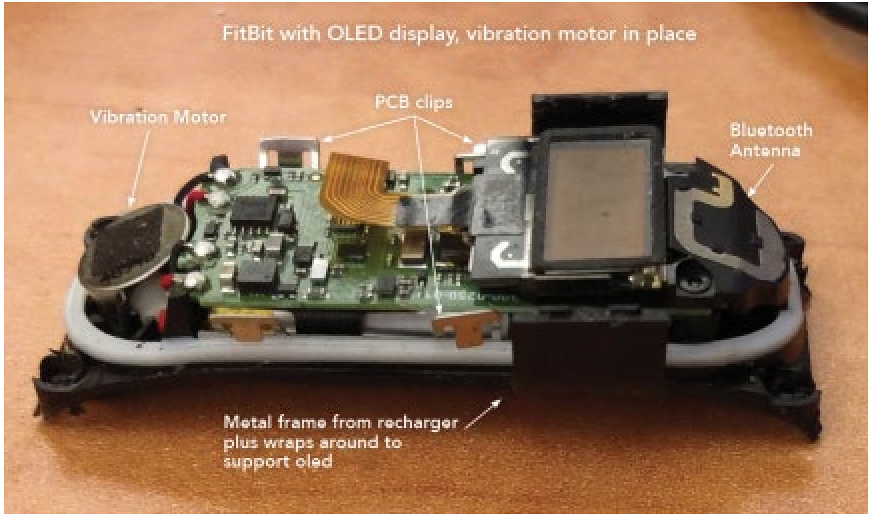
MICRO-333
Wireless Sensors DLL

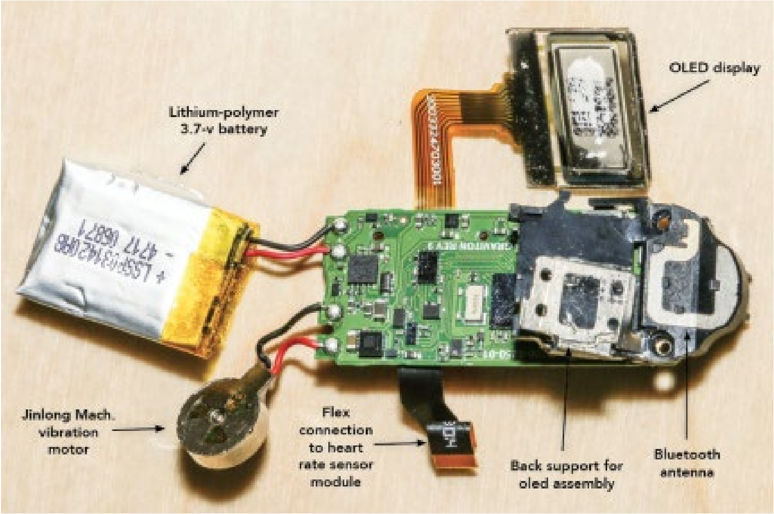
Vivek Subramanian
Professor, Microengineering
EPFL

Speaker

Modern Integrated Systems

- Silicon is king for integrating intelligence
- Lots of sensors and physical world interfaces
- Packaging and integration is challenging





So what are we going to do in the DLL?

Using a commercial Bluetooth MCU ; we will integrate with a sensing interface

You design a sensor daughter-board of your choice, design the PCB, have it fabricated, and assemble the components on it using assembly tools in the DLL building.

