# End to end developments for the Multipurpose Interferometer Array Pathfinder from the IAR Electronics Laboratory.

Hugo Command, Juan M. Gonzalez, Gastï¿½n Valdez

Instituto Argenitno de Radioastronomï¿½a (IAR)
Camino Gral. Belgrano Km 40, Berazategui, Argentina

*hcommand@iar.unlp.edu.ar; jmgonzalez@iar.unlp.edu.ar;gvaldez@iar.unlp.edu.ar*

I A R
CCT La Plata
CONICET

UNIVERSIDAD NACIONAL DE LA PLATA

COMISIÓN DE INVESTIGACIONES CIENTÍFICAS

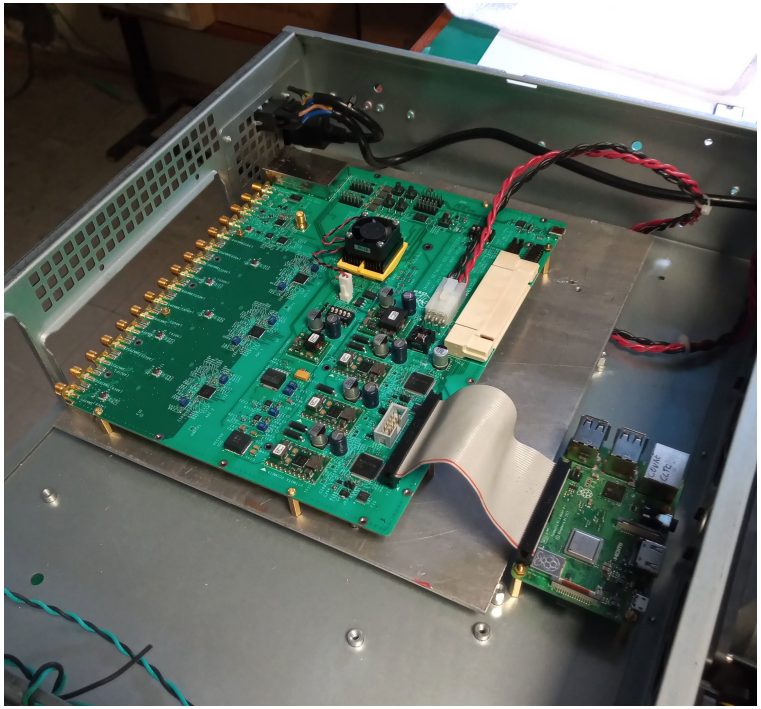
# Front-End

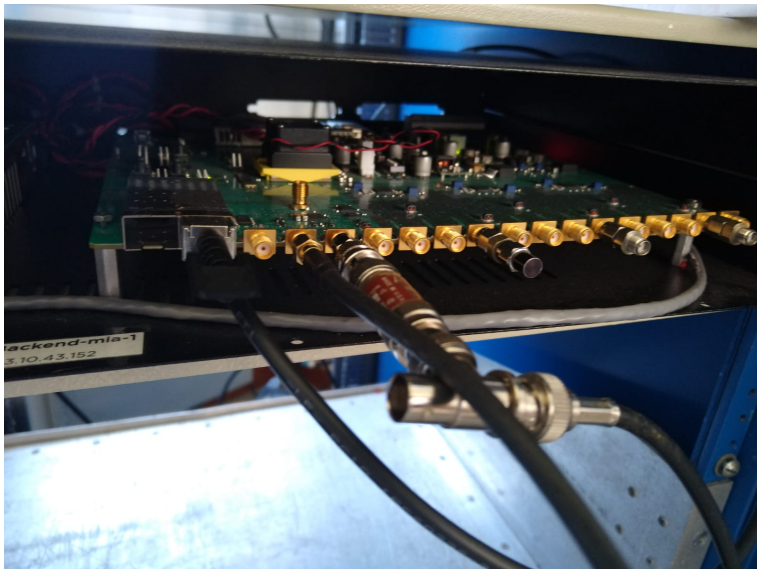
# Back-End

The Smart Network ADC Processor (SNAP) board consists of three HMCAD1511 8-bit analogue to digital converters (ADCs), capable of sampling 500 MHz for three signals per board.
The ADCs are connected to a Kintex-7 160T field-programmable gate array (FPGA),
with an associated dual 10 GbE port.
Each SNAP board is controlled via Raspberry Pi, which interacts with a controlling PC over ethernet via a series of PYTHON scripts.

