Experiment Control System Workshop, April 2023

PIPERADE EPICS Client Side

Dinko Atanasov
atanasov@lp2ib.in2p3.fr

Overview

USER operation of PIPERADE - main tasks

- Easy control of equipment - execution of ToF-ICR, PI-ICR patterns etc
- Scanning of system parameters - RFs, Steerers, Einzel lens etc
- Event Builder - consistent data tree for various measurement modes
- Visualization of data and archiving abilities
- Fast USER interaction and possibility for automation
- Rely on previous experience - MM8 and PyMassScanner

LP2i
Bordeaux

ISOL
TRAP

JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

PIPERADE Mass Scanner - PMS

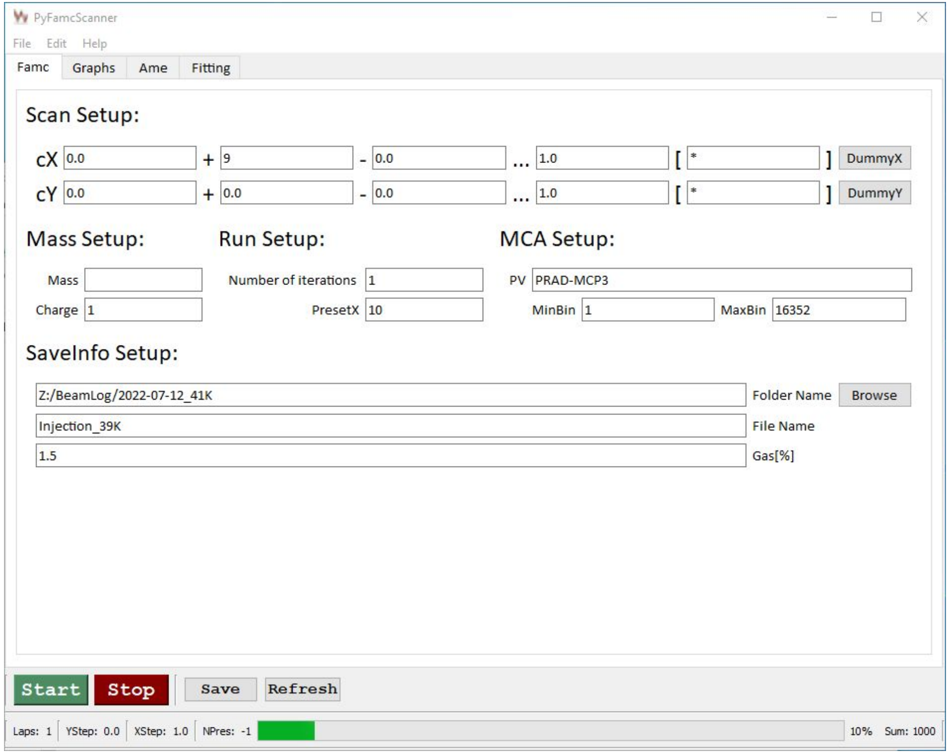
Rely on standard components and libraries:

- Python 3.X - easy to handover (presently only two developers)
- PyQt - cross-platform application development
- Pyqtgraph - Display data (interactive)
- pyEpics - communication layer
- Sphinx - documentation

PMS - USER Interface

Measurement configuration:

- Scan values - static 2D
- Mass of interest loaded from AME/NUBASE file*
- Run setup includes a Preset value
- MCA - access via EPICS PV
- Automation - switching off BL, execution of preconfigured measurements in series



Scan Setup:

cX 0.0 + 9 - 0.0 ... 1.0 [*] DummyX

cY 0.0 + 0.0 - 0.0 ... 1.0 [*] DummyY

Mass Setup:

Mass

Charge 1

Run Setup:

Number of iterations 1

PresetX 10

MCA Setup:

PV PRAD-MCP3

MinBin 1 MaxBin 16352

SaveInfo Setup:

Z:/BeamLog/2022-07-12_41K Folder Name Browse

Injection_39K File Name

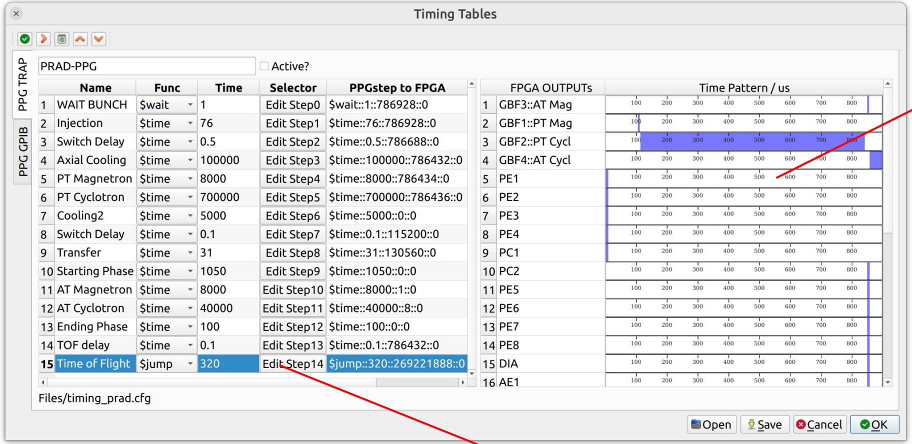
1.5 Gas[%]