We will 3D print and assemble a package

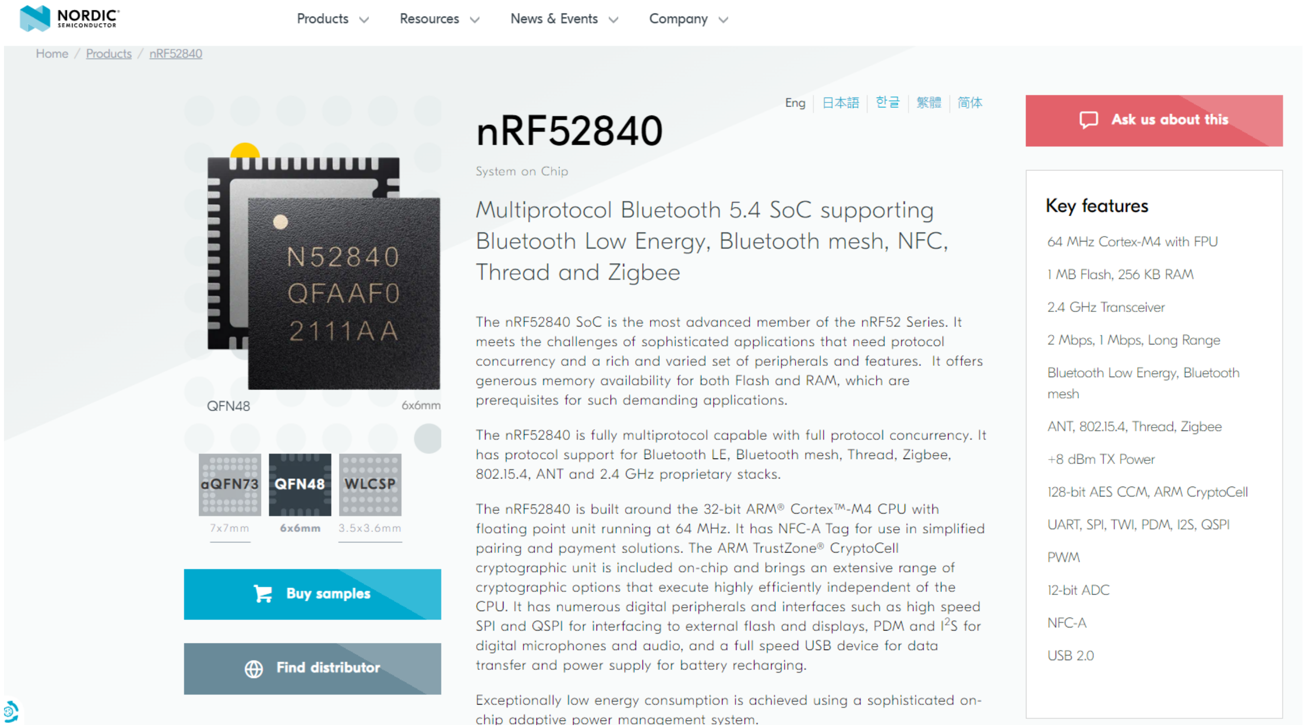
You will test it using a phone

Result: A simple integrated wireless sensor

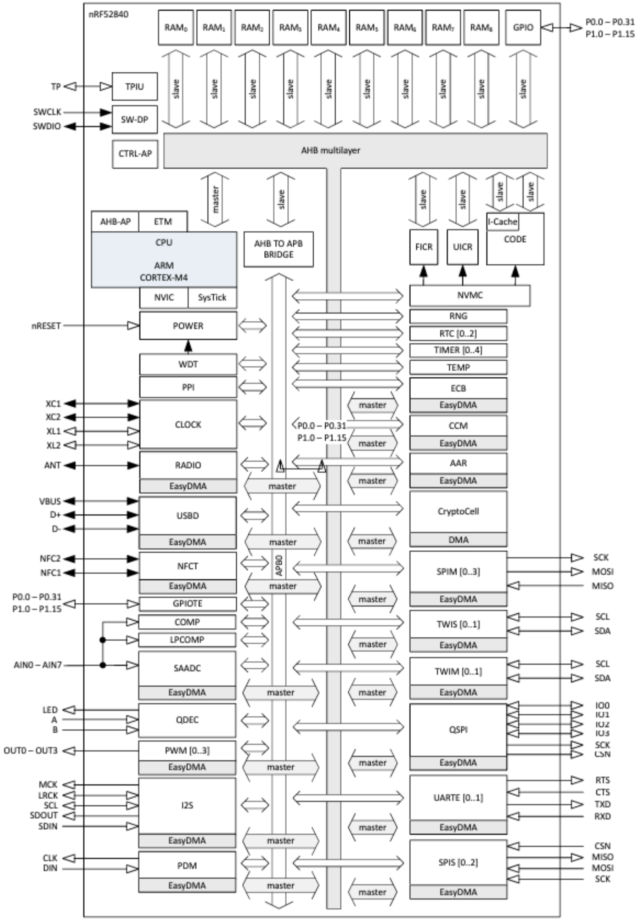
EPFL

Details: MCU

- The MCU we are using is the nRF52840 in a QFN48 package: https://www.nordicsemi.com/Products/nRF52840