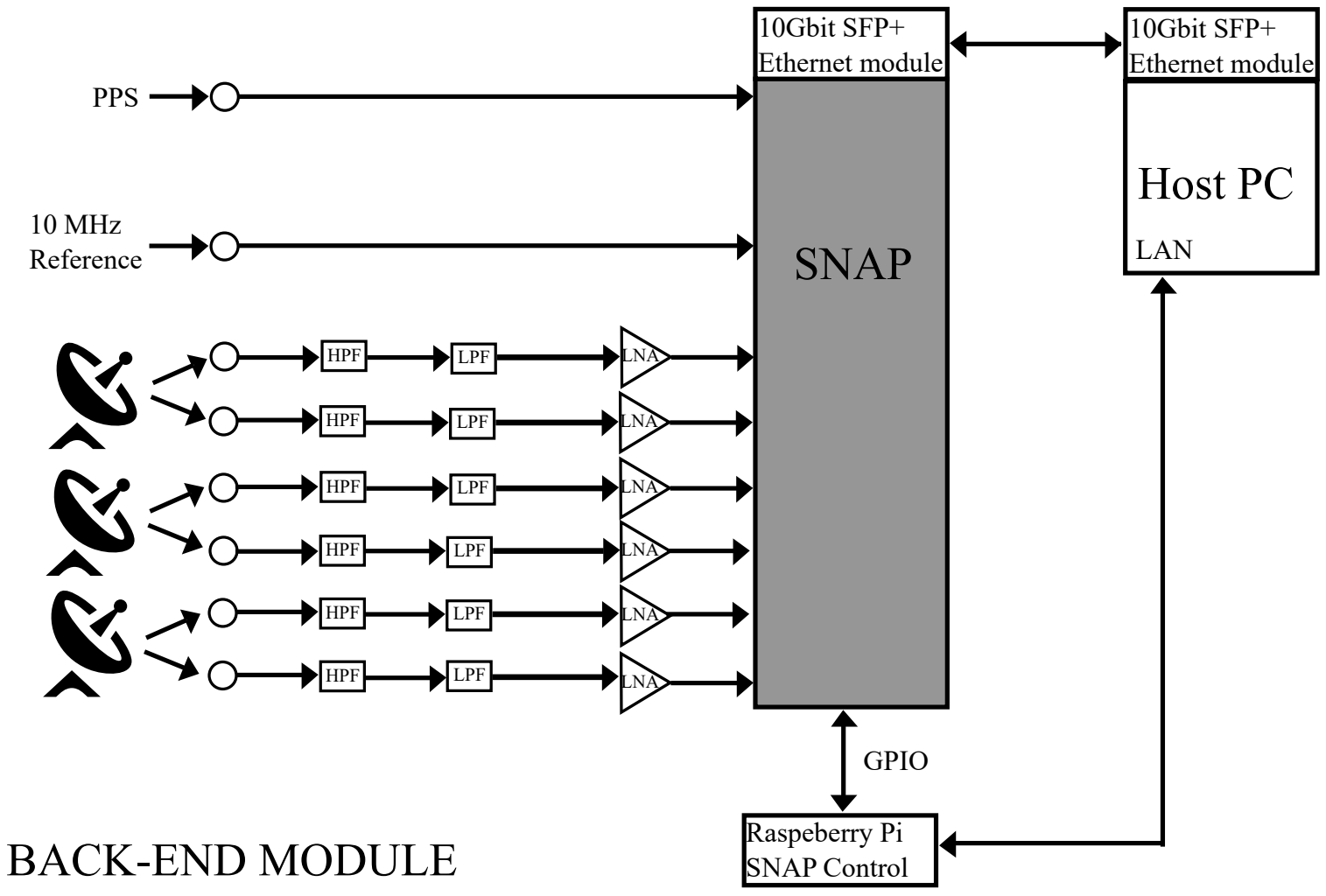
# System features

| Parameter | Value |
|---|---|
| Sample Frequency | 500 MHz |
| BW | 250 MHz |
| Centered Frequency | 1325 MHz |
| Board | SNAP (Casper) |
| Development environment | Casper Toolflow |

