UMassAmherst
The Commonwealth's Flagship Campus

Mid-Year Design Review

Team 22 - Coresidium

Advisor: Professor Siqueira

December 7, 2018

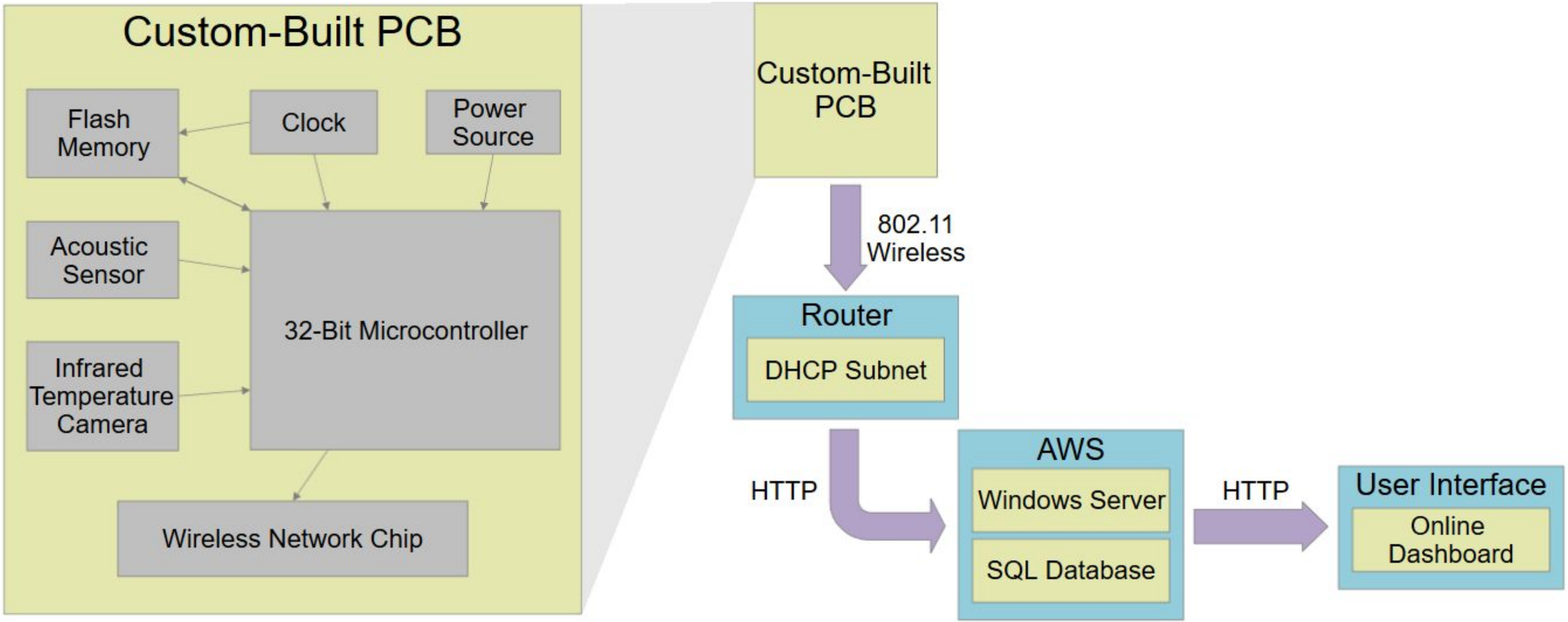
Coresidium – A School Security System

- Members
  - Valentin Degtyarev (CSE) - Acoustic Sensor
  - Andrew Eshak (CSE + EE) - Server Backend & Computation
  - Brandon Cross (CSE) - Thermal Sensor
  - Andrew LaMarche (CSE) - Communication & System Management

