Cosmic ray measurements in automation cycle using Python programming

To do: To create a program that automatically reads on line measurements via a USB cable and virtual COM port, from CosmicWatch detectors, in Python language

September 28, 2020

Project Supervisors:

PhD Marcin Bielewicz

MSc. Arkadiusz Chłopik

Paweł Pietrzak

2nd year of bachelor

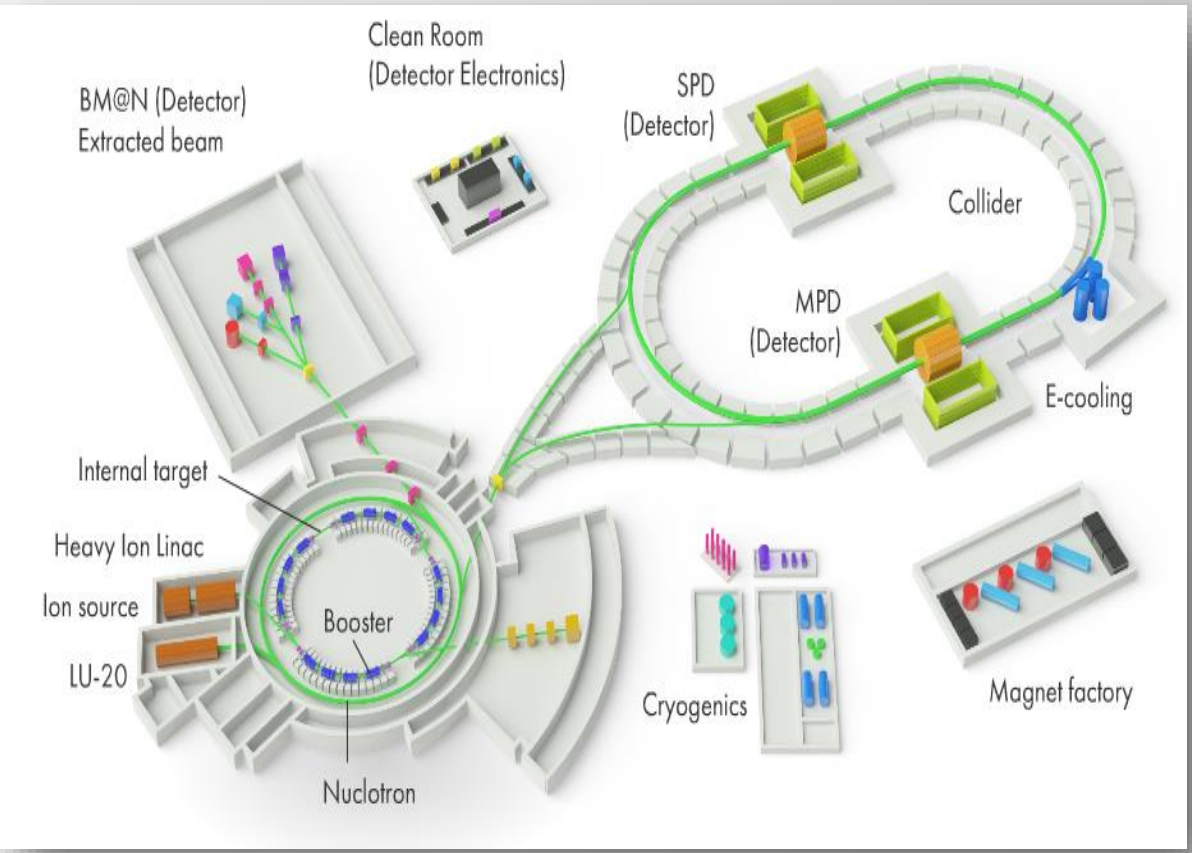
Aerospace Engineering

Faculty of Power and Aeronautical Engineering

Warsaw University of Science and Technology

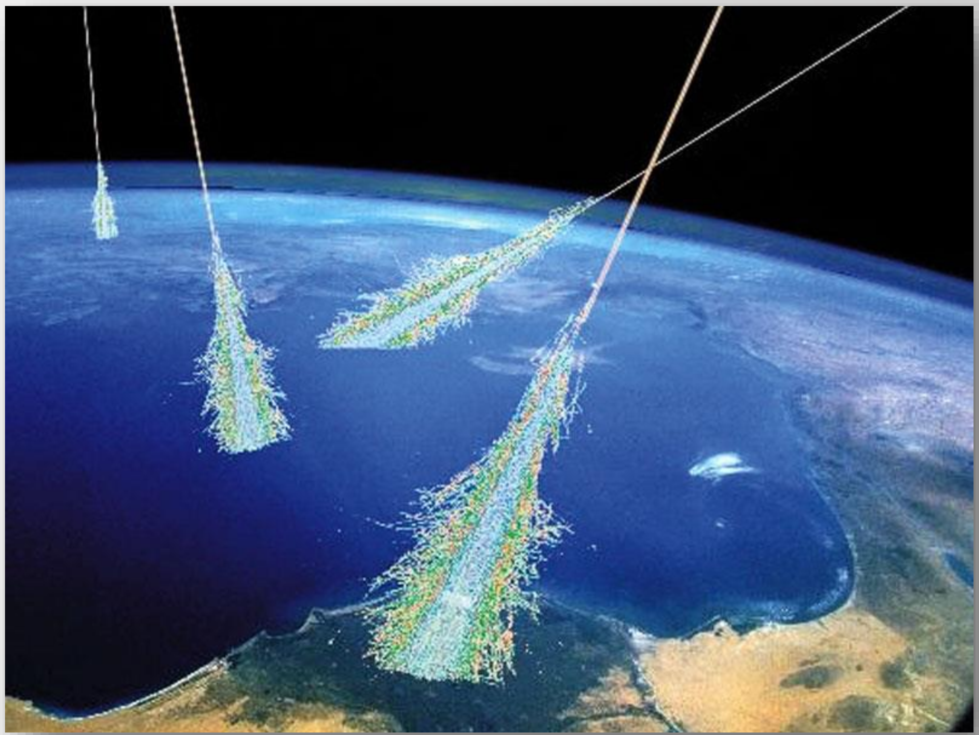