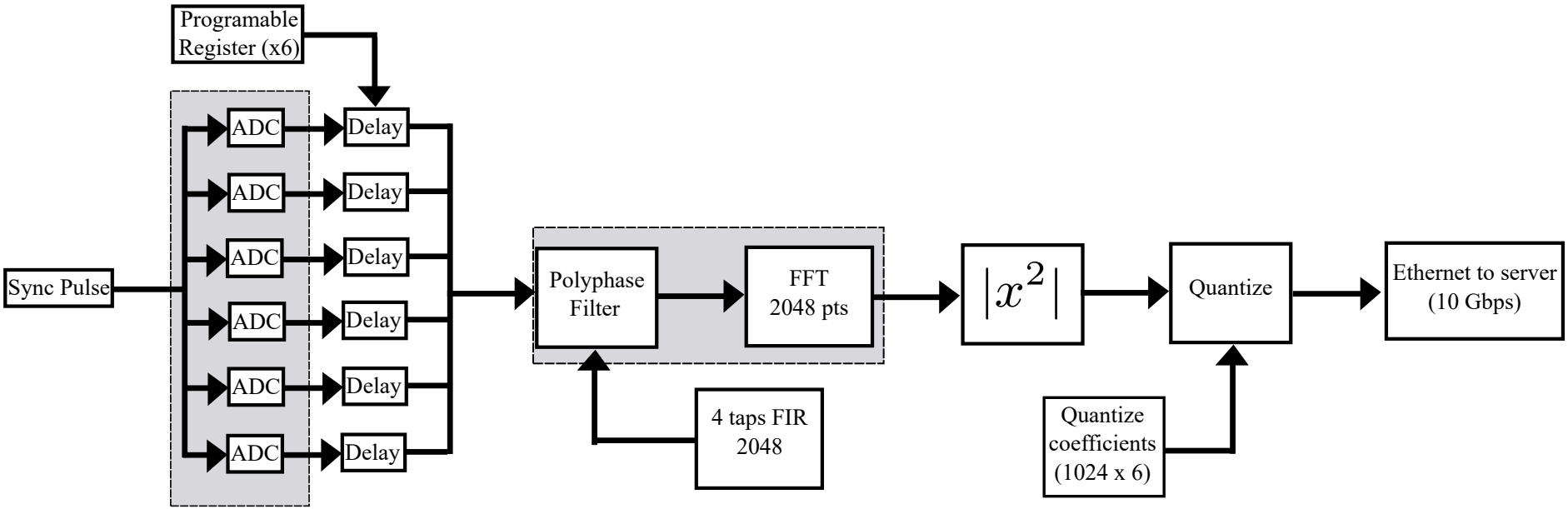




# Back-End Module



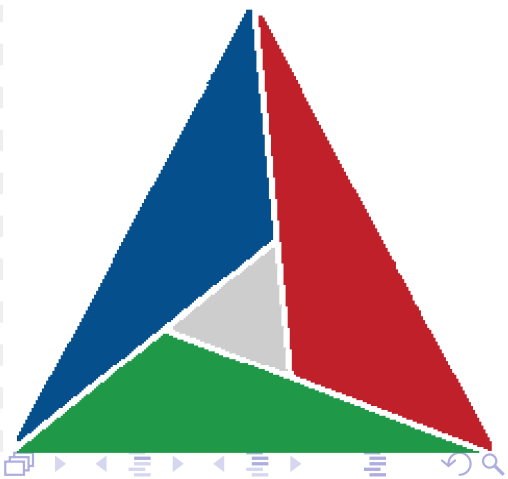
# Spectrometer for system verification



# Position Control

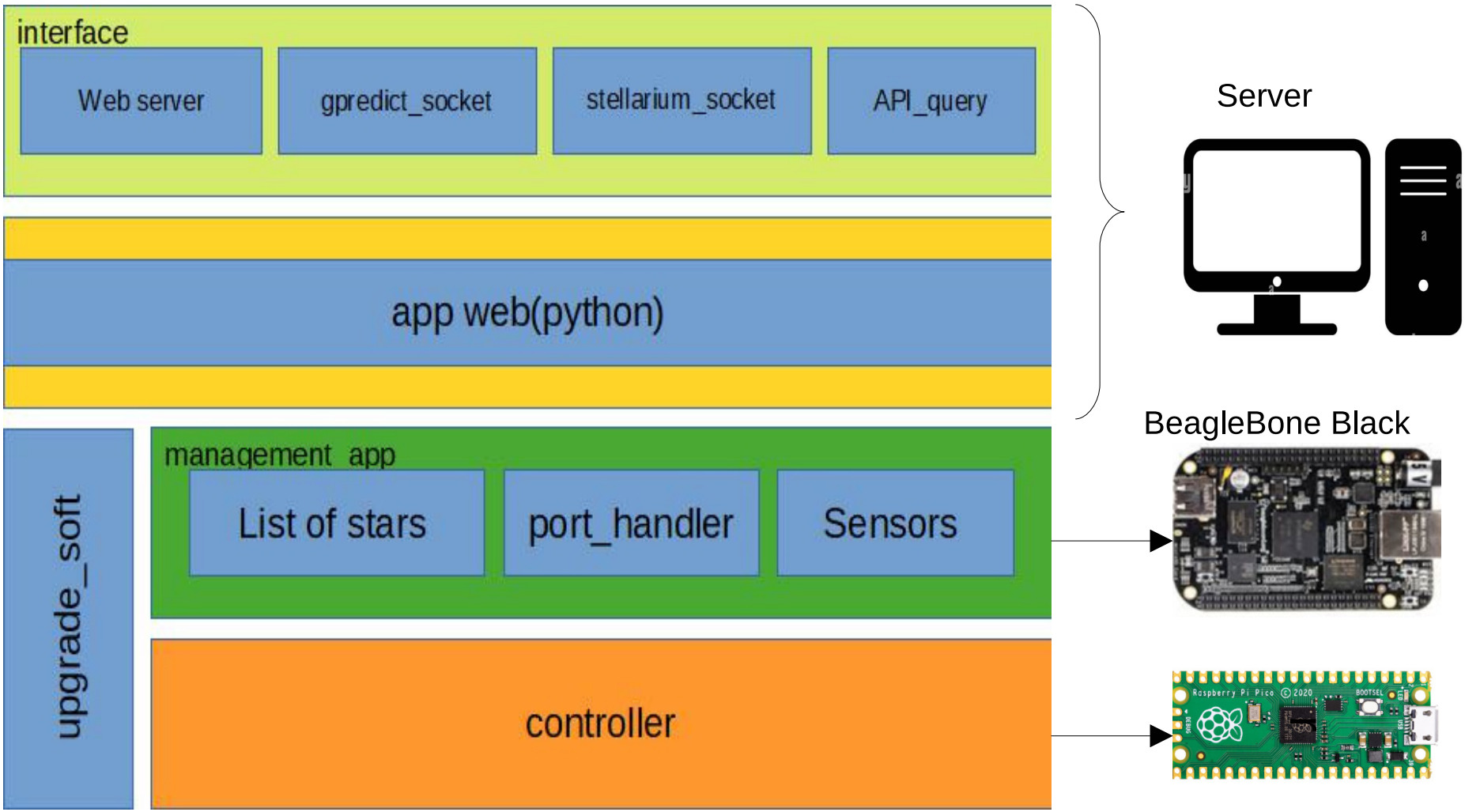
Design based open hardware and open software

- Python: Generate path trajectories stars for testing system.
- CMake : Tool for compile project and independent system build.
- Languaje programming: C/C++ and Python
- Ceedling: Framework for testing embebbed systems.