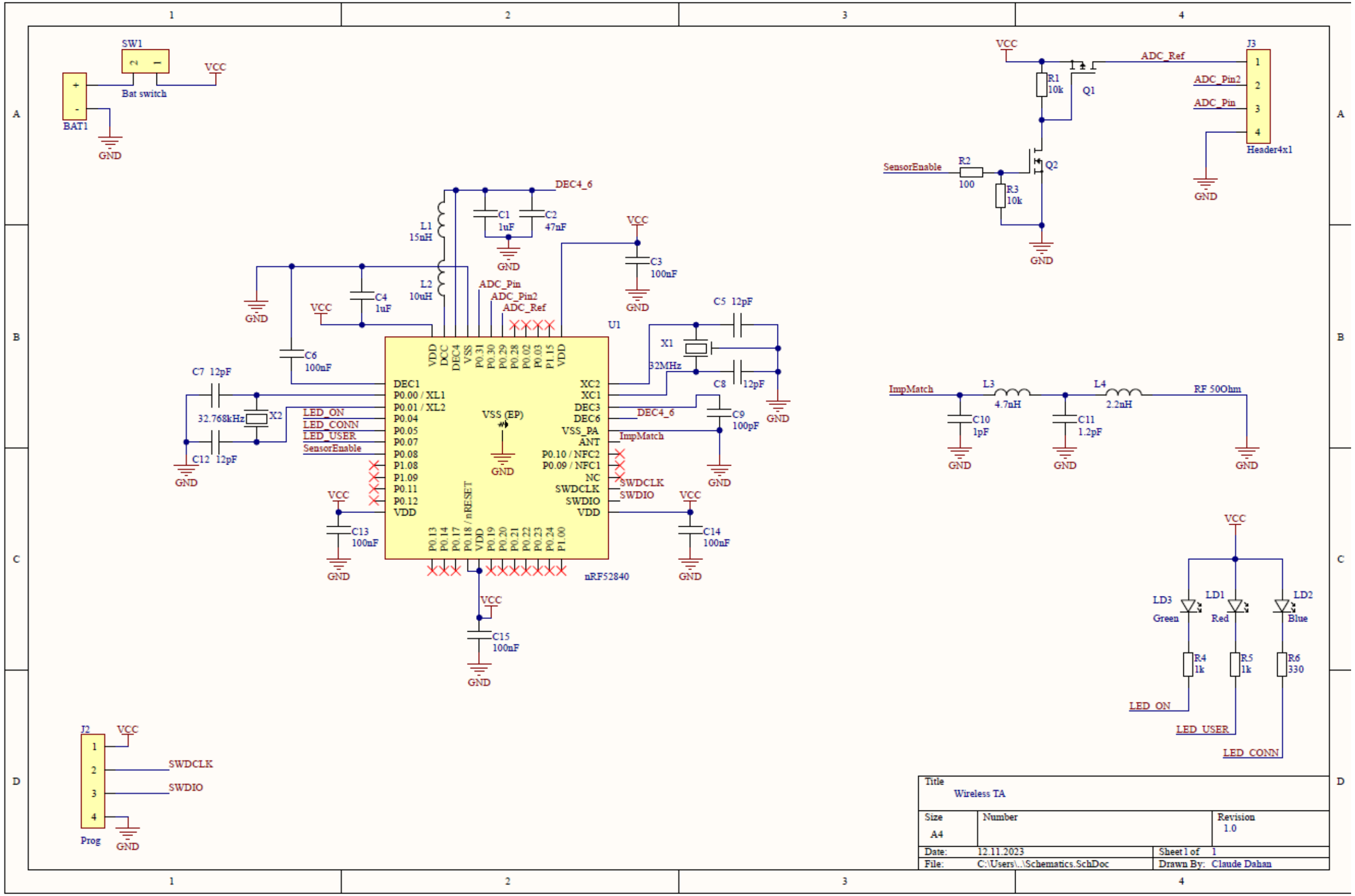


Details: MCU

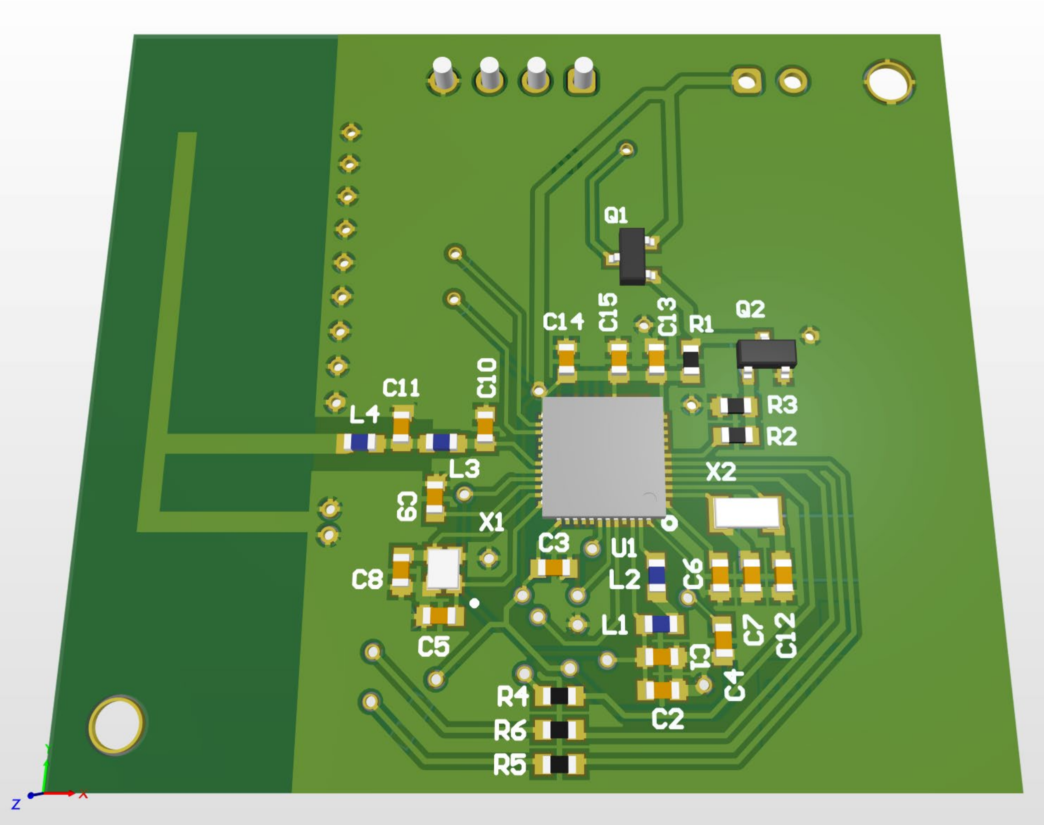


- Key points for us:
  - BLE radio integrated
    - Just need to add XTAL, antenna, some passive components, etc.
  - Analog to digital converter inputs are available to read an input voltage (e.g., from a sensor) and send the data to a phone.