Laps: 1 YStep: 0.0 XStep: 1.0 NPres: -1 10% Sum: 1000

PMS - Timing

Visual representation

Mapping to hardware

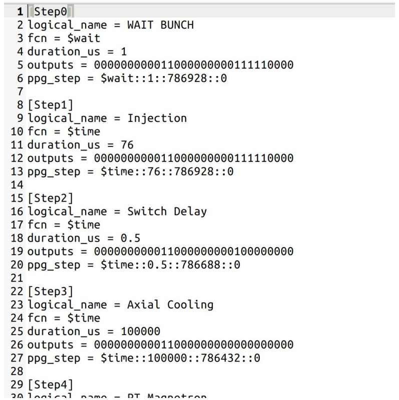


PRAD-PPG □ Active?

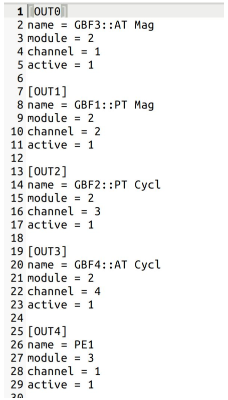
	Name	Func	Time	Selector	PPGstep to FPGA
1	WAIT BUNCH	$wait	1	Edit Step0	$wait::1::786928::0
2	Injection	$time	76	Edit Step1	$time::76::786928::0
3	Switch Delay	$time	0.5	Edit Step2	$time::0.5::786688::0
4	Axial Cooling	$time	100000	Edit Step3	$time::100000::786432::0
5	PT Magnetron	$time	8000	Edit Step4	$time::8000::786434::0
6	PT Cyclotron	$time	700000	Edit Step5	$time::700000::786436::0
7	Cooling2	$time	5000	Edit Step6	$time::5000::0::0
8	Switch Delay	$time	0.1	Edit Step7	$time::0.1::115200::0
9	Transfer	$time	31	Edit Step8	$time::31::130560::0
10	Starting Phase	$time	1050	Edit Step9	$time::1050::0::0
11	AT Magnetron	$time	8000	Edit Step10	$time::8000::1::0
12	AT Cyclotron	$time	40000	Edit Step11	$time::40000::8::0
13	Ending Phase	$time	100	Edit Step12	$time::100::0::0
14	TOF delay	$time	0.1	Edit Step13	$time::0.1::786432::0
15	Time of Flight	$jump	320	Edit Step14	$jump::320::269221888::0

	FPGA OUTPUTs
1	GBF3::AT Mag
2	GBF1::PT Mag
3	GBF2::PT Cycl
4	GBF4::AT Cycl
5	PE1
6	PE2
7	PE3
8	PE4
9	PC1
10	PC2
11	PE5
12	PE6
13	PE7
14	PE8
15	DIA
16	AE1

Files/timing_prad.cfg

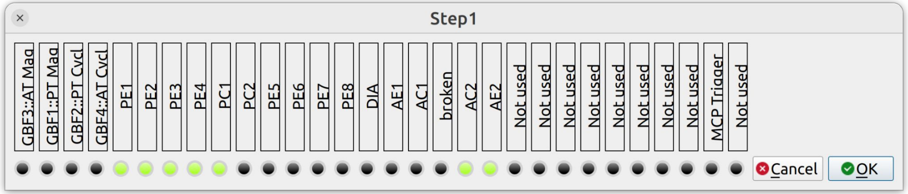


```
[Step0]
logical_name = WAIT BUNCH
fcn = $wait
duration_us = 1
outputs = 00000000001100000000111110000
ppg_step = $wait::1::786928::0

[Step1]
logical_name = Injection
fcn = $time
duration_us = 76
outputs = 00000000001100000000111110000
ppg_step = $time::76::786928::0

[Step2]
logical_name = Switch Delay
fcn = $time
duration_us = 0.5
outputs = 00000000001100000000100000000
ppg_step = $time::0.5::786688::0

[Step3]
logical_name = Axial Cooling
fcn = $time
duration_us = 100000
outputs = 00000000001100000000000000000
ppg_step = $time::100000::786432::0

[Step4]
logical_name = PT Magnetron
```



```
[OUT0]
name = GBF3::AT Mag
module = 2
channel = 1
active = 1

[OUT1]
name = GBF1::PT Mag
module = 2
channel = 2
active = 1

[OUT2]
name = GBF2::PT Cycl
module = 2
channel = 3
active = 1

[OUT3]
name = GBF4::AT Cycl
module = 2
channel = 4
active = 1

[OUT4]
name = PE1
module = 3
channel = 1
active = 1
```

