

TechyTron STEM Electronics

Classes 11-12 · Ages 16–18

Ten guided electronics experiments. Typical school-age ranges vary by learner.

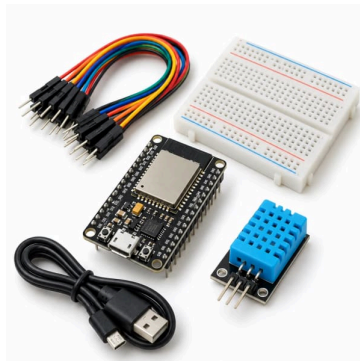
Each kit is supplied separately with the components listed in its guide.

- Local Wi-Fi Sensor Dashboard Kit
- Bluetooth LED Control Kit
- PID Light Control Lab Kit
- Solar I-V Explorer Kit
- Power and Energy Monitor Kit
- IoT Tank Level Monitor Kit
- MQTT Environment Station Kit
- Signal Sampling and Filtering Kit
- Vibration Spectrum Lab Kit
- Room Comfort Controller Kit

Local Wi-Fi Sensor Dashboard Kit

Classes 11-12 · Ages 16-18

Host a temperature and humidity dashboard on your local network.



Activity engagement: 90–150 minutes. Explore 15–30 min · Build 30–50 min · Test and discuss 45–70 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- A microcontroller can serve a web page to a browser on the same LAN.
- Data acquisition should run separately from request handling.
- Missing sensor data should remain visibly invalid.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- DHT11 — 1 sensor module · Supplied with DATA/VCC/GND labels and required data pull-up fitted.
- 10K Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- Bidirectional 3.3 V / 5 V Sensor Data Level Shifter — 1 level-shifter module

Connection guide

- DHT11 VCC → USB-powered board 5 V output, GND → GND. DATA has its supplied 10 kOhm pull-up to 5 V.
- Bidirectional data shifter HV → 5 V, LV → 3.3 V, common GND; DHT DATA → HV1, GPIO27 → LV1. Do not connect a 5 V data line directly to ESP32.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Connect the DHT11 through the supplied data level shifter.
3. Install the libraries and edit only the Wi-Fi settings in your local copy of the sketch.
4. Upload, open Serial Monitor and visit the printed local IP from a device on the same network.
5. Compare browser values with serial readings and unplug the sensor only with power off to test the unavailable-data state.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

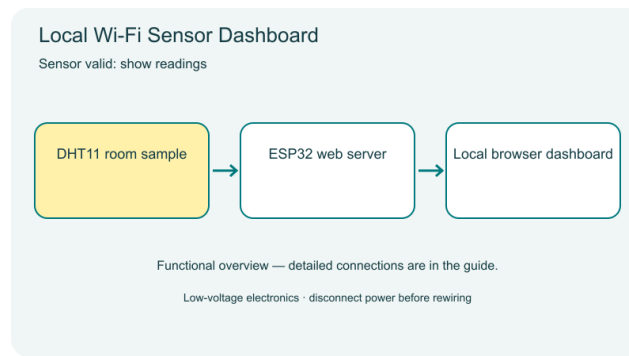
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Install from Arduino Library Manager: DHT sensor library (Adafruit), Adafruit Unified Sensor.

[Download starter sketch](#)

How this experiment works



Test challenge

Reload the page from two local devices while collecting a three-minute temperature/humidity record.

Observation task

Local sample time, browser value, serial value and any connection or sensor failure.

Expected result

The ESP32 serves current valid room readings locally; no cloud account or Internet publishing is required.

If it does not work

Use a 2.4 GHz Wi-Fi network and the correct local IP; guest-network client isolation may prevent access. Check the DHT11 level shifter.

Try next

Add JSON output with explicit units and a sample-age field.

Use and supervision

Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Bluetooth LED Control Kit

Classes 11-12 · Ages 16–18

Control an LED through a custom Bluetooth LE service.



