iHeart Sensors

Leslie A. Arriaga

Computer Information Systems + Graphic Design
Estrella Mountain Community College
June 29, 2023

Mentors: Michael Stanley, Daniel Gulick, Jean Larson, Karl Ernsberger

NSF Award 1953745
https://sensip.engineering.asu.edu/ret/
Data Storage Methods for Wireless Monitoring of Neonates for Prevention of Neurological Conditions
RESEARCH BACKGROUND

Identifying neurological conditions during the first few hours after birth is crucial for providing timely intervention. Wireless monitoring enhances neonatal care:

Enables healthcare providers to move freely while receiving real-time data.

Promotes a positive parental experience by facilitating uninterrupted skin-to-skin bonding.
Research the best method to capture and store data wirelessly transmitted from sensors to a computer. Fundamental for data gathering for next phase analysis, including Machine Learning (ML) techniques for fault detection and classification.

Past failed attempts to save the data to SD cards yielded 50%-90% data loss. SD cards were not fast enough.
RESEARCH OBJECTIVE

Hardware Modification

This project explores the viability of eliminating the “receiving” ESP32 board and SD Card from the framework by testing the following:

1. The “sending” ESP32 and sensors connect via WiFi with the “receiving” ESP32, which is serially connected to the computer.
2. The “sending” ESP32 and sensors connect directly to the computer via WiFi.
Data Storage Methods for Wireless Monitoring of Neonates for Prevention of Neurological Conditions

RESEARCH RESULTS

Method 1 – Coding a New Application

The research first explored the feasibility of writing a new computer program with C# and Visual Studio.

• Too difficult to accomplish within the set timeframe
• Limited coding skills

Method 2 – Using an Existing Application

Research pivoted to explore existing methods that could meet the objective.

WINNNER

CoolTerm
Data Storage Methods for Wireless Monitoring of Neonates for Prevention of Neurological Conditions

RESEARCH RESULTS

CoolTerm is a serial port terminal application geared towards hobbyists and professionals with a need to exchange data with hardware connected to serial ports.

Captures and saves data
PREPARATION
- Installing the Arduino IDE
- Understanding the original JSON code installed in the ESP32 boards
- Update the code to include my new WiFi information

COOLTERM SETUP
- Selecting the correct port – serial or WiFi
- Baudrate – 115,200 (matches ESP32 code)

**First Goal**
- Successful re-creation of the original hardware setup.
4 different trials conducted to ensure data is captured and saved properly.

### RESEARCH RESULTS

<table>
<thead>
<tr>
<th>Test</th>
<th>Hardware</th>
<th>Code</th>
<th>Connection</th>
<th>Results</th>
<th>Visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Single ESP32</td>
<td>Dummy code uploaded to ESP32 to simulate sensor information</td>
<td>ESP32 serially connects to the computer</td>
<td>Success CoolTerm captures and saves the data</td>
<td><img src="image1" alt="ESP32" /> <img src="image2" alt="Computer" /></td>
</tr>
<tr>
<td>2</td>
<td>1 ESP32 connected to all sensors</td>
<td>Original JSON activating all sensors</td>
<td>ESP32 serially connects to the computer</td>
<td>Success</td>
<td><img src="image1" alt="ESP32" /> <img src="image2" alt="Computer" /></td>
</tr>
<tr>
<td>3</td>
<td>2 ESP32 + all sensors</td>
<td>Original JSON on both ESP32s</td>
<td>Sensors+ESP32 wirelessly connects to receiving ESP32, serially connected to computer</td>
<td>Success</td>
<td><img src="image1" alt="ESP32" /> <img src="image2" alt="Computer" /></td>
</tr>
<tr>
<td>4</td>
<td>1 ESP32 connected to all sensors</td>
<td>Original JSON activating all sensors</td>
<td>Sensor ESP32 wirelessly connects to the computer</td>
<td>TCP CoolTerm Connection failure</td>
<td><img src="image1" alt="ESP32" /> <img src="image2" alt="Computer" /></td>
</tr>
</tbody>
</table>
Excel Macro: File Prep Option 1

Two options to organize the original JSON data (highlighted above) in preparation for ML analysis.