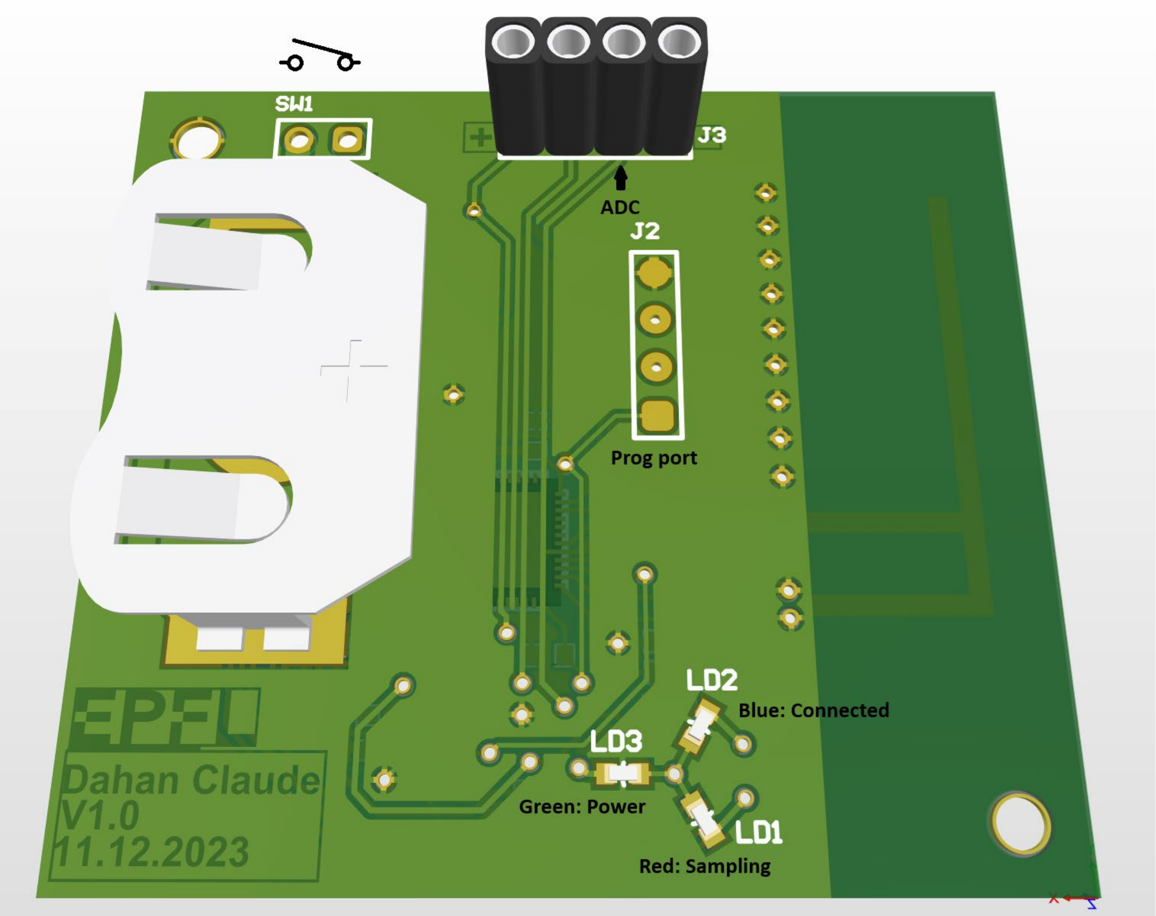
Schematic of our main board



Board component view – front side

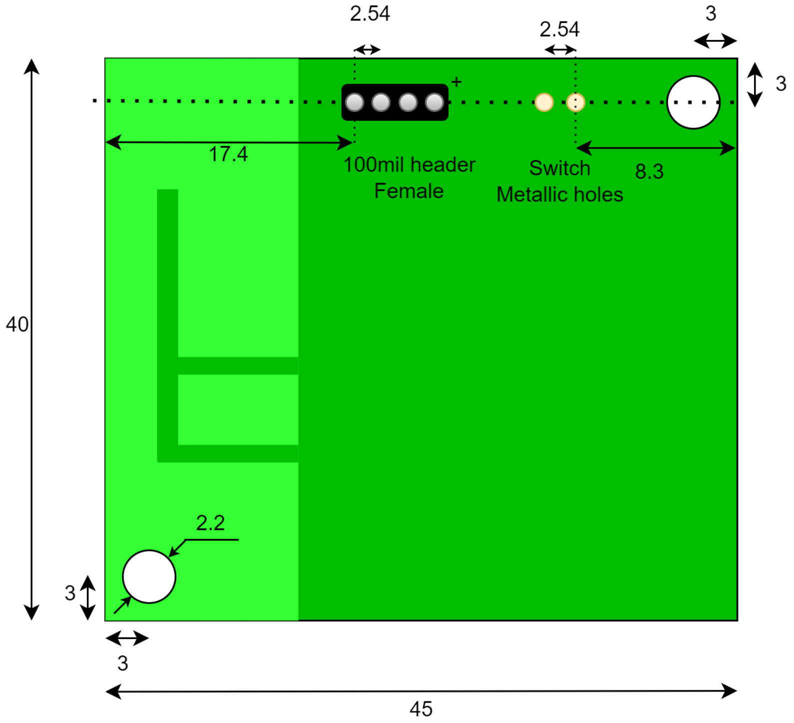


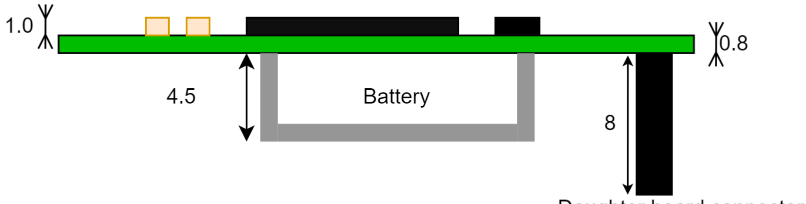
Board component view – back side



Main PCB Dimensions

in mm





Your tasks:

- Step 1 (Over the next week): Choose an application that is interesting to your group
- Step 2: Design the daughter-board circuitry
- Step 3: Design a PCB, and submit it for fabrication
- Step 4: Populate the PCB with components using tools in the DLL building
- Step 5: Test it and make sure it works
- Step 6: Design a package
- Step 7: Build it yourself or have it built using the 3D printing workshop
- Step 8: Assemble everything
- Step 9: Fully characterize the sensor performance using metrics that make sense
- Step 10: Prepare a presentation along with a short report, or prepare a more detailed report.

Steps 1 and 2: The sensor

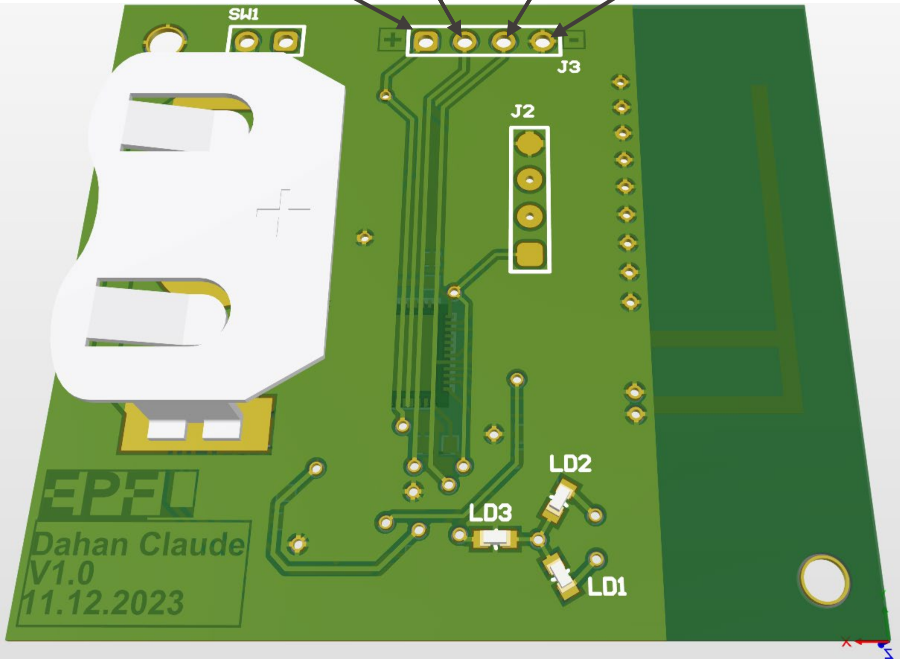
Standard 100mil female connector:

Differential mode reference

ADC input → Sensor output

Vcc

GND



Identify a sensor component that you can use for an application of your choice

16 bit value transmitted after sampling:

Single ended mode:
(default, settable by Bluetooth)

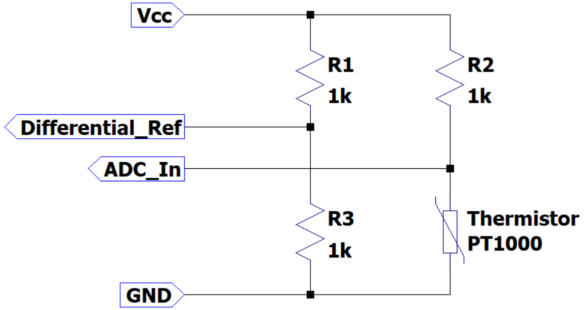
$$Result = 2^{14} \cdot \frac{ADC\_Input}{Vcc}$$

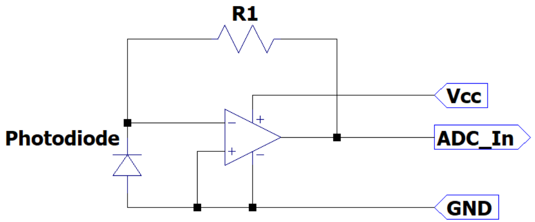
Differential mode:

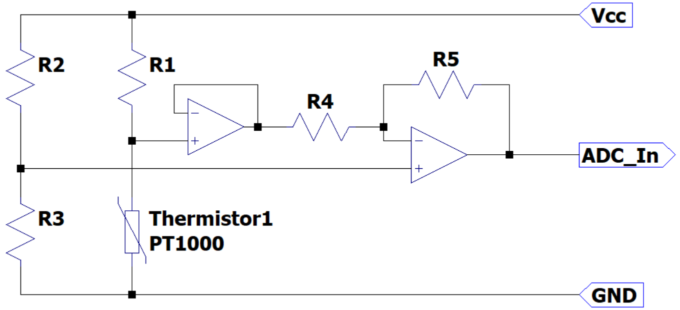
$$Result = 2^{13} \cdot \frac{ADC_{Input} - Differential\_ref}{Vcc}$$