Activity engagement: 90–150 minutes. Explore 15–30 min · Build 30–50 min · Test and discuss 45–70 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- BLE exposes data through services and characteristics.
- Received commands should be checked before changing an output.
- A classroom LED control service is not an authenticated lock or appliance controller.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- 5mm LED - Red Color — 1 LED
- 330 Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- Tactile Push Button Switch 12x12x8mm 4-Pin — 1 button

Connection guide

- GPIO25 → 330 Ohm → LED anode; cathode → GND.
- Local clear button → GPIO26 and GND, using INPUT_PULLUP.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Wire the LED and button and upload using the ESP32 core BLE library.
3. Use a BLE GATT browser app on a compatible phone; connect to TechyTron LED Lab.
4. Write UTF-8 text 1 or 0 to the characteristic shown in the guide; compare valid and invalid commands.
5. Press the local clear button and verify it switches the LED off without a phone.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

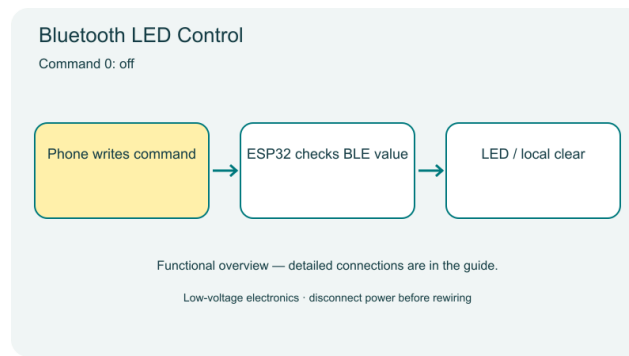
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Uses libraries supplied with the selected board core.

[Download starter sketch](#)

How this experiment works



Test challenge

Test 1, 0 and an unrelated character; reconnect and repeat.

Observation task

Command bytes, expected LED state, observed state and connection status.

Expected result

Only a leading 1 or 0 changes the LED state; the button clears it locally. No phone is included.

If it does not work

Use a BLE-capable app, select write-as-text rather than a decimal byte, and verify the correct service/characteristic.

Try next

Add notifications for confirmed state changes and discuss pairing/authentication.

Use and supervision

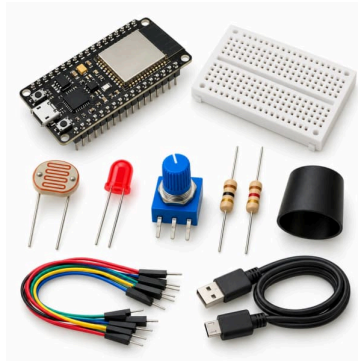
Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

PID Light Control Lab Kit

Classes 11-12 · Ages 16-18

Explore feedback control by regulating light at a sensor.



Activity engagement: 120-180 minutes. Explore 20-35 min · Build 40-60 min · Test and discuss 60-85 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- Feedback compares a measured value with a target.
- Proportional, integral and derivative terms affect response differently.
- Output and integral limits reduce saturation problems.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- LDR Sensor (Light Dependent Resistor) - 5mm — 1 photoresistor
- 10K Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- 5mm LED - Red Color — 1 LED
- 330 Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- 10 K POTENTIOMETER — 1 potentiometer
- Opaque Light-Control Shade and Sensor Mount — 1 set

Connection guide

- 3.3 V → LDR → GPIO34 node → 10 kOhm → GND.
- Pot outer pins → 3.3 V/GND; wiper → GPIO35. GPIO25 → 330 Ohm → LED anode; cathode → GND.
- Fix LED and LDR facing one another under the opaque shade; do not short their leads.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Mount the light path and wire both ADC1 inputs and PWM output.
3. Upload with Ki and Kd initially set to zero and observe response to a small target change.
4. Increase Ki modestly, then add a small Kd; compare settling time, overshoot and oscillation.
5. Temporarily shade the light path and observe recovery without changing the target.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

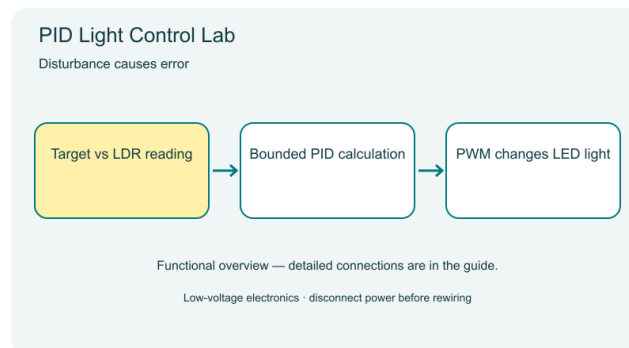
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Uses libraries supplied with the selected board core.

