ENGINEERING FOR impact

mit 6.900

Lecture 4

February 15, 2024

Make something real.

Schematic best practices

MAQS: MIT Air Quality System requirements

We developed requirements based on internal staff discussions and talking with EDS staff

1. It should accurately measure indoor air quality **
2. It should be portable ***
3. It should be possible to get the data off the device **
4. It should be a useful pedagogical exercise ***
5. It should maintain privacy *
6. It should be low cost *
7. It should be rugged and robust **
8. Multiple systems should be able to be used simultaneously ***
9. It should be easy to view the current and past data **
10. It should leverage MIT facilities **

Sentimet

MITOS

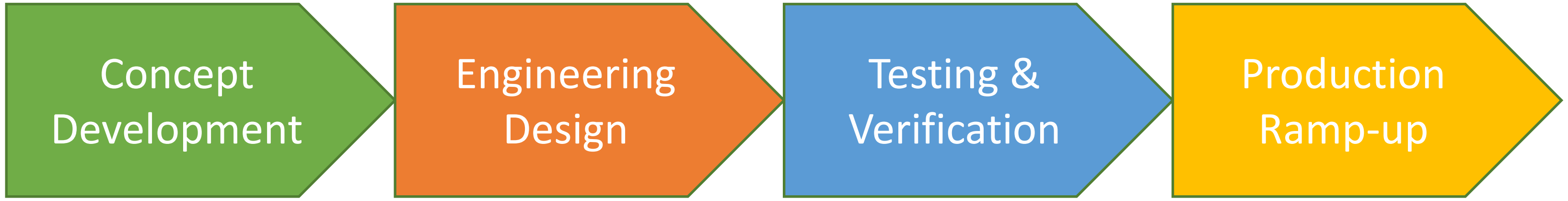
1. It should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
2. It should be able to measure how many people are in the area passing through (e.g., foot traffic) and lingering.***
3. It should operate without being connected to line voltage. ***
4. It should be portable and able to be set up by an average person in a variety of outdoor environments on the MIT campus, including on a tripod or attached to poles of various dimensions.***
5. It should be able to be physically attached to a HOBO MX2302A data logger.*
6. It should report faults, such as battery failure, falling, vandalism, etc.**
7. It should be as inexpensive as possible. *
8. Data from a sensor node should be able to be tied to a location.***
9. It should maintain privacy. ***
10. It should operate independently without user intervention for 2+ weeks.***
11. It should be rugged and able to withstand a summertime Boston-area environment (heat, rain, wind and curious people). ***
12. Multiple systems should be able to be used simultaneously. ***
13. The system should present the information on a dashboard (with real-time data outputs to a dashboard if possible), and also allow downloading of raw data.***

Miami-Dade

1. It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***
2. It should be able to measure how many people are waiting, and for how long.***
3. It should operate without being connected to line voltage. ***
4. It should be installable by a technician, and should be easy to set up.***
5. It should report faults, such as battery failure, falling, vandalism, etc.**
6. It should be as inexpensive as possible. *
7. Data from a sensor node should be able to be tied to a location.***
8. It should maintain privacy. ***
9. It should operate independently without user intervention for at least a month.***
10. It should be rugged and able to withstand shipping, setup, and operation in the Miami-Dade environment. ***
11. Multiple systems should be able to be used simultaneously. ***
12. The system should incorporate data from Swift.ly and present that information to the operator in a useful way. ***

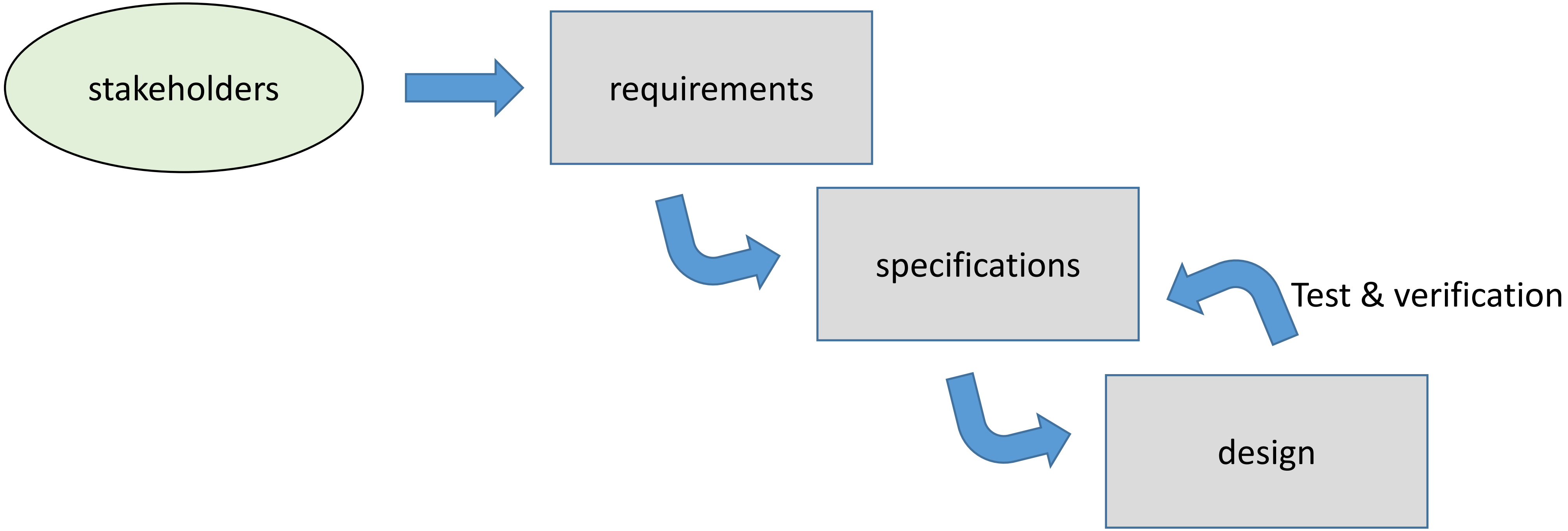
Mostly the same…could make one overall system, or two slightly different systems

Now what?



- We iterate between developing concepts, setting specs, doing design, and developing a testing plan
  - Because Sentimet already has v1 from last year, we can start from that concept and keep/change it
- Not everything will necessarily be defined yet (and thus able to be concretely specified)
  - This may not be ok for some products (aerospace, medical) but often the case for consumer, etc. ➔ Iteration can be important
- HW and SW are specified differently
  - We'll see this

Requirements, specs, and so on



Done well, a design that passes all the tests will meet the specifications and thus the requirements, making the stakeholders happy

MAQS: Power management – modeling

- Power
  - We'll do a lot more in a few weeks…
- Can we estimate the lifetime?
- What's the biggest energy consumer – usually MCU or comms
  - In our case, WiFi
  - Let's check ESP32-C3 datasheet

Table 9: Current Consumption Depending on RF Modes

Work mode	Description		Peak (mA)
Active (RF working)	TX	802.11b, 1 Mbps, @20.5 dBm	345
		802.11g, 54 Mbps, @18 dBm	285
		802.11n, HT20, MCS7, @17.5 dBm	280
		802.11n, HT40, MCS7, @17 dBm	280
	RX	802.11b/g/n, HT20	82
		802.11n, HT40	84

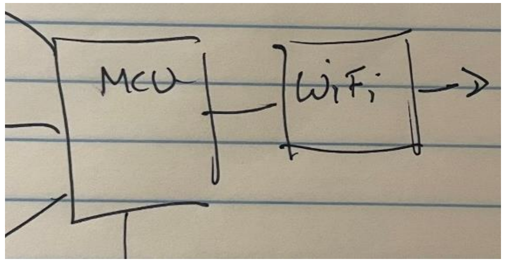
- What about battery?
  - ~infinite number of choices
  - Most common rechargeable choice these days is LiPo
  - Let's look at 18650 b/c it is used in 6.08
    - 3.7V nominal for single cell
    - Typical capacities 2200 mAh for Adafruit one

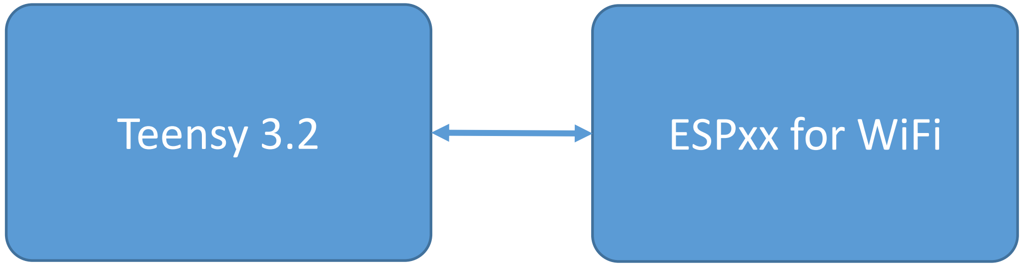
Assume 400 mA @ 3.3V consumption
Assume 2200 mAh capacity @ 3.3V [assume no energy savings for 3.7V to 3.3V conversion]

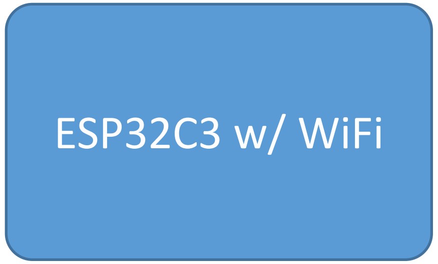
~5.5h if transmitting WiFi continuously…which we aren't going to be doing ➔ should be ok!

