

Product Engineering Program

Learn engineering by building real products - PCB to Cloud to AI

Monthly: per month, weekday or weekend batches

3 months together: for 3 months, weekday or weekend batches

Starts next week

204 milestones

WHAT YOU FINISH WITH

3-4 real industrial products, a PCB in your hand, a GitHub portfolio and a demo you can show an employer

The syllabus

Stage 1 - Build a device that works [Foundation]

Phase 1 - Plan it, and set up your tools

R&D + Project Planning

- Introduction to Product Engineering
- R&D methodology - how to research a product
- Block diagram + architecture design
- BOM preparation + component selection
- Git repository setup from Day 1
- ESP-IDF + IoT toolchain installation

Microcontroller + Embedded C

- ESP32 architecture + dual-core deep dive
- Embedded C / C++ fundamentals
- Memory layout - stack, heap, BSS
- Pointers, structs, bit manipulation
- ESP-IDF project structure + build system
- Serial monitor + basic debugging

Phase 2 - Wire up the sensors

GPIO + Sensors + Actuators

- LED + Relay - GPIO DIO programming
- LDR - ADC (light intensity sensing)
- IR Sensor - ADC (motion detection)
- Servo Motor - PWM control
- OLED Display - I2C protocol
- Combine all on breadboard: ESP32 + LED + LDR + IR + Relay + Servo

FreeRTOS + HTTP Server

- FreeRTOS task creation + scheduling
- Task stack sizing + priority management
- Run sensors + display as parallel FreeRTOS tasks
- HTTP server on ESP32 (local control)
- Hardware debug + integration testing
- Test complete product end-to-end

Wi-Fi + Provisioning

- Wi-Fi station and access-point modes on ESP32
- Wi-Fi provisioning: let the user enter Wi-Fi details from a phone
- Keep credentials in NVS flash, never in the code

- 2.4 GHz only, signal strength (RSSI) and auto-reconnect

Phase 3 - Design your own PCB

KiCad PCB Design

- Schematic capture in KiCad
- Component footprint + library management
- PCB layout + routing
- Gerber file generation
- Generate Gerber files, ready to manufacture

Git + Project Wrap Up

- Git commit discipline + branching
- GitHub README + project documentation
- Comm. protocols overview (SPI, UART, CAN)
- Prototype tested + pushed to GitHub
- Ready to move to PCB + MVP phase

Professional Git workflow

- Branches and pull requests, the way teams work
- Code review on GitHub: give and take review comments
- Coding standards for embedded C (naming, MISRA-style rules)
- Versioning firmware and writing release notes

STAGE OUTCOMES

- * Working Smart Bulb Prototype
- * FreeRTOS Firmware
- * HTTP Local Control
- * OLED Dashboard
- * Manufacture-ready PCB design
- * GitHub Repo
- * Certificate

Stage 2 - Assemble your device, and put it online [Intermediate]

Phase 1 - Assemble it, then test it

PCB Assembly

- Assemble the device on a prototype board
- Soldering with expert guidance + soldering kit
- Component assembly on PCB
- Hardware testing + validation
- Upload firmware to PCB
- Test full product on PCB

Enclosure + FreeRTOS Advanced

- Enclosure design mentorship
- Fitting PCB + product finishing
- Demo video recording
- FreeRTOS queues + semaphores (deep dive)
- Watchdog timer + error handling in RTOS
- Priority inversion + event groups

Phase 2 - Put it online

MQTT + Cloud

- MQTT protocol - publish/subscribe, QoS, topics
- MQTT broker setup (local + cloud)
- Write MQTT firmware on ESP32
- Test with readymade MQTT apps
- Going online from offline - architecture
- OTA firmware update basics

Python Dashboard + APIs

- Python web app development (Flask/FastAPI)
- Design dashboard - collect data via MQTT
- HTTP REST APIs to access device data
- Enable data storage (SQLite/PostgreSQL)

- Data visualization dashboard (charts + logs)
- Test, debug, deploy complete product

AWS IoT Core

- AWS IoT Core: Things, topics, Device Shadow and Rules
- X.509 certificates and IoT policies
- ESP32 to AWS IoT over MQTT with TLS
- Python to and from AWS IoT
- IoT Rules: store device data in a database
- Live dashboard on AWS data

OTA in production

- OTA over HTTPS with signed firmware images
- Rollback when a new firmware fails
- Rolling out an update to many devices safely