Result going from 0 to 0x3FFF in both cases, as differential mode can be both positive or negative

Steps 1 and 2: The sensor – some examples







EPFL

Steps 3: The Daughterboard

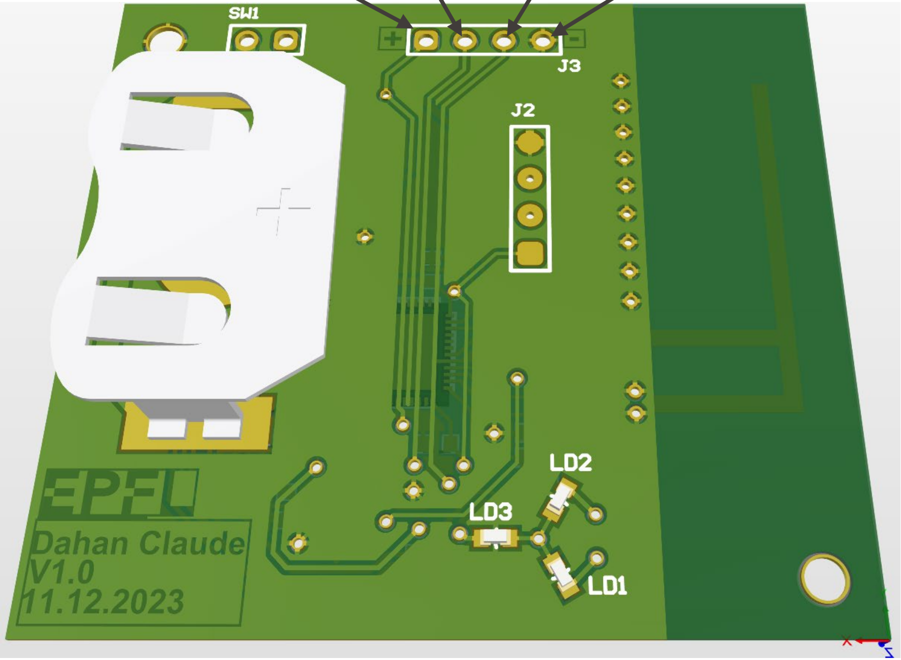
Standard 100mil female connector:

Differential mode reference

ADC input → Sensor output

Vcc

GND

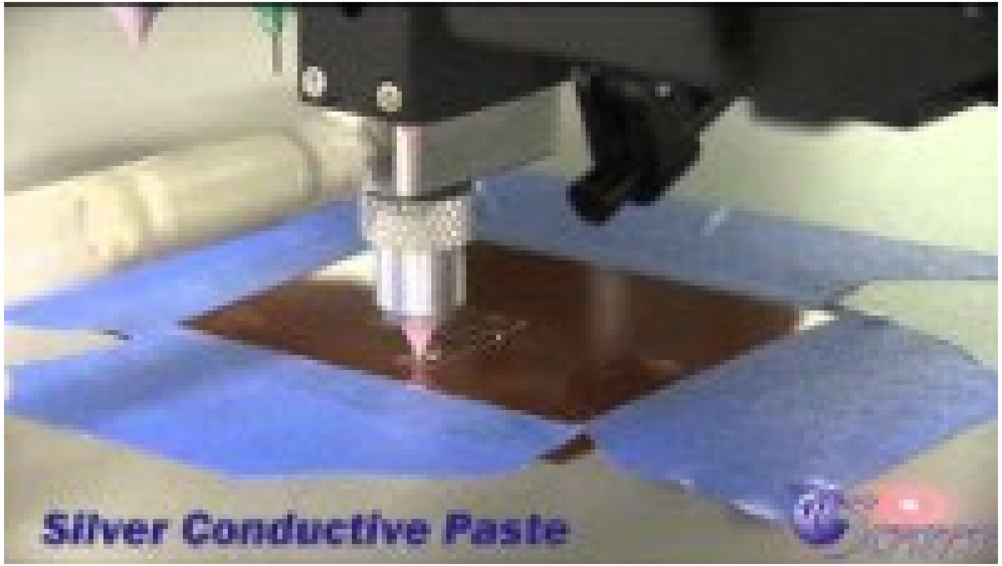


Design a daughterboard to connect to the pins, either directly or via a cable.

Think about the package while doing this design

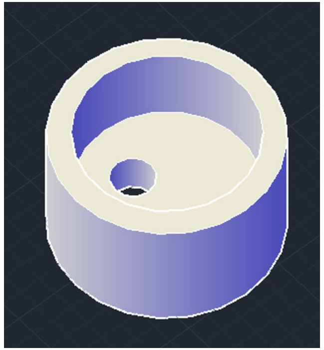
Step 4: The PCB Assembly Process

- You will use a pick and place assembly tool to populate the components on your PCB.



Step 6: Package Design

- You will use SolidWorks, AutoCAD, etc. to generate an STL file for the package. We want you to be as creative as possible and think about the implications of your choices. Therefore, we will not be giving you much guidance here, but will give you feedback on your final design



- You will submit the design via moodle for TA feedback.
- Once the TA is satisfied, we will fabricate the package for you via a 3D printing workshop.

Detailed Schedule

Date	Task
21.2.2025	Last date to drop the class
22.2.2025	Groups of 4 will be assigned randomly. I will not consider requests to pair you with a specific set of partners. However, if you have problems during the semester, you can reach out to me and I will try to help out.
28.2.2025	Present a short (5-10 minutes) summary of your proposed sensor in class (14h-18h in AAC014) for my feedback / approval.
7.3.2025	Lecture on CAD for package design 14h-17h in AAC014.
14.3.2025	Schematic review 14h-18h (schedule will be announced after sensor approval next week) in AAC020
28.3.2025	Complete PCB layout and submit for fabrication. Submit all parts requests to us so we can order the parts for you.
11.4.2025-23.5.2025	Microassembly sessions (3 groups per session; schedule will be announced after PCB submission, based on parts availability.
2.5.2025	Submit package STL file to us if you want us to fabricate it for you.
Weeks of May 26 and June 2	Scheduled short presentations, along with short report or submit long report

How to split the work

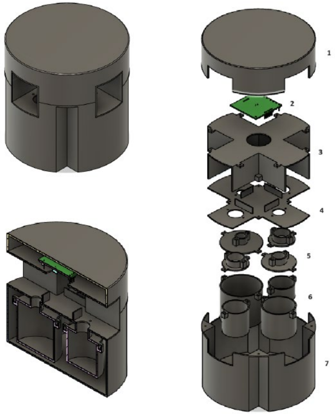
- Each group has 4 students:
- I suggest that you split work as follows:
  - Schematic design: Student 1
  - PCB layout: Student 2
  - Micro-assembly: All
  - Package design: Student 3
  - Testing: Student 4
  - Reporting: All
- All members of a group are NOT guaranteed the same grade. I will ask you questions to see evaluate your understanding and involvement in the project.