CosmicWatch at NICA

- Construction of a new accelerator complex at Joint Institute for Nuclear Research
- Multi-Purpose Detector (MPD) and MPD Cosmic Ray Detector (MCORD)
- Usage of CosmicWatch detectors for initial background cosmic ray measurements during designing and construction of MCORD detector



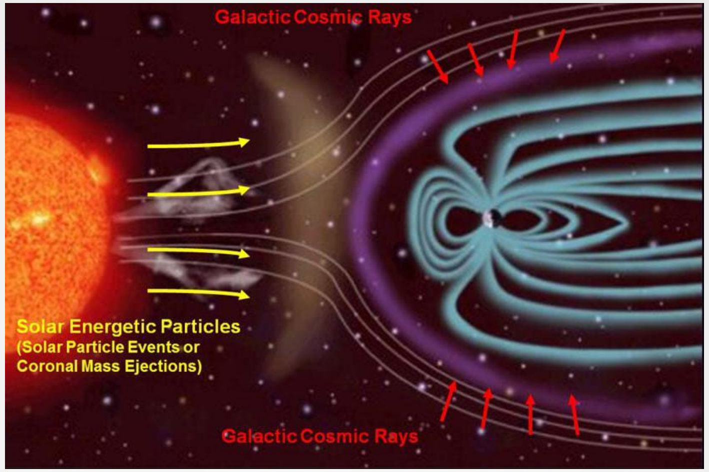
Cosmic rays

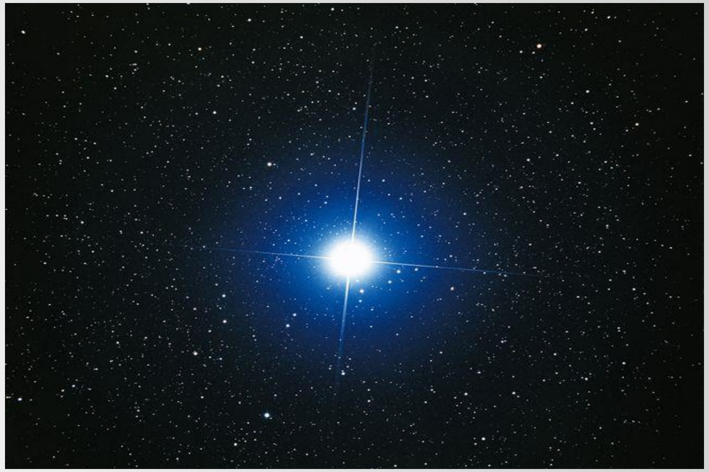
- A cosmic ray is a high-speed particles that travels throughout the universe
- There are two parts of cosmic radiation: primary and secondary



