EDVANTA

Ignite your growth, inspire your mind

Embedded Systems: A 3-Month Industry-Ready Program

A structured, hands-on online course designed for early professionals, graduate students, and career-transitioning learners who want to build real-world embedded systems skills aligned with today's industry expectations.

12 WEEKS · 3 PHASES · INDUSTRY-ORIENTED

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

WHY THIS COURSE MATTERS

Embedded Systems Are Everywhere — And Hiring Is Surging

The Market Reality

From smart home devices and wearables to automotive ECUs and industrial controllers, embedded systems power billions of devices worldwide. The global embedded systems market is projected to exceed $130B by 2028, with demand for skilled engineers outpacing supply in nearly every sector.

What This Course Solves

- Most learners know C/C++ but can't apply it to hardware constraints
- Universities teach theory; industry wants practical firmware skills
- RTOS, debugging, and low-level optimization are rare competencies
- Portfolio projects and real-world context are hard to find on your own

Who Wins With This Course

- Early-career engineers looking to specialize
- Graduate students bridging the academia-industry gap
- Software developers transitioning into firmware and hardware roles

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

PROGRAM OVERVIEW

Course Format & Weekly Structure

Every week follows a consistent rhythm — self-paced concept learning combined with live mentoring to ensure real understanding, not just passive watching.

🎥 Weekend Recorded Session

Concept-focused video modules covering core theory, tools, and demonstrations. Watch at your own pace, pause, rewind, and revisit anytime.

🎙️ Live Mentoring Session

Weekly 1:1 or small-group live sessions for doubt clearing, code reviews, case discussions, and personalized feedback from an industry expert.

🛠️ Hands-On Exercises

Each week includes practical coding tasks, hardware simulations, or firmware challenges that reinforce learning with real tools and workflows.

📦 Assignments & Mini Projects

Weekly deliverables culminate in three phase-end mini projects — each serving as a portfolio artifact you can show in job interviews.

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

PHASE 1 · WEEKS 1–4

Foundation Phase: Core Concepts & C Programming for Embedded

Build the bedrock. This phase ensures every learner — regardless of background — develops a solid, hardware-aware programming foundation before touching real peripherals.

1

Week 1 — Embedded Systems Fundamentals

- What is an embedded system? Architecture overview
- Microcontroller vs. microprocessor vs. FPGA
- Harvard vs. Von Neumann architecture
- Memory types: Flash, SRAM, EEPROM, ROM
- Introduction to development boards: Arduino, STM32, ESP32
- Setting up your toolchain: IDE, compiler, flasher

2

Week 2 — C Programming for Embedded Systems

- Data types, pointers, and memory layout in C
- Bit manipulation: masking, shifting, toggling registers
- Volatile keyword and why it matters in embedded C
- Struct, union, and typedef for hardware abstraction
- Inline functions and macros for efficiency
- Writing clean, maintainable embedded C code

3

Week 3 — GPIO, Timers & Interrupts

- GPIO configuration: input, output, pull-up/down
- Polling vs. interrupt-driven I/O — tradeoffs
- Timer modes: overflow, compare, PWM generation
- Interrupt Service Routines (ISRs): writing and debugging
- Debouncing buttons in hardware and software
- Mini project: LED dimmer with PWM and button interrupt

4

Week 4 — Communication Protocols: UART, SPI, I2C

- Serial communication fundamentals: baud rate, framing
- UART: full-duplex async communication, debugging via serial
- SPI: master/slave, clock polarity/phase, CS lines
- I2C: addressing, ACK/NACK, multi-device bus management
- When to use UART vs. SPI vs. I2C in design
- Phase 1 mini project: sensor data acquisition over I2C/SPI

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

PHASE 2 · WEEKS 5–8

Intermediate Phase: RTOS, Peripherals & Power Management

Move from bare-metal code to production-grade embedded software. This phase introduces the tools and techniques that separate hobbyists from professional embedded engineers.

