INTRODUCTION TO ESIPAP COMPUTING SESSIONS

WEDNESDAY 13 – THURSDAY 14 FEBRUARY 2019

GOALS OF THE COMPUTING SESSIONS

- Computing is required for instrumentation purposes:
  - Simulation of sensor
  - Data acquisition
  - Data analysis
  - Algorithm and reconstruction of physics objects

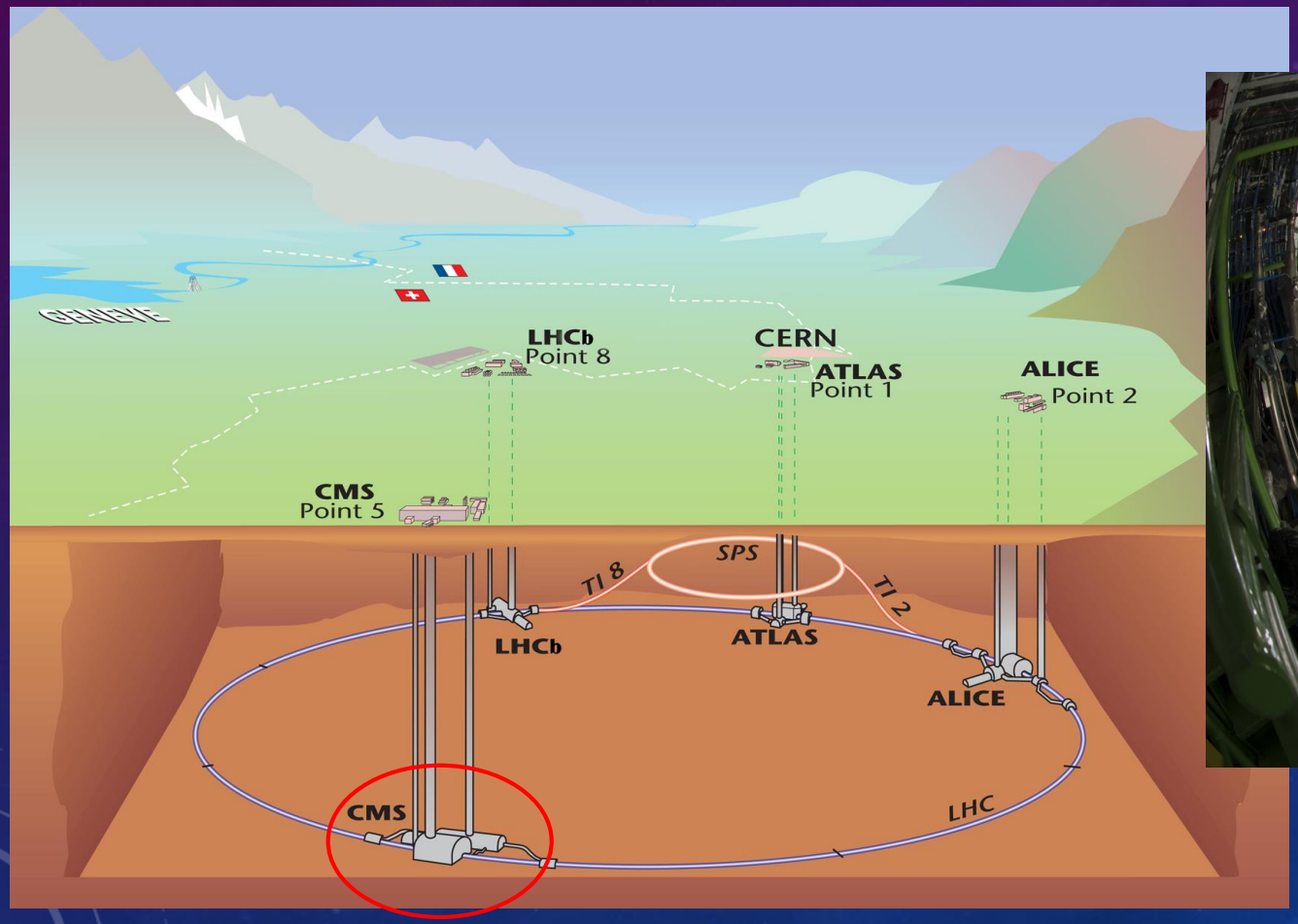
- Computing sessions target to apply your theoretical knowledge:
  - Instrumentation
  - Software programming in C++
  - Using specfic tools of high energy physics: ROOT and Geant4