MAQS: MCU & comms design

- MCU & comms
  - Details in a few weeks, but…
- Factors influencing choice
  - MCU family
  - MCU w/ or w/o integrated WiFi
  - Peripherals to connect to display, sensors, etc.
  - Price & availability
  - RAM, Flash, etc. ← SW affects HW choice!
  - Use in other classes
  - Etc..
- Examined a few options
  - ATTiny, STM32 family
  - Teensy 4.0 + WiFi module [such as ESPxx]
    - NXP IMXRT1062DVL6 w/ ARM Cortex-M7
  - ESP32C3 [MCU + WiFi]
    - Espressif RISC-V Core







	Teensy4.0 + WiFi	ESP32C3
Cost	$20+ @ 10	$2.10 @ 10
Peripherals [I2C, SPI, ADC, UART, USB]	All that are likely needed	Same
Avail	yes	yes
Used in other classes?	6.200, 6.310	6.190
Number of parts	2 [+passives]	1 [+passives]

Partial list of considerations

MAQS: MCU & comms design

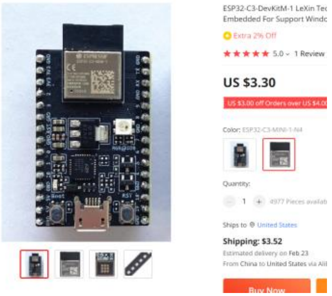
- MCU & comms programming
  - Originally wanted plain C/C++ (via ESP IDF)
- Question: how hard to get WiFi stack up-and-running on bare ESP32C3?
- Short-loop prototyping
  - Get ESP32 dev board
  - Try it out!
- Answer: pretty hard

  ➔ Incompatible with time-to-market
- Use Arduino libraries as needed [but no Arduino IDE!]

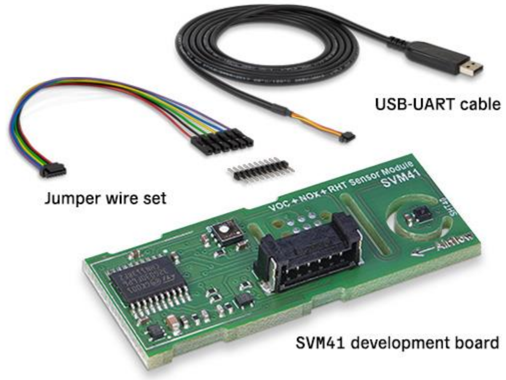
Prototyping for de-risking

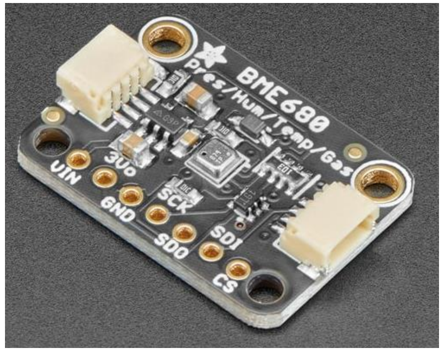
We have parts available, or can order for your team!

- Some HW aspects we can design and de-risk by research and modeling
- Many require prototyping
- Breakouts are fast/easy way to get started
  - With hardware design, firmware design
  - Breakouts are also useful inspiration when it is time to design your own board
  - **But breakouts only represent <1% of avail parts!**
  - Many/most of these are available substantially cheaper from China…but may take longer to arrive, may be *sketchy*
- There are also evaluation kits
- For other parts (such as SMT ICs), you can get adapte boards – just need to know package