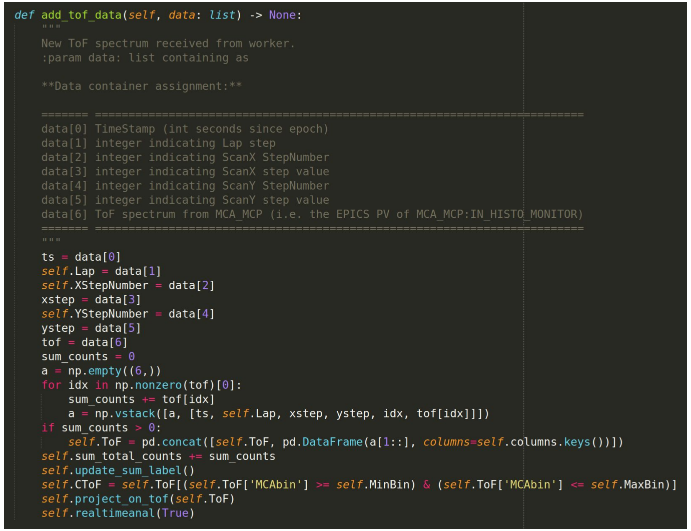
Edit individual step



PMS - Event builder and Archiving

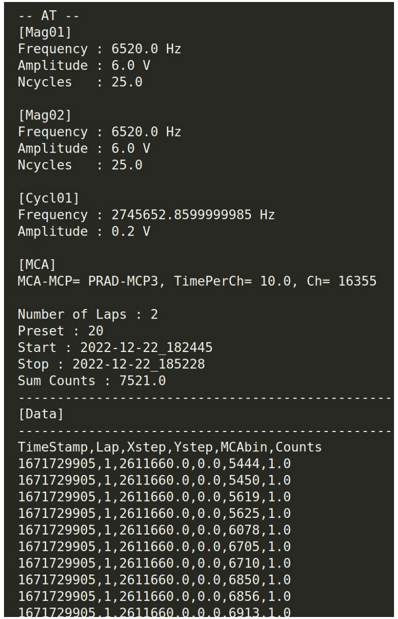
- Events received from Sequencer thread
- Filtering of usable data
- Storing the data in a Pandas container
- Archiving done via Pandas CSV

Main event building part



```
def add_tof_data(self, data: list) -> None:
    """
    New ToF spectrum received from worker.
    :param data: list containing as

    **Data container assignment:**

    ======= ========================================================================
    data[0] TimeStamp (int seconds since epoch)
    data[1] integer indicating Lap step
    data[2] integer indicating ScanX StepNumber
    data[3] integer indicating ScanX step value
    data[4] integer indicating ScanY StepNumber
    data[5] integer indicating ScanY step value
    data[6] ToF spectrum from MCA_MCP (i.e. the EPICS PV of MCA_MCP:IN_HISTO_MONITOR)
    ======= ========================================================================
    """
    ts = data[0]
    self.Lap = data[1]
    self.XStepNumber = data[2]
    xstep = data[3]
    self.YStepNumber = data[4]
    ystep = data[5]
    tof = data[6]
    sum_counts = 0
    a = np.empty((6,))
    for idx in np.nonzero(tof)[0]:
        sum_counts += tof[idx]
        a = np.vstack([a, [ts, self.Lap, xstep, ystep, idx, tof[idx]]])
    if sum_counts > 0:
        self.ToF = pd.concat([self.ToF, pd.DataFrame(a[1::], columns=self.columns.keys())])
    self.sum_total_counts += sum_counts
    self.update_sum_label()
    self.CToF = self.ToF[(self.ToF['MCAbin'] >= self.MinBin) & (self.ToF['MCAbin'] <= self.MaxBin)]
    self.project_on_tof(self.ToF)
    self.realtimeanal(True)
```