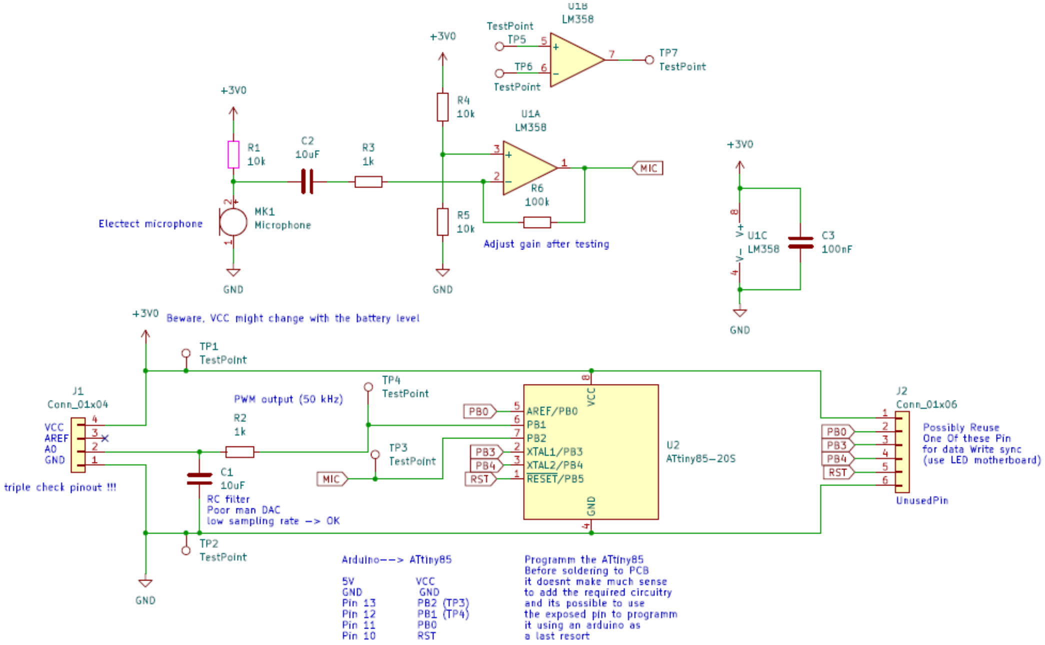
Availability for advising

- I will be available to meet by zoom or in-person most weeks on Friday afternoons.
- I can also meet with you at other times by appointment
- Claude can meet with you by appointment
- Other TAs will be present in the DLL during micro-assembly sessions

Examples from last year

Acoustic Anemometer





Examples from last year

Angle measurement sensor

2.2 Hall effect sensor

This idea was suggested by one of the TA . The idea is to use a Hall effec sensor with a magnet and have the magnet rotating, relatively to the sensoı from its north pole to south and the other way around.

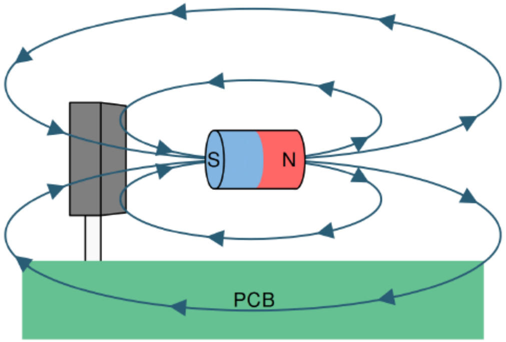
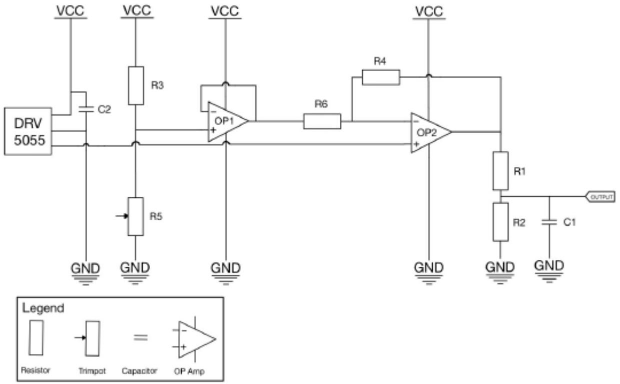
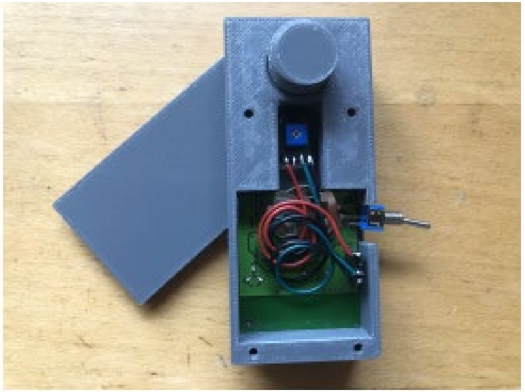


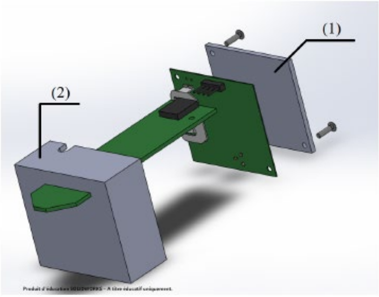
Figure 1: Scheme of the basic Hall sensor setup

The images just above comes from DVR5055 datasheet.