Primary radiation

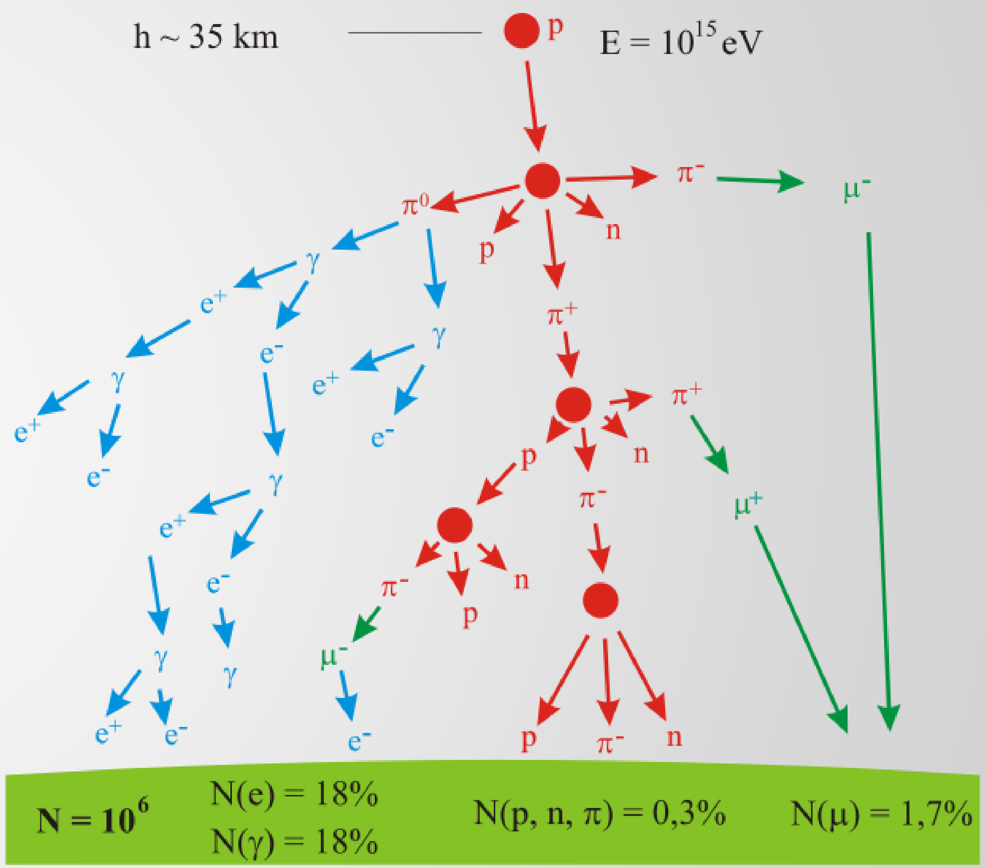
- Primary radiation:
  - 90% protons
  - 9% alpha particles
  - ~1% electrons
  - other heavier nuclei
- Primary radiation sources:
  - The Sun
  - Other stars
  - Supernovas
  - Active galactic nuclei (black holes, quasars)





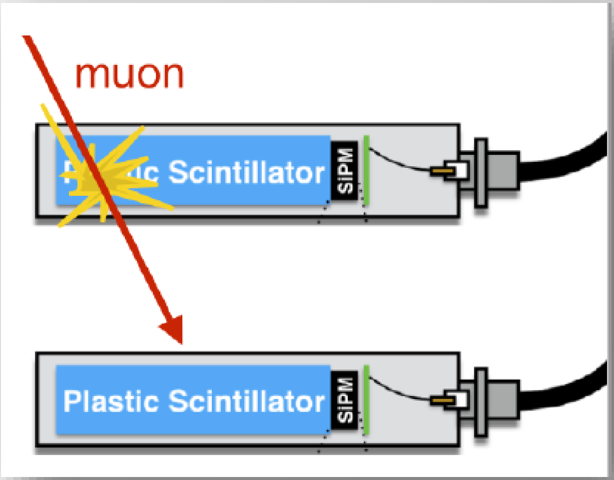
Cosmic showers

- Cosmic showers are cascades of particles and electromagnetic radiation produced in the atmosphere due to interactions of primary cosmic rays with atmospheric particles
- Secondary radiation:
  - muons
  - pions
  - neutrinos
  - electrons
  - gamma rays



GROUND LEVEL

CosmicWatch muon detectors



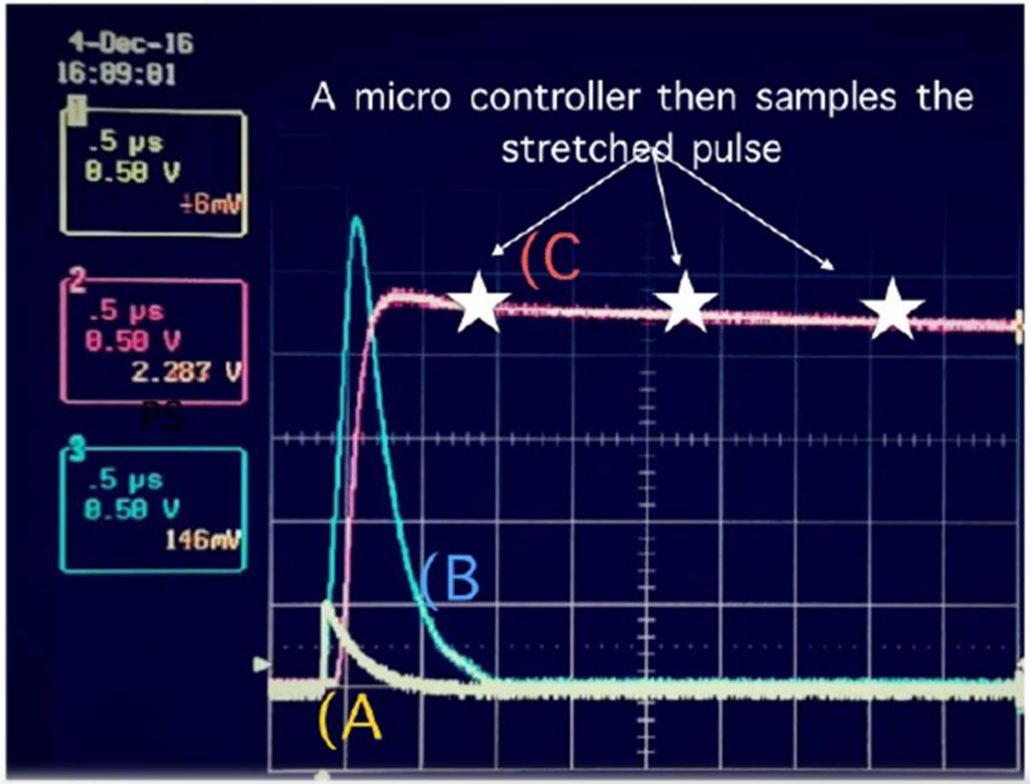
- CosmichWatch is simple, physics-motivated machine- and electronics-shop project for university students and schools
- The muons deposit 1-3 MeV per cm of scintillator.
- The measured voltage depends on the angle of the muon entering the scintillator
- The detectors can work in coincidence mode