[Download starter sketch](#)

How this experiment works



Test challenge

Use the same target step for P-only, PI and PID trials; keep geometry and room light fixed.

Observation task

Time, target, measured ADC count, duty, gains, overshoot and settling behaviour.

Expected result

The LED duty adjusts to reduce light error within available brightness and sensor limits; tuning depends on the actual setup.

If it does not work

Verify divider orientation: more LED light must increase the reading. If feedback runs away, switch off and fix orientation before increasing gains.

Try next

Add integral anti-windup conditional on output saturation and compare with the bounded-integral starter.

Use and supervision

Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Solar I-V Explorer Kit

Classes 11-12 · Ages 16-18

Measure a small solar panel under different electrical loads.



Activity engagement: 120–180 minutes. Explore 20–35 min · Build 40–60 min · Test and discuss 60–85 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- Panel voltage changes with illumination and load.
- For a known resistive load, current is approximately V/R .
- Maximum measured power is a property of the tested conditions and available load points.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- 6V - 100 mA (70 mmx70 mm) Solar panel — 1 solar panel
- DT9205A Digital Multimeter with Probes and Battery — 1 multimeter · Includes probes and meter battery. Used only on isolated low-voltage circuits.
- 100 Ohm 1 W Through-Hole Power Resistor — 1 resistor
- 220 Ohm 1 W Through-Hole Power Resistor — 1 resistor
- 470 Ohm 1 W Through-Hole Power Resistor — 1 resistor
- 1000 Ohm 1 W Through-Hole Power Resistor — 1 resistor
- 10K Ω 1/4 Watts (10 PCS) — 4 resistor(s) · Quantity is individual resistors; catalogue pack is divided by 10.

Connection guide

- Panel positive → two 10 kOhm resistors in series → GPIO34 node → one 10 kOhm → panel negative. Panel negative → ESP32 GND. A fourth 10 kOhm is a supplied spare.
- Use the 100, 220, 470 and 1000 Ohm 1 W resistors one at a time directly across the panel; the divider remains in parallel.
- Meter in DC volts mode across the panel; COM and V sockets only. Do not use current mode across the panel.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Verify the 3:1 divider before connecting the solar panel and confirm its open-circuit voltage is no higher than 9 V.
3. Power ESP32 by USB and compare ADC-derived panel voltage against the meter; adjust dividerFactor if required.
4. Under fixed illumination, disconnect the panel before changing each load; enter its measured resistance in the local sketch.
5. Record load voltage and calculate load current/power; then repeat one load with partial shading.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

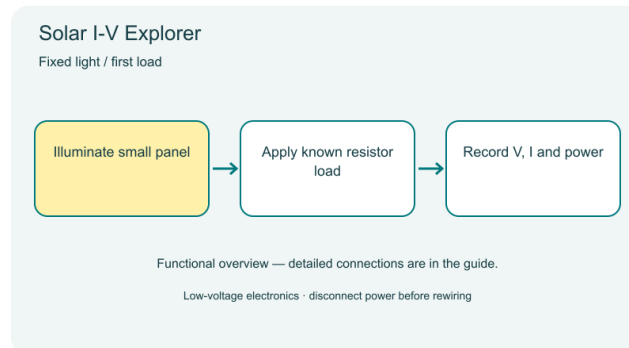
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Uses libraries supplied with the selected board core.

[Download starter sketch](#)

How this experiment works



Test challenge

Compare at least four load points under the same light, then repeat the highest measured power point in shade.

Observation task

Illumination condition, load resistance, meter voltage, ADC voltage, calculated load current and power.

Expected result

Each resistor gives an approximate operating point. $I=V/R$ refers to the selected load branch; divider current is a small additional branch.

If it does not work

Check divider orientation and stop if panel open-circuit voltage exceeds 9 V. The 1 W load resistors are required; do not substitute tiny 1/4 W parts.

Try next

Plot I against V and P against V; explain why four points do not locate the exact maximum-power point.