Phase 3 - Write it up, then present it

Technical Documentation

- Circuit diagram + PCB documentation
- Firmware documentation + Git final push
- Testing report writing
- Project PPT + technical report
- Portfolio README - industry-ready format

Pitch + Marketing

- How to pitch your product to a client
- Sales basics for engineers
- Marketing draft for LinkedIn + GitHub
- Upwork freelancing profile setup - intro
- Product 1 complete - portfolio entry done

STAGE OUTCOMES

- * PCB Assembled Product
- * MQTT Cloud Connected
- * Python Dashboard
- * REST APIs Built
- * Documentation
- * Demo Video
- * Portfolio Entry #1
- * Certificate

Stage 3 - Industrial control on STM32 [Advanced]

Phase 1 - Get STM32 running

STM32 Environment + Architecture

- STM32CubeIDE + CubeMX installation + workflow
- Cortex-M4 architecture - memory map, bus matrix
- Clock tree configuration (HSI/HSE/PLL)
- NVIC interrupt management + SysTick timing
- HAL vs LL drivers - when to use each
- DMA implementation for data transfer

STM32 Peripherals

- Interface LED (DIO) - GPIO programming
- Interface Potentiometer + LDR (ADC)
- Interface OLED Display (I2C)
- Interface Servo Motor (PWM + Timer)
- Watchdog timer + brown-out reset
- Combine all - Solar Tracker offline demo

Industrial protocols

- CAN bus: frames, IDs, bit timing and filters
- CAN on STM32 with a transceiver, sending and receiving
- RS485 and Modbus RTU: master and slave
- Reading a real Modbus device (energy meter or PLC)

Phase 2 - Run many things at once

FreeRTOS Advanced

- Task creation + preemption on STM32
- Queues - inter-task data passing
- Semaphores + mutexes - resource management
- Priority inversion detection + inheritance
- Event groups + task notifications
- Real-time timing analysis

GDB Debugging + Testing

- GDB debugging - breakpoints, watchpoints
- SWD/JTAG workflow with ST-Link
- Full product integration test offline

Testing + CI

- Unit testing embedded C (Unity / Ceedling)
- Hardware-in-the-loop testing basics
- GitHub Actions: build the firmware on every push
- Static analysis to catch bugs before the board does

Reliability

- Bootloader basics and firmware layout in flash
- Watchdog, brown-out and safe recovery
- Low-power modes and battery life

Phase 3 - Connect it to a dashboard

STM32 ESP32 UART Bridge

- UART protocol deep dive + framing
- Interface STM32 with ESP32 over UART
- Send sensor data STM32 -> ESP32
- ESP32 receives + publishes to MQTT/HTTP
- FreeRTOS tasks managing UART on both sides

HMI + Industrial Dashboard

- Design HMI dashboard over ESP32 + Python server
- Live industrial data visualization
- Positioning data display + logs
- Demonstrate Online Solar Tracker + Positioning System
- Git final push + documentation

STAGE OUTCOMES

- * STM32 Solar Tracker
- * FreeRTOS Advanced Firmware
- * UART STM32ESP32
- * Industrial Dashboard
- * GDB Debug Portfolio
- * Portfolio Entry #2
- * Certificate

Stage 4 - Make it think, then go to work [Industry-ready]

Phase 1 - Make it see

AI Home Automation (Computer Vision)

- OpenCV setup on Linux
- Camera interfacing + image capture pipeline
- Face detection + recognition basics
- Motion detection using OpenCV
- Gesture-based home control
- Trigger relay/bulb/fan via computer vision
- Connect CV output to IoT device (ESP32)
- End-to-end: Camera -> AI -> Hardware control

Industrial Object Detection

- Object detection using OpenCV (contours, HOG)

- YOLO intro - industrial use case
- Defect detection on conveyor/production line
- Counting objects on production line
- Alert system - trigger alarm via IoT on detection
- Connect detection output to MQTT/cloud
- Real industrial use case demonstration

Phase 2 - Handle real data, at scale

BLE + Advanced IoT

- BLE GATT - services, profiles, central/peripheral
- Advanced MQTT - fleet management, multi-device
- Bulk IoT data collection architecture
- OTA bulk firmware update - fleet management

AI on IoT Data

- AI on bulk sensor data - anomaly detection
- Time-series data analysis with Python + AI
- Predictive maintenance concept + demo
- Bulk IoT data management pipeline
- Real industrial AI use case end-to-end

IoT security

- TLS and certificate management for devices
- Secure boot and flash encryption on ESP32
- Provisioning many devices safely
- Data security and privacy basics