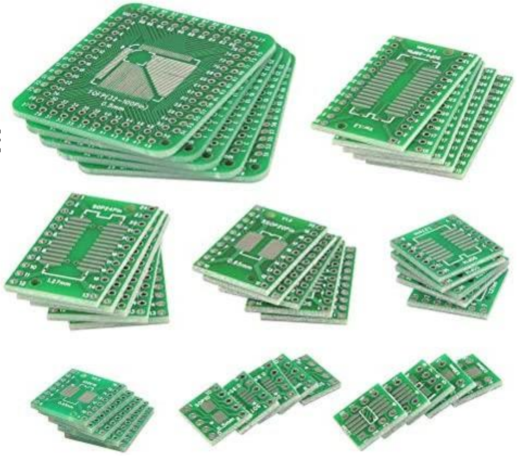
ESP32-C3 dev board
$3.30 @ 1





Adafruit Bosch BME680 breakout
$19 @ 1

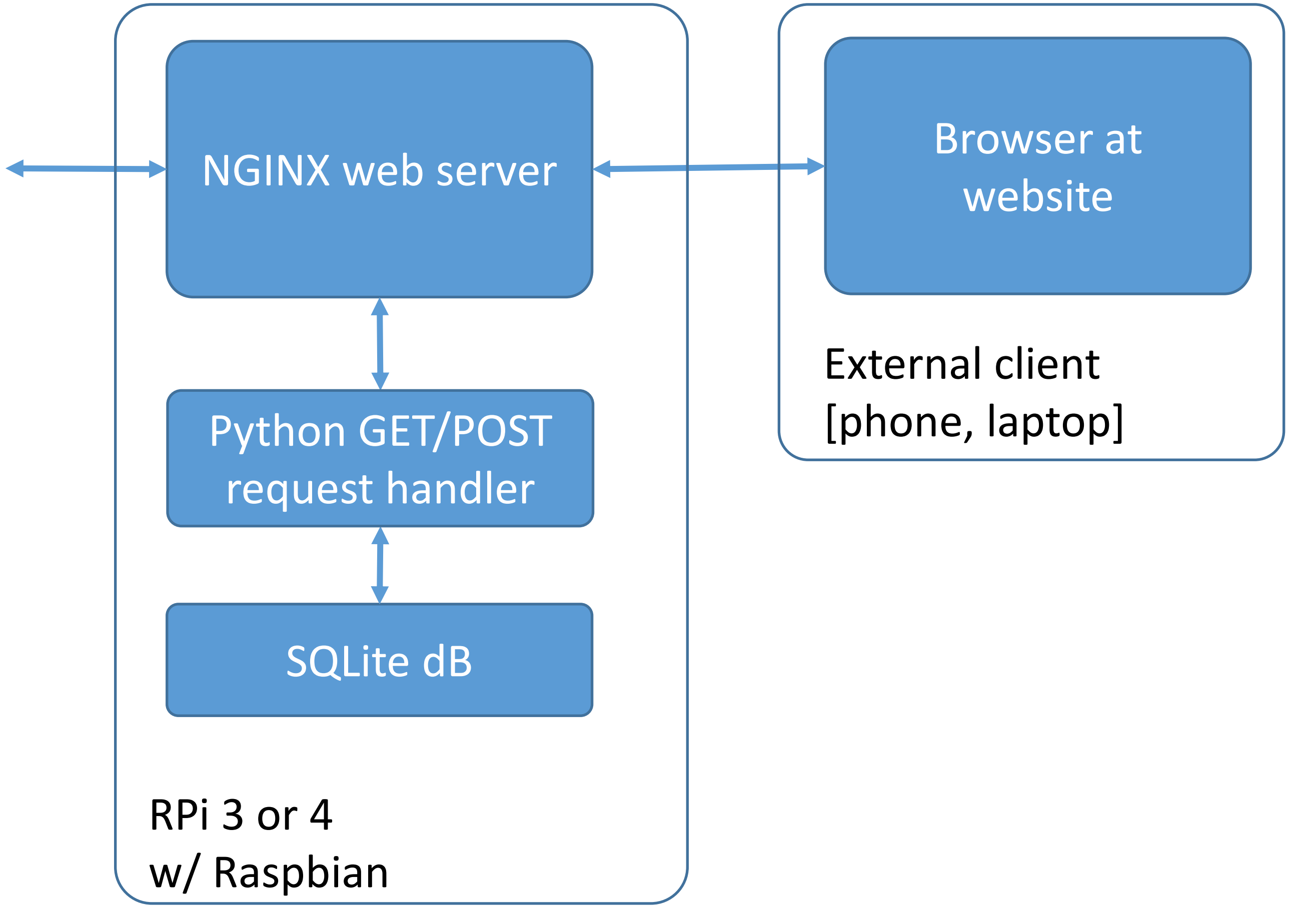
VEML7700-TT



SMD adapter boards

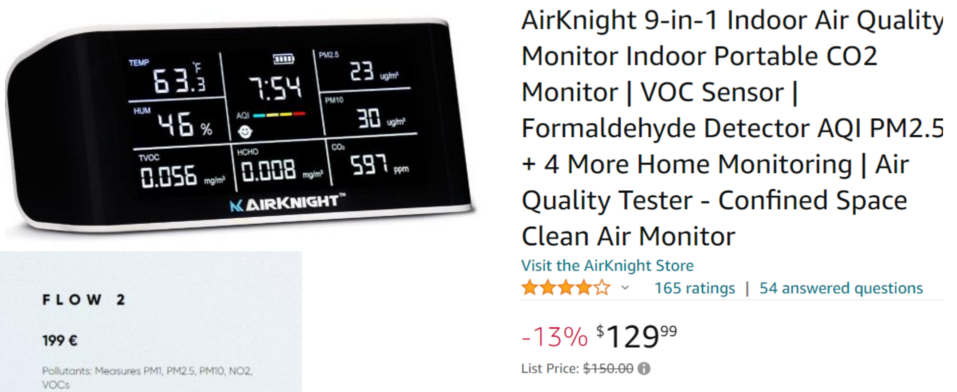
MAQS: server design

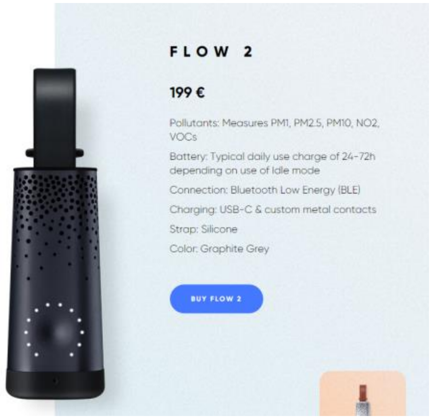
- Server-side architecture
- Server
  - Cloud provider, like AWS, GCP, etc.?
  - Virtual machines at MIT?
  - Physical machines at MIT?
- Database
  - SQLite, MySQL, etc.
- Web framework, front-end
  - Django, Flask, Plot.ly, etc.
- For all these components, pedagogical utility was the primary consideration
  - Expose students to inner workings, do not "black box" unless absolutely necessary
  - As simple as possible, yet authentic
  - Easy to manage and help debug
  - Potential to scale for future course offerings



MAQS: comms and display

- Many ways to view sensor data
  - Sensor node → on-board display
    - Could remove WiFi/comms entirely in some cases
    - Display adds cost to node
  - Sensor node → Phone→ view on app
    - Directly connect MAQS node to phone, such as Bluetooth
    - Avoid cloud server infrastructure – phone is ubiquitious
  - Sensor node → Phone/gateway → server → view on website/app
    - Use phone or other device to hand off data from sensor node to internet
    - Allows remote data retrieval, data fusion from multiple nodes/users
    - Need to maintain server, write apps for node, phone, and server
  - Sensor node → server → view on web
    - Not very common (due to power consumption)
      ...but becoming more common
    - Avoids need for app on phone, which is why we'll use it!





MAQS: specifications

Updated

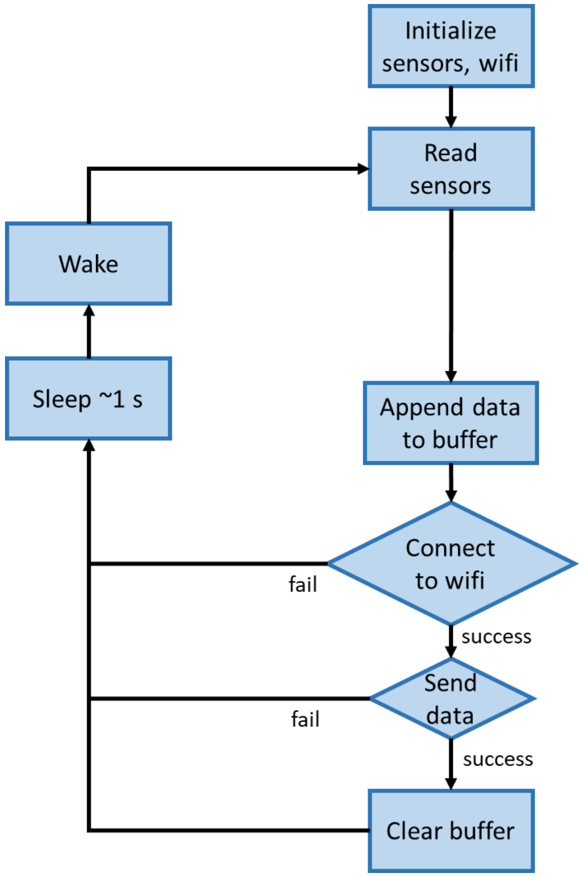
- Financial
  - BOM <= $100 for electronics components, PCB
  - BOM: TBD for enclosure , mechanical parts
  - Time to market: ~8 weeks
- Regulatory
  - FCC certification for WiFi radio module [part of ESP32C3 module]
- Industrial design
  - Weight: < 300 g [~2 iphones]
  - Size: <10 x 10 x 10 cm [kinda small]
  - Survive 12” drop onto table
  - Enclosure materials: 3DP plastics available in EDS, laser-cut plastics available in EDS
- Environmental
  - Operating temperature: 0 to 70°C [commercial temp range]
  - Humidity: 10 to 95% RH

- Engineering
  - Sensors
    - Air quality: PMS7003
      - Accuracy is not a provided spec!
    - T: 0 to 70 °C SHTC3-TR-10KS
      - Accuracy: +/- 2% RH
    - RH: 10 to 95% RH SHTC3-TR-10KS
      - Accuracy: +/- 0.2 °C
    - Measurement interval: <=10 min
    - Communications: I2C & UART
  - Compute
    - MCU: ESP32-C3-WROOM-02
    - Firmware in C/C++ w/ Arduino libraries as needed
    - How to program?
  - Comms
    - At least WiFi 802.11a/b/g/n 2.4 GHz
  - Energy management
    - LiPo battery – 18650 w/ JST-PH2 2-pin connectors
    - Lifetime between charging: >12 h
    - System voltage: 3.3 V
    - Charge from USB-micro or USB-C connection
  - Server
    - RPi 3 or 4, one for each student
    - Storage: 16 Gb SD card
    - SSH access for students, and staff
    - OS: Ubuntu, any recent LTS
    - Web server: NGINX
    - HTTPS GET/POST connections
    - DB: SQLite, at least one year of data stored

MAQS: specifications

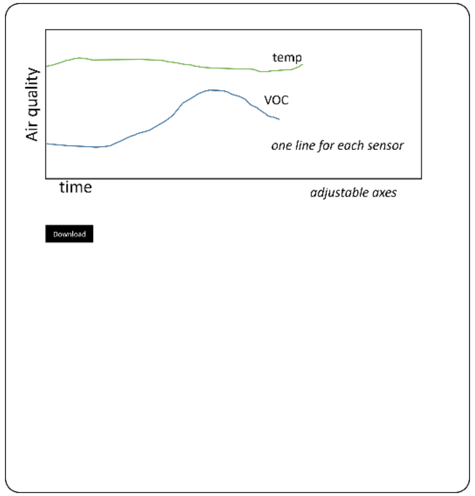
Updated

- Firmware



- Still to specify
  - How to reset?
  - Functions/APIs for sensor, display, WiFi
  - Data processing and what is transmitted
  - Sleep state, interval