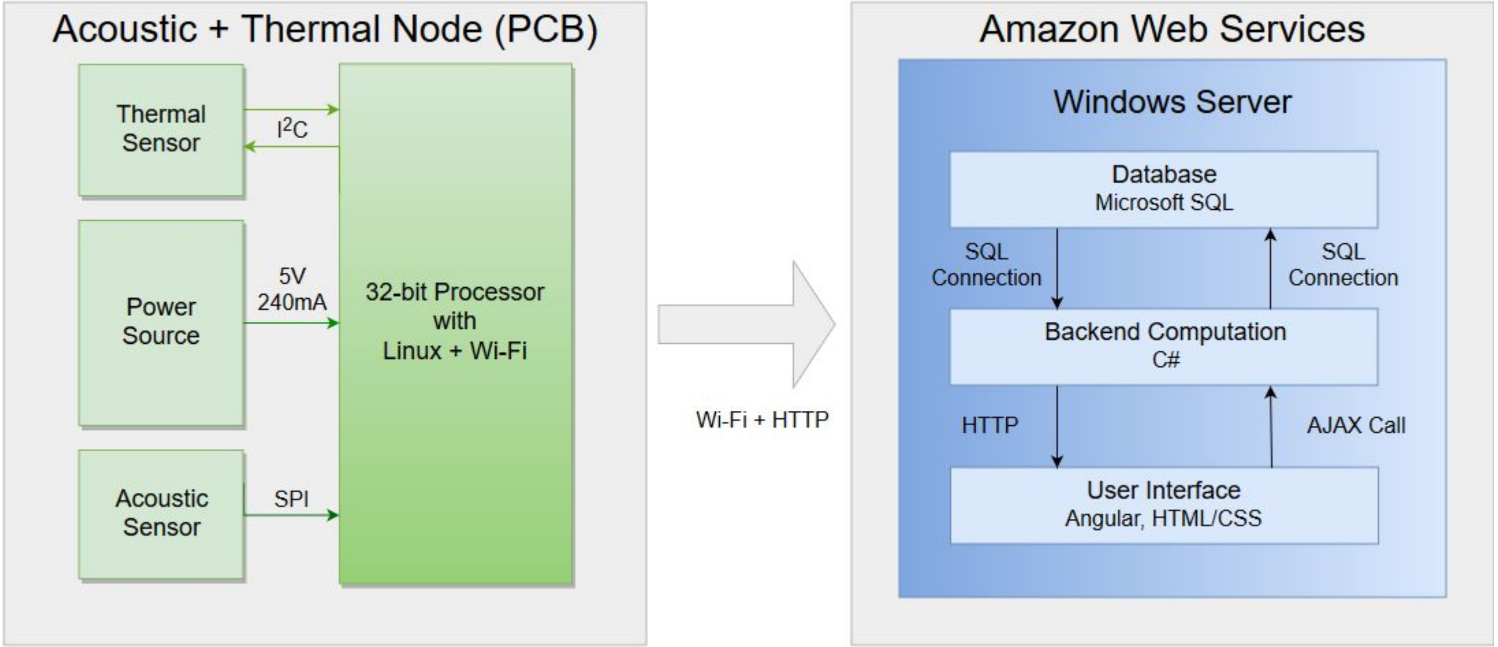
Problem Statement

- In the aftermath of a school shooting in the United States, it takes several hours for the campus to be declared safe. Due to this delay, help is not available to those who need it most. Our system aims to reduce this time by providing the relative location of where a shot has been fired. The proposed design uses infrared temperature cameras and acoustic sensors to accurately recognize an active shooting situation and notify the proper authorities.