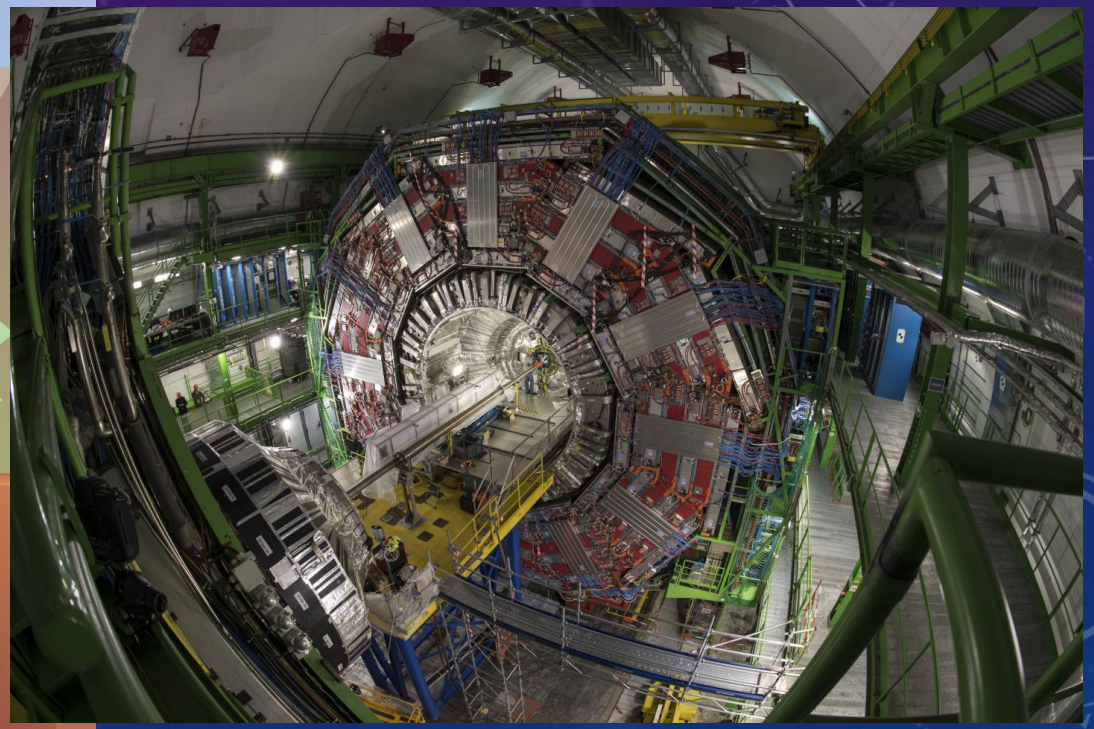
- Working by yourself and experimenting

- Getting the good practice

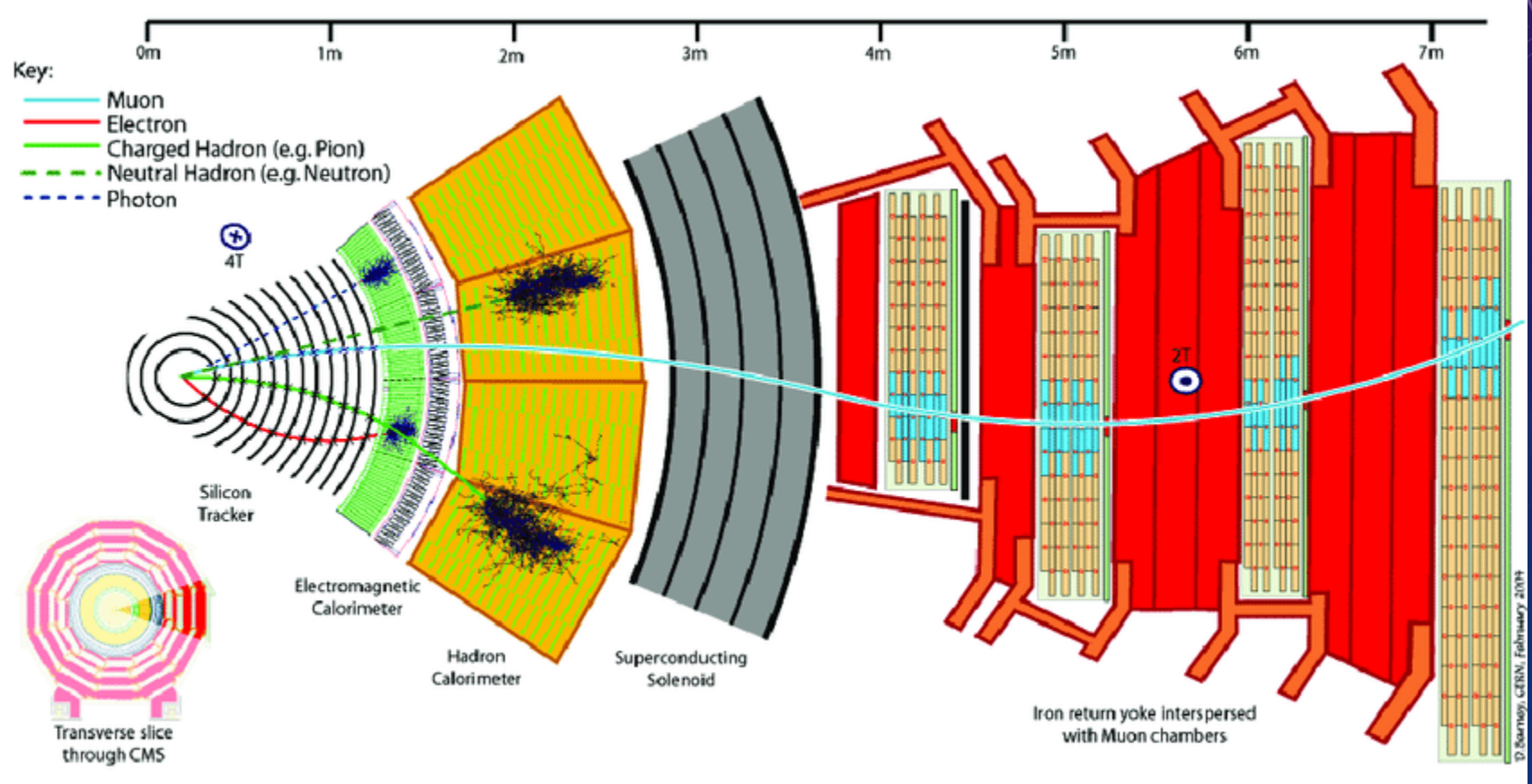
PHYSICS CONTEXT

THE CMS (COMPACT MUON SOLENOID) DETECTOR

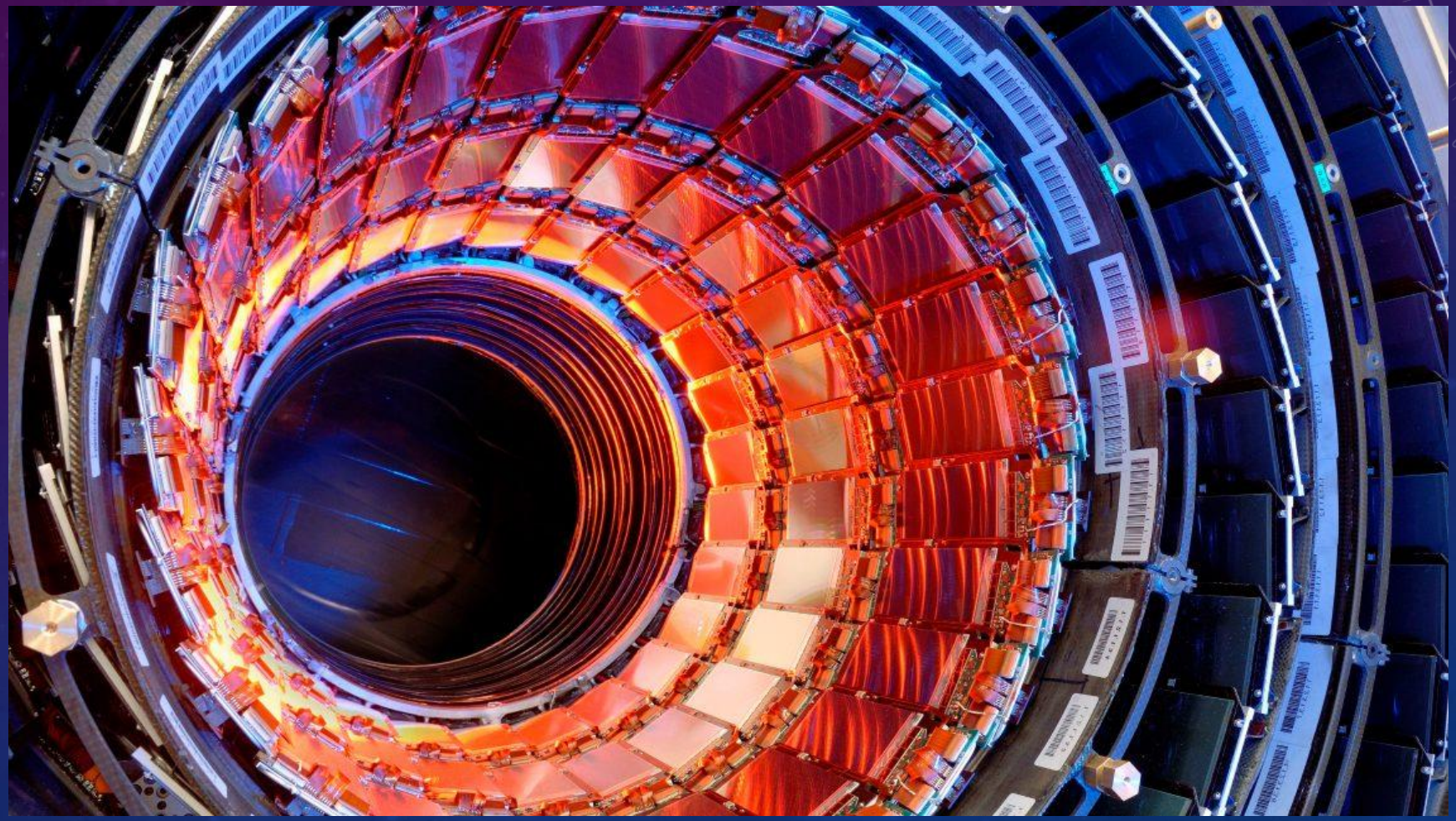




THE CMS (COMPACT MUON SOLENOID) DETECTOR



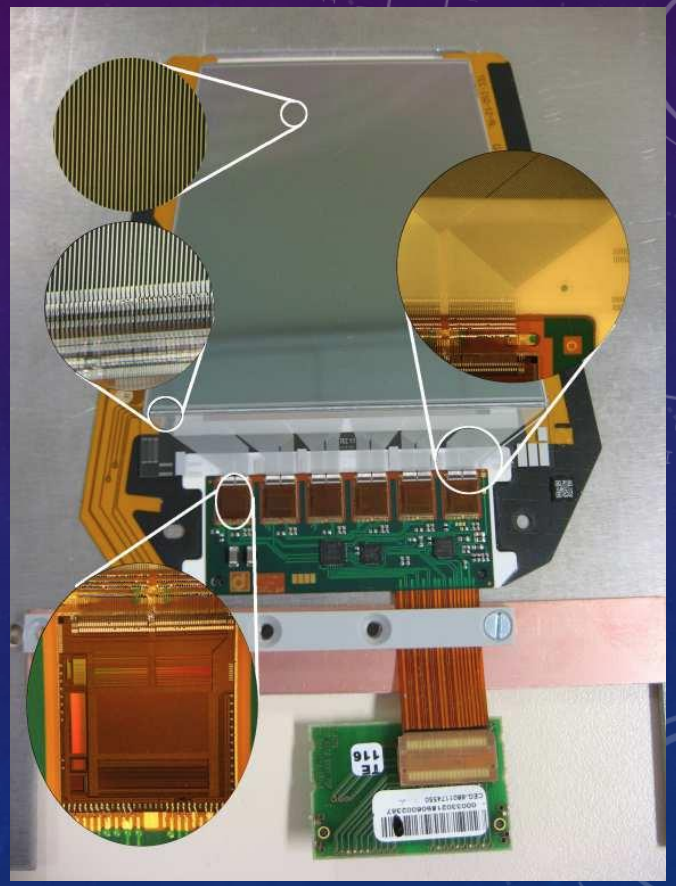
SILICON STRIP TRACKER



SILICON STRIP TRACKER

CMS silicon strip tracker in few numbers:

- 15 000 modules
- Surface: ~ 200 m^2
- 10^6 channels



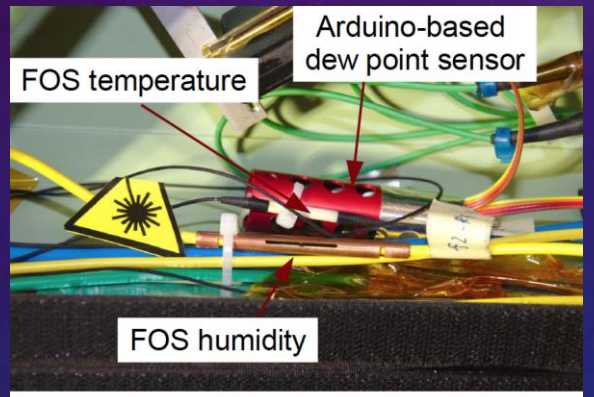
Instrumental activities

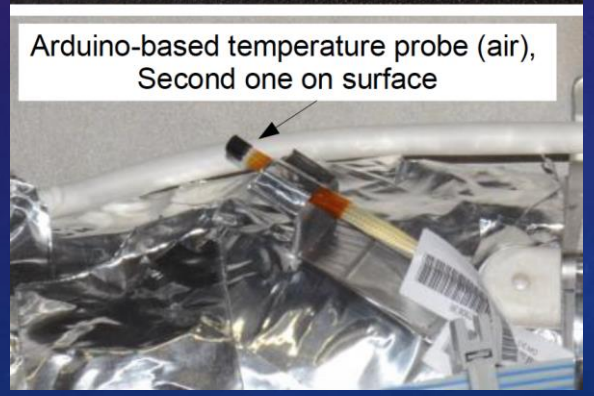
- R&D
- Construction
- Operation (online)
- Alignment & calibration
- Offline analyses
- Simulation
- Radiation damages evaluation
- …

Performances:

- Hit resolution: 20-40 µm
- Hit efficiency > 98% (at high Pile-Up)
- Timing alignment accuracy: 1ns
- …

SILICON STRIP TRACKER





During its operation it is important to monitor environment conditions:

- Temperature
  - Leakage current
  - Noise
  - Thermal dissipation
  - Radiation damages
  - …
- Humidity
  - Dew points & condensation
  - Front End electronics
  - …

Monitoring tools

Several probes are used to monitor that:

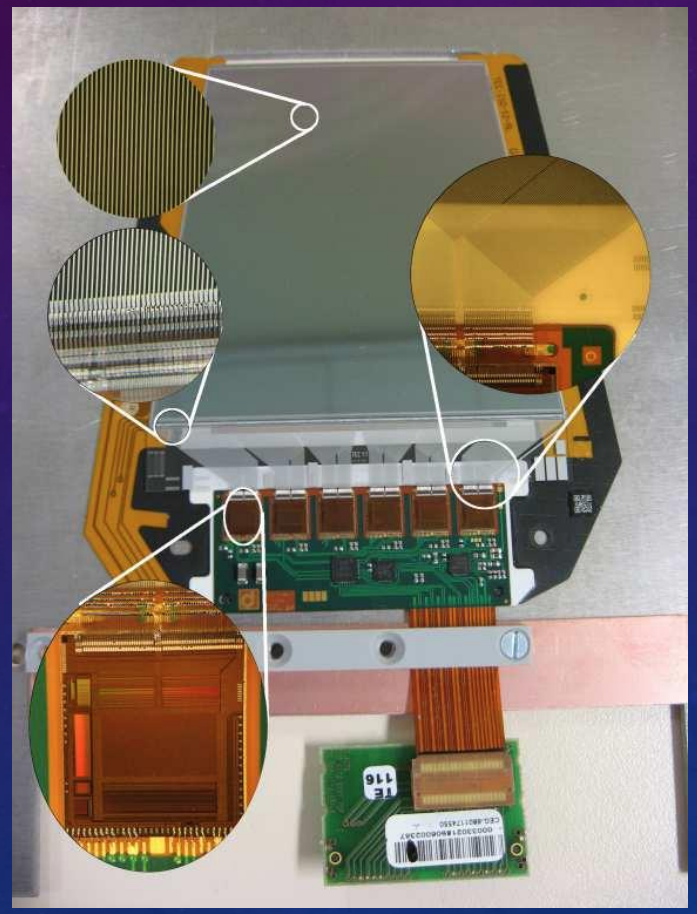
- On-board sensors
- External sensors

→ Some are ARDUINO-based!

COMPUTING SESSION AIMS

Instrumental activities

- R&D
- Construction
- **Operation (online)**
- Alignment & calibration
- **Offline analyses**
- **Simulation**
- Radiation damages evaluation
- …



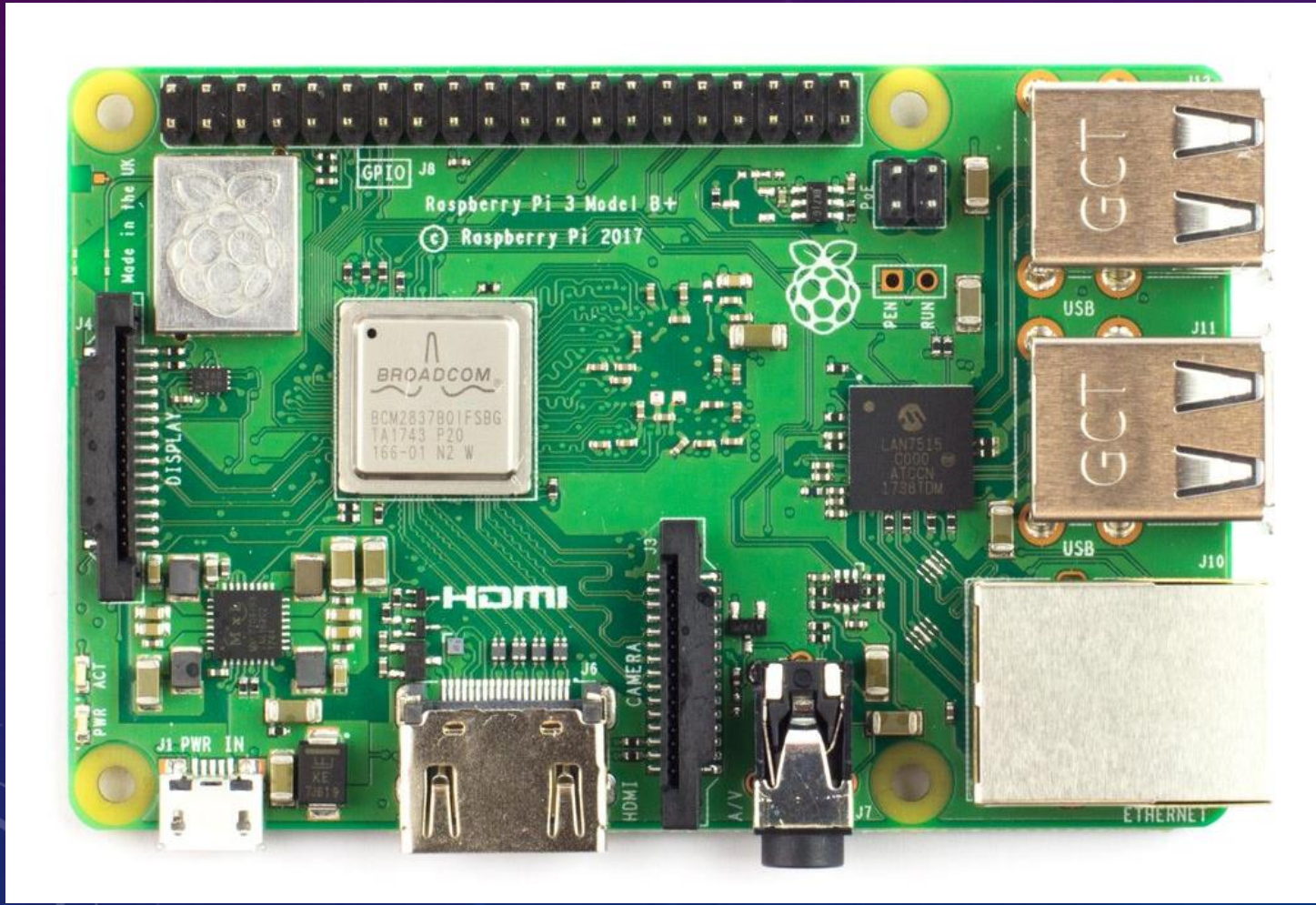
1. **Slow control**
   - Using a dedicated electronic board (Sense Hat) read by a Raspberry
     - Monitor the temperature & humidity
     - Send warning when conditions are not fulfilled

2. **Offline analyses**
   - Calibration of the temperature sensors
   - Evaluation of the sensor resolution

3. **Simulation**
   - Basic simulation with the GEANT4 package of a CMS silicon strip sensor

SETUP

THE RASPBERRY BOARD



Raspberry Pi 3 B+ motherboard

- Quad-core 64 bits processors @ 1,4 GHz
- ARM (Acorn Risc Machine) architecture used mainly in smartphones, tablets, robotics, automation

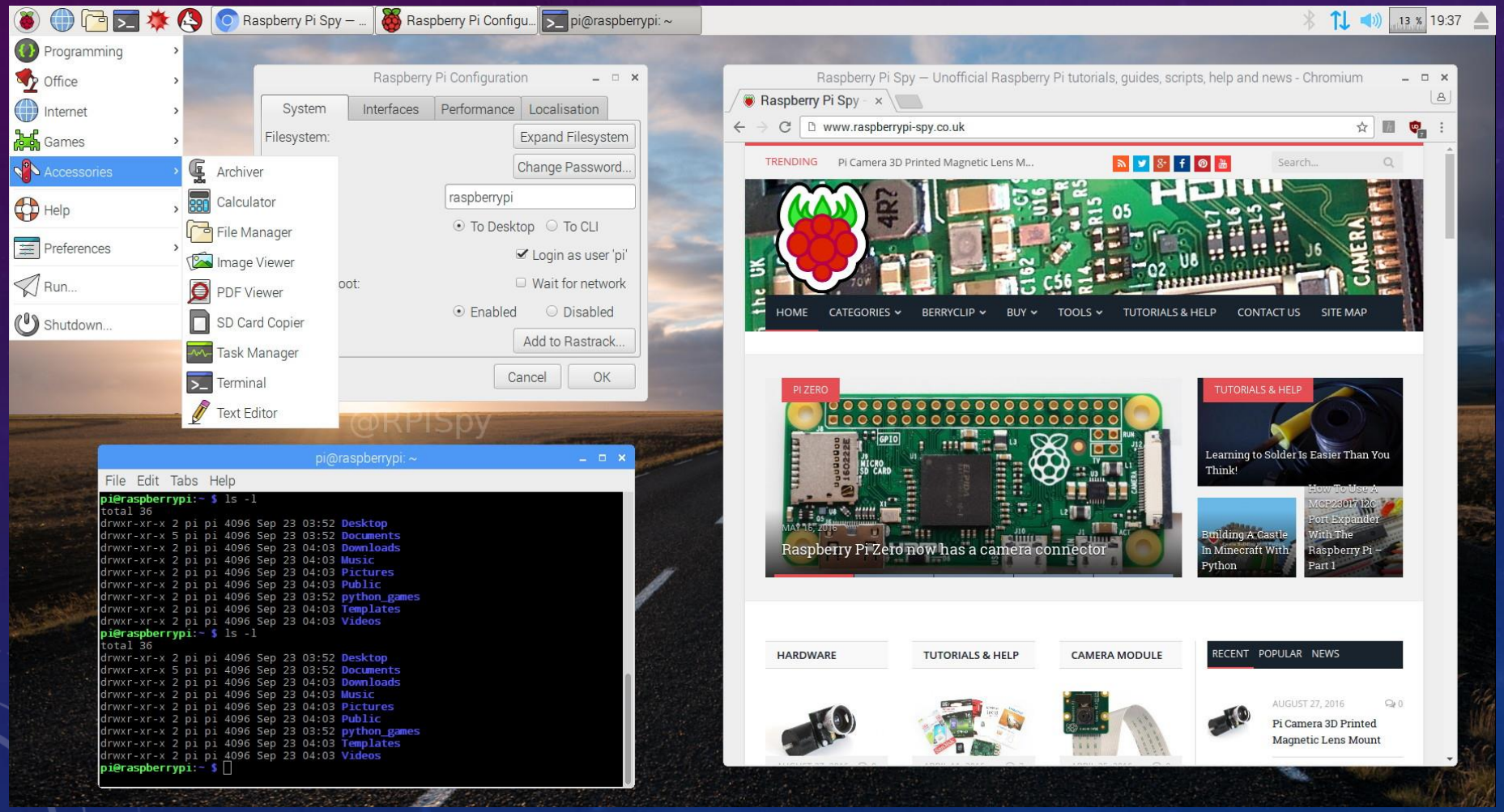
Advantages : price, flexibility, performances