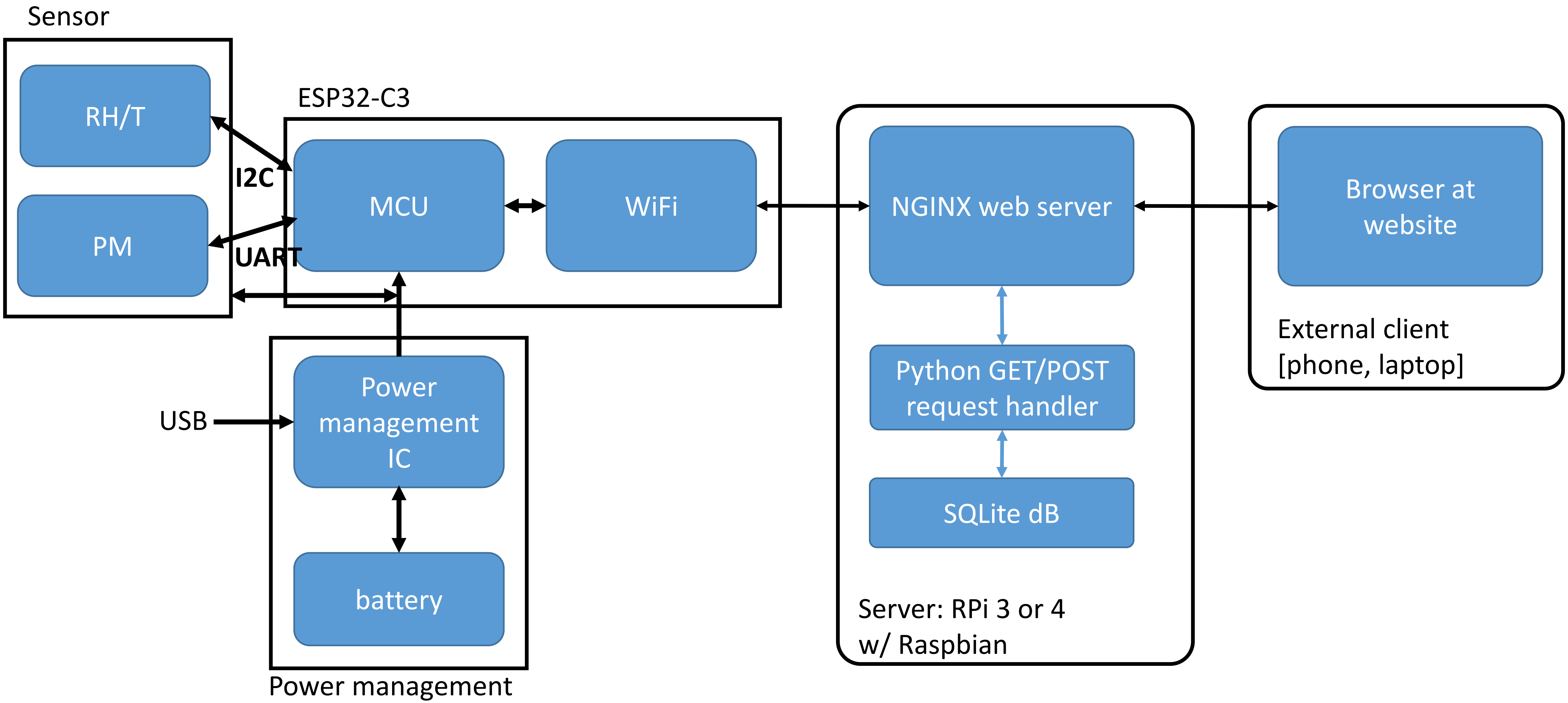
- Software [on server]
  - Store data perpetually in SQLite table, at least 1 yr of data stored
    - Fields: Index number, Timestamp, RH, T, AQ measurements
    - No location information transmitted (or stored)
  - Web front-end
    - Framework: Plot.ly



Web wireframe

MAQS: updated system diagram

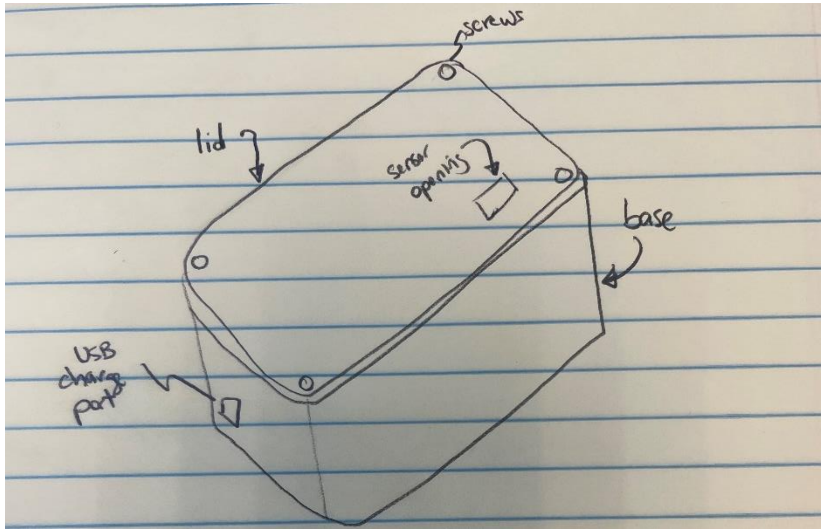
Updated



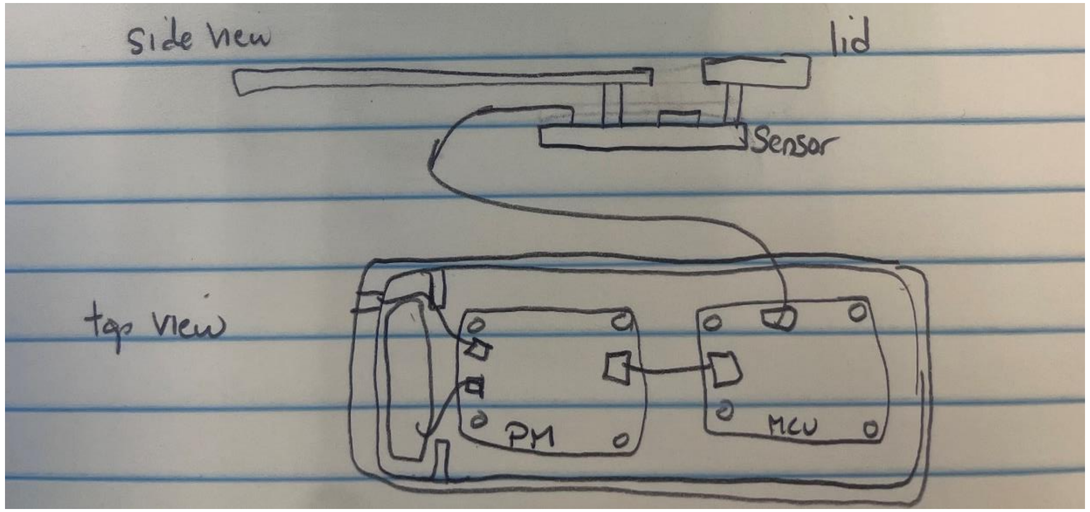
MAQS: from requirements to specifications

Updated

- Industrial design



External view



Internal view

3 subsystems ➔3 PCBs

- Still to specify
  - Status LEDs?
  - External buttons?
  - How to program?
  - What do connectors and cables look like, where are they situated?
  - How exactly to expose sensor to world

On to Sentimet!

Sentimet

MITOS

1. It should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
2. It should be able to measure how many people are in the area passing through (e.g., foot traffic) and lingering.***
3. It should operate without being connected to line voltage. ***
4. It should be portable and able to be set up by an average person in a variety of outdoor environments on the MIT campus, including on a tripod or attached to poles of various dimensions.***
5. It should be able to be physically attached to a HOBO MX2302A data logger.*
6. It should report faults, such as battery failure, falling, vandalism, etc.**
7. It should be as inexpensive as possible. *
8. Data from a sensor node should be able to be tied to a location.***
9. It should maintain privacy. ***
10. It should operate independently without user intervention for 2+ weeks.***
11. It should be rugged and able to withstand a summertime Boston-area environment (heat, rain, wind and curious people). ***
12. Multiple systems should be able to be used simultaneously. ***
13. The system should present the information on a dashboard (with real-time data outputs to a dashboard if possible), and also allow downloading of raw data.***

Miami-Dade

1. It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***
2. It should be able to measure how many people are waiting, and for how long.***
3. It should operate without being connected to line voltage. ***
4. It should be installable by a technician, and should be easy to set up.***
5. It should report faults, such as battery failure, falling, vandalism, etc.**
6. It should be as inexpensive as possible. *
7. Data from a sensor node should be able to be tied to a location.***
8. It should maintain privacy. ***
9. It should operate independently without user intervention for at least a month.***
10. It should be rugged and able to withstand shipping, setup, and operation in the Miami-Dade environment. ***
11. Multiple systems should be able to be used simultaneously. ***
12. The system should incorporate data from Swift.ly and present that information to the operator in a useful way. ***

Mostly the same…could make one overall system, or two slightly different systems

Sentimet specifications [1/2]

Ver. 1

- Financial
  - BOM:
  - COGS:
  - Time to market:
- Regulatory
- Industrial design
  - Weight:
  - Size:
  - Ruggedness:
  - Enclosure:
- Environmental
  - Operating temperature:
  - Humidity:
- Engineering
  - Sensors
  - Compute
  - Comms
  - Energy management
  - Server

Your Specs Google Doc should be broken up into these sections

Sentimet specifications [2/2]

