

The Team

Matthew Speck - Project Lead, Parts selection, PCB/Schematic

Amber Du - Data Collection, Signal analyst

Ahmed Saied - Software Development

Kevin La - Network Development









What is BLIPS?

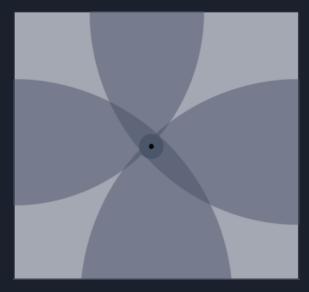
BLIPS stands for Bluetooth Low energy Indoor Positioning System. Our goal is to track the movement of doctors, nurses, and equipment in an operating room using Bluetooth Low Energy receivers that are placed in the employee's ID card holders.



A finished BLIPS badge assembly

How does BLIPS work?

- 3 6 Bluetooth beacons are placed at known locations on the ceilings of the room
- The BLIPS badge records Received Signal Strength Indicator (RSSI) values from the beacons
- RSSI Values correlate to distance to the receiver on the BLIPS badges
- Several of theses distances can be combined to calculate location
- An Inertial Measurement Unit (IMU) is used to ensure long battery life

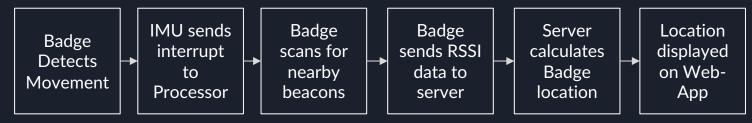


An example of a location calculation

Overview

- Location tracking of people in a room using Bluetooth Low Energy signals
 - o IMU interrupt-driven
- Sending data real time to a server, which displays data on a web application, via

WiFi



BLiPS operation flow

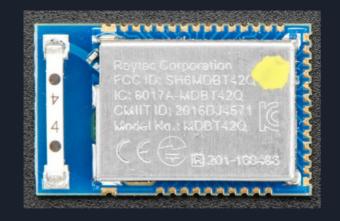
Demo



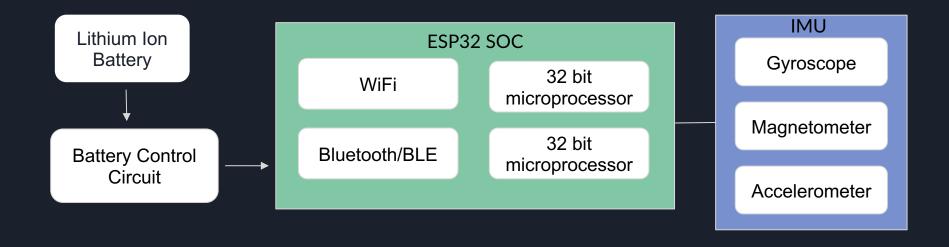
Bluetooth Low Energy (BLE) Beacons

BLE Module: Nordic nRF52832 running custom firmware

- Uses iBeacon BLE protocol for transmission
- Transmits 15 packets per second
- Strobes between signal strengths for added location accuracy
- Always on
- Powered by 2500 mAH Li-ion battery
 - O Battery life expectancy of a year from single charge
- Located in fixed known locations
 - O Ex: room corners, center of room



BLIPS Badge Block Diagram



Parts: Microprocessor

ESP32 BLE and WiFi module

- Xtensa 32 bit microprocessor
- Dual core
- 160 MHz
- 448 Kb of ROM, 520 Kb SRAM
- Ultra low power co-processor for sleep @ 8Mhz
- 150 mA max current draw



