

Supplementary Material to “Modelling a battery with an accessible and inclusive Arduino-based experimental setup”

Supplementary Material 1 – Arduino Code

```
int botaoU = 2;    // Digital input pin for voltage (U) button
int botaoI = 3;    // Digital input pin for current (I) button

int ddp1, ddp2;    // Raw ADC readings from analog pins A1 and A2
int soundU, soundI; // Frequencies to send to the buzzer for U and I
int buzzerI = 11;  // Buzzer output pin

double VA1;        // Measured voltage at A1 (in volts)
double VA2;        // Measured voltage at A2 (in volts)
double current;     // Calculated current through resistor r_2
double r_2 = 5000.00; // Known resistor value used to compute current (ohms)

void setup() {
  Serial.begin(9600); // Initialize serial communication at 9600 bps
  pinMode(botaoI, INPUT); // Configure current button pin as input
  pinMode(botaoU, INPUT); // Configure voltage button pin as input
}

void loop() {
  // Read analog inputs (0–1023)
  ddp1 = analogRead(A1); // Raw reading corresponding to voltage U
  ddp2 = analogRead(A2); // Raw reading corresponding to voltage across R2

  // Only run measurement and buzzer logic if at least one button is pressed
  if (digitalRead(botaoI) == HIGH || digitalRead(botaoU) == HIGH) {

    VA1 = (5.0 / 1023.0) * ddp1; // Convert ADC reading to voltage at A1
    VA2 = (5.0 / 1023.0) * ddp2; // Convert ADC reading to voltage at A2

    // Compute current using Ohm's Law:  $I = V / R$  (converted to mA)
    current = (VA2 / r_2) * 1000.0;

    // Print voltage and current to Serial Monitor
    Serial.print(" U(V): ");
    Serial.print(VA1, 4);
    Serial.print(" I(mA): ");
    Serial.print(current, 4);
    Serial.print(" | ");
```

```

// Convert the measured values into sound frequencies
soundU = int(500 * VA1);
soundI = int(500 * current);

// If only the voltage button is pressed: play voltage frequency
if (digitalRead(botaoU) == HIGH && digitalRead(botaoI) == LOW) {
  tone(buzzerI, soundU);
}

// If only the current button is pressed: play current frequency
if (digitalRead(botaoU) == LOW && digitalRead(botaoI) == HIGH) {
  tone(buzzerI, soundI);
}

// Print the frequency values used for debugging
Serial.print("Freq U: ");
Serial.print(soundU);
Serial.print(" | Freq I: ");
Serial.println(soundI);
}

// Stop any tone if no button combination is actively producing sound
noTone(buzzerI);
}

```