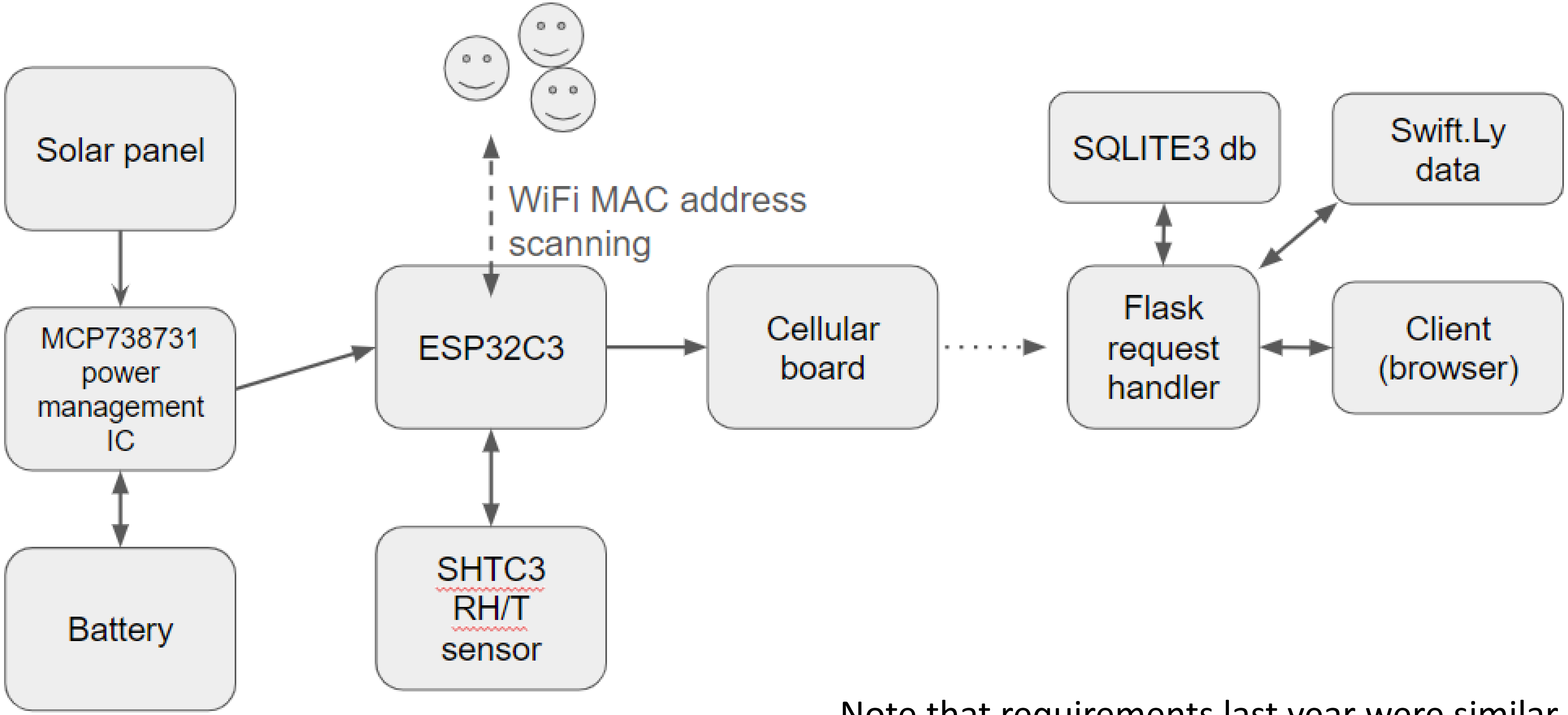
Updated

- Firmware

- Software [on server]
  - Back-end
  - Front-end

Your Specs Google Doc should be broken up into these sections

Sentimet 2023 system diagram



Note that requirements last year were similar **but not the same** as this year

Sensor subsystem

Requirements

One important change this year is a more elaborate sensor subsystem

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

Need to have: RH, T, occupancy
Like to have: sun exposure (lux), ground surface temperature
Nice to have: air pressure, air quality

Sensor subsystem

One important change this year is a more elaborate sensor subsystem

Requirements

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

Specifications

- **Engineering**
  - Sensors
    - T: SHTC3-TR-10KS
      - Range: 0 to 70 °C
      - Accuracy: +/- 2% RH
    - RH: SHTC3-TR-10KS
      - Range: 10 to 95% RH
      - Accuracy: +/- 0.2 °C

Here we have both the specifications (what we need to do) and the component (how we are going to do it) because it is so simple
But feel free to re-specify the component!

Sensor subsystem

One important change this year is a more elaborate sensor subsystem

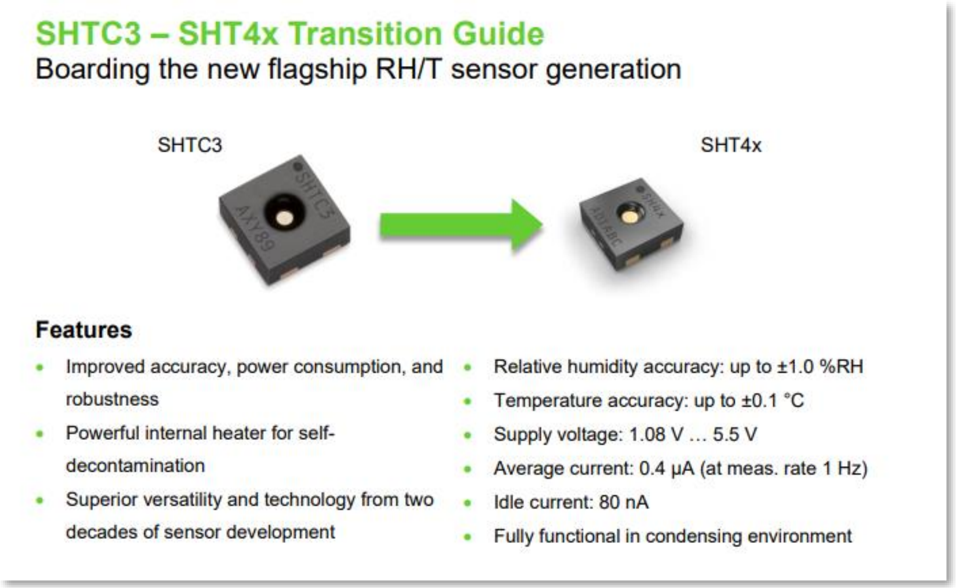
Requirements

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

Specifications

- **Engineering**
  - Sensors
    - T: SHTC3-TR-10KS
      - Range: 0 to 70 °C
      - Accuracy: +/- 2% RH
    - RH: SHTC3-TR-10KS
      - Range: 10 to 95% RH
      - Accuracy: +/- 0.2 °C

But even here it might be worth looking around a bit…



Sensor subsystem

One important change this year is a more elaborate sensor subsystem

Requirements

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

Specifications

- **Engineering**
  - Sensors
    - T: SHTC3-TR-10KS
      - Range: 0 to 70 °C
      - Accuracy: +/- 2% RH
    - RH: SHTC3-TR-10KS
      - Range: 10 to 95% RH
      - Accuracy: +/- 0.2 °C
    - Surface temperature
    - Sun exposure (lux)
      - Range: 1 lux (night) to 1e5 lux (direct sun)
    - Air pressure

Here we have a specification from research, but no solution identified

Sensor subsystem

One important change this year is a more elaborate sensor subsystem

Requirements

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

Specifications

- **Engineering**
  - Sensors
    - T: SHTC3-TR-10KS
      - Range: 0 to 70 °C
      - Accuracy: +/- 2% RH
    - RH: SHTC3-TR-10KS
      - Range: 10 to 95% RH
      - Accuracy: +/- 0.2 °C
    - Surface temperature
    - Sun exposure (lux)
      - Range: 1 lux (night) to 1e5 lux (direct sun)
    - Air pressure

Testing

Inherit spec from performance of part
Noting that this does not ensure that entire system meets spec!

Sensor subsystem

Requirements

One important change this year is a more elaborate sensor subsystem

- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

What does this mean?

Sensor subsystem

Requirements

One important change this year is a more elaborate sensor subsystem

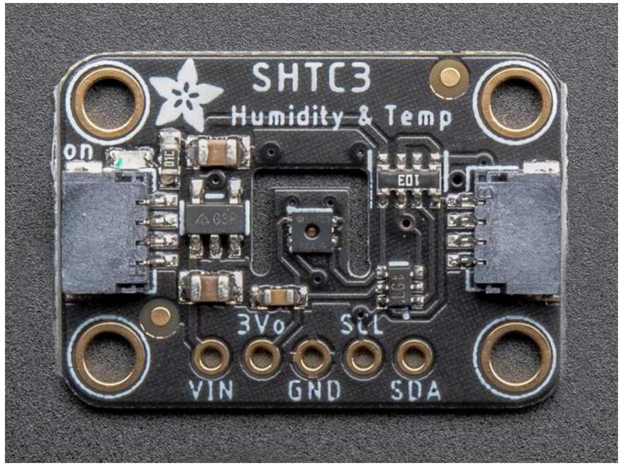
- MITOS: it should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
- M-D: It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***