1

Week 5 — Memory Management & Linker Scripts

- Stack vs. heap in embedded systems — why it matters
- Static vs. dynamic memory allocation tradeoffs
- Reading and writing linker scripts (.ld files)
- Memory-mapped I/O and register access patterns
- Understanding startup code and vector tables
- Exercise: Analyze memory map of a bare-metal STM32 project

2

Week 6 — Real-Time Operating Systems (RTOS)

- What is an RTOS? Scheduler types: preemptive vs. cooperative
- FreeRTOS: tasks, priorities, and the tick timer
- Inter-task communication: queues, semaphores, mutexes
- Avoiding priority inversion and deadlocks
- Task profiling and CPU usage analysis
- Mini project: multi-task sensor fusion using FreeRTOS

3

Week 7 — ADC, DAC & Sensor Interfacing

- ADC fundamentals: resolution, sampling rate, reference voltage
- DMA-driven ADC for efficient data acquisition
- DAC usage: audio waveforms, analog control signals
- Interfacing temperature, pressure, and IMU sensors
- Signal conditioning: filtering, scaling, and calibration
- Exercise: Build a data logger with ADC + UART output

4

Week 8 — Power Management & Low-Power Design

- Power modes: active, sleep, deep sleep, standby
- Wake sources: RTC alarm, external interrupt, watchdog
- Current profiling with a bench multimeter and energy profiler
- Low-power peripheral configuration strategies
- Battery life estimation calculations
- Phase 2 mini project: IoT node with RTOS + low-power sleep cycles

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

PHASE 3 · WEEKS 9–12

Advanced & Application Phase: Connectivity, Safety & Capstone

Apply everything in real-world contexts. This phase covers connectivity, security, production-grade practices, and culminates in a full capstone project that becomes your portfolio centerpiece.

1

Week 9 — Wireless Connectivity: BLE, Wi-Fi & MQTT

- BLE fundamentals: GATT profiles, advertising, pairing
- Wi-Fi on ESP32: station mode, AP mode, provisioning
- MQTT protocol: broker, publisher, subscriber architecture
- Sending sensor data to cloud dashboards (AWS IoT / MQTT)
- OTA firmware updates: concepts and implementation
- Exercise: Publish temperature data to cloud via MQTT

2

Week 10 — Bootloaders, Firmware Updates & Security

- Bootloader architecture: primary, secondary, recovery
- Writing a simple custom bootloader on STM32
- Secure boot and firmware signing fundamentals
- Flash memory management: wear leveling, erase cycles
- Common embedded vulnerabilities: buffer overflow, hardcoded keys
- Exercise: Implement a firmware version check + update via UART

3

Week 11 — Debugging, Testing & Code Quality

- JTAG/SWD debugging: breakpoints, watchpoints, step execution
- Logic analyzer and oscilloscope use in embedded debugging
- Unit testing embedded C with Unity or Ceedling
- Static analysis tools: PC-lint, cppcheck, Coverity basics
- MISRA-C coding standards: purpose and key rules
- Exercise: Debug a seeded bug in a provided firmware codebase

4

Week 12 — Capstone Project & Career Prep

- Capstone: Full embedded IoT system from spec to deployment
- System design document + schematic review
- Demo presentation: live device demo + code walkthrough
- Portfolio packaging: GitHub README, demo video, write-up
- Mock technical interviews: embedded-specific Q&A practice
- Final outcomes: job readiness assessment + next-step roadmap

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

TOOLS & TECHNOLOGY STACK

Everything You'll Work With in This Course

No proprietary or expensive tools. Every platform in this course is either open-source or widely used in industry, ensuring your skills transfer directly to a professional environment.

Hardware Platforms

- **STM32 Nucleo / Discovery** — ARM Cortex-M, industry standard
- **ESP32** — Wi-Fi + BLE, popular in IoT products
- **Arduino Uno/Mega** — Prototyping and early exercises
- **Raspberry Pi Pico** — Dual-core ARM, affordable and capable

Software & IDEs

- **STM32CubeIDE** — Full-featured ARM development environment
- **VS Code + PlatformIO** — Cross-platform embedded dev
- **FreeRTOS** — Open-source real-time kernel
- **GDB + OpenOCD** — Debugging via JTAG/SWD

Instrumentation & Cloud

- **Logic Analyzer (Saleae / PulseView)** — Protocol decoding
- **AWS IoT Core / HiveMQ** — MQTT broker and cloud
- **Unity/Ceedling** — Embedded unit testing framework
- **GitHub** — Version control and portfolio hosting

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

INDUSTRY EXAMPLE 1

Automotive: Engine Control Unit (ECU) Firmware Development

The Business Problem

A Tier-1 automotive supplier needed to update fuel injection timing algorithms in their ECU firmware to comply with Euro 6 emissions regulations — without hardware redesign. The firmware ran on a 16-bit Renesas microcontroller with strict real-time deadlines.