CONNECTIONS TO PERIPHERICAL DEVICES

- 4 USB ports
- 1 ethernet port
- 1 HDMI plug
- 1 GPIO (General Purpose Input/Output) port for connecting sensors
- Powered by micro USB (5V, 2.5A min)

+ WIFI
+ Bluetooth

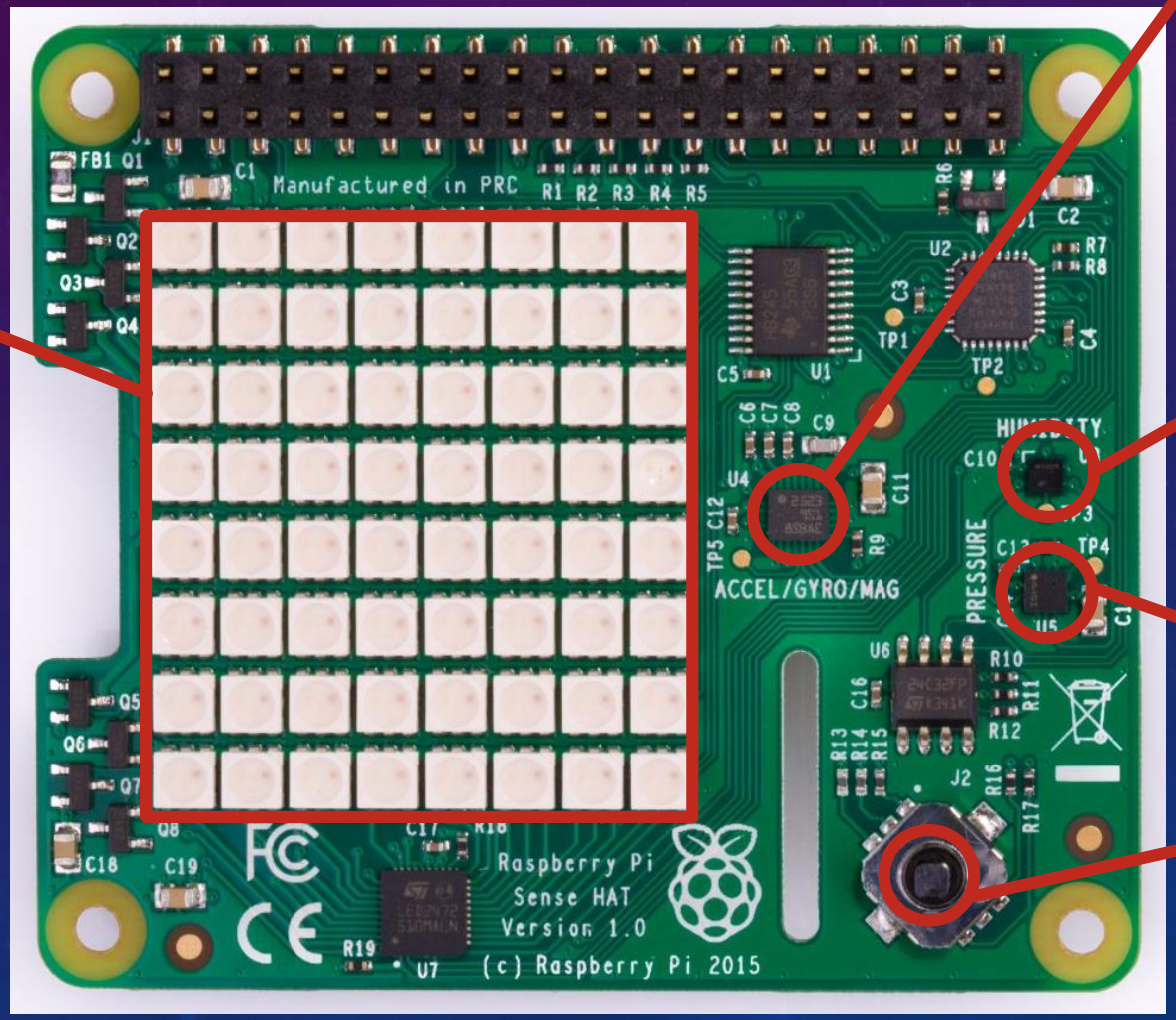
LINUX DISTRIBUTION: RASPBIAN



Stored on a micro SD card

SENSE HAT BOARD

8x8 LEDs for display



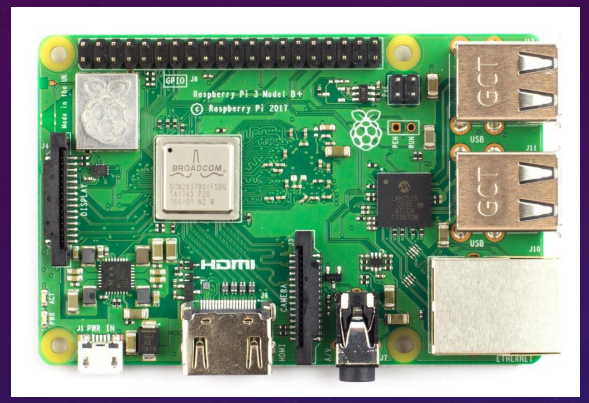
3D accelerometer, 3D gyrometer and 3D magnetometer sensor

Humidity / Temperature sensor

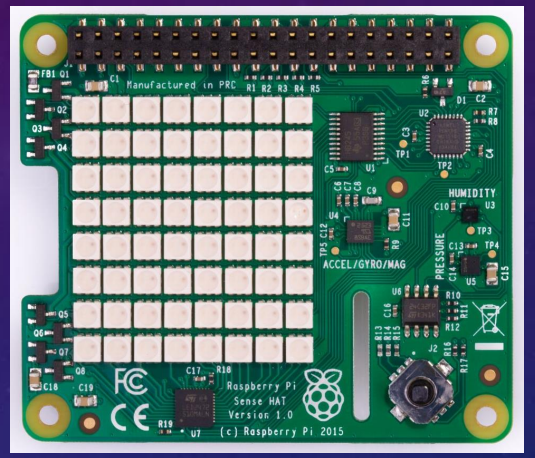
Pressure / Temperature sensor

Joystick

PRICE



Raspberry Pi 3 B+
~ 40 €



Sense Hat
~ 30 €

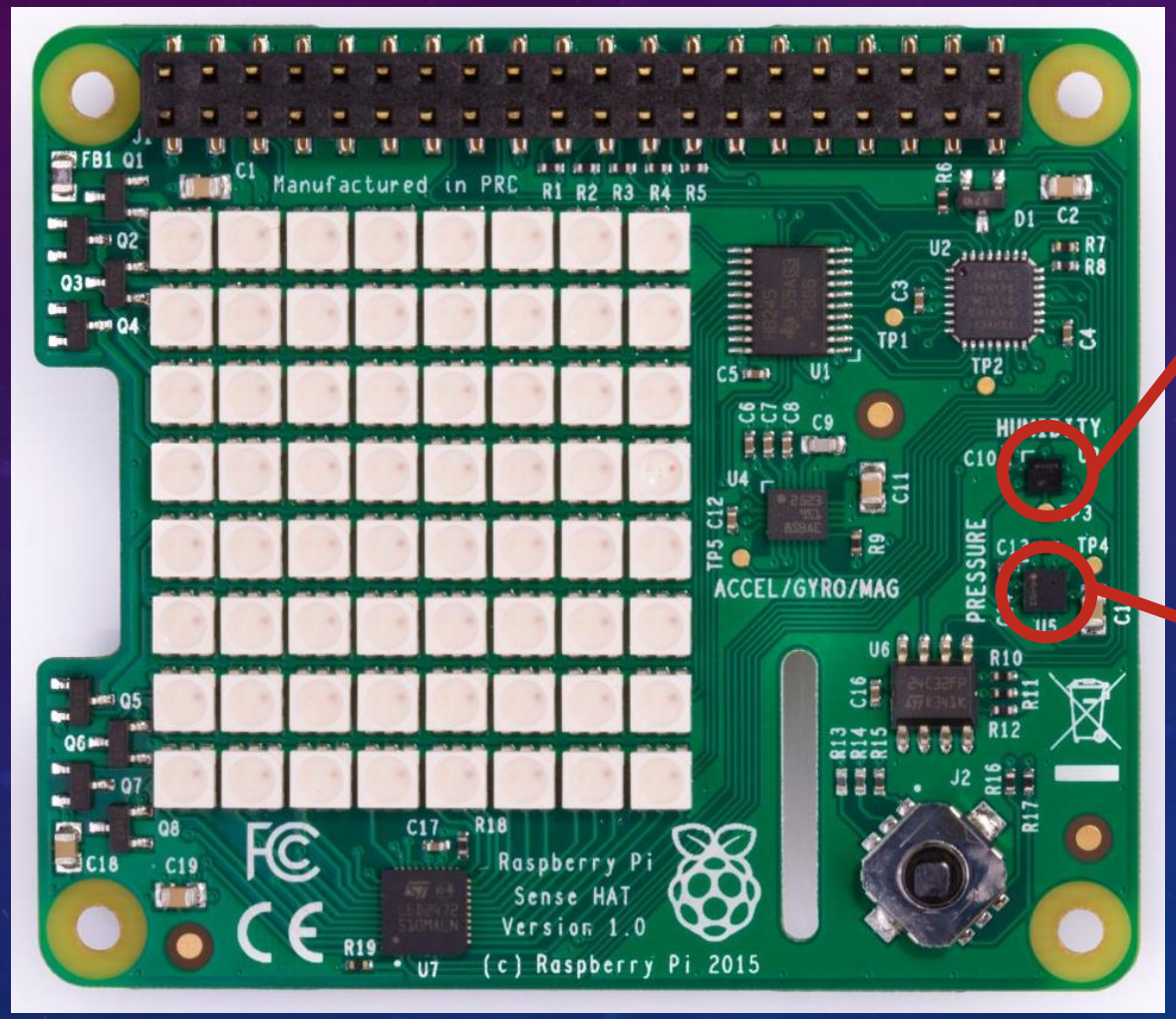
Connectors
~ 15 €

Micro SD
~ 10 €

Total: ~ 100€
(good gift for Saint-Valentin's day)