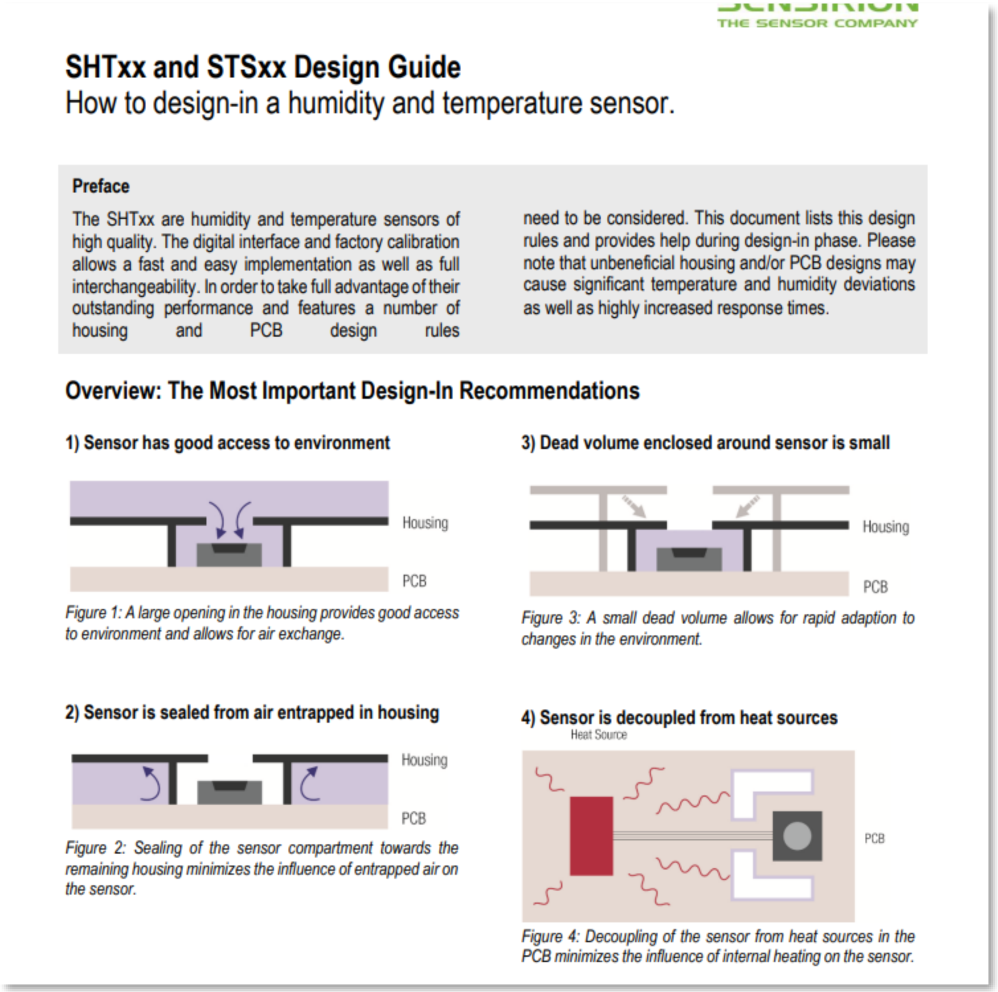
What does this mean?

Need to measure fast enough to capture changes in quantities
For weather: ?
For occupancy: ?

Sensor dynamics

- Not only does the sensor need to be fast enough, but the system around it needs to be design appropriately
- How quickly will changes in (outdoor) ambient air temperature be reflected in the SHTC3 sensor?
  - This depends on thermal coupling
- To estimate dynamics, you can
  - Use design guides from companies
  - Use models
    - Lumped-element thermal models
  - Test with prototypes





THE SENSOR COMPANY

SHTxx and STSxx Design Guide

How to design-in a humidity and temperature sensor.

Preface

The SHTxx are humidity and temperature sensors of high quality. The digital interface and factory calibration allows a fast and easy implementation as well as full interchangeability. In order to take full advantage of their outstanding performance and features a number of housing and PCB design rules need to be considered. This document lists this design rules and provides help during design-in phase. Please note that unbeneficial housing and/or PCB designs may cause significant temperature and humidity deviations as well as highly increased response times.

Overview: The Most Important Design-In Recommendations

1) Sensor has good access to environment

Housing
PCB

Figure 1: A large opening in the housing provides good access to environment and allows for air exchange.

2) Sensor is sealed from air entrapped in housing

Housing
PCB

Figure 2: Sealing of the sensor compartment towards the remaining housing minimizes the influence of entrapped air on the sensor.

3) Dead volume enclosed around sensor is small

Housing
PCB

Figure 3: A small dead volume allows for rapid adaption to changes in the environment.

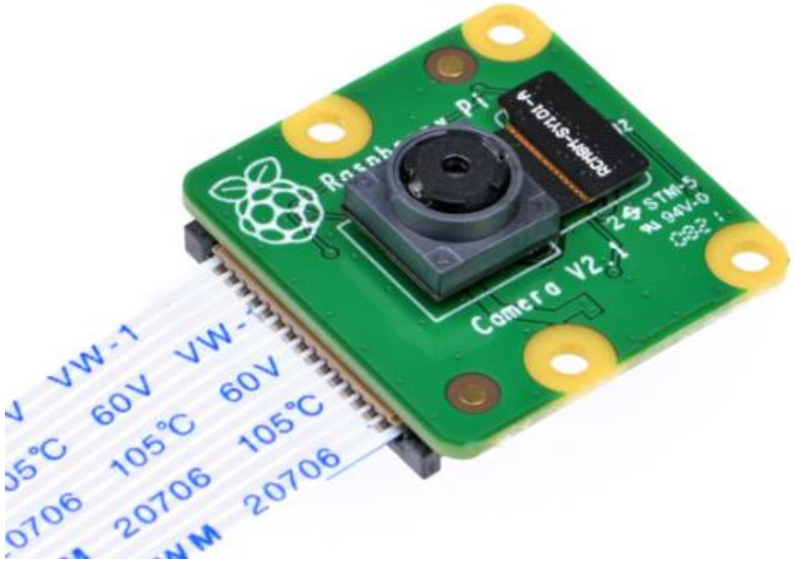
4) Sensor is decoupled from heat sources

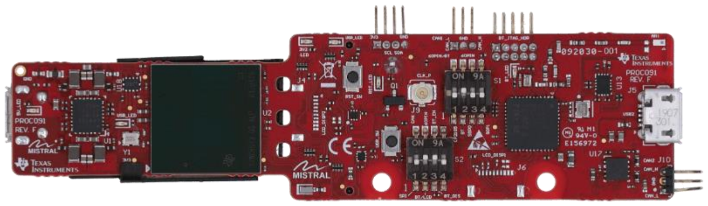
Heat Source
PCB

Figure 4: Decoupling of the sensor from heat sources in the PCB minimizes the influence of internal heating on the sensor.

Sensor subsystem: occupancy

- This is going to be challenging
- Some concepts explored last year
  - Camera
  - mmWave radar
  - User interaction (buttons or QR code on pole)
  - WiFi or BLE beacon counting