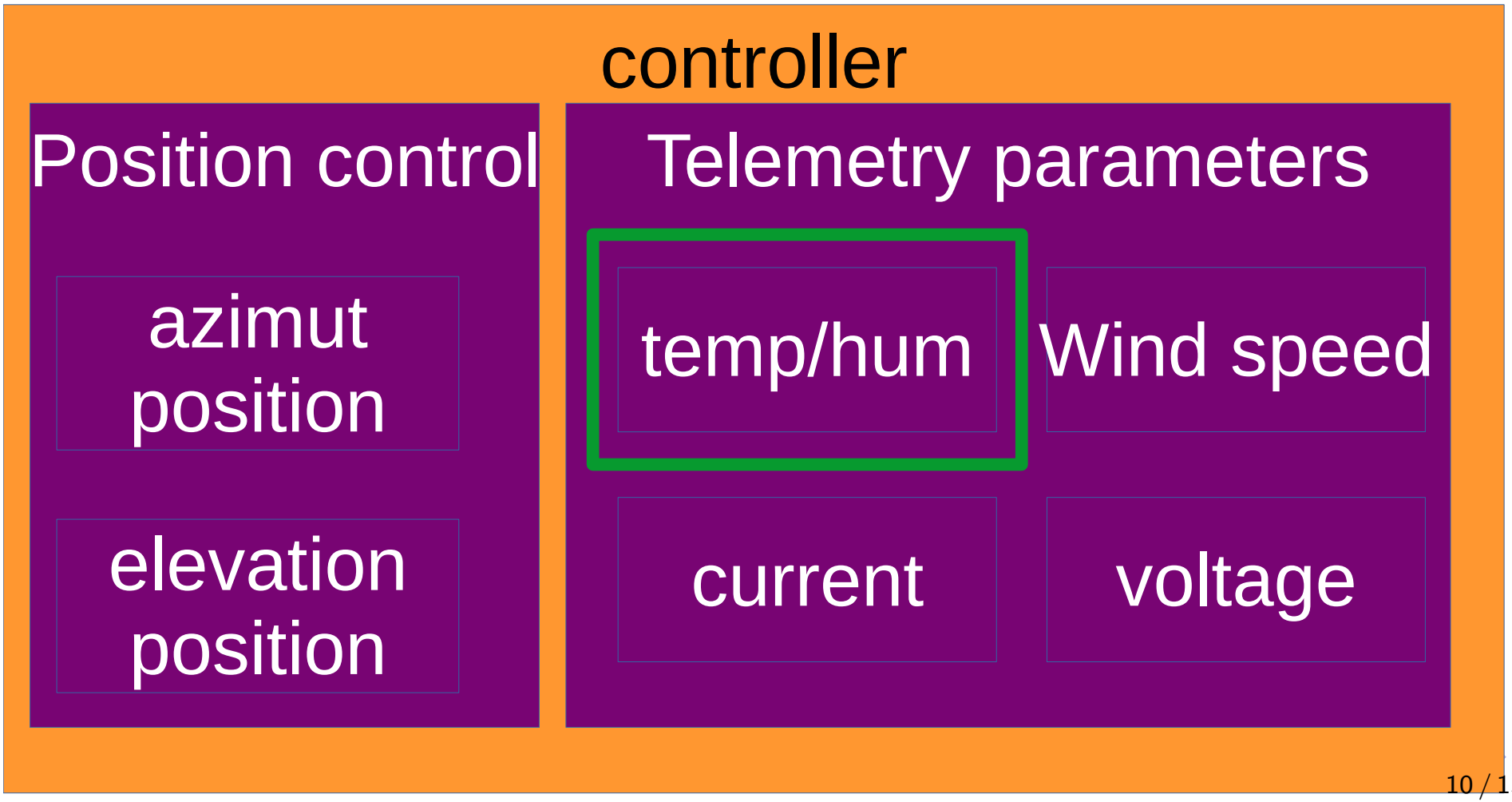
# Position Control

The software design using architecture of layers.



