Optimizing Cage Change Schedules Through Waste-Associate Gas Measurements in Mouse Housing
Methodology and Supplemental Information
Bioscience Support Facility – University of Toronto

Section 1: Materials and Methods

Animals and Caging

Male and female adult C57BL6 mice (Source: Jackson Laboratory) were housed in individually ventilated cages (IVCs) within a DGM14 IVC rack (Tecniplast, 140 cage capacity). Each cage consisted of a GM500SUED polysulfone cage body with autowatering grommet and a GM500TOPSEALSU Sealsafe Plus flat top equipped with a 0.2 µm microbiological filter and retainer. Cages measured 391 mm (W) × 199 mm (D) × 160 mm (H) for an approximate volume of 12.4 Liters.

The HEPA-certified IVC rack system was maintained with a BOX100WF blower unit (100/110V, 50/60 Hz), maintaining 70 air changes per hour (ACH) of HEPA-filtered air.

Adult mice (10-48 weeks old) of both sexes were selected to produce a more diverse and paradigmatic representation of typical vivarium operations.

Animal Husbandry

Mice were group-housed by sex (i.e.. Cages had either only male mice or only female mice) in cages of 2-4 and provided *ad libitum* LabDiet 5001 – Laboratory Rodent Diet and reverse osmosis water. Food was provided both on the bottom (floor) of the cage and in the food hopper of the wire bar lid for ease of access. Corncob bedding (Approximately 1 Liter of autoclaved ¼" corncob bedding) and 8 grams of crinkle paper was placed in each cage.

A 12:12-hour light:dark cycle with the light-phase beginning at 07:00 local time was used to ensure that air quality sampling was performed in the light-phase.

The room housing the animals was set at a minimum of 15 ACH, with temperature set at 22°C, and Relative Humidity set at 40-60% RH. Environmental conditions were verified using the Bioscience Support Facility's Environmental Monitoring System (EMS)

Building the Calibration Curve

To build the Calibration Curve between the 10 static, variously-soiled mouse cages were measured via the “ammonia” channel of a calibrated MiniRAE 3000 Photoionization Detector (PID) (100ppm isobutylene) (Correction Factor to ammonia: 10.9).

The MiniRAE 3000 PID was turned on and given 10 minutes to stabilize in a room with clean air, and a two-point calibration (zero/span calibration in clean air followed by span calibration with 100ppm isobutylene). Once calibrated, the MiniRAE 3000 PID’s probe was inserted into a cage via the water grommet, given 1 minute to stabilize, and a recording was taken. The probe was then removed and given 1 minute to stabilize in clean air before being reinserted into the cage for a total of 5 measurement points.

After 5 measurements were performed using the MiniRAE PID, MQ-135 readings were taken. The MQ-135 was turned on, provided 10 minutes to stabilize in a room with clean air, and inserted into a cage via the water grommet. The unit was given 1 minute to stabilize, and 30 recordings (one per second) were taken. The probe was then removed and given 1 minute to stabilize in clean air before being reinserted into the cage for a total of 5 measurement points.

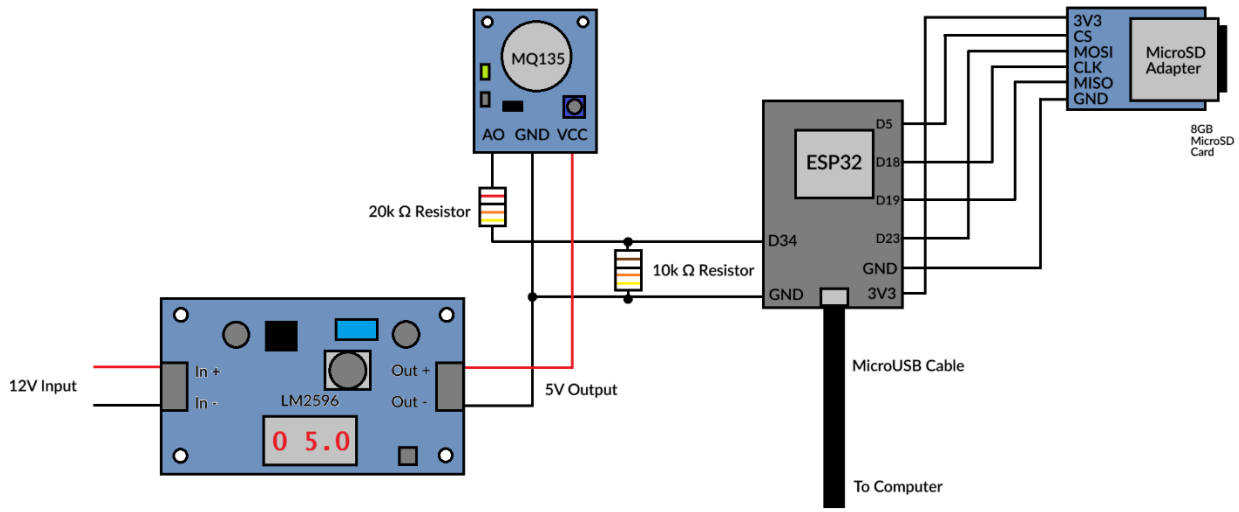
Live-Cage Sampling Design

Measurements were taken using an MQ-135 Metal Oxide Semiconductor (MOS) sensor connected to an ESP32 microcontroller board, with data stored to a 32GB MicroSD card. The MQ-135 and connected ESP32 were turned on, provided 10 minutes to stabilize in a room with clean air, and inserted into a mouse cage via the water grommet. The unit was given 1 minute to stabilize, and 30 recordings (one per second) were then taken. The probe was then removed and given 1 minute to stabilize in clean air before being reinserted into the cage for a total of 5 measurement points (150 recordings).