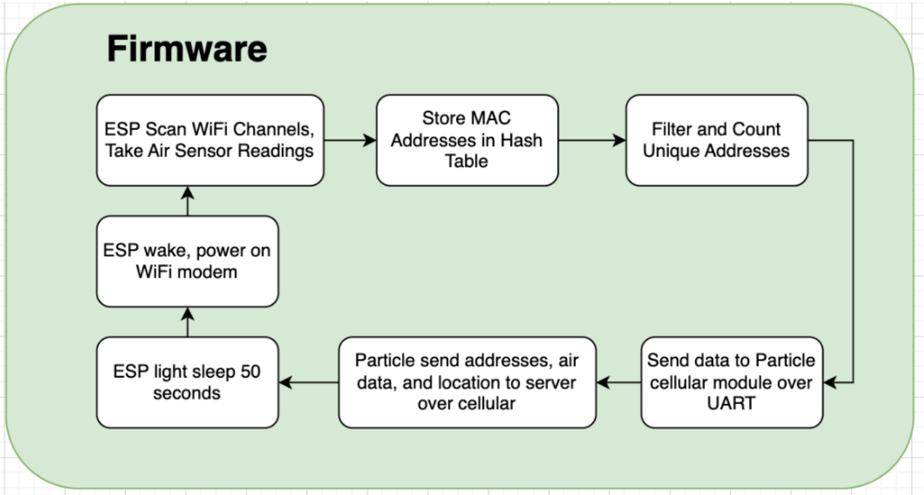




TI mmWave sensor board

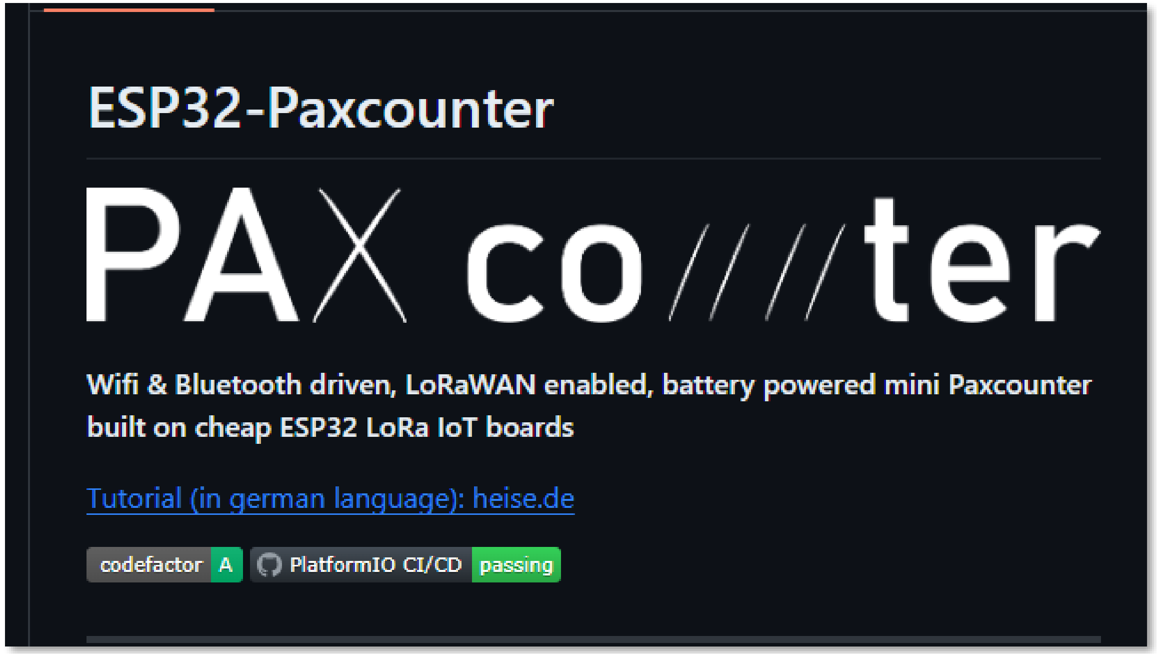
Sensor subsystem: occupancy

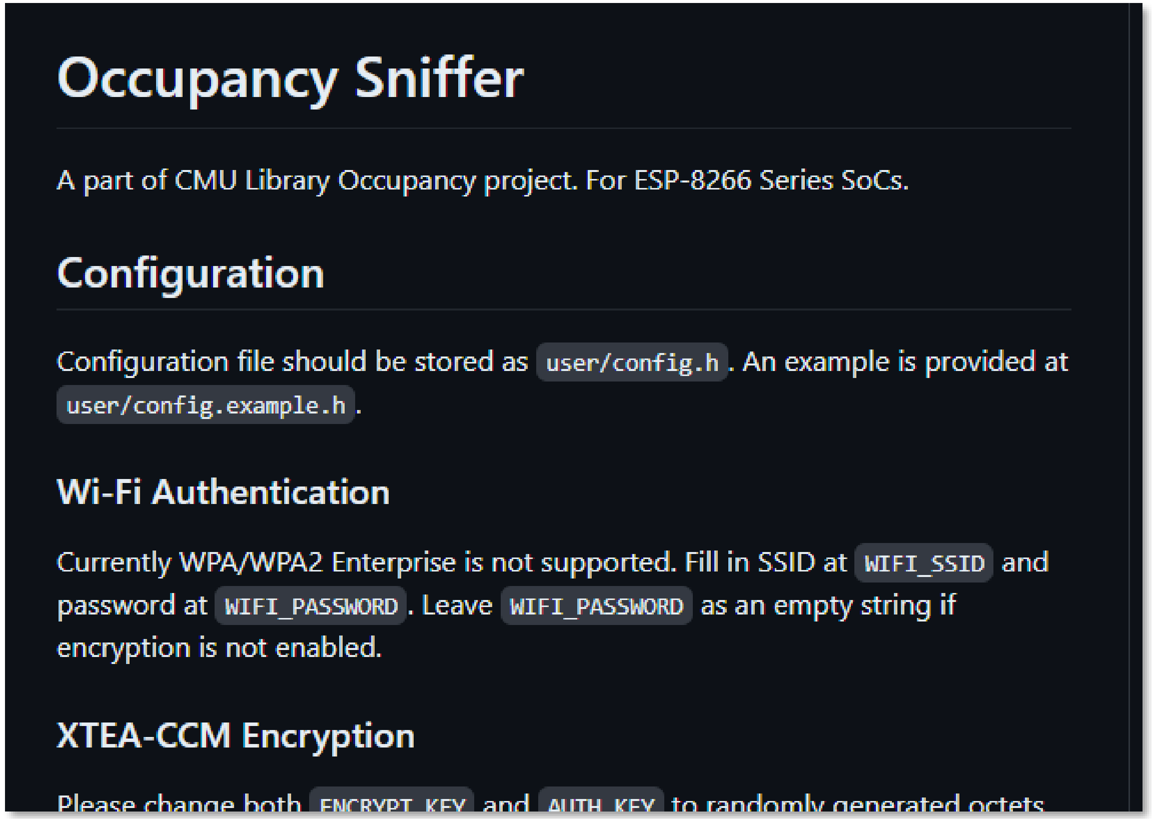
- WiFi or BLE beacon counting
  - For your phone to know which access points (for WiFi) or other devices (for BLE) are around, it sends advertisements/beacons out periodically announcing its presence
    - The beacon will include the device’s MAC address, which is unique
  - APs can see and record these beacons
    - They also measure the signal strength of the received beacon, which gives a “rough” indication of distance
  - So counting MAC addresses gives an estimate of number of devices and therefore estimate of number of people
  - These systems are used commercially



Sensor subsystem: occupancy

- WiFi or BLE beacon counting
  - Easy conceptually, tricky to implement and test
  - Some prior examples you can leverage (but not copy!)





Sensor subsystem: occupancy

- WiFi or BLE beacon counting
  - Why is testing difficult?
  - How to get ground truth data?
    - True ground truth is number of people → how to get that?
    - But even getting "true number of devices" is not trivial
  - Going from number of MACs (w/ RSSI) to number of people in area is going to take some work

Think about this carefully and be ready for extensive testing in March

Power management

It should operate independently without user intervention for 2+ weeks.***

It should operate without being connected to line voltage. ***

It should operate independently without user intervention for at least a month.***

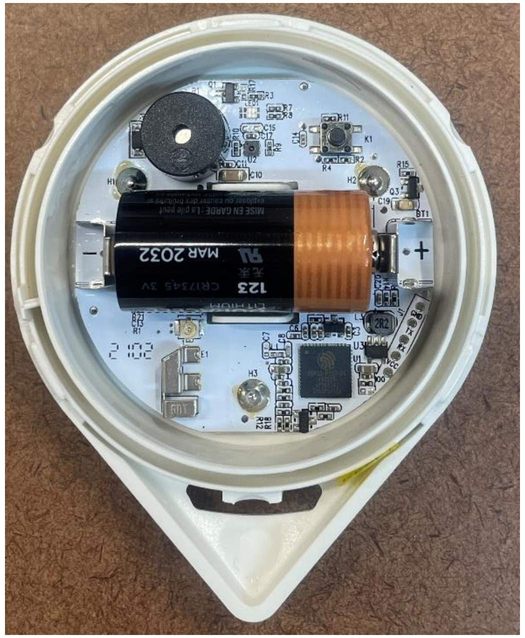
It should operate without being connected to line voltage. ***

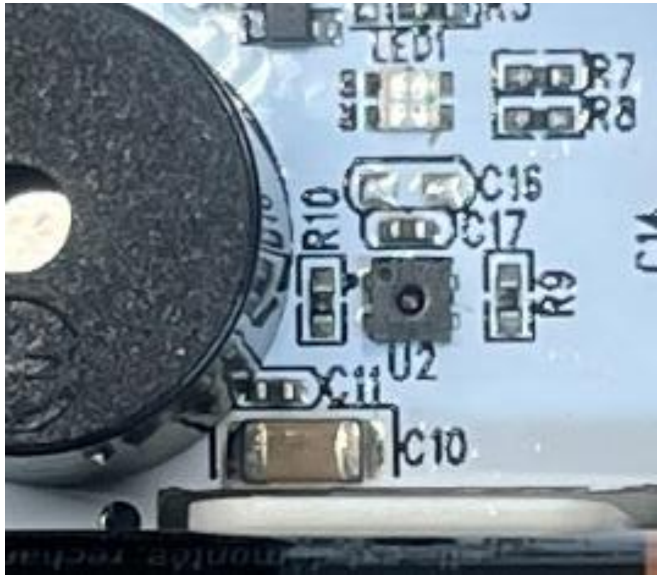
- Implies battery-powered
- Last year teams used solar panel to recharge battery
- That’s not the only potential design