Hardware of CosmicWatch

- Plastic scintillator
- SiPM (6mm x 6mm)
- Amplifier with shaper
- Pulse stretcher
- Arduino Nano Board
- SD CARD Reader/Writer
- OLED Display
- Serial data transmission
- COM Port usage

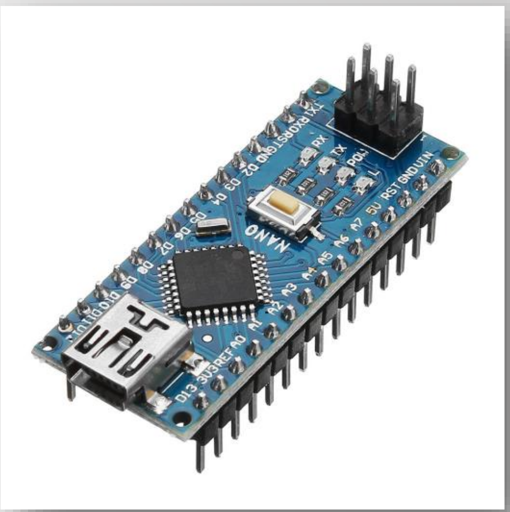
Silicon photomultiplier

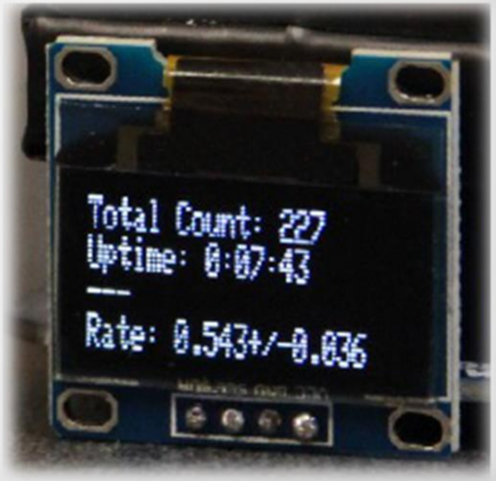
- Silicon photomultipliers, often called SiPM, are solid-state single-photon-sensitive devices
- SiPM produces current pulse (A) in response to absorption of photons emitted by the scintillator
- This pulse is then amplified, output signal (B) has much bigger amplitude, but is still too short to be detected by an analog to digital converter (ADC)
- Pulse stretcher detects the peak amplitude of the amplified pulse and extends its duration (C), so it can be processed by a microcontroller's 10-bit ADC



Arduino Nano Board

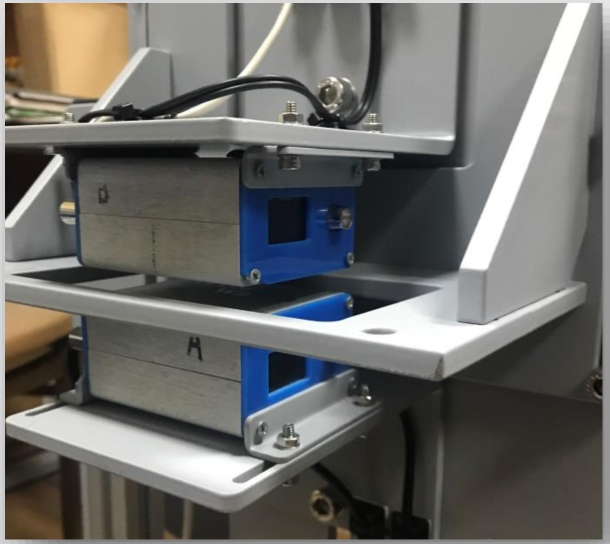
- The Arduino is used to perform several tasks:
  - Set the trigger threshold on the ADC
  - Measure the pulse amplitude from the peak detector circuit
  - Convert the SiPM analog signal to digital
  - Record the time of the event and dead time between events
  - Control the OLED screen and LED light
  - Send the data via USB bridge to a computer (serial COM port)