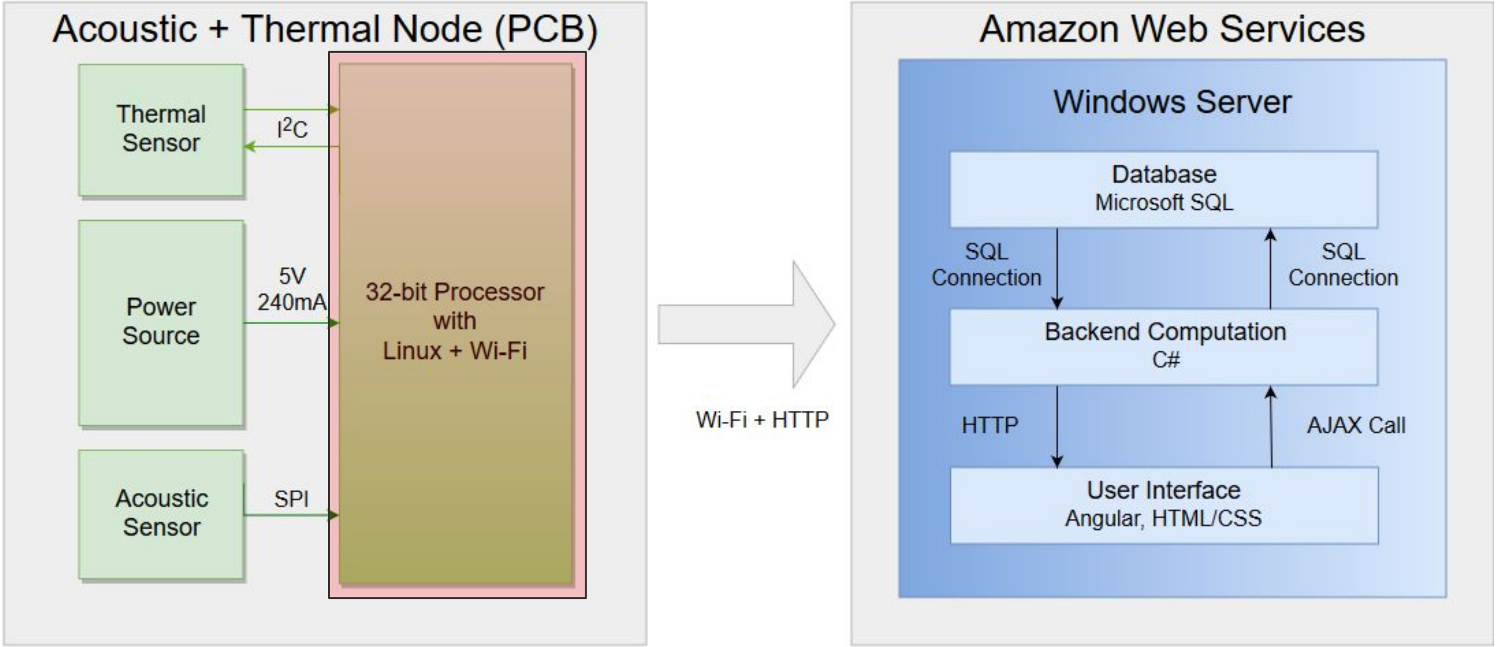
Original Block Diagram



Proposed Block Diagram

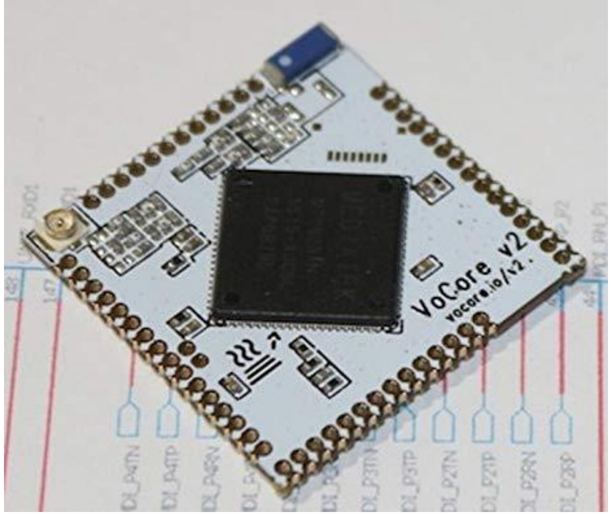


VoCore v2 32-bit Processor



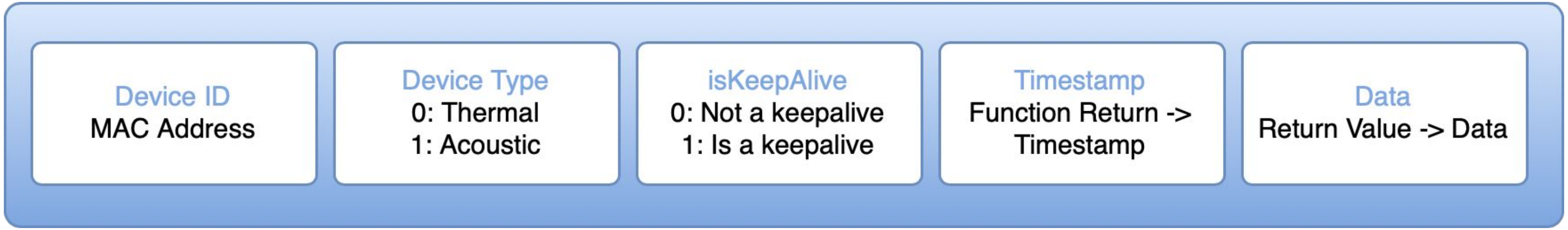
VoCore v2 32-bit Processor

- MIPS architecture
- Prebuilt with Linux and Wi-Fi
- 580 MHz clock speed
- Handles on-chip data processing
- Communicates data to server via HTTP



VoCore Boot Configuration

- 5 element struct
  - Load Device ID from config file
  - Set device type
  - Set isKeepAlive to 0 by default
  - Timestamp and Data come from each processor respectively