SENSORS AND SIGNAL CONDITIONING

SENSORS TO STUDY

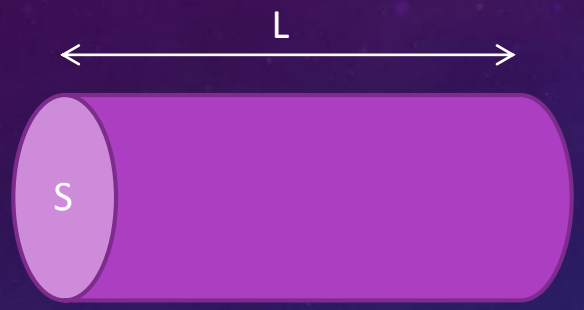


Humidity / Temperature sensor

Pressure / Temperature sensor

HOW TO MEASURE ….?

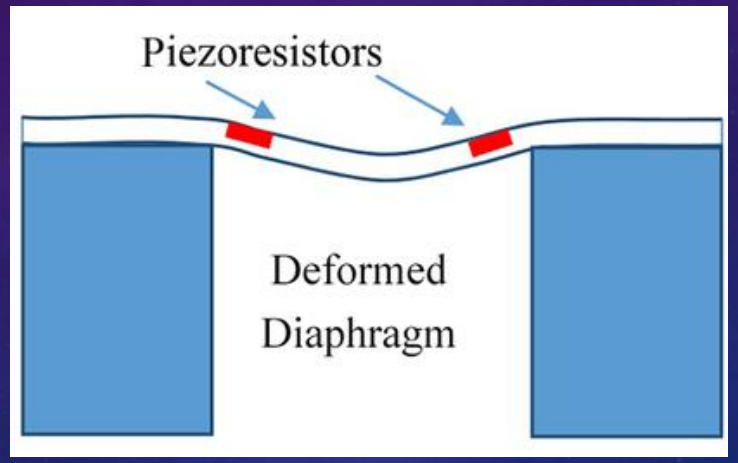
Temperature



$$R = \rho \frac{S}{L}$$

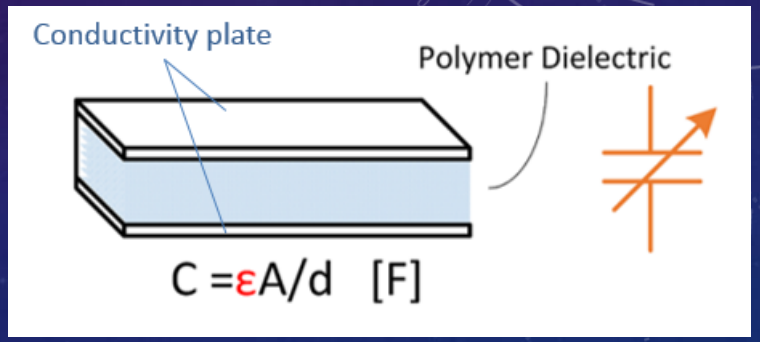
Material resistivity ρ depends on temperature.

Pressure



Piezoresistive effect:
a change in resistivity when a stress is applied.

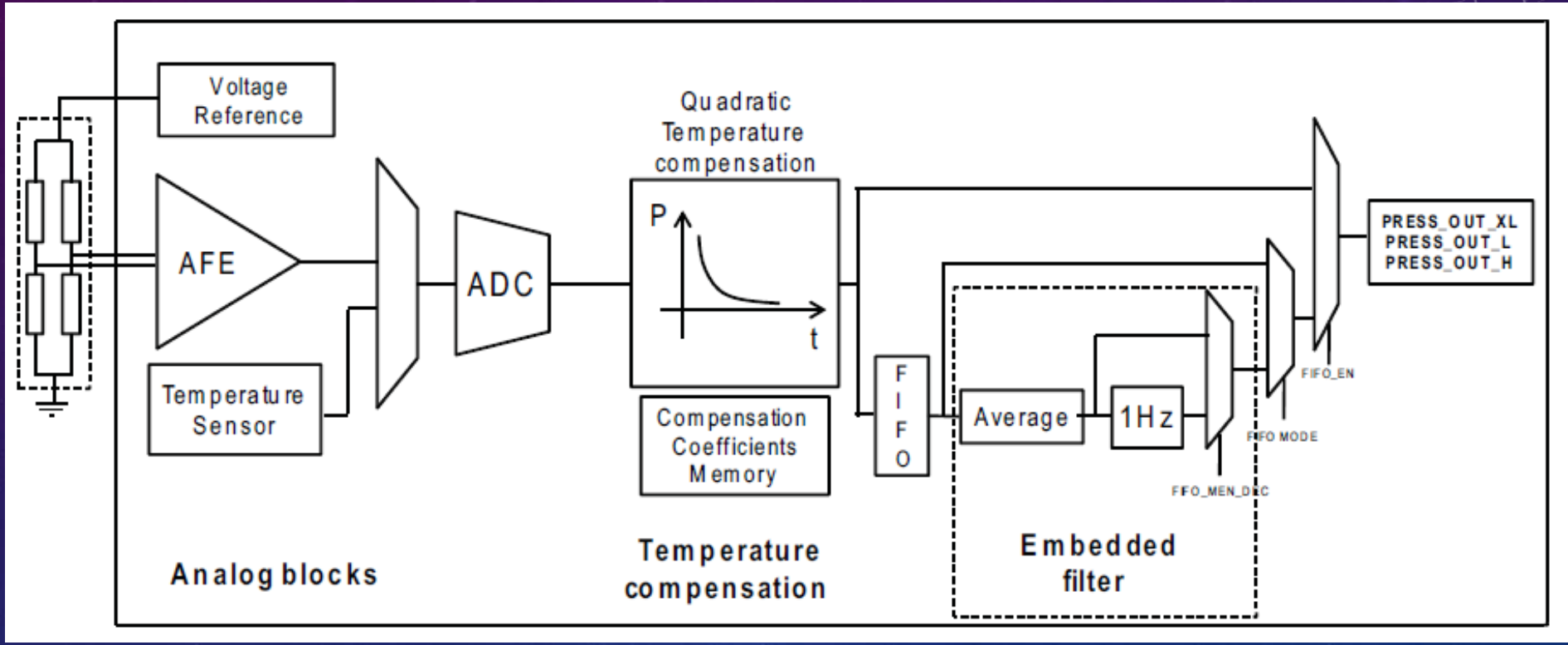
Humidity



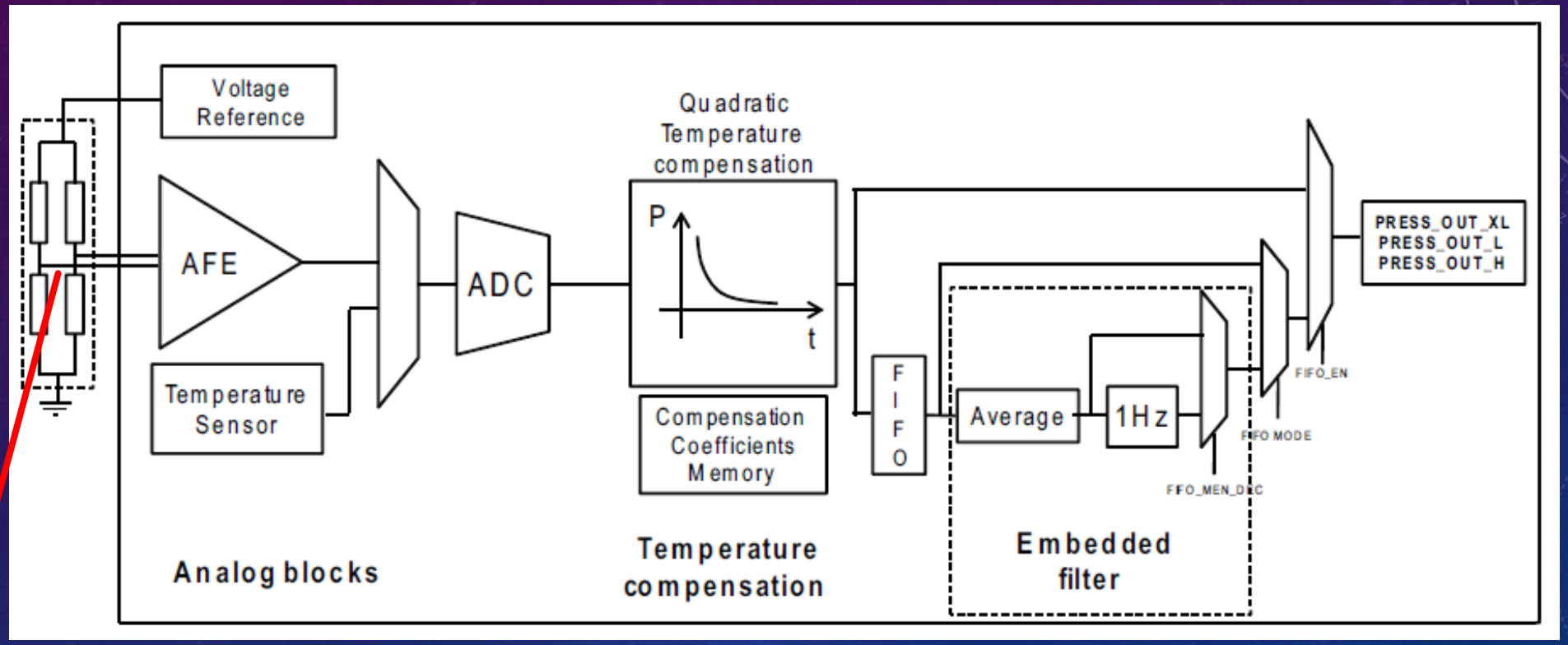
Dielectric material absorbs water molecules until equilibrium