RESEARCH RESULTS

Python: File Prep Option 2

Two options to organize the original JSON data (highlighted above) in preparation for ML analysis.

```python
[6]:
import pandas as pd
import json

file_path = 'text.txt'  # Replace with the actual file path

data_list = []  # List to store individual JSON statements

with open(file_path, 'r') as file:
    for line in file:
        json_data = json.loads(line.strip())  # Load JSON from each line
        data_list.append(json_data)  # Append JSON to the list

df = pd.DataFrame(data_list)  # Create a DataFrame from the list of JSON statements

display(df)
```

<table>
<thead>
<tr>
<th>timer</th>
<th>ECG</th>
<th>PPG</th>
<th>T</th>
<th>CO2</th>
<th>pH</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>61255</td>
<td>16878</td>
<td>163</td>
<td>25</td>
<td>402</td>
</tr>
<tr>
<td>1</td>
<td>61255</td>
<td>554</td>
<td>163</td>
<td>25</td>
<td>402</td>
</tr>
<tr>
<td>2</td>
<td>61255</td>
<td>-11854</td>
<td>163</td>
<td>25</td>
<td>402</td>
</tr>
<tr>
<td>3</td>
<td>61271</td>
<td>-17594</td>
<td>159</td>
<td>25</td>
<td>402</td>
</tr>
<tr>
<td>4</td>
<td>61271</td>
<td>-15088</td>
<td>159</td>
<td>25</td>
<td>402</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>732</td>
<td>69369</td>
<td>-12475</td>
<td>159</td>
<td>25</td>
<td>406</td>
</tr>
<tr>
<td>733</td>
<td>69369</td>
<td>-10764</td>
<td>159</td>
<td>25</td>
<td>406</td>
</tr>
<tr>
<td>734</td>
<td>69389</td>
<td>1985</td>
<td>163</td>
<td>25</td>
<td>406</td>
</tr>
<tr>
<td>735</td>
<td>69389</td>
<td>15617</td>
<td>163</td>
<td>25</td>
<td>406</td>
</tr>
<tr>
<td>736</td>
<td>69389</td>
<td>23915</td>
<td>163</td>
<td>25</td>
<td>406</td>
</tr>
</tbody>
</table>

737 rows x 6 columns
Objective accomplished!

1. CoolTerm captures and stores wirelessly transmitted data from the sensors (see diagram).
2. Unable to eliminate the second ESP32 board.
3. Both the Excel Macro and Python code developed are essential for future ML data analysis.
Students are introduced to embedded sensors through a micro lecture and various hands-on activities/assignments.

<table>
<thead>
<tr>
<th>#</th>
<th>Learning Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Recall and define key vocabulary words related to embedded sensors.</td>
</tr>
<tr>
<td>2</td>
<td>Utilize the downloaded app to capture their own heartbeats using embedded sensors.</td>
</tr>
<tr>
<td>3</td>
<td>Explain the concept of at least one embedded sensor, its functionalities, and evaluate the limitations of using the embedded sensor in a real-life scenario.</td>
</tr>
<tr>
<td>4</td>
<td>Apply data visualization techniques in Excel to plot the captured heartbeat data.</td>
</tr>
<tr>
<td>5</td>
<td>Generate a 3D object and print it in the Makerspace.</td>
</tr>
<tr>
<td>6</td>
<td>Design and deliver a slide presentation showcasing their knowledge of embedded sensors, the heartbeat graph, and their 3D printed creation.</td>
</tr>
</tbody>
</table>
In-Class Activities

#1 Micro Lecture

• What are sensors?
• Vocabulary
• Showcase my SenSip experience

#2 Activity – Vocab Game
#3 Activity - Heartrate

- Students download Phyphox App
- Students take own ECG with phone accelerometer
- Export data
LESSON OBJECTIVES

Assignments

#2 Graph heart rate data

Excel App

One cycle needed for 3D Print
#3 (optional)
Prepare Heartbeat Cycle to 3D print

Tinkercad App
LESSON OBJECTIVES

Assignments

#4 Project Showcase

PowerPoint App
CONCLUDING REMARKS

Project Next Steps

• Sensors to send data wirelessly direct to the computer
• Use sensors to obtain vital signs data
• ML analysis

Lesson Next Steps

• Transfer content to my LMS
• Teach the students and do the activities
Key research - skills gained (theory/software)?

• Experience with Arduino IDE, JSON, Python, sensors, ESP32
• Understanding of sensors
• Introduction to machine learning

Self-assessment – what did you get out of this experience?

• Research experience
• Learned about the other research at ASU
• Met some cool peeps in my cohort
CONCLUDING REMARKS

Program Observations/Recommendations

• Mentor introduction 1 week before start of program
• Assign projects on day 1 or before
  • Didn’t know what we were doing for 2 weeks
  • Start project work on day 1
• 1st week of lectures – consider learning experience
Thank you!
Questions?

Leslie Arriaga
Leslie.arriaga@estrellamountain.edu