Position Tracking Using WiFi

Tyler Bayne, Nathan Conrad, and Mitchell Woolley
Advisor: Dr. Ralph Tanner
Sponsor: Dr. John Kapenga
Contents

- Need Statement
- Scope
- Specifications
- Design Concept
- Methods
- Performance
- Unexpected Problems
- Comparing Performance to Original Specifications
- Recommendations
- Things We Learned
- Questions
A system is needed to track workplace items and employees, as well as report emergencies inside a building. It is necessary for the system to be able to determine the exact room each item or employee is located in.
Scope

- Physical Device
  - Microcontroller
  - User Interface
- Central Server
  - Display a map with device locations
Specifications

- Physical Characteristics
  - The device should be relatively small
  - A final version will be attached to an identification badge worn by employees

- Functionality
  - The device should consume a low amount of power
  - The device should be in a low power state when it is not reporting its location

- Portability
  - The device should be portable
  - The device should be able to operate for an 8 hour workday
Specifications (cont.)

- Communication
  - The device should be able to communicate on the 802.11 g and g bands

- User Interface
  - The device should allow users to send simple messages to the Central Server about an emergency and/or their location

- Position Tracking
  - The device should periodically report its position to the Central Server
  - The location of each device should be shown on a map of the building

- Cost
  - The device should be inexpensive
Design Concept

Device (Arduino Uno) → Central Server → Redpin Server
Design Concept (cont.)

- Schematic
Design Concept (cont.)

- Redpin
  - Open source
  - Initial calibration phase
    - Receives numerous measurements of signal strengths at different locations
    - Creates “fingerprint” for each location
  - Calculating location
    - Receives new signal strength measurement
    - Uses “fingerprints” to calculate current location
Design Concept (cont.)

- All devices are green
- No emergencies
Design Concept (cont.)

- D has an emergency
- Changes to red
Methods

- Central Server
  - Written in Java
  - NetBeans IDE used to write the server
- Arduino Uno Software
  - Written in C
  - Arduino IDE used to write the software and download to the Arduino Uno
- Version Control
  - Git used for main version control
  - BitBucket used to host the Git repository and easily share with entire team
Performance

- Testing the existing Redpin Android app and server
- Parkview
  - No additional access points
  - With additional access points
- House
Performance

Diagram:
- Kitchen
- Living Room
- Bathroom
- Bedroom
- Office
- Bedroom
- Access point main level
- Access point basement
Performance
Performance
Video
Unexpected Problems

- Issues with original microcontroller MSP430 and WiFi shield CC3000
- Arduino WiFi Shield can only locate a maximum of 10 networks
- Redpin triangulation was not consistent at Parkview
- Arduino WiFi Shield cannot return router specific identification (BSSID)
# Low Power

<table>
<thead>
<tr>
<th>Device</th>
<th>Power Consumption (mA)</th>
<th>Duty (8 hrs)</th>
<th>Total (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atmega328</td>
<td>5.2</td>
<td>100%</td>
<td>5.2</td>
</tr>
<tr>
<td>AT32UC3A1256</td>
<td>20</td>
<td>100%</td>
<td>20</td>
</tr>
<tr>
<td>HDG104 WLAN (Tx)</td>
<td>178</td>
<td>3.33%</td>
<td>5.93</td>
</tr>
<tr>
<td>HDG104 WLAN (Sleep)</td>
<td>0.06</td>
<td>96.67%</td>
<td>0.06</td>
</tr>
</tbody>
</table>

average current draw: 31.19133333

# of transmits per minute: 2
Duration of transmit (sec): 1

# Low Power

<table>
<thead>
<tr>
<th>Device</th>
<th>Power Consumption (mA)</th>
<th>Duty (8 hrs)</th>
<th>Total (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atmega328 (Idle)</td>
<td>1.2</td>
<td>90.00%</td>
<td>1.2</td>
</tr>
<tr>
<td>Atmega328 (Active)</td>
<td>5.2</td>
<td>10.00%</td>
<td>0.52</td>
</tr>
<tr>
<td>AT32UC3A1256 (Idle)</td>
<td>14</td>
<td>93.33%</td>
<td>14</td>
</tr>
<tr>
<td>AT32UC3A1256 (Active)</td>
<td>20</td>
<td>6.67%</td>
<td>1.334</td>
</tr>
<tr>
<td>HDG104 WLAN (Sleep)</td>
<td>0.06</td>
<td>96.67%</td>
<td>0.06</td>
</tr>
<tr>
<td>HDG104 WLAN (Tx)</td>
<td>178</td>
<td>3.33%</td>
<td>5.93</td>
</tr>
</tbody>
</table>

average current draw: 23.04533333

# of transmits per minute: 2
Duration of transmit (sec): 1
Comparing Performance to Original Specifications

<table>
<thead>
<tr>
<th>Specification</th>
<th>% Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relatively small device</td>
<td>85%</td>
</tr>
<tr>
<td>Low power consumption</td>
<td>50%</td>
</tr>
<tr>
<td>Portability</td>
<td>100%</td>
</tr>
<tr>
<td>Long operating life</td>
<td>75%</td>
</tr>
<tr>
<td>Communicate using 802.11 b/g</td>
<td>100%</td>
</tr>
<tr>
<td>Device sends emergency message</td>
<td>50%</td>
</tr>
<tr>
<td>Device periodically reports location</td>
<td>100%</td>
</tr>
<tr>
<td>Locations of devices shown on map</td>
<td>100%</td>
</tr>
<tr>
<td>Inexpensive</td>
<td>50%</td>
</tr>
<tr>
<td>Total Percent Complete</td>
<td>75%</td>
</tr>
</tbody>
</table>
Recommendations

- Make the device smaller
  - Design a printed circuit board that only has the necessary components
- Add two-way communication between the devices and Central Server
  - Devices receive messages from central server alerting them of a button press from another device
- Add extra components to the device
  - Liquid-Crystal Display (LCD)
    - Display emergency messages to the user
  - Vibration motor
    - Alert user of an incoming message
- Modify the Arduino WiFi Shield’s firmware and libraries to allow more than 10 access points per scan result and include BSSID in the result
Acknowledgements

- Dr. Ralph Tanner for being our faculty advisor
- Dr. John Kapenga for being our sponsor
- Dr. Damon Miller for teaching ECE 4810 Senior Design I
- Dr. Massood Atashbar for teaching ECE 4820 Senior Design II
Any Questions?