Use and supervision

Use only the supplied 6 V/100 mA panel, no batteries or charging circuits. Keep panel open-circuit voltage at or below 9 V for this divider. Switch off/disconnect before changing loads; 1 W resistors may become warm.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Power and Energy Monitor Kit

Classes 11-12 · Ages 16–18

Measure low-voltage load current, power and accumulated energy.



Activity engagement: 105–150 minutes. Explore 20–30 min · Build 35–50 min · Test and discuss 50–70 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- Current sensing is inserted in series with a load.
- Power and energy use different units.
- Energy is estimated by integrating power over elapsed time.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- CJMCU-219 INA219 I2C Interface No Drift Bi-directional Current / Power Monitoring Sensor Module — 1 sensor module
- 5mm LED - Red Color — 1 LED
- 330 Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- 5 V USB Power Lead with Screw-Terminal Output — 1 lead
- 5V 2A Adapter (Without Cable) - 10W Adapter for Raspberry Pi 3 — 1 power supply · Enclosed regulated 5 V adapter; adult connects the closed adapter, never exposed mains wiring.

Connection guide

- INA219 VCC → 3.3 V, GND → ESP32 GND, SDA → GPIO21, SCL → GPIO22.
- Supplied separate 5 V load supply positive → INA219 VIN+; VIN- → 330 Ohm → LED anode; LED cathode → load supply GND.
- Load-supply GND → ESP32 GND. Keep its +5 V separate from USB-powered board +5 V.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Wire the current path in series and check it with power disconnected.
3. Install libraries, upload and enable the separate 5 V load supply.
4. Observe volts, milliamps, milliwatts and accumulated milliwatt-hours for two minutes.
5. Disconnect only the load branch with power off, restart and compare the near-zero-load reading.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

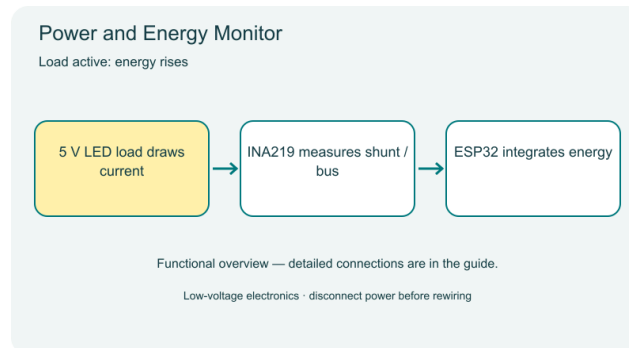
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Install from Arduino Library Manager: Adafruit INA219, Adafruit BusIO.

[Download starter sketch](#)

How this experiment works



Test challenge

Compare integrated energy with average measured power multiplied by the test duration in hours.

Observation task

Elapsed seconds, load voltage, current mA, power mW and energy mWh.

Expected result

Energy accumulates while a valid non-negative load power is measured and resets on board restart.

If it does not work

Check VIN+/VIN- orientation and common grounds. Do not wire the current path across a supply as a short.

Try next

Estimate uncertainty from current-sensor offset and sampling interval; this is not a billing meter.

Use and supervision

Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

IoT Tank Level Monitor Kit

Classes 11-12 · Ages 16–18

View a calibrated non-contact level model over local Wi-Fi.



Activity engagement: 90–150 minutes. Explore 15–30 min · Build 30–50 min · Test and discuss 45–70 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- A distance measurement can be mapped to a relative level.
- Empty and full reference distances depend on installation geometry.
- Invalid echoes must remain invalid rather than appearing full.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- HC-SR04 Ultrasonic Distance Sensor Module — 1 sensor module
- 1K Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- 1.8 k Ω 1/4 W Through-Hole Resistor — 1 resistor

Connection guide

- HC-SR04 VCC → board USB 5 V output; GND → GND; TRIG → GPIO26.
- ECHO → 1 k Ω → GPIO34 node → 1.8 k Ω → GND. The divider protects the ESP32 input from the 5 V echo.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Build and inspect the echo divider before connecting ECHO.
3. Point the dry sensor at a movable flat card; measure your empty and full reference distances with a ruler.
4. Enter reference distances and local Wi-Fi settings, upload and open the printed local IP.
5. Move the target between marked positions, then remove it to test the invalid-echo state.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