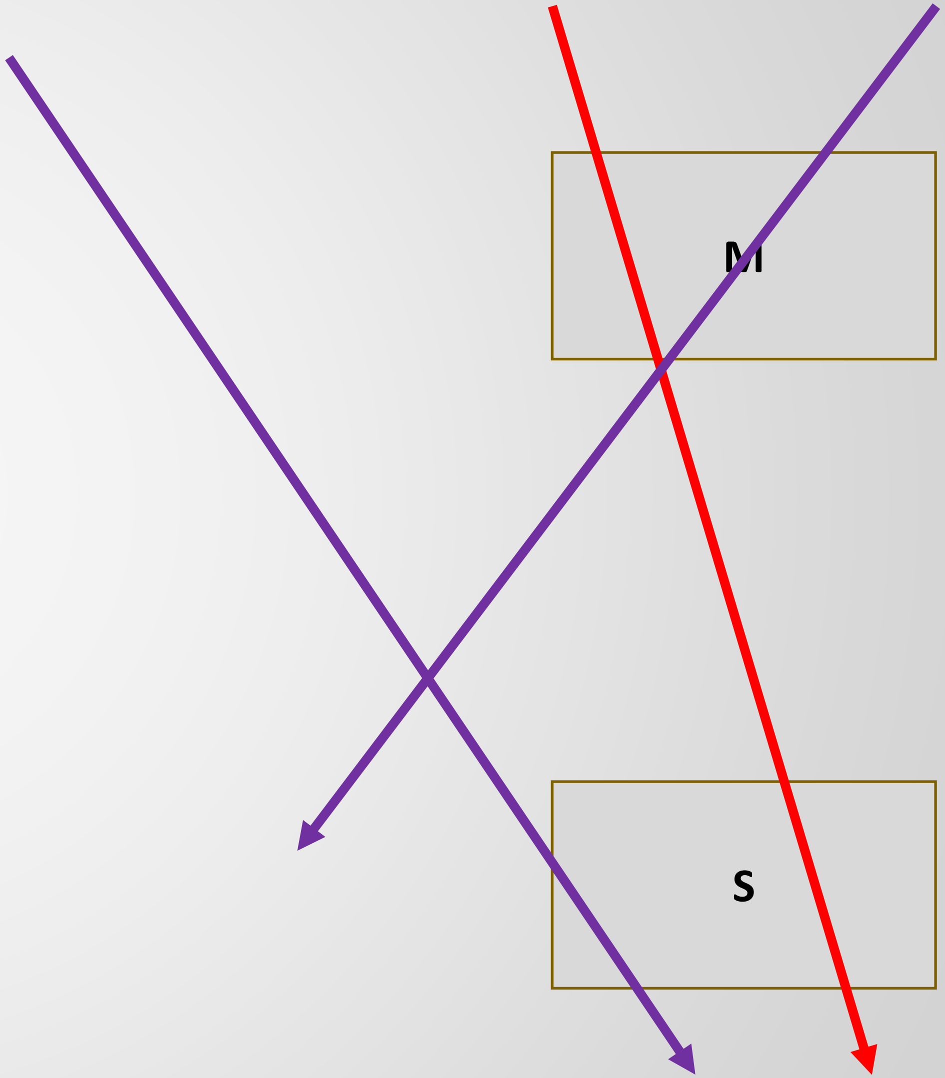




Coincidence mode

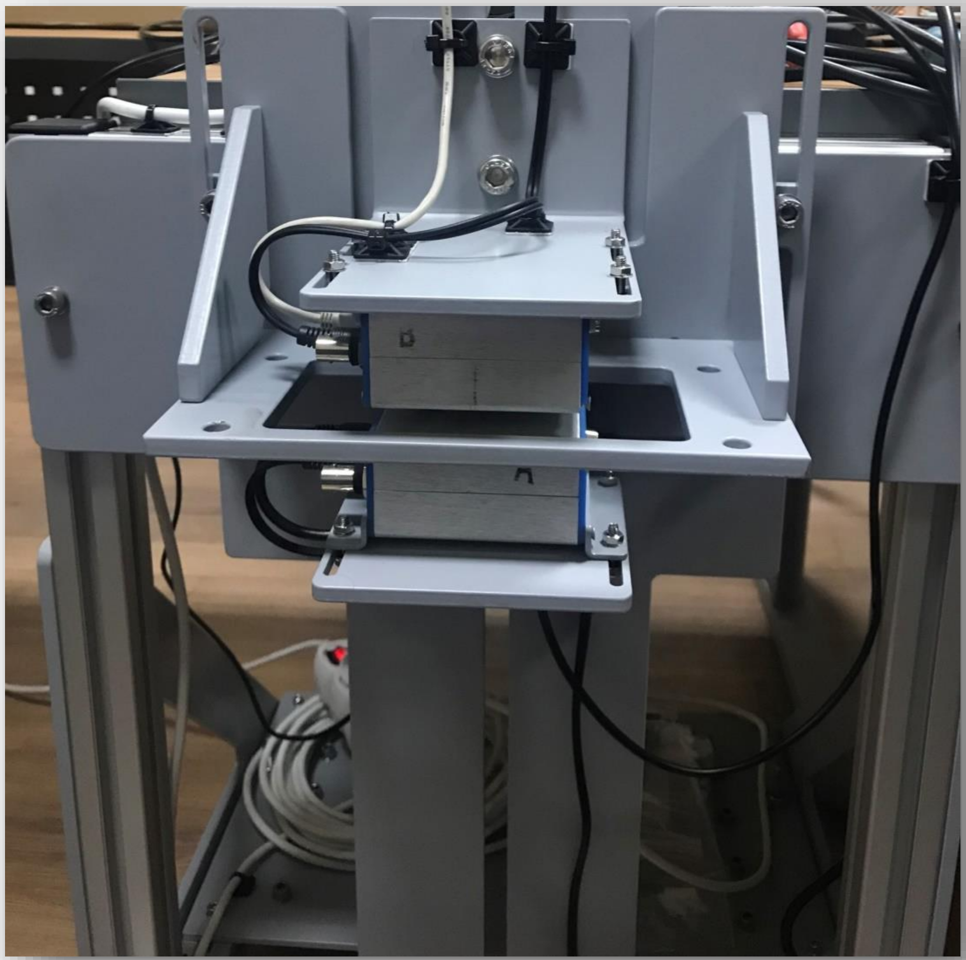
- A pair of detectors can be connected in coincidence mode
- Each device gets assigned one of two roles:
    - Master, which will register all detections
    - Slave, which will only register common detections