Examples from last year



Soil Moisture Sensor

Wireless sensor practicals | MICRO-333

Group 9 | MT-BA6

3.3 Schéma et liste des composants

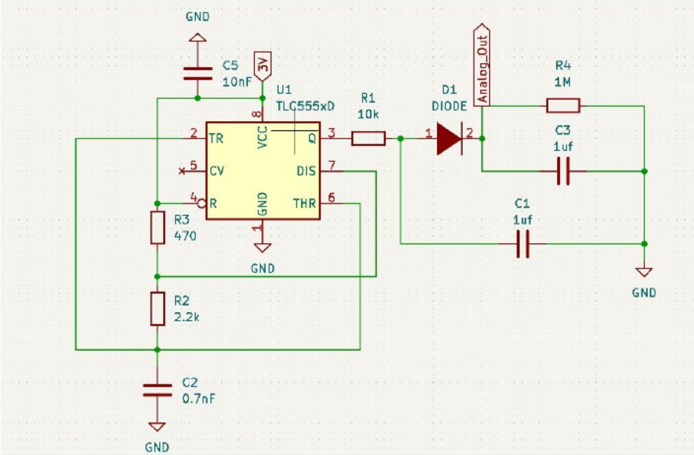


FIGURE 1 – Schéma de la carte du capteur et des composants

Examples from last year

Humidity sensor report

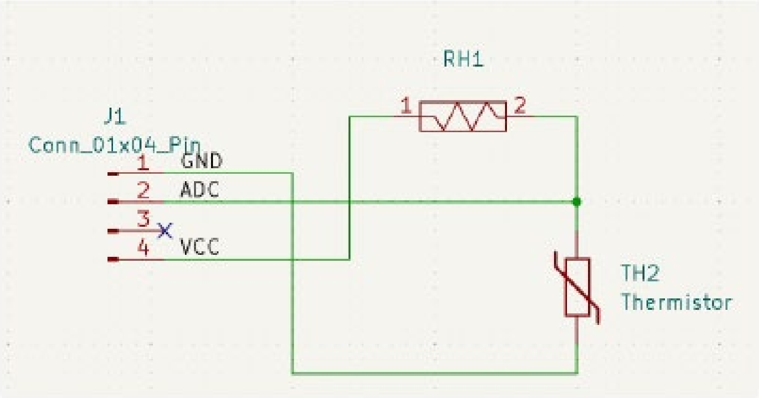
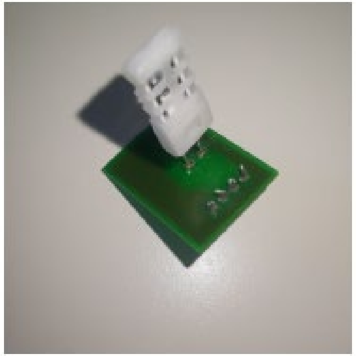


Figure 5: Electrical shematic

Figure 8: Our sensor at different stages of assembly

(a) Front of PCB



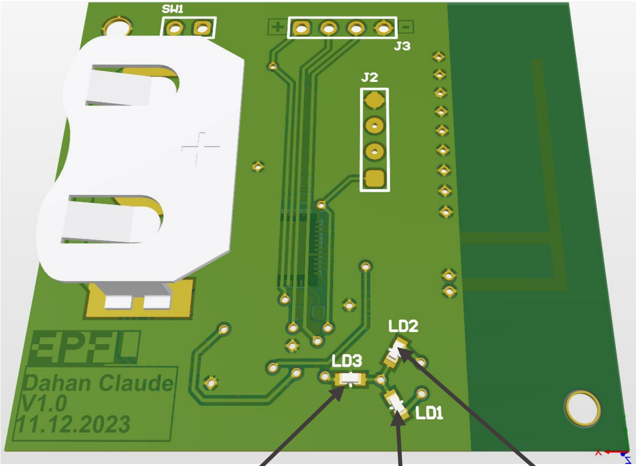
(b) Back of PCB

How to choose a project

- Be adventurous!
  - You will not be penalized if your final project doesn't work… In that situation, I will mainly grade you on your design concept and attempt to debug what happened.
- Think deeply
  - Consider the details of the schematic (voltage and current requirements, physical size, package interactions, etc.)
  - Consider the implications on the package.
- Ask me for advice
  - Put together a few ideas if you are unable to come up with a single "great" idea and ask for our feedback. I can meet with you by zoom throughout the week.

More details: Using the board

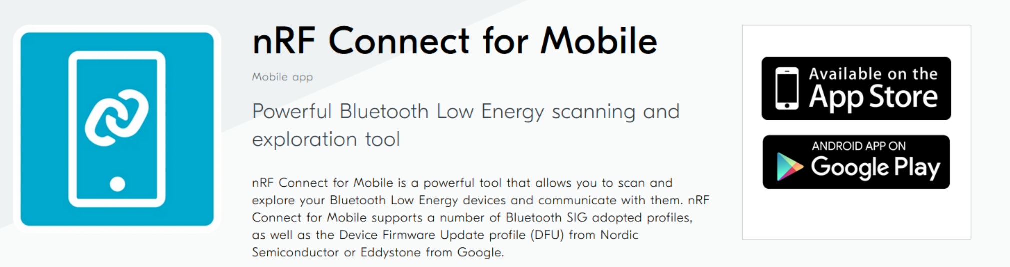
Any Bluetooth Low Energy app would work.
Proposed solution is nRF Connect, from the MCU manufacturer:



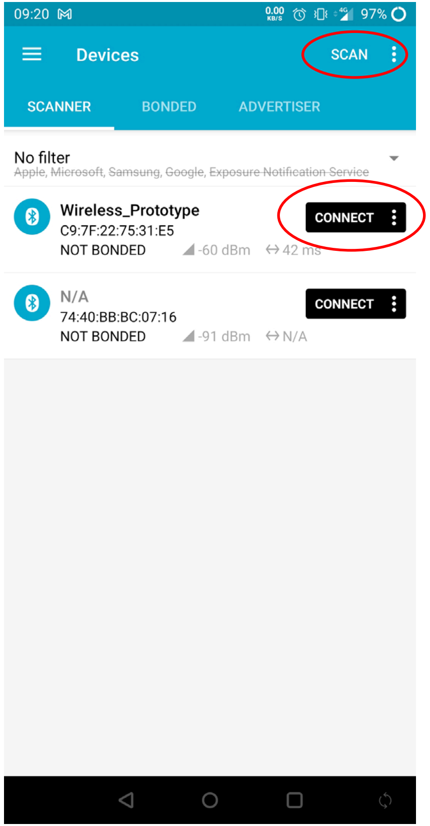
Green
Power ON

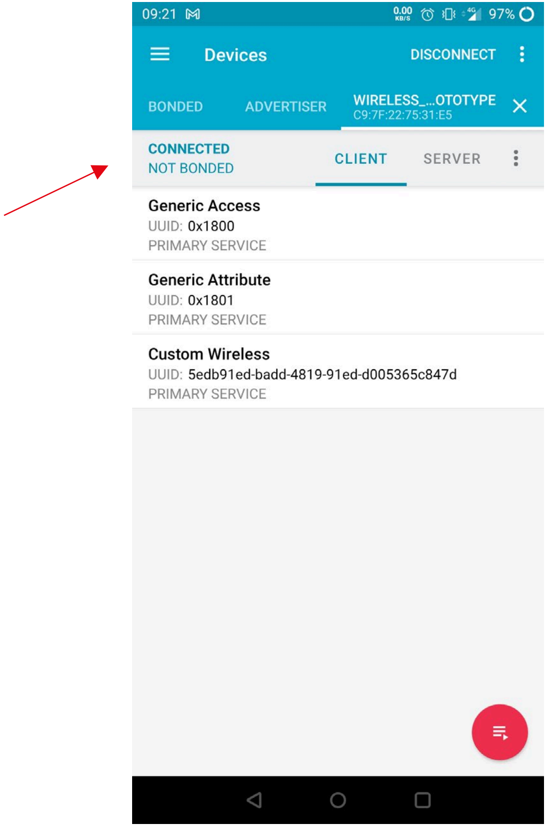
Red
Toggles at data transmission

Blue
Bluetooth connected



Using the app

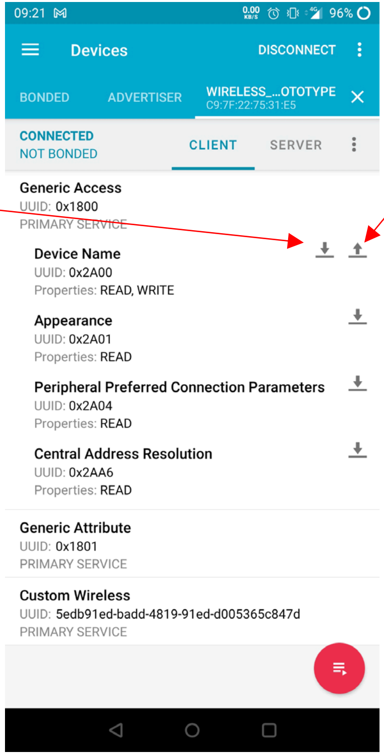




Mostly Bluetooth specificities.
Of interest to us:
- Generic Access → Name, to rename the peripheral
- Custom Wireless → Custom service for the course

Using the app

Read current name



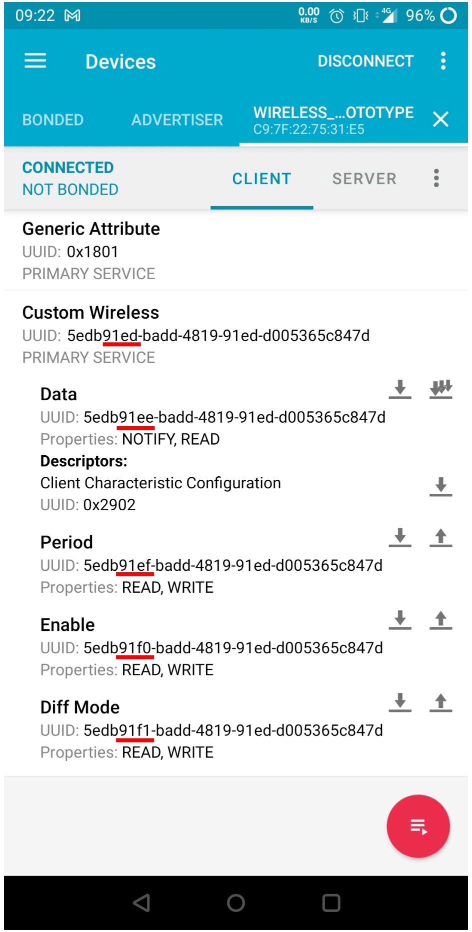
Write new name
Max 20 characters
Stored in memory, will be used for advertisement (here instead of “Wireless_Prototype”)

Using app

On first connection, only the UUIDs will be displayed

Press and hold on the app to edit name for each UUID as follows:

The app then keeps in memory which UUID corresponds to what



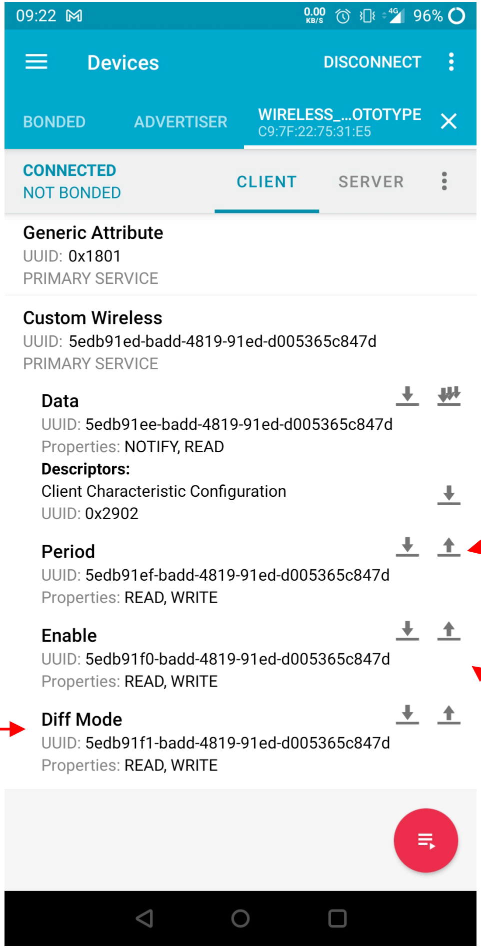
Read value
Current sampling period, last data recorded, …

Write value
Change sampling period, enable recording, …

Subscribe for notifications
Value automatically read on modification, no need to manually read

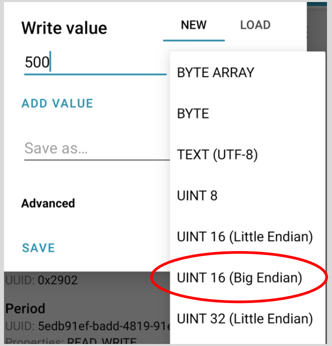
EPFL

Using app



16bit value, from 10 to 60,000
Sampling period, in milliseconds
(Write either in hexadecimal, or as “UINT16 Big Endian” decimal value)

500ms sampling:



8bit value
Differential mode if not equal to 0
(Data then becomes signed)

8bit value
Sampling enabled if not equal to 0
Cannot be enabled if invalid period

Known issues

- Due to handshake at OS level on phones, “Disconnect” from app does not actually fully disconnect boards on some phones, as seen by blue LED remaining ON. Either restart the board, or turn OFF and ON the Bluetooth on your phone
- ADC can under/overflow in single ended mode, as in Data can jump from 0 to 0x3FFF and vice versa. Take care to correctly interpret those values
- ADC is quite noisy. When designing your sensor, use the widest voltage range as possible, or use some data processing like oversampling

… and that is the end of the DLL