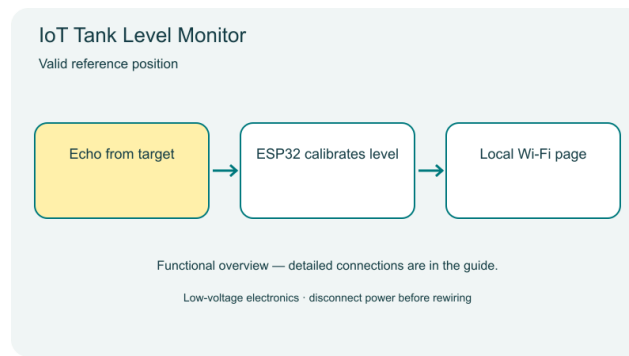
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Uses libraries supplied with the selected board core.

[Download starter sketch](#)

How this experiment works



Test challenge

Compare browser percentage at empty, halfway and full reference positions, with five trials each.

Observation task

Target distance, valid/invalid echo, relative percentage and browser result.

Expected result

The local page shows a calibrated relative level model; it is not a direct volume or waterproof installation measurement.

If it does not work

Verify the echo divider, common ground and dry flat target. HC-SR04 modules are not waterproof; keep them away from splashes.

Try next

Add sample-age and confidence fields before considering an actual tank installation.

Use and supervision

Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

MQTT Environment Station Kit

Classes 11-12 · Ages 16-18

Publish environmental readings to a local MQTT broker.



Activity engagement: 120–180 minutes. Explore 20–35 min · Build 40–60 min · Test and discuss 60–85 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- MQTT separates publishers from subscribers through topics.
- Sensor identity and valid data types matter.
- Reconnect logic and local credentials are part of a working networked system.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- Bme280 Embedded High Precision Atmospheric Pressure Height Sensor Module — 1 sensor module · 3.3 V I2C supply and pull-ups. BME280 includes humidity; BMP280 is not an equivalent substitute.

Connection guide

- BME280 VCC → 3.3 V; GND → GND; SDA → GPIO21; SCL → GPIO22. Use BME280, not a BMP280 substitute.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Connect the 3.3 V sensor and install the listed libraries.
3. On a computer/local classroom server, prepare an MQTT broker and subscriber on the same private LAN.
4. Enter local Wi-Fi, broker host and the classroom broker credentials in your own sketch copy.
5. Upload, subscribe to techytron/lab/environment, and compare received JSON with serial readings.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

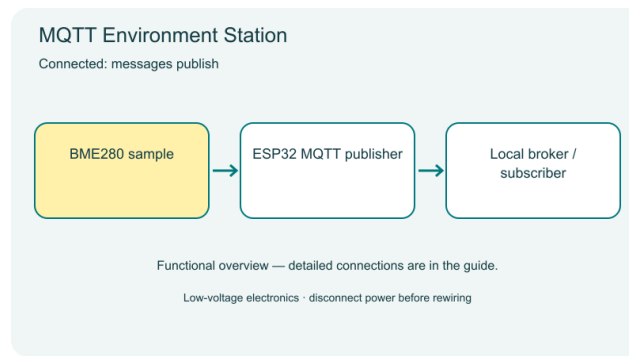
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Install from Arduino Library Manager: Adafruit BME280 Library, Adafruit Unified Sensor, Adafruit BusIO, PubSubClient.

[Download starter sketch](#)

How this experiment works



Test challenge

Disconnect and restore Wi-Fi once; check reconnection and the return of valid messages.

Observation task

Message time, topic, temperature, humidity, pressure and reconnect events.

Expected result

Valid BME280 measurements publish as JSON on the configured local topic; no broker or cloud subscription is supplied.

If it does not work

Check 0x76/0x77 address, 2.4 GHz Wi-Fi, broker access/credentials and firewall rules. Keep the teaching broker on the private LAN.

Try next

Add unique device IDs, retained-message policy and a sample timestamp; compare MQTT with direct HTTP.

Use and supervision

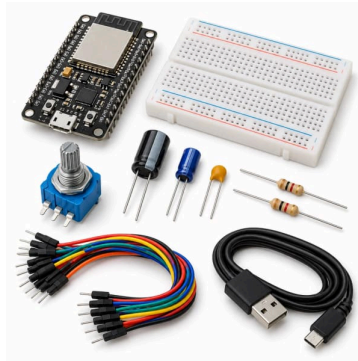
Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Signal Sampling and Filtering Kit

Classes 11-12 · Ages 16-18

Compare raw, RC-filtered and digitally averaged signals.



