# Noise Sensor Test – Three-Sensor Comparison (Open, Small Hole, Large Hole)

#### **Preparation Work:**

* **Understand Speaker Orientation:**
  + Confirm the orientation of the Bluetooth speaker to ensure optimal sound reception.
  + Test different angles and distances to verify that the sound reaches all three noise sensors effectively.
  + Perform preliminary sound checks to confirm that the speaker produces consistent sound levels (**20 dB to 90 dB**) at the intended setup.
  + Adjust the speaker position if necessary to optimize sound delivery to all three sensors.

#### **Setup:**

* Use **three noise sensors** for the test (**Open, Small Hole (Microphone Size), Large Hole (1-Inch Diameter)**).

#### **Test Duration:**

* Conduct noise logging for a duration of **2 hours**.

#### **Sound Source:**

* Use a Bluetooth speaker connected to an app to generate sounds ranging from **20 dB to 90 dB** during the test.

#### **Data Collection:**

* Log noise data from all **three sensors** throughout the **2-hour duration**.

#### **Time Offset Compensation:**

* Adjust for any time offsets between the **three** sensors to ensure proper synchronization.

#### **Data Analysis:**

* Use the **previously developed R script** to analyze the data.
* Calculate the following parameters to compare the sensors:
  + **R²**
  + **Slope**
  + **Intercept**
  + **Correlation coefficient**
* Perform **pairwise comparisons** to evaluate consistency between each pair of sensors.

#### **Outcome:**

* Assess whether the **three sensors** provide **comparable** measurements by analyzing the calculated values.

**Data Collection:**

* One **timestamp** corresponds to **three sensor readings** (one per sound sensor), logged together in a single row.  
  **Format:**  
  timestamp, sensor1\_value, sensor2\_value, sensor3\_value
* The **loop runs continuously**, collecting and writing data in **batches of 10 readings** per cycle.
* Designed for **real-time logging** of 3 sound sensor values to the SD card with **minimal delay** between readings and writes.
* **Buffering** is implemented using arrays:
  + sensorTimestamps[10]
  + sensor1Readings[10]
  + sensor2Readings[10]
  + sensor3Readings[10]  
    These arrays temporarily store 10 readings before writing them all at once to the SD card.

**Data Analysis:**

DATA ANALYSIS SUMMARY – SOUND SENSOR CORRELATION

DATA INPUT AND PREPROCESSING

- Data is read from a file named 'LOG.TXT'

- File format: comma-separated

- Columns: Timestamp, Sensor1, Sensor2, Sensor3 (raw ADC values)

- Reference voltage (VREF) is set to 6.14 V

**CONVERSIONS PERFORMED**

1. ADC to Voltage Conversion:

Formula: (ADC Value / 32768.0) \* VREF

Applied to:

- Sensor1\_Voltage

- Sensor2\_Voltage

- Sensor3\_Voltage

2. Voltage to dB Conversion:

Formula: Voltage \* 50

Applied to:

- Sensor1\_dB

- Sensor2\_dB

- Sensor3\_dB

**ANALYSIS PERFORMED**

- Linear regression is computed for sensor pairs

- Model: y = slope \* x + intercept

- Extracted statistics:

- Slope

- Intercept

- R-squared

- Correlation Coefficient (square root of R-squared)

**PLOTS GENERATED**

Each of the following sensor pairs is analyzed in 3 formats:

1. **dB vs. dB**

- Sensor1\_dB vs Sensor2\_dB

- Sensor1\_dB vs Sensor3\_dB

- Sensor2\_dB vs Sensor3\_dB

2. **Voltage vs. Voltage**

- Sensor1\_Voltage vs Sensor2\_Voltage

- Sensor1\_Voltage vs Sensor3\_Voltage

- Sensor2\_Voltage vs Sensor3\_Voltage

3. **Raw ADC vs. Raw ADC**

- Sensor1 vs Sensor2

- Sensor1 vs Sensor3

- Sensor2 vs Sensor3