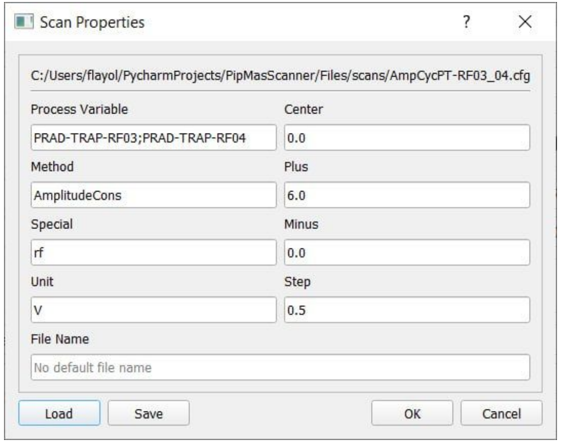
Part of a record



```
-- AT --
[Mag01]
Frequency : 6520.0 Hz
Amplitude : 6.0 V
Ncycles   : 25.0

[Mag02]
Frequency : 6520.0 Hz
Amplitude : 6.0 V
Ncycles   : 25.0

[Cycl01]
Frequency : 2745652.8599999985 Hz
Amplitude : 0.2 V

[MCA]
MCA-MCP= PRAD-MCP3, TimePerCh= 10.0, Ch= 16355

Number of Laps : 2
Preset : 20
Start : 2022-12-22_182445
Stop : 2022-12-22_185228
Sum Counts : 7521.0
-----------------------------------------------------
[Data]
-----------------------------------------------------
TimeStamp,Lap,Xstep,Ystep,MCAbin,Counts
1671729905,1,2611660.0,0.0,5444,1.0
1671729905,1,2611660.0,0.0,5450,1.0
1671729905,1,2611660.0,0.0,5619,1.0
1671729905,1,2611660.0,0.0,5625,1.0
1671729905,1,2611660.0,0.0,6078,1.0
1671729905,1,2611660.0,0.0,6705,1.0
1671729905,1,2611660.0,0.0,6710,1.0
1671729905,1,2611660.0,0.0,6850,1.0
1671729905,1,2611660.0,0.0,6856,1.0
1671729905,1,2611660.0,0.0,6913,1.0
```

PMS - Parameter optimization

- Scanning of multiple PVs, namely electrostatic elements, RF Gen etc
- Special methods available - Timing, RF phase

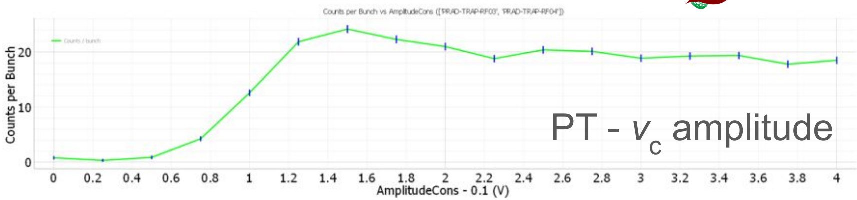


C:/Users/flayol/PycharmProjects/PipMasScanner/Files/scans/AmpCycPT-RF03_04.cfg

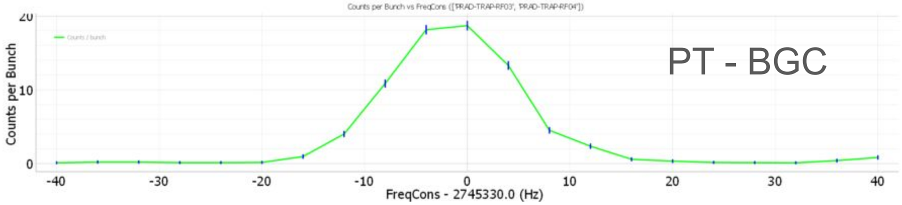
Process Variable	Center
PRAD-TRAP-RF03;PRAD-TRAP-RF04	0.0
Method	Plus
AmplitudeCons	6.0
Special	Minus
rf	0.0
Unit	Step
V	0.5