Project goals

- Automation of the CosmicWatch data gathering process
- Make CosmicWatch measurements more reliable and convenient
- Archiving and visualisation of collected data



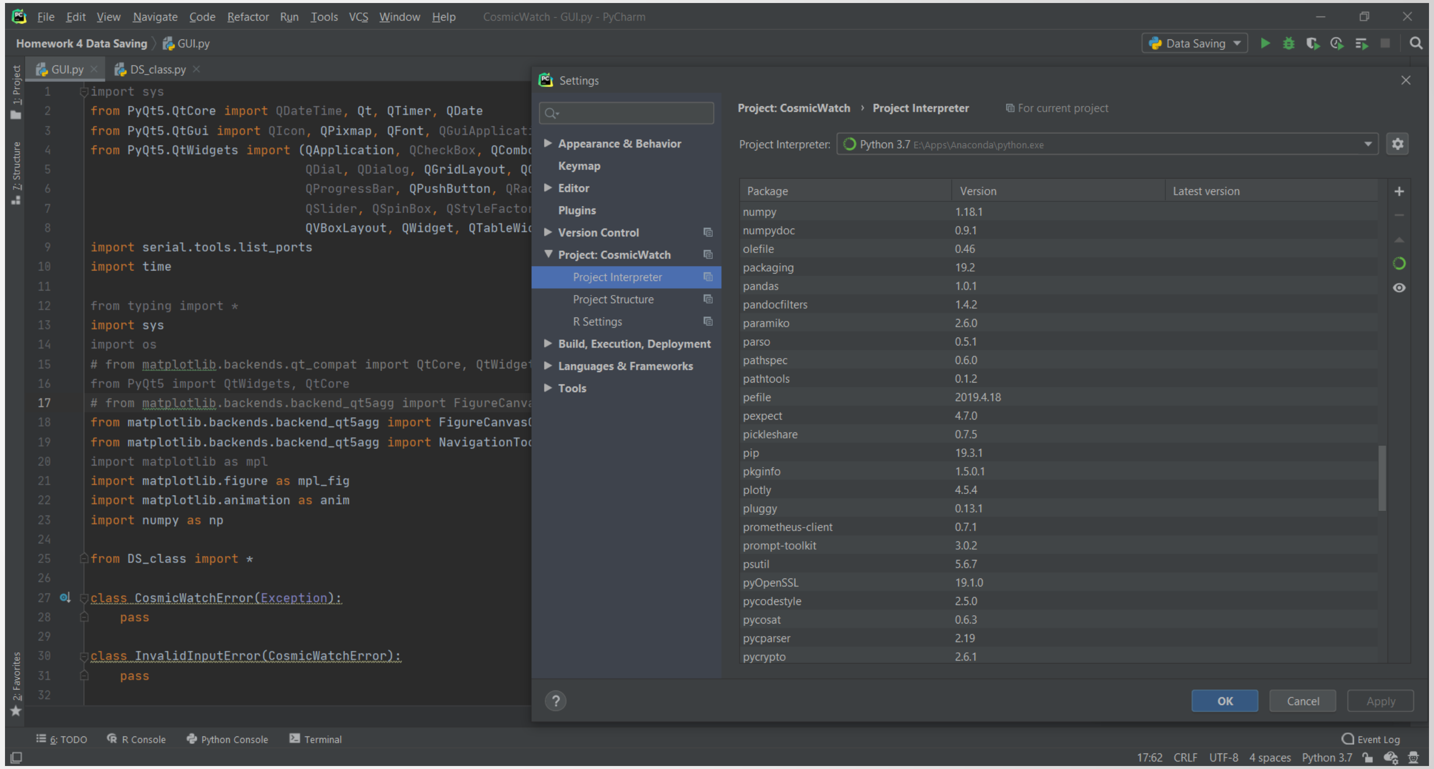
Data saving

- Standard saved data include:
  - Event number
  - Time stamp in milliseconds
  - Digital signal value from 0 to 1023
  - SiPM pulse amplitude
  - Deadtime in milliseconds
  - Temperature in Celsius degree
- Significant data that may be added include:
  - Time of measurements start
  - Detector mode (Master/Slave)
  - Time stamp for each detection
  - Distance and angle between detectors

```
##########################################################################################
### CosmicWatch: The Desktop Muon Detector
### Questions? saxani@mit.edu
### Distance: 5 cm; Angle: 0.0 degrees
### Comp_date Comp_time Event Ardn_time[ms] ADC[0-1023] SiPM[mV] Deadtime[ms] Temp[C]
##########################################################################################
DetectorID: NCBJ_026
DetectorMode: Slave
2020-09-07 10:42:59.356382 1 3448 158 27.53 0 23.15
2020-09-07 10:43:03.717653 2 7813 204 34.73 5 23.05
2020-09-07 10:43:05.712893 3 9810 163 28.20 12 22.94
2020-09-07 10:43:19.964938 4 24074 192 32.67 23 23.58
2020-09-07 10:43:21.706669 5 25816 344 64.67 28 23.15
2020-09-07 10:43:23.174349 6 27286 369 73.42 34 23.15
2020-09-07 10:43:30.204000 7 34322 158 27.53 47 22.94
```

Used software

- Python 3.7 programming language
- PyCharm Community Edition 2020
- Python modules:
  - PySerial
  - PyQt5
  - NumPy
  - Matplotlib
  - Pandas



Python 3.7

- Python is an interpreted high-level programming language with dynamic semantics. It supports object-oriented and functional programming