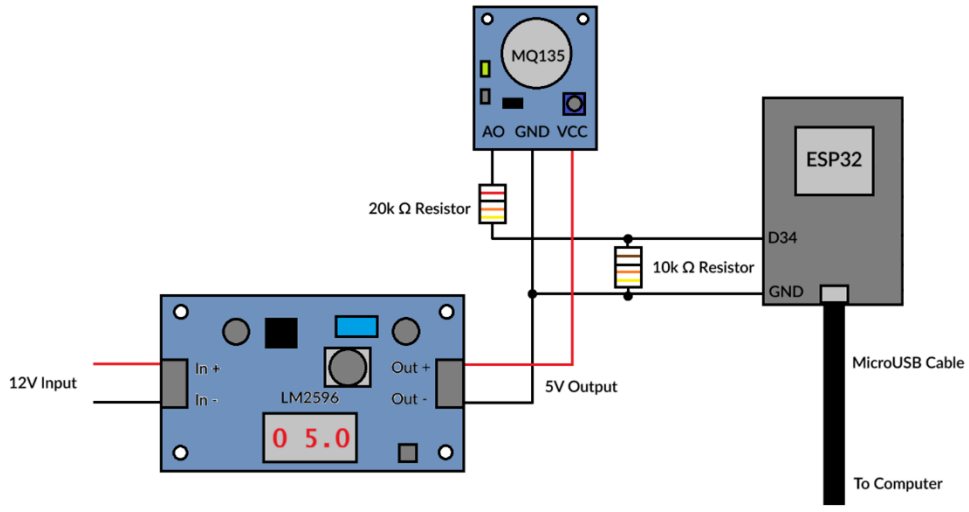
MQ-135 readings were taken at 11:00 AM on the day of sampling. Instead of following the same cage over time, we sampled different cages between 3–7 days post–cage-change to ensure broader representation of the facility. The same cages were never measured more than once. Cages contained 2, 3, or 4 mice, and each time point reflected a different cage randomly selected from the rack.

Section 2: Wiring Diagram of the MQ-135 and ESP32

MQ-135 and ESP32 With Onboard MicroSD Card (Data Recorded to MicroSD)



MQ-135 and ESP32 Without Onboard MicroSD Card (Data Recorded to Computer)



Section 3: MQ-135 and ESP32 Code

Code for the MQ-135 and ESP32 With Onboard MicroSD Card

```
#include <SPI.h>
#include <SD.h>

#define MQ135_AO 34
#define SD_CS 5

unsigned long startTime;

void setup() {
  Serial.begin(115200);

  if (!SD.begin(SD_CS)) {
    Serial.println("SD Card initialization failed!");
    return;
  }
  Serial.println("SD Card initialized.");

  startTime = millis();

  File logFile = SD.open("/baseline_log.csv", FILE_READ);
  if (logFile) {
    if (logFile.size() == 0) {
      logFile.close();
      File writeFile = SD.open("/baseline_log.csv", FILE_WRITE);
      if (writeFile) {
        writeFile.println("Time (hh:mm:ss),Analog Value");
        writeFile.close();
      }
    } else {
      logFile.close();
    }
  }
}

void loop() {
  int sensorValue = analogRead(MQ135_AO);

  unsigned long elapsedTime = millis() - startTime;
  int hours = (elapsedTime / 3600000) % 24;
  int minutes = (elapsedTime / 60000) % 60;
  int seconds = (elapsedTime / 1000) % 60;

  char timestamp[12];
  sprintf(timestamp, "%02d:%02d:%02d", hours, minutes, seconds);

  Serial.print("Time: ");
  Serial.print(timestamp);
  Serial.print(" | Analog Value: ");
  Serial.println(sensorValue);

  File logFile = SD.open("/baseline_log.csv", FILE_WRITE);
  if (logFile) {
    logFile.seek(logFile.size());
    logFile.print(timestamp);
    logFile.print(",");
    logFile.println(sensorValue);
    logFile.close();
  } else {
    Serial.println("Error opening CSV file!");
```

```
  }

  delay(1000);
}
```

Code for the MQ-135 and ESP32 Without Onboard MicroSD Card

```
#define MQ135_AO 34

unsigned long startTime;

void setup() {
  Serial.begin(115200);
  startTime = millis();
}

void loop() {
  int sensorValue = analogRead(MQ135_AO);

  unsigned long elapsedTime = millis() - startTime;
  int hours = (elapsedTime / 3600000) % 24;
  int minutes = (elapsedTime / 60000) % 60;
  int seconds = (elapsedTime / 1000) % 60;

  char timestamp[12];
  sprintf(timestamp, "%02d:%02d:%02d", hours, minutes, seconds);

  Serial.print(timestamp);
  Serial.print(",");
  Serial.println(sensorValue);

  delay(1000);
}
```