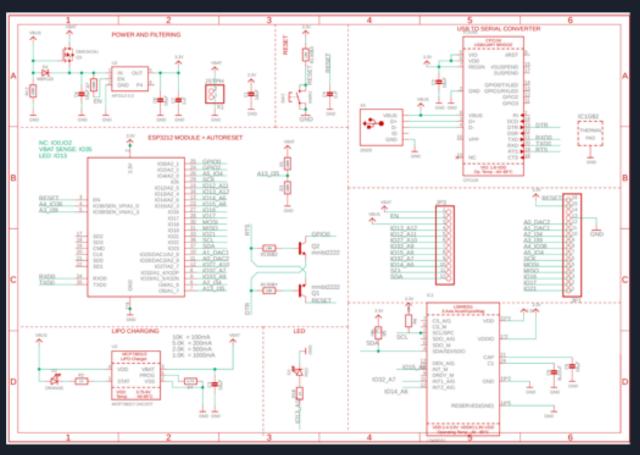
Parts:IMU

LSM9DS1

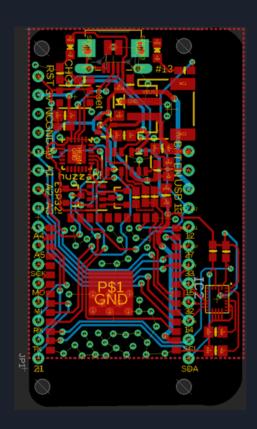
- IMU chip, contains gyroscope, accelerometer, magnetometer
- Connects through I2C
- 9 Degrees of freedom
- Used to wake CPU from low power state
- 15 mA max current draw



Schematic



PCB





Software Development (Badge side)

- Badges use an interrupt driven software structure
 - High energy processor state is costly, so we only transmit/collect on a change in position
- Accomplished via the inertial measurement unit (IMU) on our board
 - O Using accelerometer/gyroscope, we can tell when a badge has shifted its location drastically

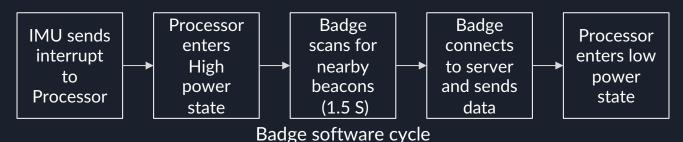


IMU Dev board

Interrupt Handling

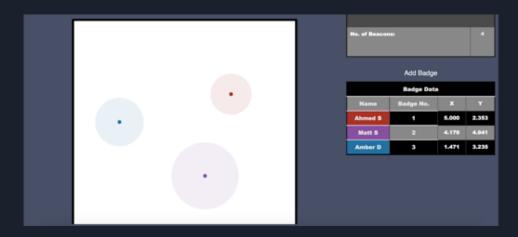
- Interrupt handling is done through RTOS (real time operating system)
- An interrupt is triggered by the LSM9DS1 IMU by setting a pin high, which wakes the ESP32 so data collection can begin.





User Interface

- Our interface is a MeteorJS web-app running on a separate server
- A user can add a new badge, give it a name, and have it assigned a badge number so that it would appear on the web app display
- Position is recorded (in meters) at all times whenever badges are present, as seen through the X and Y coordinates
- Entry and exit to the room can also be read based on if we are properly receiving bluetooth data on the badge side



Client - Server Communication

Client Side:

- ESP32 (badge) CPU has a built in WiFi module
 - O Once data is recorded we send it off to be dealt with on the server side
 - High energy state is minimized
- Client Side code is compartmentalized and cleans up data as much as possible
- We use TCP to transmit data from client to server
- Each TCP packet includes data for every beacon it can see (less transmissions)

```
payload: 1:4=52,0.-52,-11.-50,-13.-52,-11.-52,-11.-52,-11.-52,-11.-49,-14./3=-72,0.-73,10.-71,8.-76,13./1=-66,0.-64,1.-61,-2.-66,3.-66,3.-66,3.-65,2./:0.35.105.80.115.76.=-93.-92.-93./176.90.218.52.214.207.=-71.-71.-70.-70.-70./26.245.50.145.16.129.=-92.-95.-95.-93.-93./160.4.96.97.47.243.=-88.-87.-89.-89./176.78.38.41.100.34.=-95./0.17.50.87.166.203.=-56.-57.-56.-55.-55./248.245.50.145.16.129.=-94./192.86.39.7.63.10.=-90.-90.-92./28.73.123.57.160.204.=-90.-92.-90.-90./192.197.34.2.188.156.=-90./
```