- Main Python features:
  - Open source project
  - Totally free
  - Easy to read code syntax
  - Extensive error handling capabilities
  - Simple debugging
  - Huge number of modules

PyCharm

- PyCharm is an Integrated Development Environment (IDE) for many languages including Python
- It has an intelligent code editor with error detection and fast refactoring
- Also an integrated debugger, a memory usage profiler, a built-in terminal and a test runner

PySerial

- PySerial is a library, which provides support for serial communication
- We use it to communicate with CosmicWatch detectors over COM ports

PyQt5

- PyQt is a module allowing usage of the Qt framework in Python language
- Qt is a cross-platform framework used to construct graphical user interfaces (GUI) and provides a simple way to implement many interactable UI elements, like buttons, lists or charts and many more

NumPy

- NumPy is a package for scientific computing. Its main feature is a support for multidimensional arrays, matrices and mathematical operations on them

Matplotlib

- Matplotlib is a comprehensive library for creating static, animated and interactive visualizations
- It supports many operating systems and frameworks for creating the GUIs
- Allows to apply many customization options
- We use it for plotting data charts

Pandas

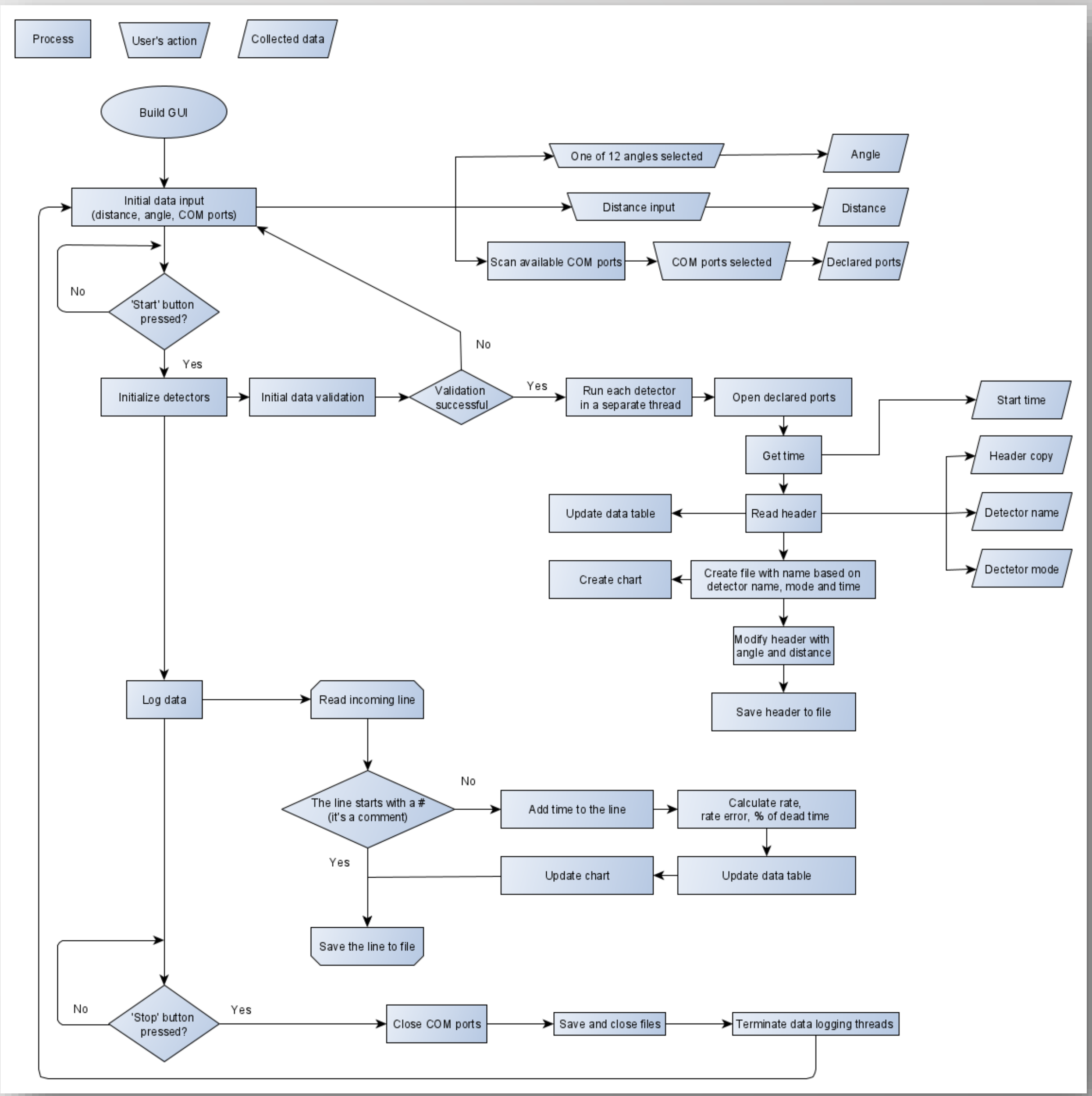
- Pandas is a library for data manipulation and analysis
- It provides a DataFrame object for efficient data manipulation, supports reading and writing to most popular data file formats and other tools allowing operations on data sets

