies**, and Vicon motion capture system, which supported successful **data collection** for three robotics experiments and sharpened instrumentation and debugging skills.
- Utilized rapid-prototyping tools such as **3D printing**, **soldering** stations, and **CNC cutters** to design and build custom hardware components, enabling two robotics prototypes to be completed on schedule.
- Collaborated with robotics lab members to integrate hardware using **SPI** and **UART** interfaces and to develop control software in **C++** and **Python**, which reduced system-integration debugging time and improved project workflow.
- Presented the Semi-Autonomous Robot Car and Gesture-Based Drone Control projects at **undergraduate research conferences**.

Mercer University Academic Resource Center

Teaching Assistant

Jan. 2025 – Dec. 2025

Macon, GA

- Facilitated group physics sessions several times a week for **50+ students**, raising average grade outcome by a **full letter grade**.

PROJECTS

[Quadcopter & RC System](#) | *ESP32-C3, ESP-NOW, DShot600, MPU6050*

July 2026 – Present

- Architecting a custom micro quadcopter (**2" props**, 2S brushless) with a flight controller built on the **Arduino-ESP32** framework, designing **complementary-filter IMU** sensor fusion and cascaded **PID control** targeting **250–500Hz** attitude stabilization.
- Designed a **two-node RC system** with a **custom-built ESP32 remote** and an **ESP-NOW** radio link at **50–100Hz**, including an arm/disarm state machine and link-loss failsafe with automatic motor cutoff.
- Engineering the full electrical system, integrating a **4-in-1 DShot600 ESC** via the **DShotRMT** library and **MPU6050 IMU** sensing, performing component selection and thrust-to-weight budgeting across an **18-item** bill of materials.

[AI-Powered Wearable Health Monitor](#) | *ESP32, FreeRTOS, React Native, Groq Llama 3, KiCad*

May 2026 – Present

- Designed and fabricated a **custom 2-layer PCB** in **KiCad** (JLCPCB, hand-soldered assembly) integrating the **ESP32** and **3 biometric sensors** into a compact wearable form factor.
- Engineered ESP32 firmware under **FreeRTOS** to sample sensors over **I2C** at **100Hz**, streaming structured **JSON** vitals over **BLE** to a custom **React Native iOS** app with live dashboards.
- Built an **AI agent** on **Groq Llama 3** with persistent session memory, analyzing real-time vitals to generate personalized health feedback.

[8-Bit CPU](#) | *VHDL, Intel FPGA, Quartus Prime*

Oct. 2025 – Nov. 2025

- Designed and implemented a functional **8-bit CPU** in **VHDL** on an **Intel FPGA**, architecting an **ALU**, register file, FSM-based instruction decoder, and ROM-based instruction memory to execute a custom **19-instruction ISA**.
- Validated correct arithmetic and branching behavior through **hand-assembled machine code** programs, verifying execution accuracy across edge cases via **simulation** and **live FPGA testing**.
- Implemented **.mif-based ROM** initialization for persistent instruction storage, demonstrating end-to-end control of instruction **fetch**, **decode**, and **execute** pipeline stages.

[Semi-Autonomous Robot Car](#) | *Raspberry Pi, Arduino, OpenCV, Python*

May 2025 – Aug. 2025

- Engineered a voice- and vision-controlled autonomous vehicle on a **Raspberry Pi** that tracks and follows a user in real time using offline speech recognition via **Vosk** and visual tracking via **OpenCV**, eliminating cloud dependency.
- Architected a **serial communication bridge** between the Raspberry Pi and **Arduino** to translate high-level decisions into low-level **PWM motor** and **servo control** signals, achieving smooth real-time response.
- Optimized the sensor processing and control pipeline to minimize end-to-end command latency, enabling reliable **follow-me behavior** through tuned computer vision parameters and **control loop timing**.

TECHNICAL SKILLS

Programming: C, C++, Python, VHDL, Embedded C, Java, MATLAB, C#

Embedded & Hardware: FPGA (Intel/Altera), ESP32, Arduino, Raspberry Pi, TM4C1294, I2C, SPI, UART, BLE, FreeRTOS

Software: Quartus Prime, Code Composer Studio, KiCad, Git, Linux, LTspice, Visual Studio Code, Cursor, Claude Code