→ change the electrical conductivity ε [in S/m]

PRESSURE / TEMPERATURE SENSOR PROCESS

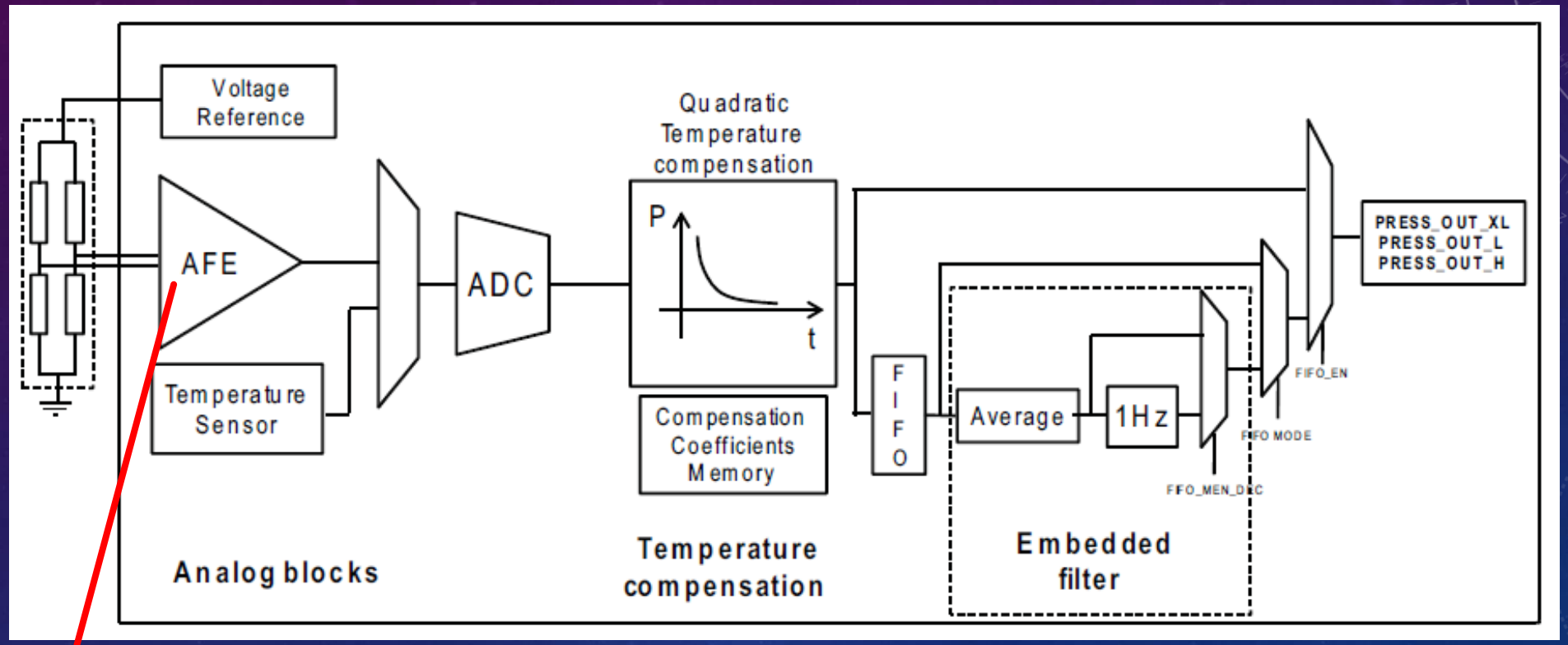


PRESSURE / TEMPERATURE SENSOR PROCESS



Wheatstone bridge
for translating change of resitance into change of tension

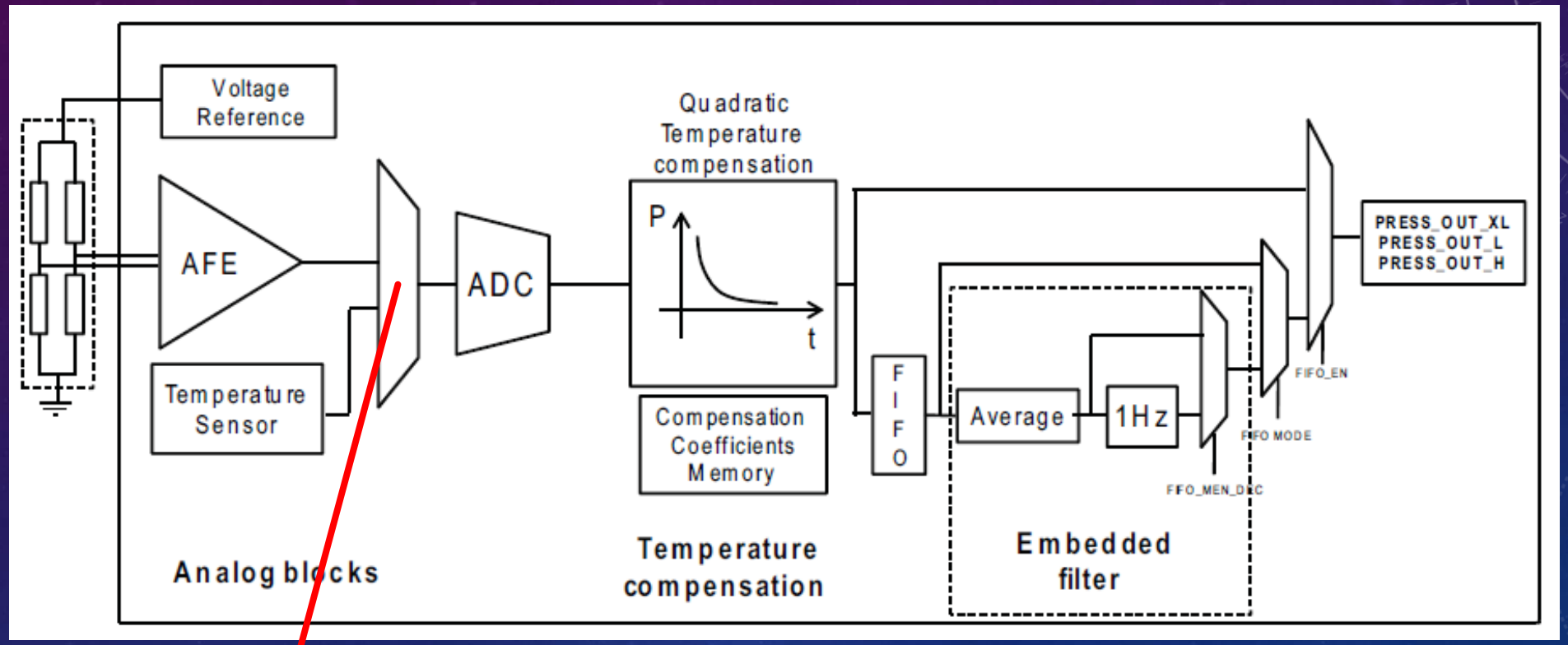
PRESSURE / TEMPERATURE SENSOR PROCESS



Analogic Front-End

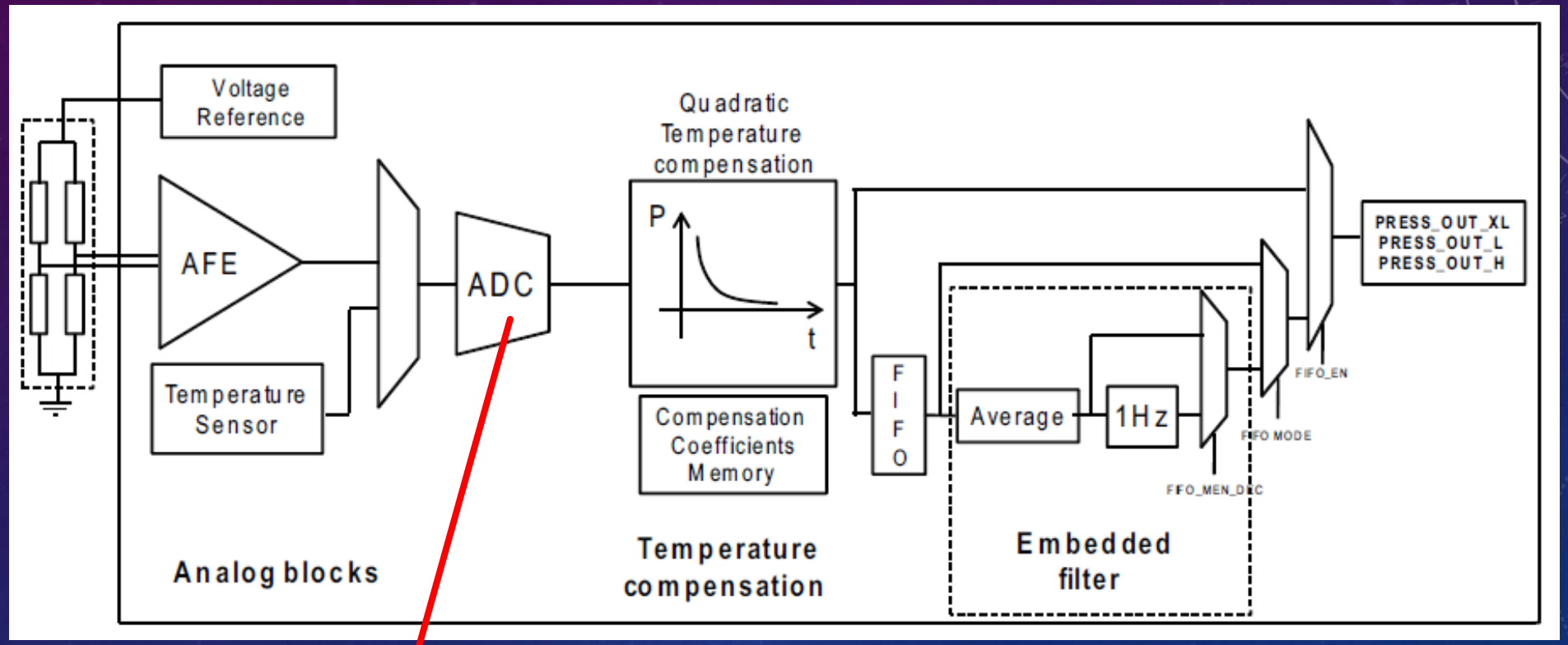
- Small signal voltages vs noise floor
- Amplifying signal and removing noise