Client - Server Communication

Server-Side

- Receives RSSI values and IMU data from the badges and stores it
- Processes the data received, and displays it on a map showing where the badges are
- Constantly process new data received, continuously updating location of badges

```
I20190522-20:37:39.970(-7)? Beacon ID: 4
I20190522-20:37:39.970(-7)? 2.2387211385683394
120190522-20:37:39.984(-7)? A badge has signaled to disconnect
I20190522-20:37:39.985(-7)? Socket fully closed.
I20190522-20:37:39.985(-7)? Raw: -85 Diff: 6
I20190522-20:37:39.986(-7)? 2.411908026006927
I20190522-20:37:39.988(-7)? 3.79029267078917
I20190522-20:37:39.989(-7)? Beacon ID: 3
I20190522-20:37:39.991(-7)? MAC address: 12.128.99.173.132.210.
I20190522-20:37:50.704(-7)? Server connected on: ::ffff:192.168.0.101 from ::ffff:192.168.0.100
I20190522-20:37:50.704(-7)? Sent message to badge
I20190522-20:37:50.705(-7)? ---- Web App Received Data ----
120190522-20:37:50.705(-7)? 1:4=-93.14./:12.128.99.173.132.210.=-61.-62.-65.-69.-67./
I20190522-20:37:50.707(-7)? Raw: -93 Diff: 14
I20190522-20:37:50.707(-7)? 5.011872336272722
I20190522-20:37:50.710(-7)? MAC address: 12.128.99.173.132.210.
I20190522-20:37:50.710(-7)? RSSI 0: -61
I20190522-20:37:50.711(-7)? RSSI 1: -62
I20190522-20:37:50.713(-7)? RSSI 2: -65
I20190522-20:37:50.716(-7)? RSSI 3: -69
I20190522-20:37:50.717(-7)? RSSI 4: -67
I20190522-20:37:50.722(-7)? A badge has signaled to disconnect
```

An example of the server side terminal

Challenges

We faced many challenges in developing this project, here are some we worked on solving:

- BLE receivers need an almost direct line of sight to receive signals
- Unstable signal strength from BLE
- BLE signal does not drop consistently with distance
- Problems with BLE signals reflecting off walls gives inaccurate RSSI readings
- Implemented Wi-Fi based tracking as an attempt to improve accuracy
 - O Wi-Fi doesn't send data management frames often enough to calculate distance reliably from multiple routers
- Inability to accurately track badges located in the center of the room

Problems with RSSI: Unstable Strength

Problem: Wild variations in signal strength

 Bluetooth signal strength naturally fluctuates wildly, also very easily influenced by external conditions

Solution: Kalman Filtering

 A way of normalizing the received signal strengths





Problems with RSSI: No consistent power drop with distance

Problem: BLE Signal strength drops inconsistently

Solution: artificially attenuate the signal strength by strobing the transmit strength

- Every third signal sent changes power level
- -8, -12, -16 Db transmission levels
- Each power level gives hard cutoff for how close to transmitter badge is



Problems with Wi-Fi based tracking

In addition to BLE tracking we implemented a Wi-Fi sniffing based tracking system. This however had its own complications:

- Tracking was accomplished via measuring received power of Wifi management frames
 - O Too few management frames received at any given time led to inability to accurately find location
- Wi-Fi sniffing extended the time the badge processor was in a high power state
 - Shorter battery life
 - O Halved number of locations able to be recorded by server in a given time

BLIPS Successes

Despite the stated issues with the project here is what is successfully accomplished:

- Interrupt driven badge system for whole day battery life
- Tracking when people enter and leave the room
- Accurately tracking BLIPS badge wearers who are located close to BLE emitters
- Simultaneously tracking multiple BLIPS badge wearers in the same room



Indoor Positioning System

Thanks (again) to:

- Our Sponsor: Arthrex
- Our Instructor : Yoganada Isukapalli
- Our TA's:
 - O Brandon Pon
 - Carrie Segal