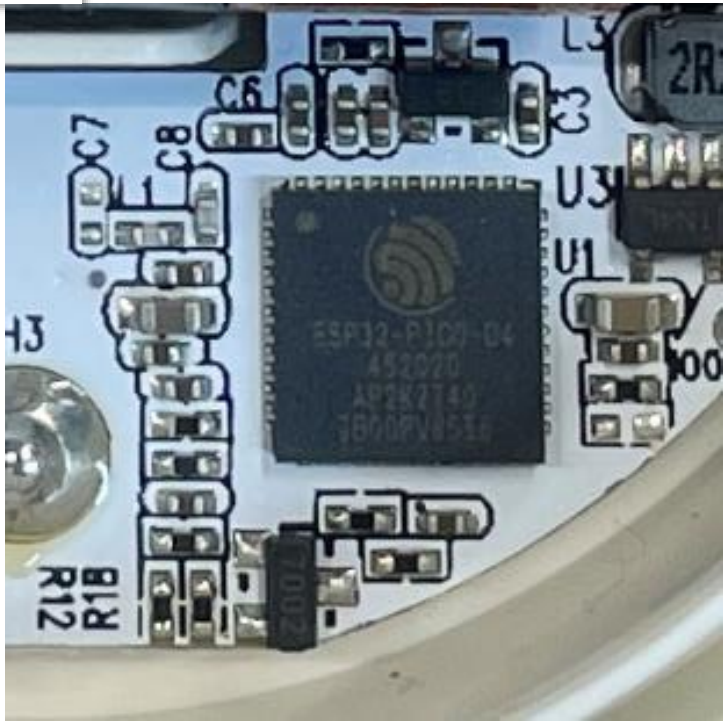
Power management

- Flo by Moen Smart water detector
  - Leak detector, also RH/T

- Smartphone app available for iOS and Android that provides customizable alerts for:
  - -Moisture Detection through base unit or included remote sensing disc
  - -Humidity Range: customizable between 0-100% Relative Humidity (RH)
  - -Room Temperature: customizable between 0 to 140 deg F / -18 to 60 deg C
  - -Battery Level: customizable between 0-100%
- Automatic water shutoff feature: if a detector senses moisture, it can trigger the smart water shutoff to close (only available if the Flo Smart Water Monitor and Shutoff is installed on the main water supply line)
- Battery type: CR123A Lithium Cell (lasts up to 2 years, included with detector)

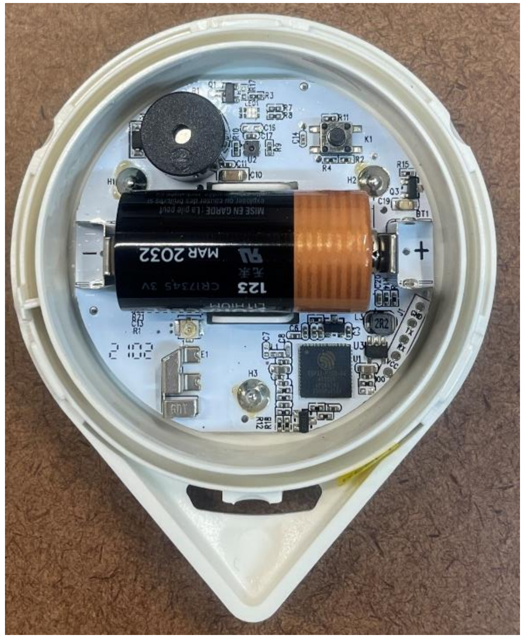


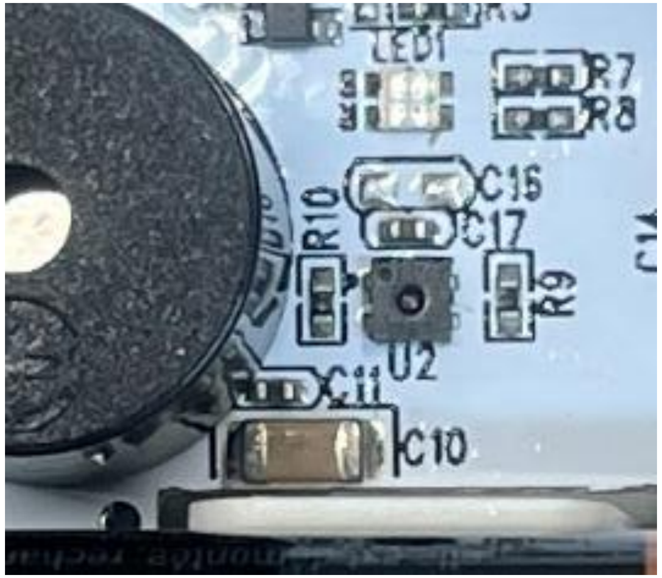


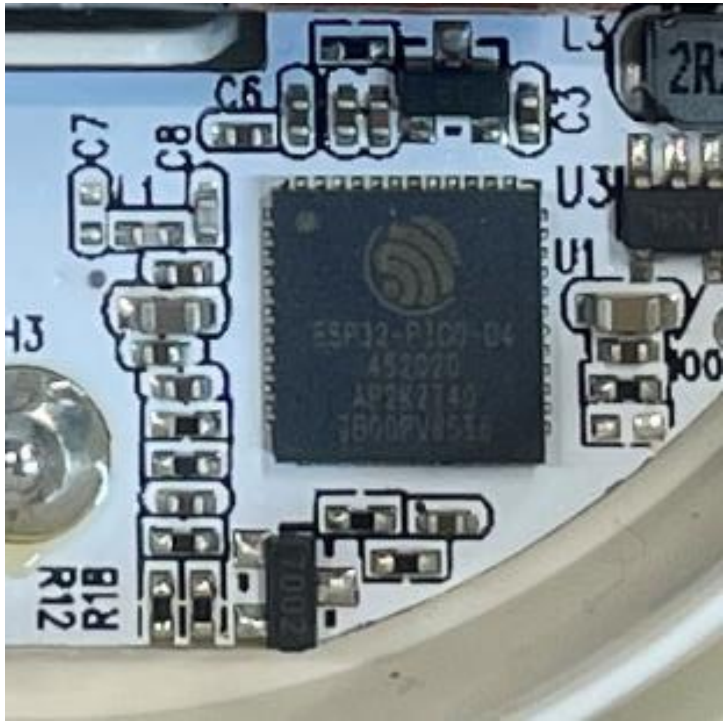


Power management

- Flo by Moen Smart water detector
  - Leak detector, also RH/T
  - Connects to WiFi, uploads every few minutes
  - Battery is 3V@1470 mAh, lasts around 1 y in my house







Power management

- Solely use battery power
  - Probably lower BOM cost
  - Fewer parts to break, simpler industrial design
  - More work on firmware
  - Limited lifetime (but might be long enough)
- Solar charger
  - Likely higher cost
  - More complicated industrial design
  - FW is simpler
  - Lifetime potentially indefinite

Sentimet

MITOS

1. It should measure the local weather, at least temperature and humidity and ideally also sun exposure and ground surface temperature and air pressure, all with dynamics appropriate for the use case.***
2. It should be able to measure how many people are in the area passing through (e.g., foot traffic) and lingering.***
3. It should operate without being connected to line voltage. ***
4. It should be portable and able to be set up by an average person in a variety of outdoor environments on the MIT campus, including on a tripod or attached to poles of various dimensions.***
5. It should be able to be physically attached to a HOBO MX2302A data logger.*
6. It should report faults, such as battery failure, falling, vandalism, etc.**
7. It should be as inexpensive as possible. *
8. Data from a sensor node should be able to be tied to a location.***
9. It should maintain privacy. ***
10. It should operate independently without user intervention for 2+ weeks.***
11. It should be rugged and able to withstand a summertime Boston-area environment (heat, rain, wind and curious people). ***
12. Multiple systems should be able to be used simultaneously. ***
13. The system should present the information on a dashboard (with real-time data outputs to a dashboard if possible), and also allow downloading of raw data.***

Miami-Dade

1. It should measure the "heat experience" at each bus stop, at least temperature and humidity, but also could include air quality, all with dynamics appropriate to the use case.***
2. It should be able to measure how many people are waiting, and for how long.***
3. It should operate without being connected to line voltage. ***
4. It should be installable by a technician, and should be easy to set up.***
5. It should report faults, such as battery failure, falling, vandalism, etc.**
6. It should be as inexpensive as possible. *
7. Data from a sensor node should be able to be tied to a location.***
8. It should maintain privacy. ***
9. It should operate independently without user intervention for at least a month.***
10. It should be rugged and able to withstand shipping, setup, and operation in the Miami-Dade environment. ***
11. Multiple systems should be able to be used simultaneously. ***
12. The system should incorporate data from Swift.ly and present that information to the operator in a useful way. ***

Mostly the same…could make one overall system, or two slightly different systems

Sentimet specifications [1/2]

Ver. 1

- Financial
  - BOM:
  - COGS:
  - Time to market:
- Regulatory
- Industrial design
  - Weight:
  - Size:
  - Ruggedness:
  - Enclosure:
- Environmental
  - Operating temperature:
  - Humidity:
- Engineering
  - Sensors
  - Compute
  - Comms
  - Energy management
  - Server

Your Specs Google Doc should be broken up into these sections

Sentimet specifications [2/2]

Updated

- Firmware

- Software [on server]
  - Back-end
  - Front-end

Your Specs Google Doc should be broken up into these sections