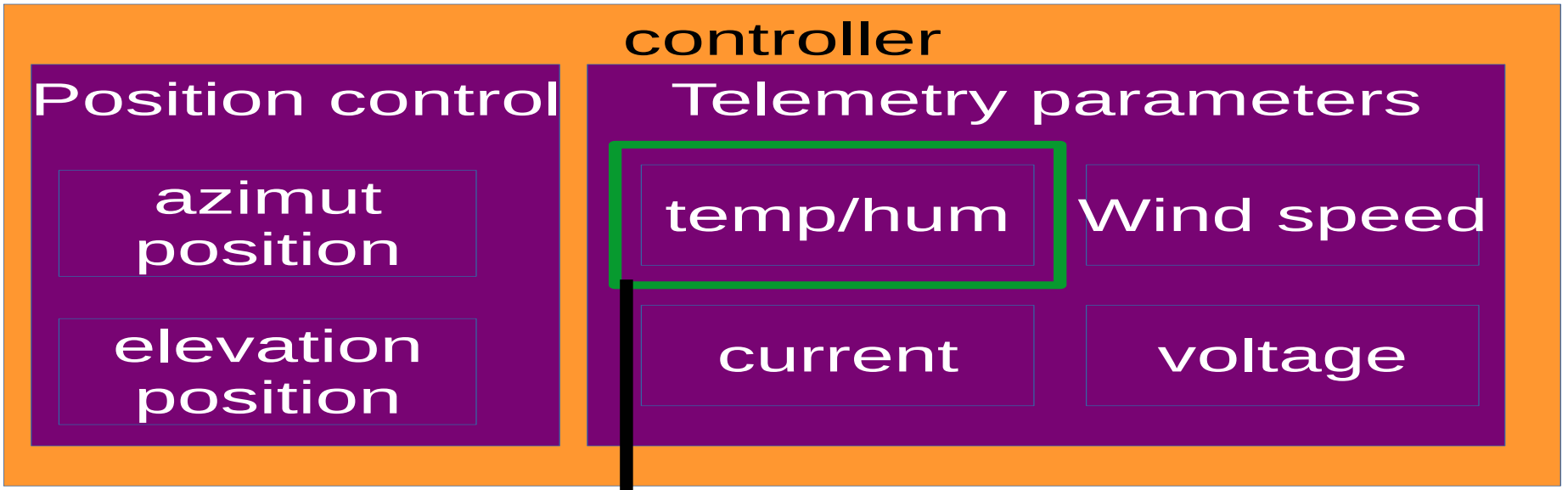
# Position Control

The controller layer obtains environmental parameters and applies the control algorithm. A distributed system has been created using Raspberry Pi Pico (RP2040 microcontroller).



# Position Control

Example reports generate automatically using ceedling (reports for Code Coverage using Ceedling).




**GCC Code Coverage Report**

Directory: src/
Date: 2022-11-07 16:38:26
Coverage: low: ≥ 0% medium: ≥ 75.0% high: ≥ 90.0%

| | Exec | Total | Coverage |
|---|---|---|---|
| Lines: | 27 | 27 | 100.0% |
| Functions: | 7 | 7 | 100.0% |
| Branches: | 2 | 2 | 100.0% |

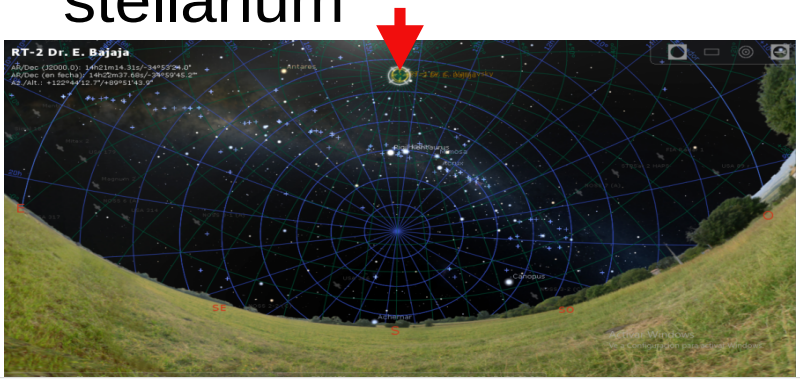
List of functions

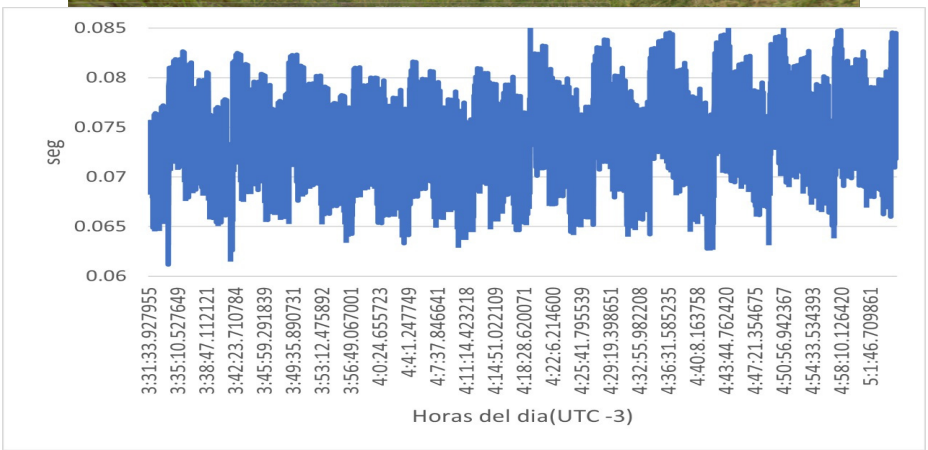
| File | Lines | | Functions | | Branches | |
|---|---|---|---|---|---|---|
| dht11.c | 100.0% | 27 / 27 | 100.0% | 7 / 7 | 100.0% | 2 / 2 |