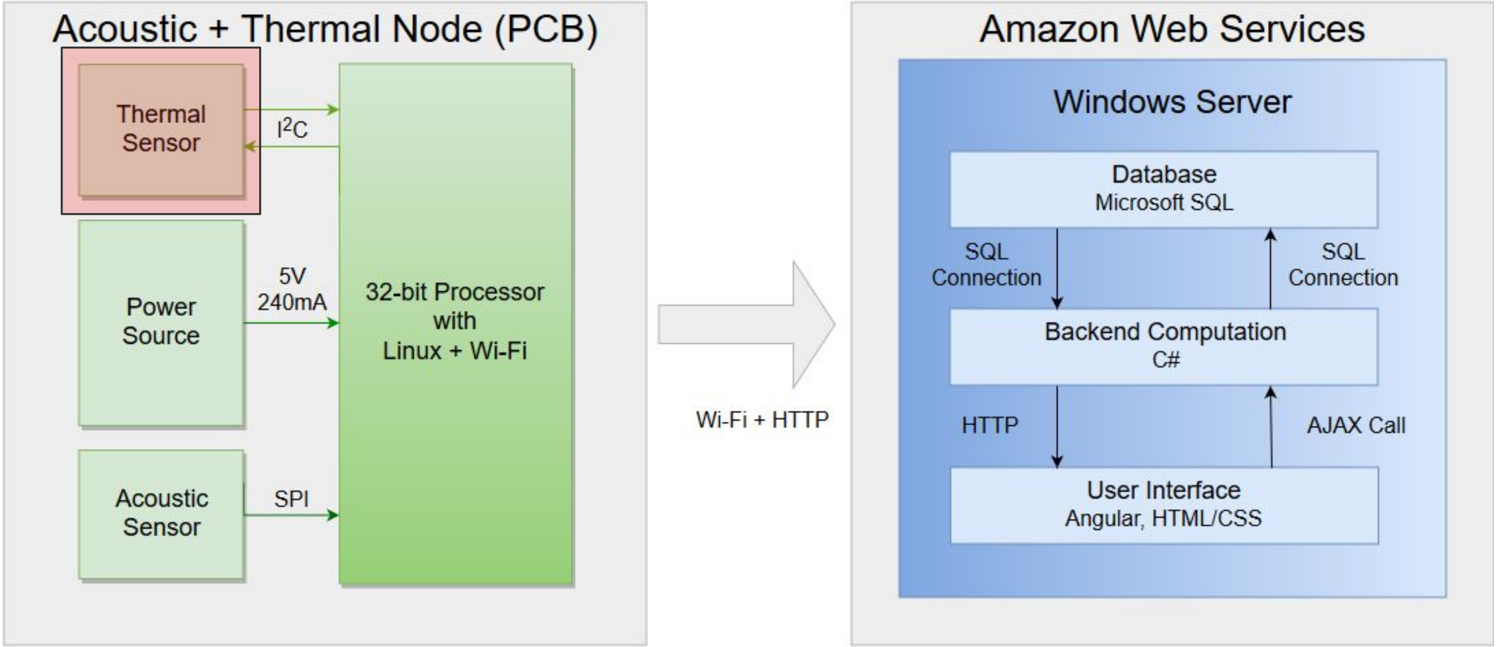


Device ID	Device Type	isKeepAlive	Timestamp	Data
MAC Address	0: Thermal 1: Acoustic	0: Not a keepalive 1: Is a keepalive	Function Return -> Timestamp	Return Value -> Data

VoCore Communication & Return Processing

1. Audio/thermal processor return to main
2. Convert all struct values to HTTP GET string
3. Create socket with client.c
4. Send string to backend server
5. Wait for OK response
6. Call audio/thermal processor function

MLX90640 Thermal Sensor



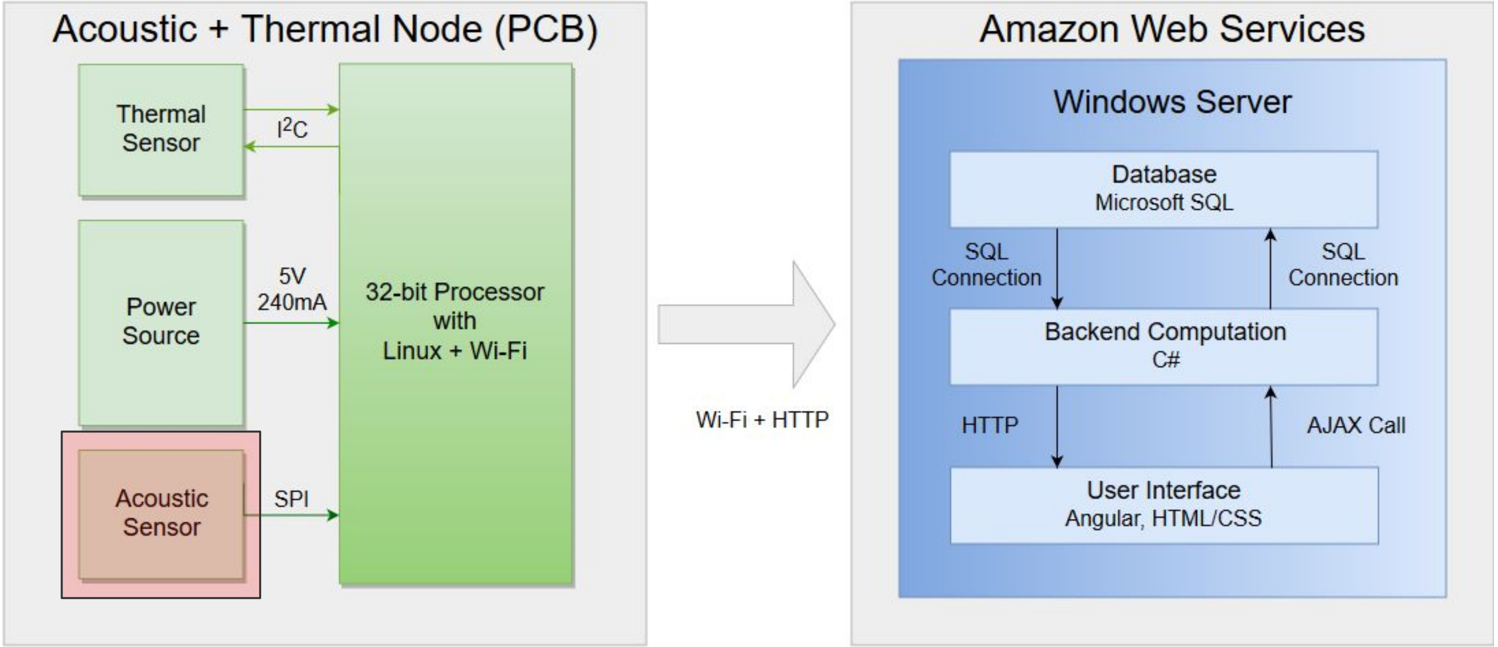
MLX90640 Thermal Sensor

- Thermal detection of barrel upon firing
- Gunshot heats rifle barrel to ~150ºC
- 32x24 resolution
- 110° x 75° viewing angle
- 32 frames per second (Raspberry Pi)
- -40ºC to 300ºC