File Name

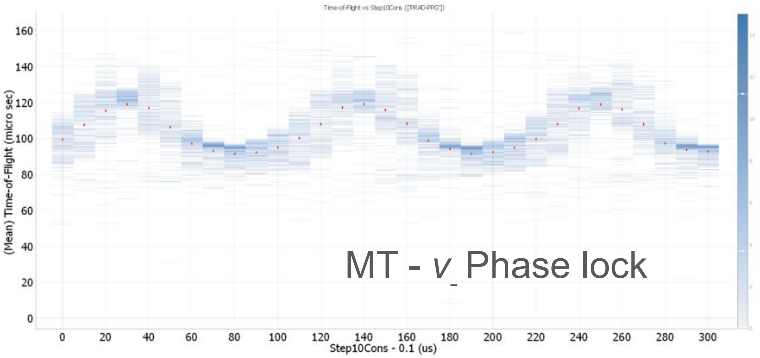
No default file name



PT - v_c amplitude

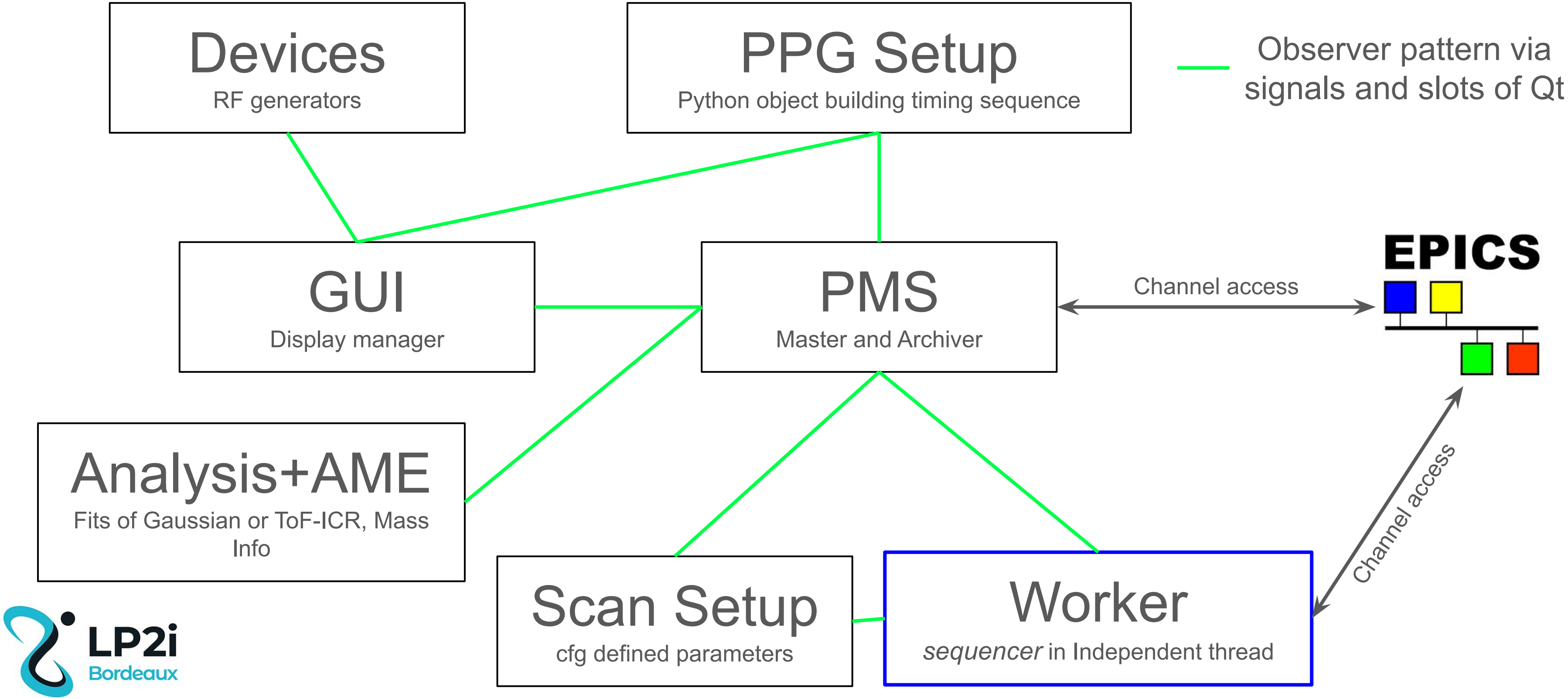


PT - BGC



MT - v_- Phase lock

PMS - behind the scenes



Devices

RF generators

PPG Setup

Python object building timing sequence

Observer pattern via signals and slots of Qt

GUI

Display manager

PMS

Master and Archiver

Analysis+AME

Fits of Gaussian or ToF-ICR, Mass Info

Scan Setup

