⚡ BlitzTech Learn

Getting Started with Arduino UNO

Your first 30 minutes with an Arduino UNO, install the IDE, wire an LED, upload your first sketch. Zero electronics experience required.

Beginner | 30 min | Arduino | Jun 26, 2026 | blitztechelectronics.co.zw/learn/getting-started-with-arduino-uno/

The Arduino UNO is the single best entry point into electronics for makers, students, and engineers in Zimbabwe. This guide walks you from unboxing to a blinking LED in roughly half an hour. Everything here works identically whether you bought from BlitzTech or anywhere else, but we're the only Zimbabwean shop that stocks the UNO with same-day Harare delivery and a 24-hour replacement policy if your board is dead-on-arrival.

1 What you'll need

- Arduino UNO R3 (BlitzTech stock code: ARD-UNO-R3)
- USB cable, type A to type B (printer cable)
- A computer running Windows, macOS, or Linux
- 1× LED (any colour) and 1× 220 Ω resistor, or skip both and use the on-board LED on pin 13
- (Optional) A breadboard and 2 jumper wires

2 Install the Arduino IDE

Download Arduino IDE 2.x from arduino.cc/en/software. The IDE is free and runs on Windows, macOS, and Linux. During install, accept any driver prompts, these are the FTDI/CH340 USB-to-serial drivers Arduino needs to talk to the board over USB.

3 Connect the board

Plug your Arduino into your computer with the USB cable. Power LED on the board should light up solid green or orange. In the Arduino IDE go to Tools → Board → Arduino UNO. Then Tools → Port and pick the COMx (Windows) or /dev/tty.usbmodem... (macOS/Linux) entry that appeared.

4 Upload the Blink sketch

In the Arduino IDE go to File → Examples → 01.Basics → Blink. A small program loads. Click the upload button (right-arrow icon at the top). After 5-10 seconds the on-board LED next to pin 13 will start blinking once per second. That's your first program running on real silicon you control.

```
// File: Blink.ino
void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_BUILTIN, HIGH);  // turn LED on
  delay(1000);                       // wait 1 second
  digitalWrite(LED_BUILTIN, LOW);   // turn LED off
  delay(1000);                       // wait 1 second
}
```

5 Where to go next

- Modify the delay() values, change 1000 to 100 for a fast blink, 5000 for a slow one.

- Wire an external LED to pin 13 via the 220 Ω resistor, exact same code, brighter blink.

- Try File → Examples → 01.Basics → Fade for a gentle PWM brightness ramp.

- Read the BlitzTech guide 'Reading a DHT22 Sensor with Arduino' for your first input device.