Thermal Algorithm (thermal.cpp)

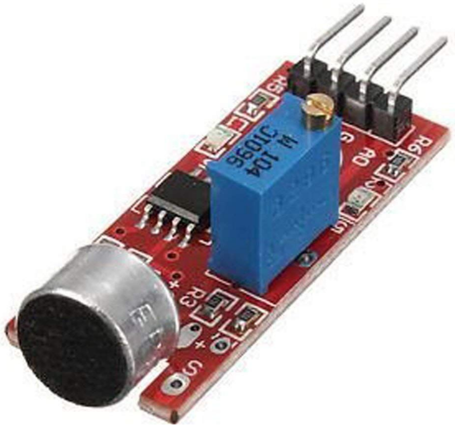
1. Configure MLX90640 parameters
   - 32 FPS, 18-bit quantization
2. Pull an image from sensor EEPROM
3. Store highest pixel value
4. Value exceeds custom threshold?
   - Yes: Proceed to step 5
   - No: Repeat steps 2-4
5. Timestamp
6. Save highest value
7. Return control to main

KY-038 Acoustic Sensor



KY-038 Acoustic Sensor

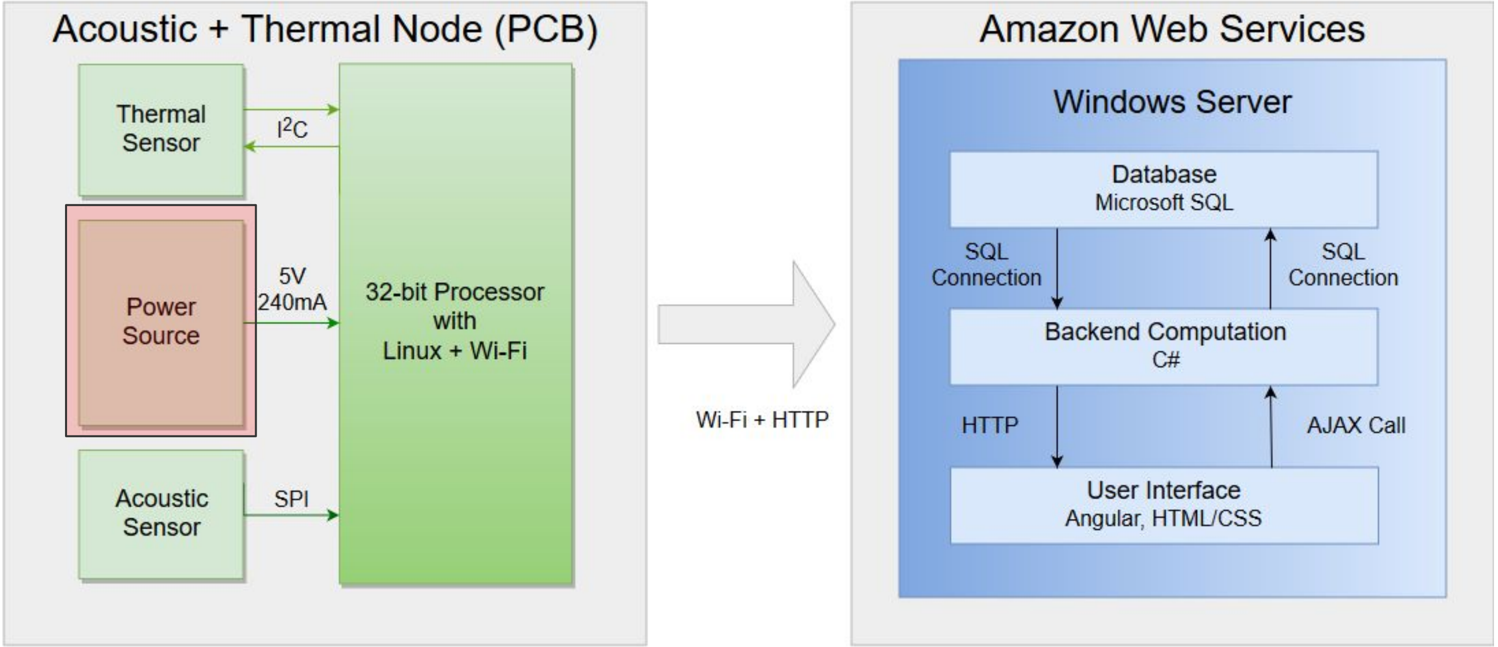
- Large sound sensor to detect gunshot
- 130 dB peak detection
- Gunshot between 150 dB and 190 dB
- 21,000 samples per second (Raspberry Pi)
- Output analog sound amplitude