Activity engagement: 105–150 minutes. Explore 20–30 min · Build 35–50 min · Test and discuss 50–70 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- Analog and digital filters operate at different points in a signal path.
- Averaging reduces variation but delays a changing signal.
- Sampling rate and filter bandwidth must be considered together.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- 10 K POTENTIOMETER — 1 potentiometer
- 0.1 μ F CAPACITOR (5 PCS) — 3 capacitor(s) · Quantity is individual capacitors; catalogue pack is divided by 5.
- 1K Ω 1/4 Watts (10 PCS) — 2 resistor(s) · Quantity is individual resistors; catalogue pack is divided by 10.
- 10 μ F CAPACITOR (5 PCS) — 1 capacitor · Quantity is individual capacitors; catalogue pack is divided by 5.

Connection guide

- Pot outer pins $\rightarrow$ 3.3 V/GND; wiper $\rightarrow$ GPIO35 for the raw measurement.
- Wiper $\rightarrow$ two 1 kOhm resistors in series $\rightarrow$ GPIO34; capacitor from GPIO34 to GND. Start with 100 nF.
- Compare 100 nF with the supplied 10 μ F electrolytic (positive to GPIO34); two extra 100 nF capacitors can be tested in parallel.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Build both ADC1 inputs, checking that neither can exceed 3.3 V.
3. Upload and view raw, RC and 16-sample-average channels in Serial Plotter.
4. Turn the dial slowly, then make repeatable small step changes.
5. Power off before changing the capacitor; compare ripple and settling delay for the two values.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

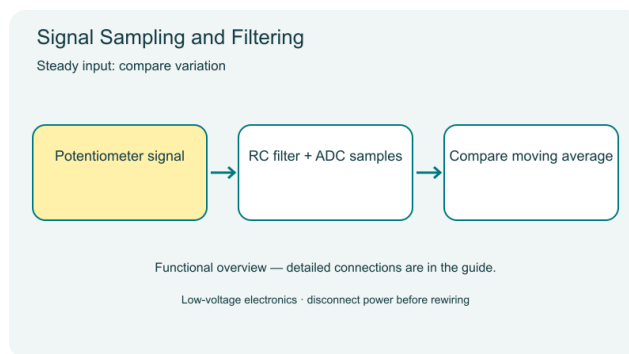
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Uses libraries supplied with the selected board core.

[Download starter sketch](#)

How this experiment works



Test challenge

Compare the same dial step with 100 nF and 10 uF; estimate filter delay without assuming the knob is an ideal signal generator.

Observation task

Capacitance, sample interval, raw/RC/average values, settling time and stable-signal variation.

Expected result

The RC and moving-average channels respond more smoothly; larger filtering causes more lag.

If it does not work

Check the shared ground and electrolytic polarity. Keep GPIO35 on the raw wiper side of the resistors.

Try next

For 2 kOhm and 10 uF, estimate $f_c = 1/(2\pi RC)$, about 8 Hz, and compare with the 100 Hz sampling rate.

Use and supervision

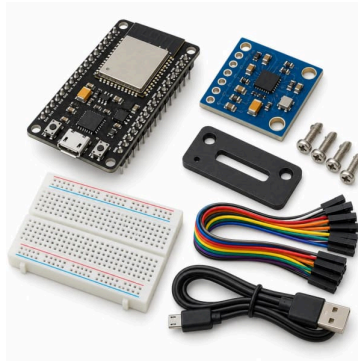
Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Vibration Spectrum Lab Kit

Classes 11-12 · Ages 16-18

Capture vibration data and inspect its frequency spectrum.