How Course Concepts Applied

- Interrupt-driven ADC sampling for crankshaft position sensor
- RTOS task scheduling to meet hard real-time deadlines (<1ms)
- CAN bus communication with engine management system
- MISRA-C compliance and static analysis tools (Polyspace)
- OTA-style bootloader for dealer-side calibration updates

Measurable Business Impact

18%

Fuel Efficiency Gain

Improved injection timing reduced fuel consumption significantly

$2M

Compliance Cost Avoided

No hardware redesign required — firmware-only fix

Role of Early Professionals

- Writing and unit-testing individual firmware modules
- Running HIL (Hardware-in-Loop) simulation test cases
- Documenting register configurations and timing diagrams

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

INDUSTRY EXAMPLE 2

Medical Devices: Wearable Continuous Glucose Monitor (CGM)

The Business Problem

A medtech startup developing a wearable CGM patch needed firmware that sampled glucose biosensor data every 5 minutes, stored readings locally, and transmitted them via BLE to a companion mobile app — all while running on a coin cell battery for 14 days.

How Course Concepts Applied

- Low-power design: deep sleep between sampling windows
- ADC calibration and signal filtering for biosensor accuracy
- BLE GATT profile for standardized health data transmission
- FreeRTOS for managing sampling, storage, and BLE tasks
- Secure firmware signing to meet FDA cybersecurity guidance

Tools & Technologies Used

- Nordic nRF52840 (ARM Cortex-M4 + BLE SoC)
- Zephyr RTOS with BLE stack
- Energy Profiler (Otii Arc) for battery life validation
- Python scripts for sensor calibration curve fitting

Measurable Impact

- **Battery life achieved:** 15.2 days (exceeded 14-day target)
- **Data accuracy:** ±5 mg/dL — within clinical acceptance range
- **Time to FDA submission:** Reduced by 3 weeks through clean firmware audit trail

Early Professional's Role

Junior firmware engineers owned the BLE GATT service implementation and wrote automated regression tests for the ADC sampling pipeline.

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

INDUSTRY EXAMPLE 3

Industrial IoT: Predictive Maintenance on Factory Equipment

The Business Problem

A manufacturing plant was experiencing unplanned downtime on CNC machines, costing ~$50K per hour in lost production. The engineering team deployed vibration and temperature sensor nodes on each machine to detect abnormal signatures before failure occurred.

How Course Concepts Applied

- High-speed ADC sampling for vibration (accelerometer via SPI)
- Edge DSP: FFT-based frequency analysis on-chip (ARM CMSIS-DSP)
- MQTT over Wi-Fi to push anomaly alerts to SCADA dashboard
- FreeRTOS managing concurrent sampling and communication tasks
- OTA firmware updates to tune thresholds without downtime

Measurable Business Impact

73%

Downtime Reduction

Unplanned stoppages dropped dramatically in 6 months

$1.2M

Annual Savings

Maintenance costs and production loss savings combined

Role of Early Professionals

- Integrating sensor libraries and validating ADC accuracy
- Writing MQTT publisher code and testing cloud connectivity
- Tuning FFT thresholds based on field data collected on-site

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

SKILLS MAPPING

Your Embedded Systems Skill Journey

This course is designed so that every week moves you measurably forward on the skill ladder — from writing basic C to delivering production-grade firmware that passes code review.

🔰 Beginner (Weeks 1–2)

Understand microcontroller architecture, write embedded C, perform bit manipulation, configure a toolchain, and blink an LED with confidence.

🔧 Developing (Weeks 3–5)

Implement GPIO, timers, interrupts, and serial protocols (UART/SPI/I2C). Understand memory layout, linker scripts, and hardware abstraction layers.

