Team Turn Up

December 4, 2018
Team Members

Nicholas Kafasis
CSE

Harold Healy
EE

Ryan Walsh
CSE

Rahaun Perkins
CSE
Problem Statement

- In environments with dynamic noise levels, frequent volume adjustments for Speakers and TVs are a nuisance. Our system will be an intermediary device that regulates the volume of such audio devices based on levels of ambient noise in a room.
Motivations

- Automation is in high demand
  - Temperature
  - Brightness
- Speakers can be found almost anywhere
  - TVs
  - Computers
  - Cars
System Requirements

- Easy to use phone app will allow user to set fixed level of audibility above/below ambient noise.
- System will not exceed max volume setting
- System will not react suddenly to isolated loud noises
- System will function in multiple locations within desired room
System Specifications

- Maintain signal + noise-to-signal ratio within threshold set by user (range of 1.1-2)
- Reaction time for maintaining ratio is based on sensitivity set by user
  - Moving average weights
    - Low sensitivity: Signal Intensity = \[ (.001)a + (.999)b \]
    - High sensitivity: Signal Intensity = \[ (.01)a + (.99)b \]
    - Where \( a \) is the intensity of the newest sample set, and \( b \) is the current value of the moving average.
  - Sample interval of power calculations at 512 samples
- System will work within a 20 ft. radius of microphone
Previous Block Diagram
Current Block Diagram
Proposed CDR Block Diagram
MDR Deliverables

**Proposed:** No App, Variables will be set manually in code.

**Delivered:**
Sensitivity:
Weighted Average Constants:
- Average Signal Intensity = \[(.01)a + (.99)b\]
  - \(a\) = avg. of most recent window
  - \(b\) = avg. calculated so far
- Calculation Window: 512 samples @44100Hz~0.01s

Ratio Threshold: set to 1.5 for MDR Demo purposes
MDR Deliverables

**Proposed**: Single speaker system

**Delivered**: Oontz Angle 3

- 10W Peak Output Power
- Lightweight and Portable
- D/A conversion directly on Pi
MDR Deliverables

**Proposed**: Wired connections

**Delivered**: Microphone connected directly to Raspberry Pi (MXL-AC404 Conference Microphone)

- 25 ft. range
- 180° range
MDR Deliverables

**Proposed**: Proof of captured signals and computed calibration graph, additional information graphs

**Delivered**:

- Calibration graph
- Mic intensity & expected mic intensity over time
- Moving averages of mic intensity & expected mic intensity over time
- Ratio over time
- Scale factor over time
Calibrate.py

Main calibration procedure:
1. Play predetermined calibrate sound
2. Record microphone signal as the sound plays
3. Calculate RMS over the calibrate sound and mic pickup signal
   - RMS taken over 512 samples at a time
4. Obtain model of expected mic pickup intensity as a function of input signal intensity
   - Linear relationship

\[
RMS = \sqrt{\frac{x_1^2 + x_2^2 + \ldots + x_N^2}{N}}
\]
Main listen procedure:
1. Obtain calibration graph
   - Either run calibration function or load previous calibration graph
2. Play live input signal and record sound from mic
   - Live input signal constantly multiplied by scale factor with default value 1
3. Continuously compute RMS values of input signal and mic signal
   - Compute expected mic intensity with input signal intensity & calibration model
4. Continuously compute the moving averages of mic intensity and expected mic intensity

5. Continuously compute ratio of mic intensity average to expected mic intensity average

6. Follow this condition:
   - If: ratio rises above threshold → increase scale factor
   - Otherwise:
     - If: scale is larger than 1 → decrease scale factor
     - Otherwise: keep scale factor at 1
Mic Pickup Intensity vs. Input Signal Intensity

- Intensity Data Points
- Best Fit Line
- $y = 0.1983x + 86.7288$

Relative Fraction of Maximum Intensity

Relative Fraction of Maximum Intensity
MDR Deliverables: Data Plots

Moving Averages of Mic Intensity & Expected Mic Intensity over Time

- Mic Intensity Moving Average
- Expected Mic Intensity Moving Average
MDR Deliverables: Data Plots

Scale Factor and Avg. Mic Intensity to Avg. Expected Mic Intensity Ratio over Time
## Project Breakdown and Responsibilities

<table>
<thead>
<tr>
<th>Task name</th>
<th>Start date</th>
<th>End date</th>
<th>Assigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>TurnUp SDP</td>
<td>10/22/2018</td>
<td>04/07/2019</td>
<td></td>
</tr>
<tr>
<td>Development</td>
<td>10/22/2018</td>
<td>04/07/2019</td>
<td></td>
</tr>
<tr>
<td>Core Software Design</td>
<td>10/22/2018</td>
<td>03/13/2019</td>
<td></td>
</tr>
<tr>
<td>Calibration</td>
<td>10/22/2018</td>
<td>11/11/2019</td>
<td></td>
</tr>
<tr>
<td>Real Time Audio Monitoring</td>
<td>11/12/2018</td>
<td>03/13/2019</td>
<td></td>
</tr>
<tr>
<td>Adjustment Algorithms</td>
<td>11/12/2018</td>
<td>03/13/2019</td>
<td></td>
</tr>
<tr>
<td>Website Development</td>
<td>12/01/2018</td>
<td>12/19/2018</td>
<td>Nick Kafasis</td>
</tr>
<tr>
<td>Wireless Mic Integration</td>
<td>12/01/2018</td>
<td>12/29/2018</td>
<td>Rahaun Perkins</td>
</tr>
<tr>
<td>PCB Design</td>
<td>12/10/2018</td>
<td>01/31/2019</td>
<td>Harold Healy</td>
</tr>
<tr>
<td>PCB Integration</td>
<td>02/01/2019</td>
<td>03/10/2019</td>
<td>Harold Healy</td>
</tr>
<tr>
<td>App Development</td>
<td>02/01/2019</td>
<td>03/29/2019</td>
<td>rtwalsh</td>
</tr>
<tr>
<td>Refinement</td>
<td>02/01/2019</td>
<td>04/07/2019</td>
<td>Rahaun Perkins</td>
</tr>
<tr>
<td>Enclosure Development</td>
<td>03/11/2019</td>
<td>04/07/2019</td>
<td>Harold Healy</td>
</tr>
</tbody>
</table>
Gantt Chart
CDR Deliverables

- Basic IOS App that will allow user to choose sensitivity and threshold based with sliders (Ryan)
- System will have the full range of sensitivity (1-10) based on weights of weighted average and size of calculation window (Ryan/Nick)
- System will have full range of thresholds (Expected range of (1.1 - 2) with actual values TBD) (Ryan/Rahaun)
Microphone will communicate wirelessly with Pi (Rahaun)

Analog Amplification with Digitally Programmable Gain (Harry)
  • (Instead of Digital Scaling of Samples)
Design Validation

- Graphically show that system increases gain after X seconds of Mic intensity above threshold R
- Consumer satisfaction survey
Challenges Moving Forward

- Integrating the wireless microphone
  ○ Making sure data is transferred fast enough

- Determining reasonable user preferences through research and polling

- Delivering a clear and concise user interface
Demo
Thank you

Questions?