PRESSURE / TEMPERATURE SENSOR PROCESS



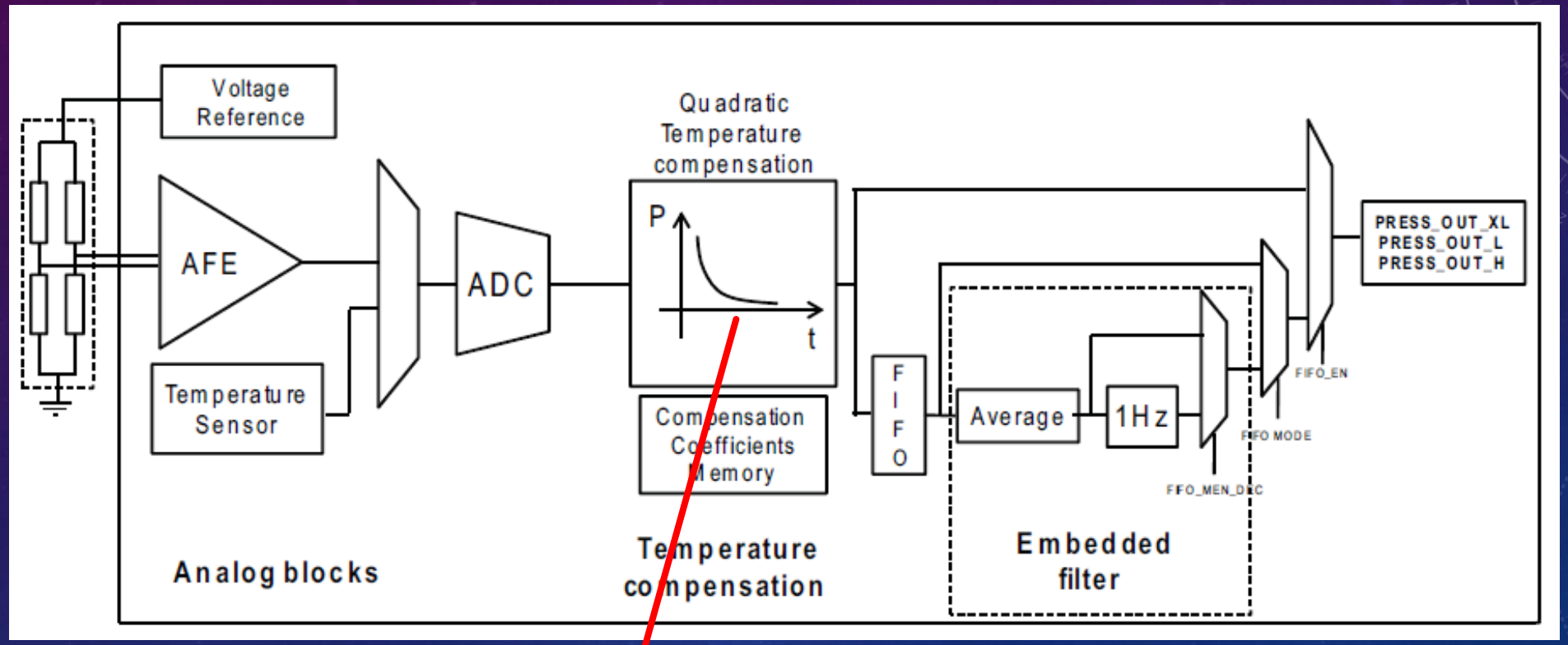
Multiplexer
Treating pressure and temperature measures by the same channel

PRESSURE / TEMPERATURE SENSOR PROCESS



Analogic to Digital converter
Digitalization of the measure

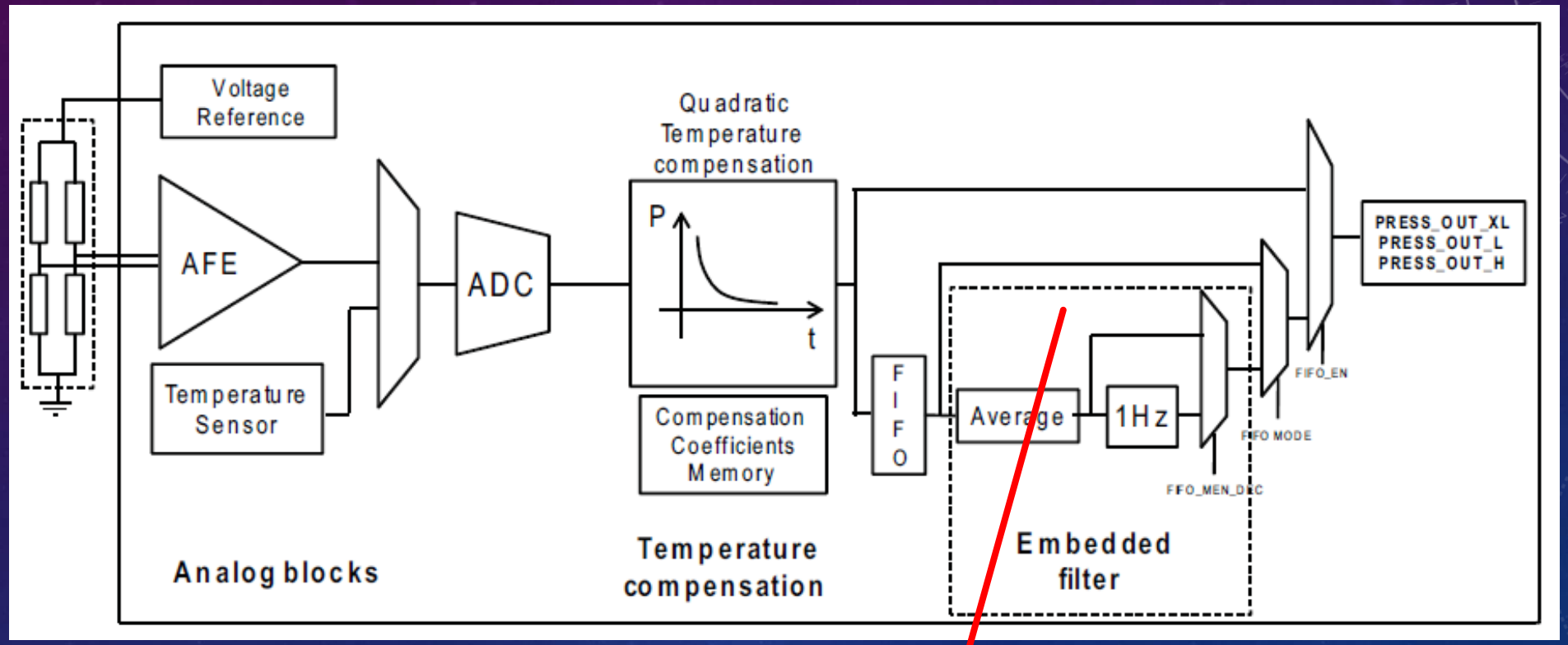
PRESSURE / TEMPERATURE SENSOR PROCESS



Temperature compensation
Piezoresistivity depends on T
→ Need to compensate this effect