Phase 3 - Turn it into work

Technical Documentation + Portfolio

- SRS / PRD writing for real products
- Block diagram + DFD + system architecture docs
- BOM costing + sourcing for production
- Testing report + client handover package
- GitHub portfolio finalization - both products
- LinkedIn profile + resume for embedded roles

Career + Freelancing + Startup

- Mock interviews with industrial experts - real feedback
- Job referrals to industry network
- Upwork freelancing profile + first proposal
- Startup mentorship - how to build a product company
- Industrial Expert Sessions - real industry insights
- Group Discussions + Peer Reviews

Production engineering

- BOM costing for 100 and 1,000 units
- Design for manufacture and test (DFM / DFT)
- EMI / ESD basics and certification (CE, BIS) overview
- Test jig and production flashing

Working like an engineer

- Agile sprints and estimating your own tasks
- Tracking work in GitHub Projects or Jira
- Handling changing requirements with a client

STAGE OUTCOMES

- * AI Smart Home Product
- * Industrial Object Detection
- * IoT AI Pipeline
- * Freelancing Setup
- * Placement Assistance
- * Job Referrals
- * Community Access - Lifetime
- * Certificate

Plans and fees

Monthly (Pay one month at a time)

Rs 9,999 + GST per month

Rs 11,799 incl. 18% GST

- + Roadmap: one stage a month, starting with your first device and your own PCB design
- + Mini lab set-up kit
- + Live classes (online / offline)
- + Live projects
- + Doubt support
- + Learning resources
- + Certificate
- + Access to recorded lectures

3 months together (Kit + placement, pay once)

Rs 29,997 + GST for 3 months

Rs 35,396 incl. 18% GST

- + Roadmap: all 4 stages - 3-4 products, STM32 and edge AI
- + Mini lab set-up kit
- + Placement assistance
- + Live classes (online / offline)
- + Live projects
- + Doubt support
- + Learning resources
- + Certificate
- + Access to recorded lectures

Batches: weekdays Mon-Fri 1.5 hrs a day, or weekends Sat-Sun 3 hrs a day. Pay month by month, one stage at a time, or pay 3 months together and save. The mini lab set-up kit comes with every plan. Scholarship test: up to Rs 3,000 off your first month. Seats left in this batch: 10.

About Altrobyte Lab

Altrobyte Lab is the training arm of Altrobyte Automation, which builds IoT products and automation for industry. What you learn here is what we ship: telematics, industrial dashboards and connected devices.

Clients and partners

Vedanta (BALCO), Tata Steel, Continental Conveyors, Unitech AI Solutions, My e-Kisan, Pi by 3, Renu Factory Automation, Nanotech Solutions, Oxbow Intellect, Anedya, nlink, Dhartea, NaviGo

Training partners

NIT Warangal, IIT Bombay, ITM University Gwalior, SISTec Bhopal, Banasthali Vidyapith, Chameli Devi Group (CDGI)

Your trainer

Pawan Meena - founder and lead trainer. Builds embedded and IoT products for industry.

Sr. IoT Engineer: CAN, telematics and GPS for smart vehicles at Eicher Motors (Heximpact Tech); CAN-based vehicle tracking (Navigo Tech Lab). R&D Engineer at Scientech; embedded work at Mahindra TEQO and DRMZ Automation.

Has taught at NIT Warangal (Faculty Development Programme on IoT and AI), Banasthali Vidyapeeth, an IIT Bombay robotics workshop, ITM University Gwalior and SISTEC Bhopal.

Why Altrobyte

- You build real products: you design your own PCB, put your device online and make it think.
- Live classes online or offline in Indore, recorded lectures and doubt support.
- Online and offline classes with a mini lab set-up kit in every plan, and an AI lab trainer that checks each step on your own board.
- You leave with 3-4 products, a GitHub portfolio, demo videos and a certificate; the 3-month plan adds placement assistance.

Book your seat

See the roadmap, fees and EMI, check your profile and enrol online. A mentor will help you choose a plan - no pressure.

Enrol: altrobytelab.com/roadmap

WhatsApp / call a mentor: +91 72229 77927

YouTube: youtube.com/@pawanmeena.official | Indore and online

Contact

Email: pawan@altrobytelab.com

Visit: 5th Floor, Gravity Mall, Mechanic Nagar, Indore 452010, India

Website: altrobytelab.com

Altrobyte Lab - altrobytelab.com/roadmap

Programme duration, project scope and technology coverage may vary by batch and learner progress.