SYSTEMS · TAUGHT WITH ARDUINO

Embedded / IoT

Microcontrollers, circuits, sensors and radios. You build a device on a breadboard, keep it responsive without blocking, and get its readings onto a network while running on a battery.

Foundational | 4 phases | 18 sessions | 27 contact hours

Syllabus

4 phases, 18 sessions. Each phase ends in something you have built.

1 Make Something Move

Get one LED, one button, and one sensor working, and understand the circuit behind each.

5 sessions
$72 alone

1.1 What a Microcontroller Actually Is

Pins, flash, RAM, and why this chip has no operating system to catch your mistakes.

FREE PREVIEW

1.2 Toolchain and First Upload

Install Arduino IDE 2, add the ESP32 board core, and get a blink onto real hardware or into Wokwi.

1.3 Volts, Amps, and Ohm's Law

Read a schematic, size a resistor, and use a multimeter before you power anything up.

1.4 Digital In and Out

GPIO modes, pull-up resistors, switch bounce, and why a floating input reads garbage.

1.5 Analog In, PWM Out

Read a potentiometer with the ADC, dim an LED, and use the serial monitor to see what the board thinks is happening.

By the end: **A breadboard circuit and sketch that reads a sensor and drives an output, handed in with the schematic you drew for it.**

2 Code That Does Not Block

Write firmware that tracks several things at once instead of stopping dead at every delay().

4 sessions
$72 alone

2.1 C for People Who Have Not Written C
Types, integer overflow, arrays, and functions, on a chip with kilobytes of RAM.

2.2 millis() Instead of delay()
Time things by checking elapsed milliseconds so the board can still read a button while an LED blinks.

2.3 State Machines
Describe a device as states and transitions, then write the loop that walks them.

2.4 Interrupts and Debouncing
Write a short ISR, mark shared variables volatile, and get a button that registers exactly one press.

By the end: **One device running three independent timed behaviours with no delay() anywhere in the sketch.**

3 Sensors, Buses, and Numbers You Trust

Talk to real sensor chips over a bus and turn their output into readings that mean something.

4 sessions
$72 alone

3.1 Reading a Datasheet
Find the pinout, the voltage range, and the one register you need inside a 300-page PDF.

3.2 I2C, SPI, and UART
Wire a bus, address a device, and scan an I2C line when nothing answers.

3.3 Libraries, Displays, and Distrusting Both
Drive an I2C OLED from a library, read that library's source, and handle the errors it quietly swallows.

3.4 Calibration, Noise, and Logging
Average out noise, apply an offset, convert to real units, and write readings as CSV with a timestamp.

By the end: **A data logger producing timestamped CSV from at least two calibrated sensors, plus a plot of a full day of readings.**

4 Get It Off the Desk

Put the device on a network and a battery, then keep it running when nobody is watching it.

5 sessions
$72 alone

4.1 Wi-Fi on the ESP32
Join a network, survive a dropped connection, and keep credentials out of the repository you push.

4.2 HTTP and JSON
Call a web API from the device, parse what comes back, and post your own readings to a server.

4.3 MQTT and a Dashboard
Brokers, topics, retained messages, and a Node-RED dashboard that moves when you breathe on the sensor.

4.4 Radios Beyond Wi-Fi
Bluetooth LE, 802.15.4, and where Thread and Matter actually sit in a 2026 smart home.

4.5 Batteries, Sleep, and Staying Up
Deep sleep, measured current draw rather than datasheet current draw, and watchdog plus reconnect logic for a device you cannot reach.

By the end: **A battery-powered node publishing sensor readings over MQTT to a dashboard, still reporting after a week unattended.**

Tools you will use

Arduino UNO R4 WiFi | ESP32-S3 / ESP32-C3 dev board | Arduino IDE 2.3 + Arduino CLI | arduino-esp32 core 3.x | C / C++ | Wokwi | Mosquitto (MQTT) | Node-RED

What you will build

1. **Plant Monitor**
   Soil, light, and a dashboard that updates live
2. **Battery Logger**
   One sensor, thirty days, one battery
3. **Reaction Timer**
   Interrupts, a display, and a high score that survives reset

Registration

Phase 1: Make Something Move	$72
Phase 2: Code That Does Not Block	$72
Phase 3: Sensors, Buses, and Numbers You Trust	$72
Phase 4: Get It Off the Desk	$72
Whole track, all 4 phases	**$240**

Buying every phase separately costs $288, so the whole track saves $48.

No payment is taken online. Registering creates an order and payment instructions are sent with it. Access opens once payment is confirmed.