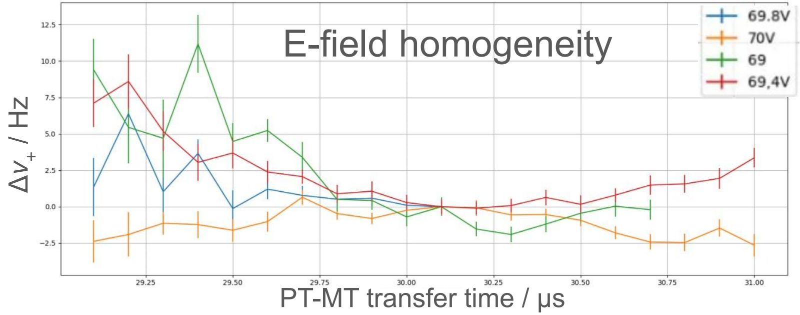
cfg defined parameters

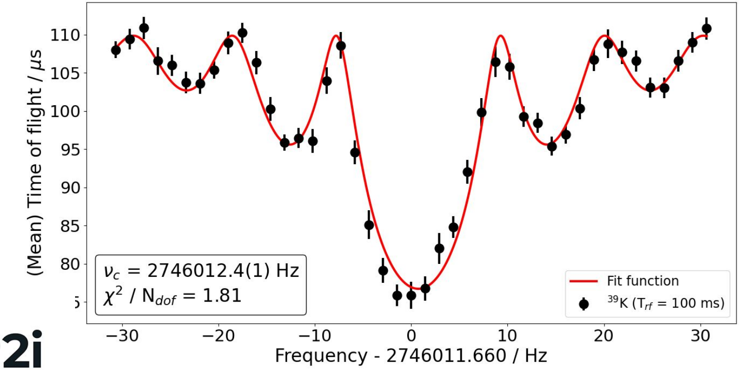
Worker

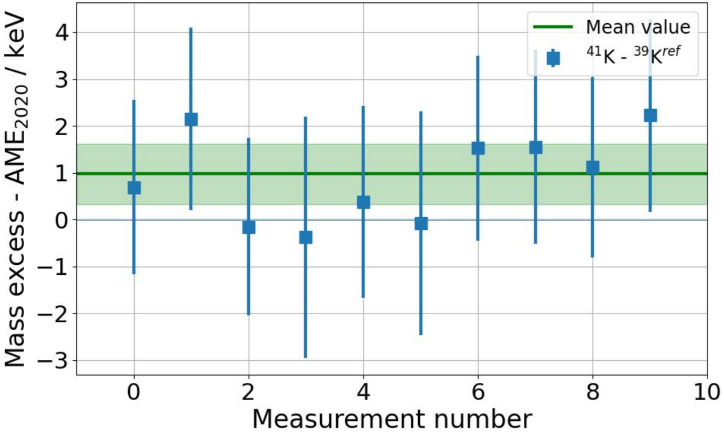
sequencer in Independent thread

ToF-ICR commissioning

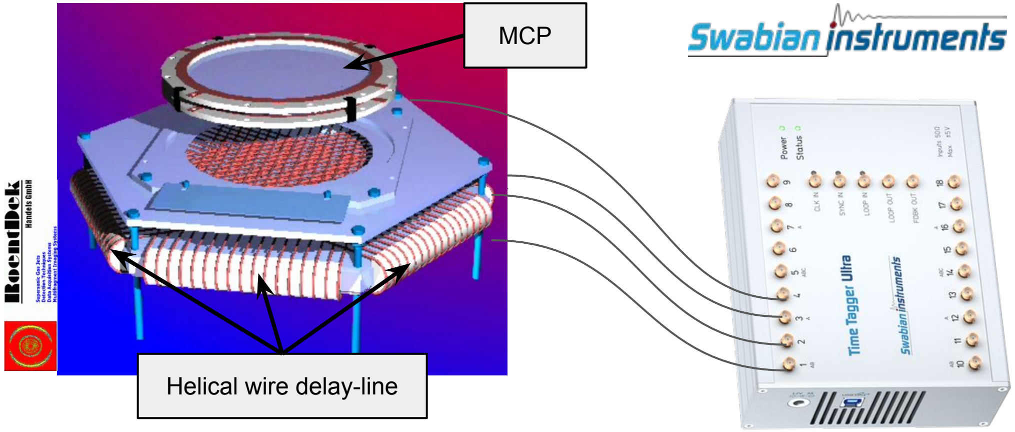
- Cross-checks measurement of $^{39\text{-}41}$K
- Automatic switching between isotopes
- Loading of Archived files and E-B fields
- Verification of fitting routines of ToF-ICR