# Position Control

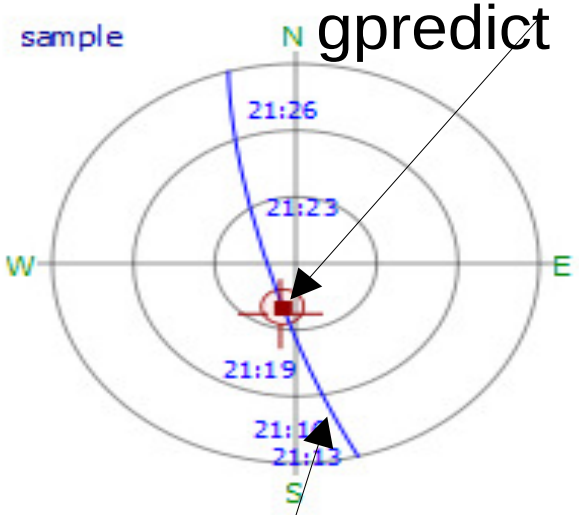
Connect antenna pointing to Gpredict and Stellarium software. Compute sidereal time using eight 8 bits microcontroller for a test and error obtained is 0.07 s .RP2040 is a 32 bits it is expected to improve the error. Testbench using astropy and algoritms for a calculus sidereal time.

antenna pointing
stellarium



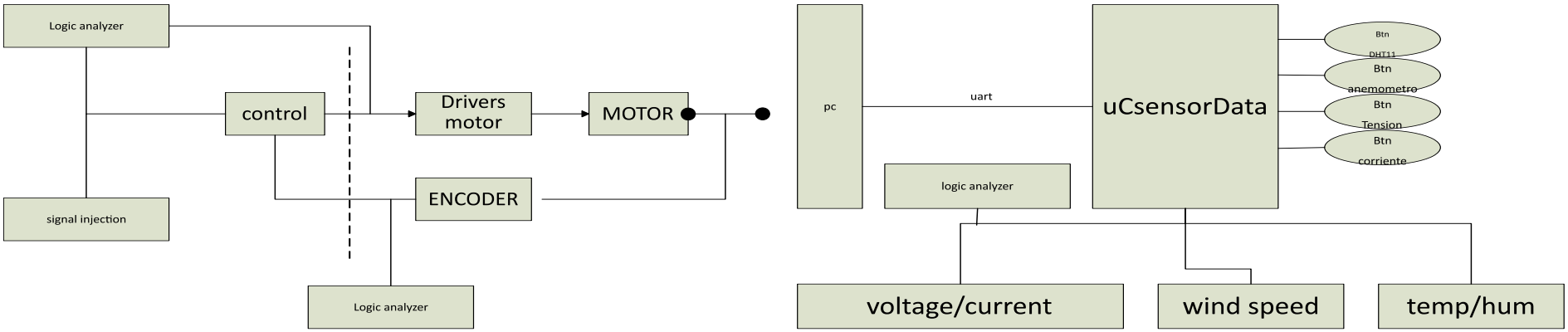

antenna pointing
gpredict




trajectory path

# Position Control

Tests for a final system controller.




Next steps:

- create a software diagrams (UML,state machines, etc) for a total system
- Talk to system management to define terminal interfaces.
- Create a test for a diagrams software.
- Integration systems with a mechanical movement antenna and test using a radiosource.
- Create repository to share with the community(TBD)

# Conclusiones

1. C1

# Conclusiones

1. C1
2. C2

# Conclusiones

1. C1
2. C2
3. C3

# Conclusiones

1. C1
2. C2
3. C3
4. C4

**Questions** $\cdots$