Sound Algorithm (audio.c)

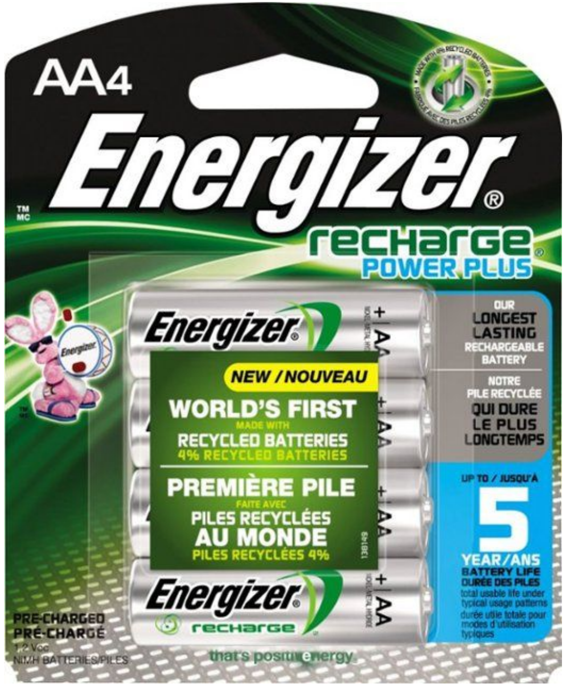
1. ADC converts amplitude to 10-bit encoding (0 - 1023)
2. Sound exceeds custom threshold bounds?
   - Yes: Proceed to step 3
   - No: Repeat steps 1-2
3. Timestamp
4. Sample 4000 points
5. Save highest value
6. Return control to main

Power Source

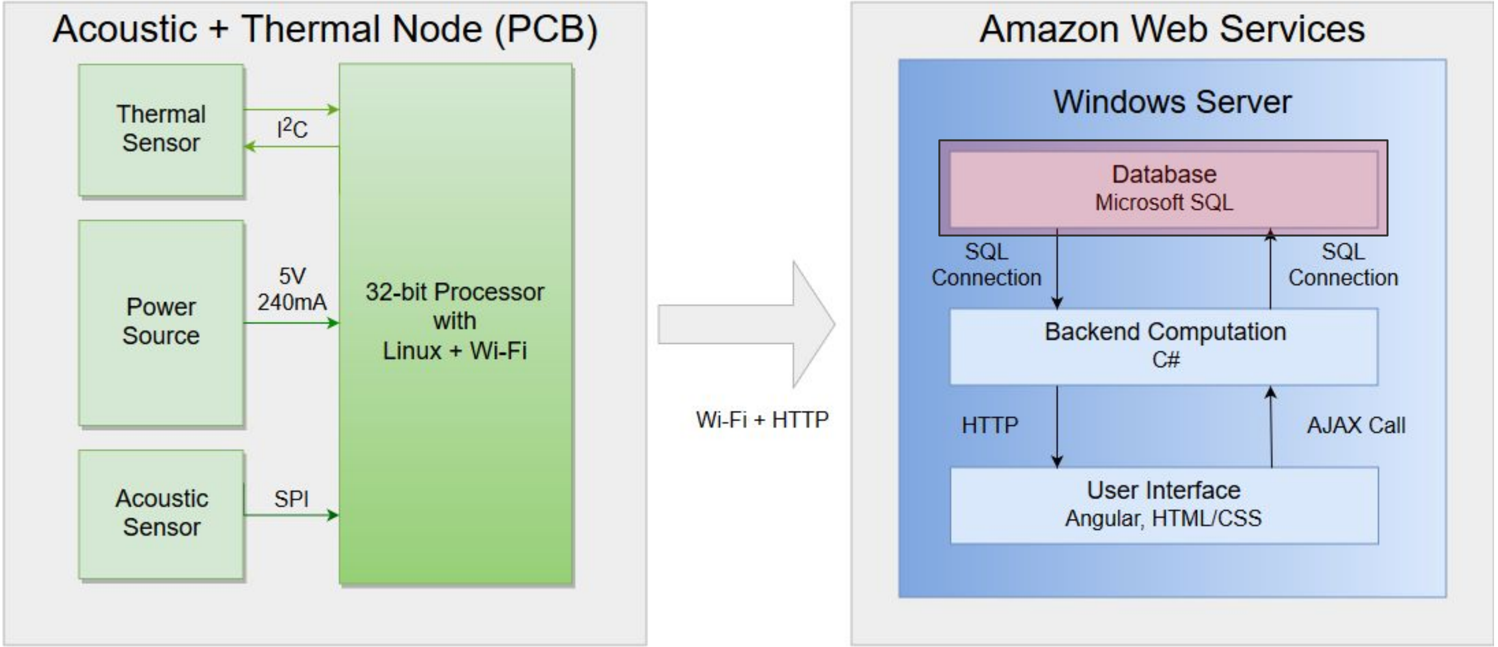


Power Source

- Processor requires 5V and max ~240mA
- Thermal sensor requires max ~25mA
- Acoustic sensor requires max ~1mA
- 4 AA batteries in series
- Would last roughly 30 hours
- Portability and ease-of-use