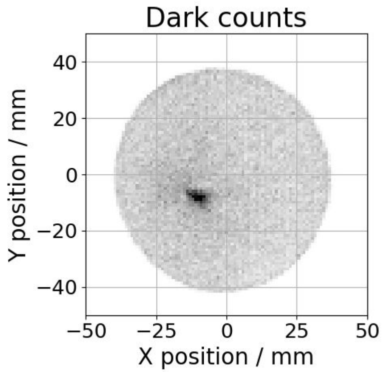




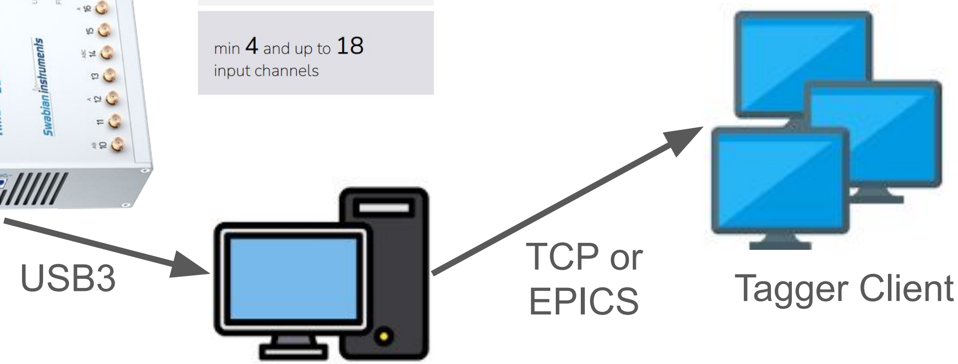
PI-ICR - TDC developments



Performance	Value
8 ps RMS jitter	42 ps RMS jitter
70 M tags/s data transfer rate over USB 3	
min 4 and up to 18 input channels	



Rate of ~ 1 MHz
Delay line signals FWHM ~ 10 ns

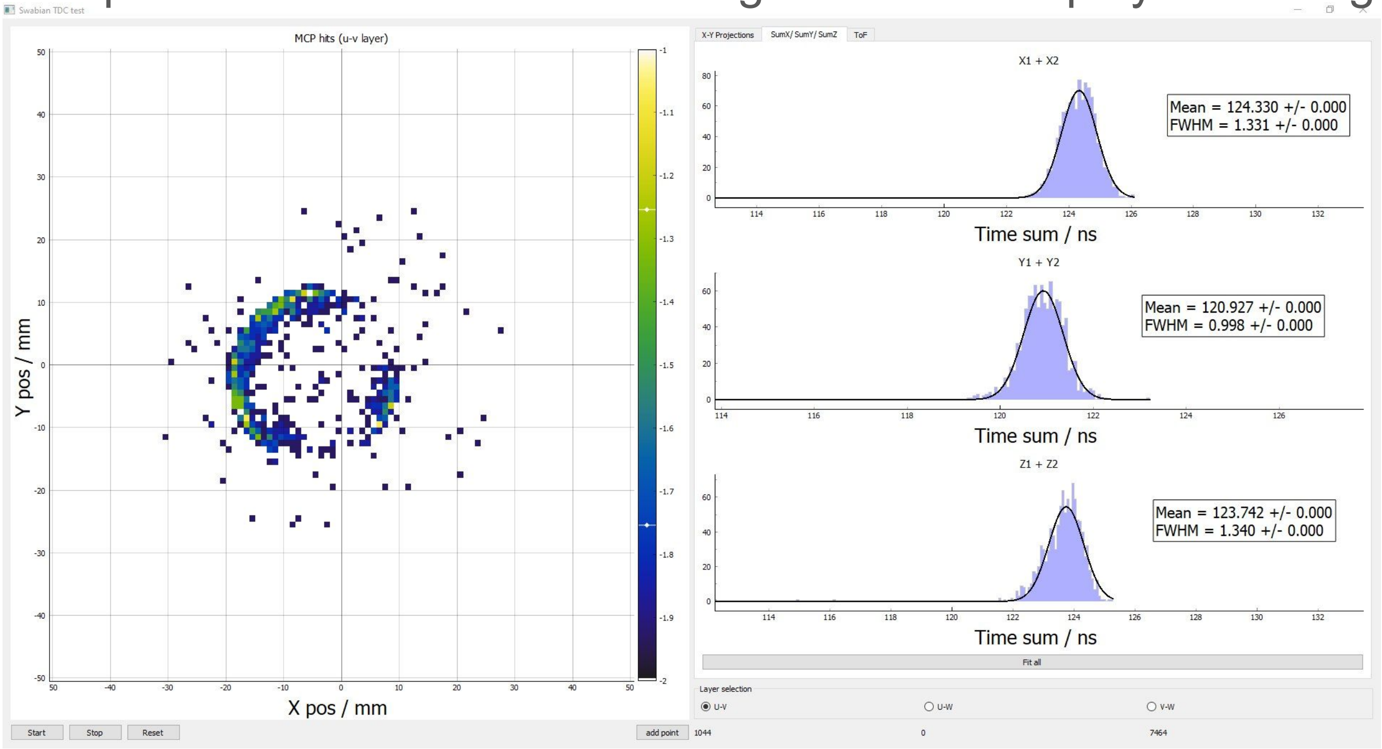


Tagger Server
Python standalone

S. Eliseev et al., Phys. Rev. Lett. 110, 082501 (2013)
http://www.roentdek.com/info/movies/hex_80.htm
https://www.swabianinstruments.com/time-tagger/