Calibration settings

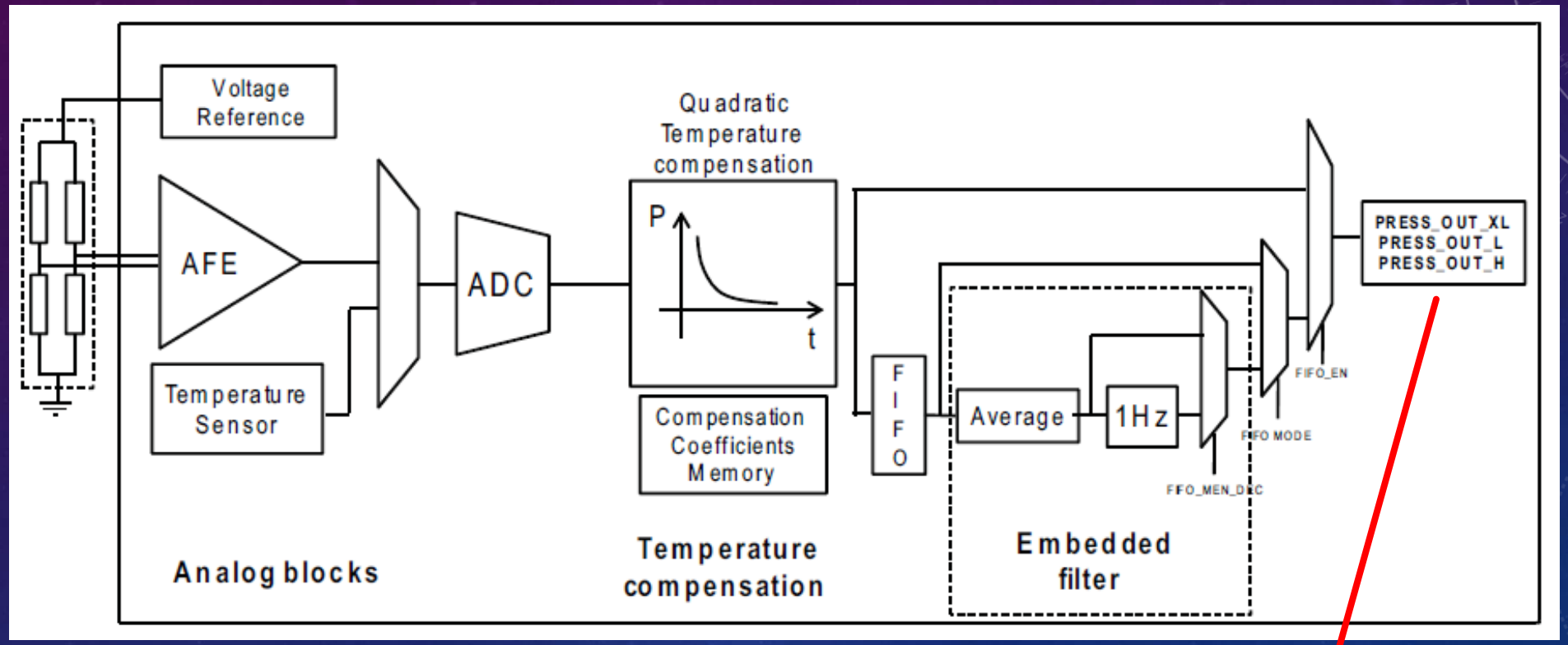
PRESSURE / TEMPERATURE SENSOR PROCESS



Calibration settings

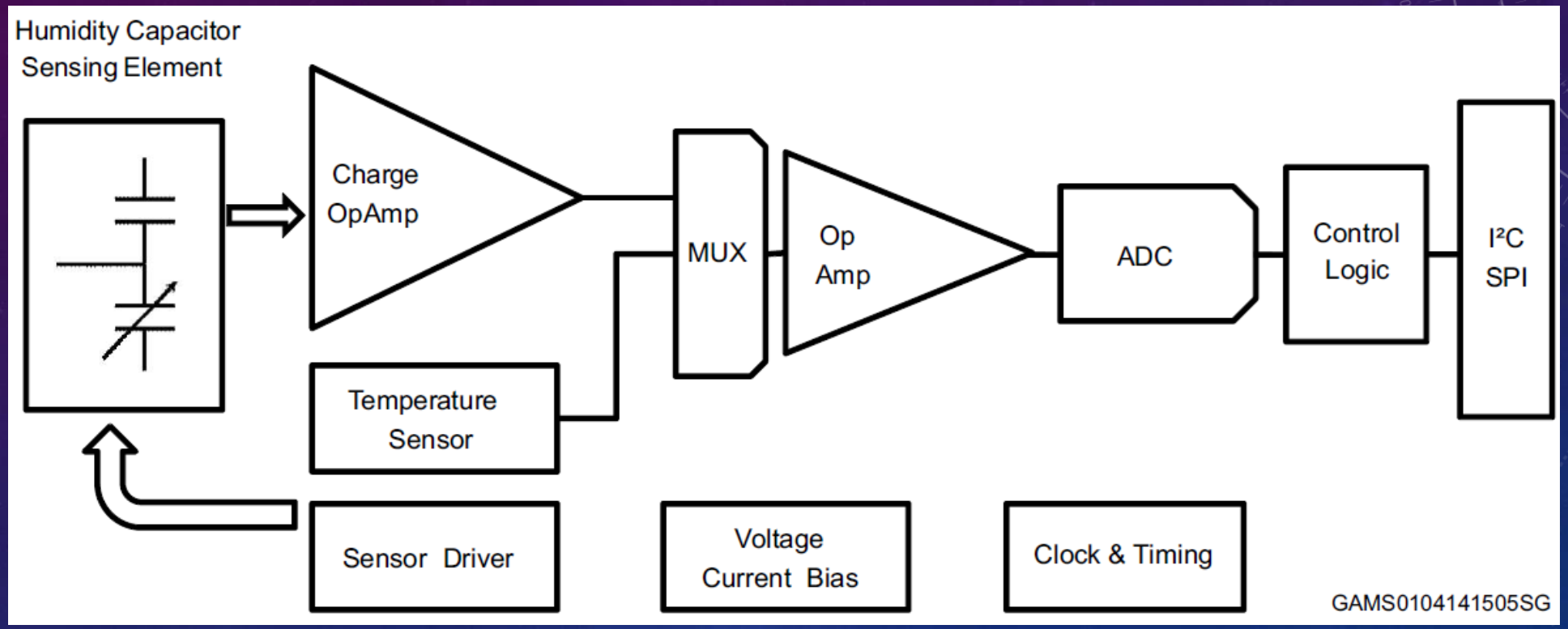
Average
A maximum of 32 successive measurements are done (~1s) and an average value is computed.

PRESSURE / TEMPERATURE SENSOR PROCESS



Data transfer
Data are sent to the Raspberry via the GPIO port with the protocol I2C

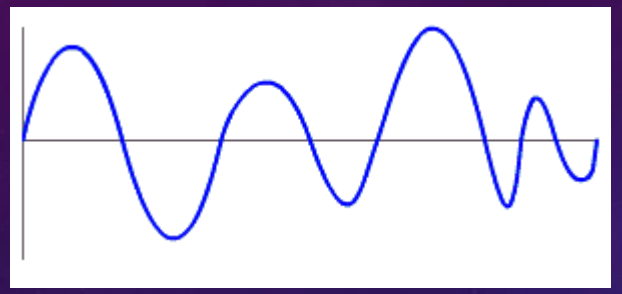
HUMIDITY / TEMPERATURE SENSOR PROCESS



One logic part is missing in this schema:
Translation of the tension to
temperature and relative humidity.

Calibration settings

ADC RESOLUTION FOR TEMPERATURE SENSOR



analogic

ADC

24 bits

0	1	0	1	1	1	0	1	0	0

digital

- Operating range of the sensor: [260 hPa to 1260 hPa] where the sensor is relevant and reliable
- Conversion pressure to measure: measure = pressure × 4096
- Number of bits for coding the maximum value 1260 hPa → measure = 5 160 960
  → N = 23 bits because $2^{22}-1 <$ measure $< 2^{23}-1$ …. but not standard: using 24 bits – ADC
- Full range : [0 hPa to 4096 hPa]
- Sensitivity : 4096 hPa / $2^{24}-1$ = 0,00024 hPa

SUMMARY ON ADC SENSITIVITY

	Pressure - Temperature sensor		Humidity - Temperature sensor	
	Pressure	Temperature	Humidity	Temperature
Operating range	260 hPa to 1260 hPa	-30°C to +105°C	0% to 100%	-40°C to +120°C
Full scale	0 hPa to 4096 hPa	-30°C to +110°C	Linear interpolation, depending of the calibration coefficients	
ADC resolution	24 bits	16 bits	16 bits	16 bits
Sensitivity	0,00024 hPa	0,002 °C	0,004 %	0,016 °C

ORGANIZATION

ORGANIZATION IN SESSIONS

9:00 12:15 14:00 17:15 17:20 18:50

Wednesday

Session 1

- Introduction
- Data acquisition

Session 2

Developing a C++ class

Session 3

Combining classes

Thursday

Session 4

Analyzing data with ROOT

Session 5

- Simulating particle interaction with GEANT4
- Summary

ONE STUDENT, ONE RASPBERRY, ONE PC

First Name	Family Name	Email address	Raspberry ID	PC ID

https://docs.google.com/spreadsheets/d/11kUn_jwpypZLoiuCy6KbL9pOtJx85zv2GIgtNICG9CQ/edit?usp=sharing

SAVING YOUR PRODUCTION

- End of session 2
- End of session 4

} Sending your code to the supervisors for assement

- Using a web service: www.wetransfer.com
  - destination: eric.conte@iphc.cnrs.fr
  - author: filling your address email
- A URL link is created: put it on the following spreadsheet

https://docs.google.com/spreadsheets/d/1QF1hbePGmzLfQ4jQGMD180Ix8dLivxRkPI5sIzGXBO8/edit?usp=sharing

SKILL ASSESSMENT

Computing sessions 2019: assessment skill list

Skill category	Minimum	Satisfying	Very satisfying
1. Knowing C-programming basics	• Writing a "Hello World!" program • Asking questions to the user • Writing functions		
2. Using the standard library	• Using std::cout, std::string, std::fstream	• Using std::vector, std::stringstream and cmath.	• Using algorithms, iterators and manipulators.
3. Writing a C++ class	• Writing a simple class with: constructor without and with arguments, destructor, mutators, accessors and "print" function. • Instantiating and testing the implemented class.	• The class is operational: it fills up the primary aim.	• Implementing operator overloading and copy constructor. • Using properly the reserved keyword "const".
4. Coding algorithms	• Algorithms work and give the correct results.	• The code is robust: it is protected against bad inputs. • Managing properly the dynamic memory allocation (delete).	• The code is efficient: efforts are achieved for saving time.

1 / 2

- Evaluation over 8 categories
- For validating the module
  - Minimum level must be reached for all the 8 categories
  - Satisfying level for at least 3 categories