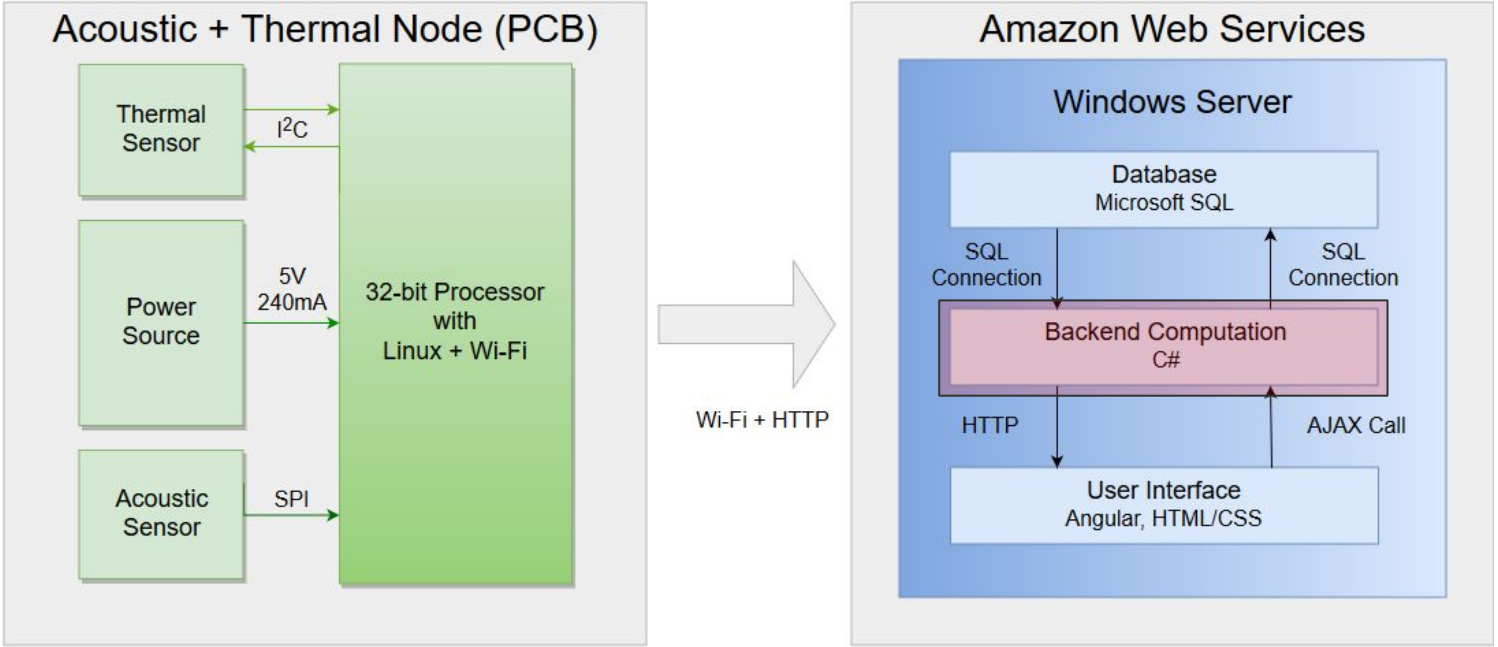
Database


Acoustic + Thermal Node (PCB)
Thermal Sensor
I²C
Power Source
5V
240mA
32-bit Processor with Linux + Wi-Fi
Acoustic Sensor
SPI
Wi-Fi + HTTP
Amazon Web Services
Windows Server
Database
Microsoft SQL
SQL Connection
SQL Connection
Backend Computation
C#
HTTP
AJAX Call
User Interface
Angular, HTML/CSS

Database

- Microsoft SQL based
- Stores user account information, location mappings and incident log
- Will implement user access control