Program goals

- Read serial data from two CosmicWatch detectors set up in coincidence mode
- Live display of data:
  - Detector name and status
  - Pulse amplitude
  - Event time and number
  - Rate and rate error
- Graphical display of data collected by each detector
- Allow to input user data
  - Angle
  - Distance
  - COM Port selection

How does the program work?

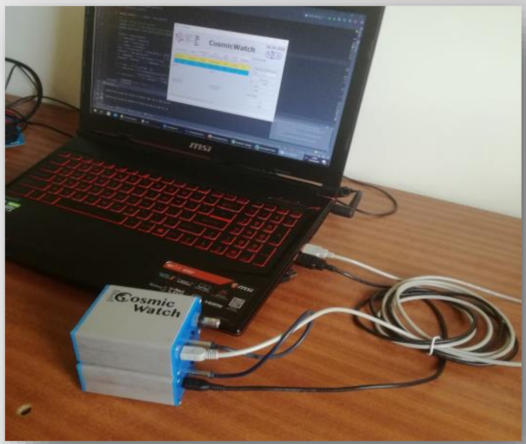
- The program block diagram
- Four main stages:
  1. Collecting initial data
  2. Initializing detectors
  3. Logging data loop
  4. Safe program stop



Program demonstration

Cooperation with Mrs. Justyna Jaczewska from NCBJ for program compatibility

- Mrs. Jaczewska is an author of a similar Java application designed to be used by high school students
- Establishing a convention to make programs compatible on data level
  - File name formatting
  - Order of default data columns
  - Data delimiter
- The ideas and programmatic solutions exchange
- Differences between our Arduino programs
  - Example: printing „Detector ID: XYZ” vs „Device’s Name: XYZ”



Detectors connected in coincidence mode

```
##################################################################################
### CosmicWatch: The Desktop Muon Detector
### Questions? saxani@mit.edu
### Comp_date Comp_time Event Ardn_time[ms] ADC[0-1023] SiPM[mV] Deadtime[ms] Temp[C]
##################################################################################
Device's name: NCBJ_021
2020-08-17 14-24-13-724964 1 4478 103 22.23 0 22.51
2020-08-17 14-24-13-960334 2 4714 121 23.66 5 22.29
2020-08-17 14-24-18-581812 3 9340 287 50.23 10 22.62
```

Old header

```
##################################################################################
### CosmicWatch: The Desktop Muon Detector
### Questions? saxani@mit.edu
### Distance: 5 cm; Angle: 0.0 degrees
### Comp_date Comp_time Event Ardn_time[ms] ADC[0-1023] SiPM[mV] Deadtime[ms] Temp[C]
##################################################################################
DetectorID: NCBJ_026
DetectorMode: Slave
2020-09-01 20:00:15.698426  1 13234 192 32.67 2962 22.72
2020-09-01 20:00:50.889173  2 48460 159 27.66 9375 23.58
```

Header modified for uniformity

Program development

- Saving log file with:
  - Time log of all user actions
  - Program warning, alarms and displayed errors
- New functions:
  - Displaying multiple waveforms on a single graph
  - New types of charts (for data analysis)
- Error handling
- Software testing and error debugging

References

M. Bielewicz et al., MCORD - MPD Cosmic Ray Detector a new features, EPJ Web of Conferences 204, 07016 (2019)

Lectures of Dr. M. Bielewicz at Astronomical Camp 23.VIII.2019

https://github.com/spenceraxani/CosmicWatch-Desktop-Muon-Detector-v2/blob/master/Instructions.pdf

https://www.python.org/doc/essays/blurb/

https://www.jetbrains.com/pycharm/

https://wiki.python.org/moin/PySerial

https://www.qt.io/

https://pypi.org/project/PyQt5/

https://www.riverbankcomputing.com/software/pyqt/

https://matplotlib.org/

https://numpy.org/doc/stable/user/whatisnumpy.html

https://pandas.pydata.org/about/index.html

https://en.wikipedia.org/wiki/Pandas_(software)