Activity engagement: 120–180 minutes. Explore 20–35 min · Build 40–60 min · Test and discuss 60–85 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- Sampling interval determines the usable frequency range.
- Removing the mean separates gravity/bias from changing acceleration.
- A spectrum describes the measured signal, not a certified machine-health diagnosis.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- MPU6050 3-Axis Gyroscope/Accelerometer Sensor Module — 1 sensor module · Supplied regulated GY-521 breakout: 5 V-compatible VCC and 3.3 V I2C, with fitted labelled header.
- Inertial Sensor Mounting Plate with Screws — 1 set

Connection guide

- Supplied regulated GY-521 VCC → board USB 5 V output; GND → GND; SDA → GPIO21; SCL → GPIO22. I2C pull-ups on this breakout are 3.3 V.
- Attach the board to the supplied flat mounting plate with insulated spacers; do not short contacts against metal.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Mount the sensor rigidly and upload the 100 Hz serial capture sketch.
3. Save at least 256 consecutive CSV samples for a stationary trial and a gentle periodic hand-motion trial.
4. Run the included standard-library Python analysis script on each saved CSV file.
5. Compare time spacing, estimated sample rate and the main non-DC frequency bin; repeat with a different hand-motion rate.
6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

Programming resources

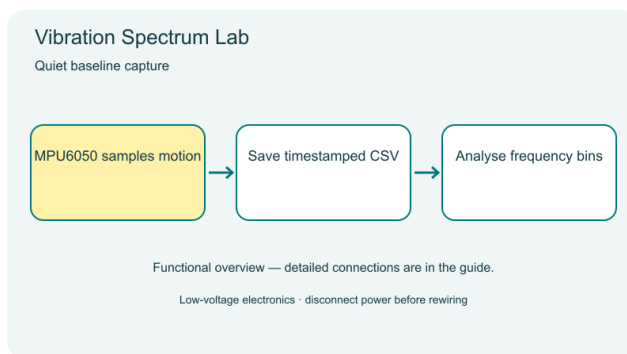
Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Install from Arduino Library Manager: Adafruit MPU6050, Adafruit Unified Sensor, Adafruit BusIO.

[Download starter sketch](#) · [Download spectrum analysis](#)

Save CSV with the header ms,ax,ay,az. Run `python vibration-spectrum-lab-analysis.py capture.csv > spectrum.csv` on your computer. This analysis uses the Python standard library.

How this experiment works



Test challenge

Compare a quiet baseline and two gentle periodic motions, avoiding impacts that saturate the ± 2 g range.

Observation task

Capture label, timestamp spacing, sample rate, dominant frequency and spectral magnitude.

Expected result

A finite-window spectrum can reveal repeated motion; resolution is sample rate divided by sample count.

If it does not work

Export only CSV rows with the header ms,ax,ay,az. Large timing gaps or loose mounts change the result; there is no forced-vibration actuator in this kit.

Try next

Compare a rectangular window with the supplied Hann window and discuss leakage.

Use and supervision

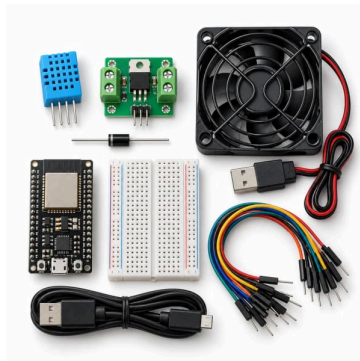
Use only the supplied low-voltage DC parts. Disconnect power before rewiring; check polarity and labelled pinouts. No soldering or exposed mains connections. Adult guidance is recommended.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.

Room Comfort Controller Kit

Classes 11-12 · Ages 16-18

Control an enclosed low-voltage fan with temperature hysteresis.



Activity engagement: 120–180 minutes. Explore 20–35 min · Build 40–60 min · Test and discuss 60–85 min. Times are estimates for guided use.

You will need: A computer with a USB port and Arduino IDE is required and is not included. Wi-Fi/BLE experiments also require the local network or phone specified in the guide.

What you learn

- A GPIO controls a driver; it does not power a motor.
- Separate on/off thresholds reduce rapid switching.
- Invalid sensor data should lead to a defined fail-safe output.