⚙️ Intermediate (Weeks 6–8)

Deploy FreeRTOS-based multi-tasking firmware. Interface analog sensors via ADC/DAC with DMA. Implement low-power design with sleep/wake cycles.

🌐 Advanced (Weeks 9–11)

Build connected IoT firmware using BLE and Wi-Fi. Write and test bootloaders. Apply MISRA-C standards and use JTAG debugging and static analysis tools professionally.

✅ Job-Ready (Week 12)

Deliver a complete capstone system with documentation, live demo, and GitHub portfolio. Pass mock embedded technical interviews with confidence.

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

CAREER ALIGNMENT

Job Roles This Course Prepares You For

Embedded systems skills are in demand across automotive, medical, consumer electronics, industrial, and aerospace sectors. Here are the roles you'll be positioned for after completing this program.

Firmware Engineer

Write, test, and maintain embedded firmware on microcontrollers for consumer electronics, IoT, or industrial devices. Entry-level to mid-level. Most direct match to this course.

Embedded Software Engineer (Automotive)

Develop ECU software, AUTOSAR components, or ADAS firmware using MISRA-C and ISO 26262 frameworks. Strong growth in EV and ADAS sectors.

IoT Firmware Developer

Build connected device firmware for smart home, healthcare, or industrial platforms. Expertise in BLE, MQTT, and cloud integration is highly valued.

Embedded Systems Test Engineer

Validate firmware using HIL systems, unit testing frameworks, and automated regression suites. A critical role in safety-critical product lines.

BSP / Platform Engineer

Develop board support packages, device drivers, and HAL layers for new hardware platforms. Requires strong knowledge of memory maps, linker scripts, and RTOS internals.

Embedded Security Engineer

Identify and mitigate firmware vulnerabilities, implement secure boot, and apply cryptographic libraries. One of the fastest-growing niches in embedded.

EDVANTA

YOUR ROADMAP AFTER COMPLETION

What Comes Next: Your Post-Course Action Plan

Finishing the course is the beginning, not the end. Here is a clear 90-day post-course roadmap to turn your skills into a job offer or a promotion.

Pro Tip: Hiring managers in embedded systems pay close attention to GitHub activity and real hardware demos. A well-documented project with a short demo video is worth more than 10 certifications on your resume.

EDVANTA

Ignite your growth, inspire your mind

KEY TAKEAWAYS

What You Walk Away With

Industry-Grade Firmware Skills

Write production-quality embedded C code with clean structure, low memory footprint, and real-time responsiveness — skills interviewers actively test for.

Full-Stack Embedded Knowledge

From bare-metal register manipulation to RTOS, wireless connectivity, and secure OTA updates — you'll understand the full embedded software stack.

3 Portfolio-Ready Projects

Each phase ends with a mini project. Combined with the capstone, you'll have four documented, demonstrable projects to show during job interviews.

Job-Ready Positioning

Skills mapped directly to six high-demand embedded engineering roles across automotive, IoT, medical, and industrial sectors.

Mentorship & Community

Weekly live sessions with an industry expert ensure you're never stuck. Small-group learning creates a peer network that outlasts the course itself.

Real-World Context

Three detailed industry case studies (automotive, medical, industrial) show exactly how your skills map to real business problems and measurable outcomes.

EDVANTA

EDVANTA

Ignite your growth, inspire your mind

Ready to Build the Skills That Industry Is Hiring For?

Embedded systems engineering is one of the most durable and in-demand disciplines in tech — and this course is designed to get you there in 12 focused, practical weeks. Whether you're aiming for your first embedded role or transitioning from software, this program gives you the foundation, the portfolio, and the confidence to compete.

12 Weeks · 70+ Topics Covered

Structured phase-by-phase curriculum from fundamentals to production-grade advanced skills.

Live Mentoring Every Week

Get personalized guidance from an industry expert — not just pre-recorded videos. Your questions get answered in real time.

4 Portfolio Projects + Capstone Demo

Leave with proof of your skills: documented projects on GitHub, a live device demo, and interview-ready talking points.

Industry-Aligned from Day One

Tools, workflows, and standards used in this course are the same ones used in automotive, medical, and industrial embedded teams worldwide.

EDVANTA