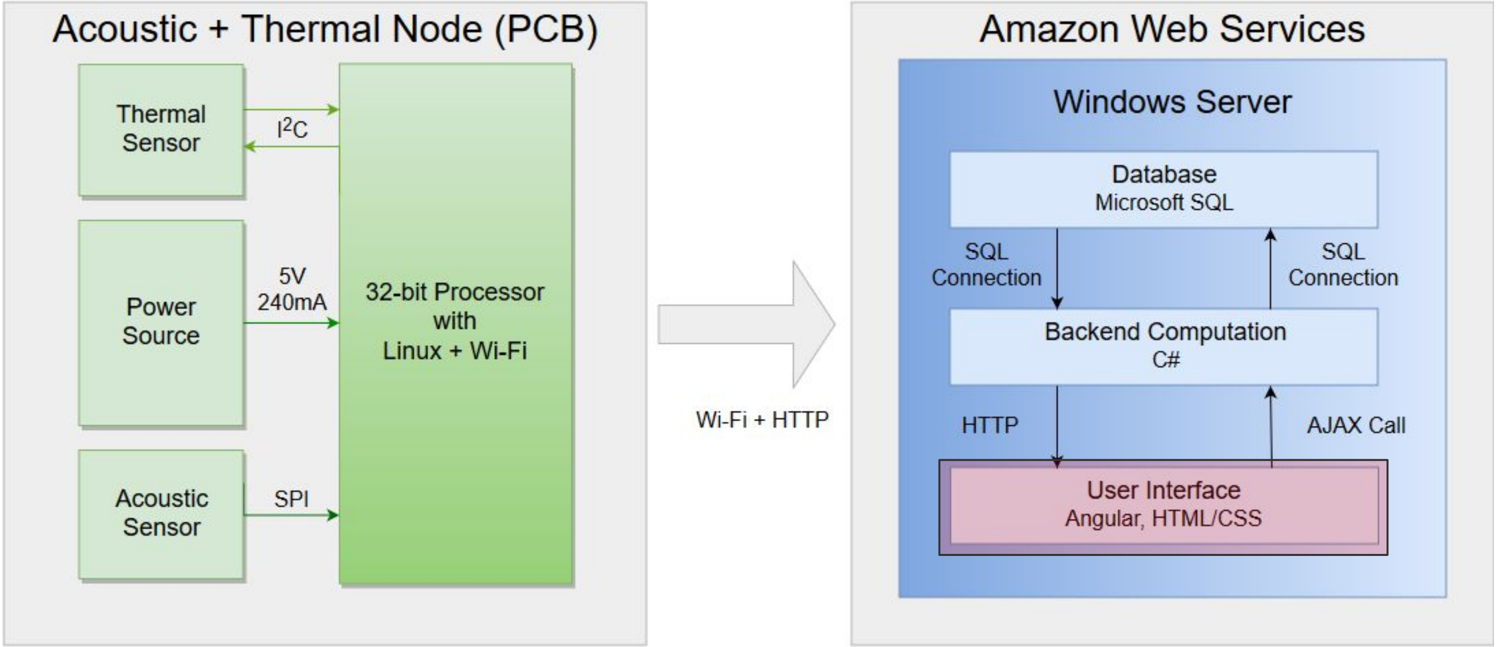
API


Acoustic + Thermal Node (PCB)
Thermal Sensor
I²C
Power Source
5V
240mA
32-bit Processor with Linux + Wi-Fi
Acoustic Sensor
SPI
Wi-Fi + HTTP
Amazon Web Services
Windows Server
Database
Microsoft SQL
SQL Connection
SQL Connection
Backend Computation
C#
HTTP
AJAX Call
User Interface
Angular, HTML/CSS

API

- Implemented in C#
- HTTP based
- Communication to the modules and to the dashboard
- Performs queries on the database as necessary
- backend.coresidium.com

Dashboard


Acoustic + Thermal Node (PCB)
Thermal Sensor
I^2C
Power Source
5V
240mA
32-bit Processor with Linux + Wi-Fi
Acoustic Sensor
SPI
Wi-Fi + HTTP
Amazon Web Services
Windows Server
Database
Microsoft SQL
SQL Connection
SQL Connection
Backend Computation
C#
HTTP
AJAX Call
User Interface
Angular, HTML/CSS

Dashboard

- Implemented in Angular, HTML/CSS
- Receives data from API
- Communicates location information to user
- dashboard.coresidium.com

Testing Procedure

- Pop 25 balloons from 12 feet away from 4 acoustic sensors
  - Repeat test in environment with different ambient noise
  - Record ratio between success rate and total attempts
  - Varying balloon size
  - Direct line of sight
- Ignite a lighter 25 times from 3 feet away from thermal sensor for 0.5 seconds

Results

- Acoustic Sensors Test #1 (No ambient noise)
  - Average success rate of acoustic sensors - 86%
- Acoustic Sensors Test #2 (With ambient noise)
  - Average success rate of acoustic sensors - 80%
- Thermal Sensor Test
  - Average success rate of thermal sensor - 68%

Expenses (per device)

VoCore v2:	$20.00
Acoustic Sensor (CMA-4544PF-W):	$0.94
Thermal Sensor (MLX90640):	$44.58
Analog to Digital Converter (MCP3008):	$2.26

Module Cost:	$67.78
PCB Fabrication:	$8.00
Total:	**$75.78**

Challenges

- Low thermal camera resolution
- Variation in acoustic sensor configurations

MDR Deliverables

✓ Acoustic module identifies simulated gunshot with 65% accuracy

✓ Thermal camera recognizes objects above 120°C for 0.5 seconds

✓ Store data in SQL, compute location and coordinate between modules

✓ Simple online dashboard with relative location of threat (floor & side)

? Find acceptable insulation for microphone

Proposed CDR Deliverables

- Eliminate breakout boards for acoustic/thermal sensors
- Implement keepalive for each node
- Voting/redundancy system
- Improve dashboard UI
- Functional VoCore

Experiments

- Sound signature/Fourier transforms
- Sound sampling/file storage
- Temperature and dB data conversion
- Thermal video storage

DEMO