Included components

- Small Breadboard / Mini Solderless Board - 400 points — 1 breadboard
- Jumper Wire Combo - Male to Male, Male to Female and Female to Female Breadboard Connecting Wires (10 Qty Each) — 1 set
- Illustrated Activity Guide and Observation Sheets — 1 set
- Fitted Headers and Labelled Connection Leads — 1 set
- ESP32S 38Pin Development Board (WIFI+BLUETOOTH) — 1 controller board · Classic ESP32/WROOM development board, 3.3 V GPIO; USB powers its on-board regulator.
- MICRO USB DATA CABLE — 1 data cable
- DHT11 — 1 sensor module · Supplied with DATA/VCC/GND labels and required data pull-up fitted.
- 3.3 V Logic-Level MOSFET Fan Driver Module — 1 driver module
- 1N4007 DIODE (5 PCS) — 1 diode
- Enclosed 5 V Cooling Fan with Safety Grilles — 1 fan · Catalogue 5 V fan + INR 25 protective grille/enclosure allowance.
- Enclosed Regulated 5 V Fan Supply with Fitted Lead — 1 power supply
- 10K Ω 1/4 Watts (10 PCS) — 1 resistor · Quantity is individual resistors; catalogue pack is divided by 10.
- Bidirectional 3.3 V / 5 V Sensor Data Level Shifter — 1 level-shifter module

Connection guide

- DHT11 VCC → USB-powered board 5 V output, GND → GND. DATA has its supplied 10 kOhm pull-up to 5 V.
- Bidirectional data shifter HV → 5 V, LV → 3.3 V, common GND; DHT DATA → HV1, GPIO27 → LV1. Do not connect a 5 V data line directly to ESP32.
- GPIO25 → labelled 3.3 V-compatible driver IN; driver signal GND → ESP32 GND and external supply GND.
- Fan + → separate 5 V supply +; fan - → driver switched OUT-; follow the driver power-terminal labels. Flyback diode across fan: banded cathode to +5 V, anode to fan -.
- Keep the external +5 V separate from the USB-powered ESP32 +5 V. The fan remains inside both safety grilles.

Steps to make it

1. Disconnect all power and identify the supplied part labels and pinouts before wiring.
2. Inspect the enclosed fan, diode orientation and driver labels before enabling power.
3. Upload and record the baseline room temperature. Choose on/off thresholds two degrees apart around the intended test range.
4. Change the sensor temperature gently using hand warmth; observe the fan across both thresholds.
5. With power off, unplug sensor DATA; restart to check the defined fan-OFF error behaviour.

6. Change only the named parameter for the extension, repeat the controlled trials, save observations and disconnect power afterward.

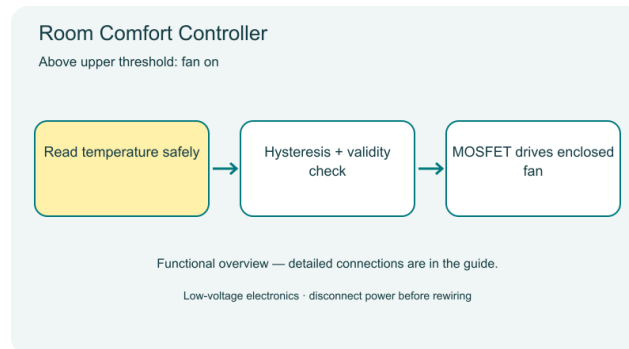
Programming resources

Arduino IDE: install esp32 by Espressif Systems 3.x; select ESP32 Dev Module for the supplied classic WROOM board. Serial Monitor: 115200 baud. GPIO is 3.3 V only.

Install from Arduino Library Manager: DHT sensor library (Adafruit), Adafruit Unified Sensor.

[Download starter sketch](#)

How this experiment works



Test challenge

Log temperature and fan state through a warming/cooling cycle; compare with a single-threshold decision.

Observation task

Time, temperature, on/off thresholds, fan state and invalid-sensor events.

Expected result

Valid temperature above the upper threshold turns the fan on; below the lower threshold turns it off. Invalid data turns it off and reports an error.

If it does not work

Check the external fan supply, common ground, active-high driver input and DHT data level shifter. Never connect a fan directly to a GPIO.

Try next

Add manual override with a timeout and distinguish a classroom control model from a safety-critical controller.

Use and supervision

Use only the supplied enclosed 5 V fan and driver. Keep both grilles fitted, leave the fan unloaded, and disconnect power before wiring. No mains relays, heaters or exposed blades.

Cover images illustrate component types. Flow animations explain principles and are not recorded prototype tests. Use the exact connection labels supplied with the kit.