PI-ICR - Image display

Client performs the reconstruction algorithm and displays the image



X1 + X2: Mean = 124.330 +/- 0.000
FWHM = 1.331 +/- 0.000

Y1 + Y2: Mean = 120.927 +/- 0.000
FWHM = 0.998 +/- 0.000

Z1 + Z2: Mean = 123.742 +/- 0.000
FWHM = 1.340 +/- 0.000

Layer selection

● U-V ○ U-W ○ V-W

1044 0 7464

Conclusions and Outlook

- Developed python-based client software PMS
- ToF-ICR measurements integrated and optimization continue
- PI-ICR developments are well under way
  - Delay line of HEX detector installed and working within specifications
  - Next step is the integration of the reconstruction package into PMS
- Adjusting the PMS Archiver to handle PI-ICR and ToF-ICR data

Thank you

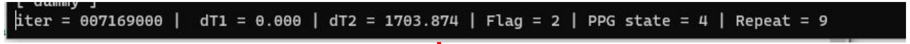
PIPERADE team:
P. Alfaurt, P. Ascher, D. Atanasov, L. Daudin, M. Flayol, M. Gerbaux, S. Grévy, M. Hukkanen, A. Husson, B. Lachacinski and A. de Roubin

PMS developm
M. Flayol and D. Atanasov

PPG speed via EPICS

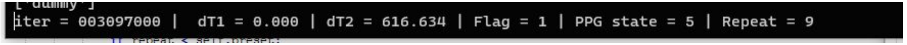
TRAP-PPG Labview Control software:
an execution step which loads the timing pattern to the FPGA waits at the end for one second.

The function was executed at each START event, i.e. if one clicks on the start button or



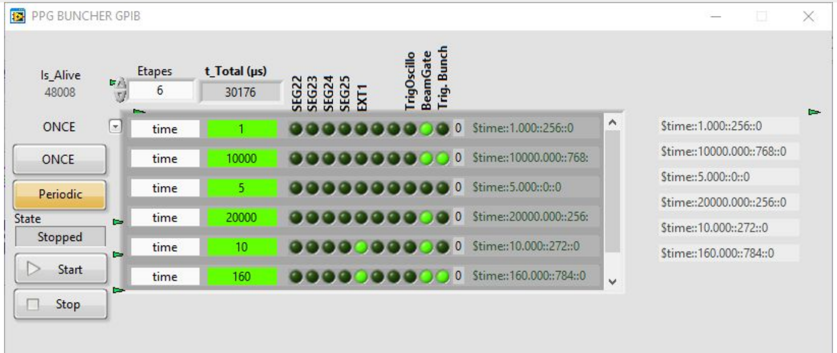
```
iter = 007169000 |  dT1 = 0.000 | dT2 = 1703.874 | Flag = 2 | PPG state = 4 | Repeat = 9
```

Reduction of wait function from 1s to 10ms



```
iter = 003097000 |  dT1 = 0.000 | dT2 = 616.634 | Flag = 1 | PPG state = 5 | Repeat = 9
```

dT2 shows the time delta between EPICS caput command to start and EPICS camonitor (callback function)



PPG BUNCHER GPIB

Is_Alive 48008

Etapes 6

t_Total (µs) 30176

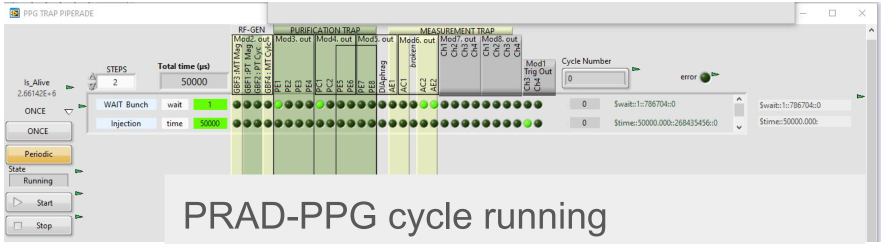
ONCE

State Stopped

time	1	$time::1.000::256::0
time	10000	$time::10000.000::768:
time	5	$time::5.000::0::0
time	20000	$time::20000.000::256:
time	10	$time::10.000::272::0
time	160	$time::160.000::784::0

$time::1.000::256::0
$time::10000.000::768::0
$time::5.000::0::0
$time::20000.000::256::0
$time::10.000::272::0
$time::160.000::784::0

GPIB-PPG cycle running at ~30 ms total time.



PPG TRAP PIPERADE

Is_Alive 2.66142E+6

STEPS 2

Total time (µs) 50000

Cycle Number 0

ONCE

State Running

WAIT Bunch	wait	1	0	$wait::1::786704::0
Injection	time	50000	0	$time::50000.000::268435456::0

$wait::1::786704::0
$time::50000.000:

PRAD-PPG cycle running
with a wait on GPIB-PPG trigger*
Cycle length is 50 ms