The (upcoming) digital real-time positron lifetime measurement of EPOS

A. Krille[1] R. Krause-Rehberg[1] F. Becvar[2] G. Brauer[3]

[1]Fachbereich Physik, Martin-Luther-University Halle

[2]Department of Low-Temperature Physics, Charles University Prague

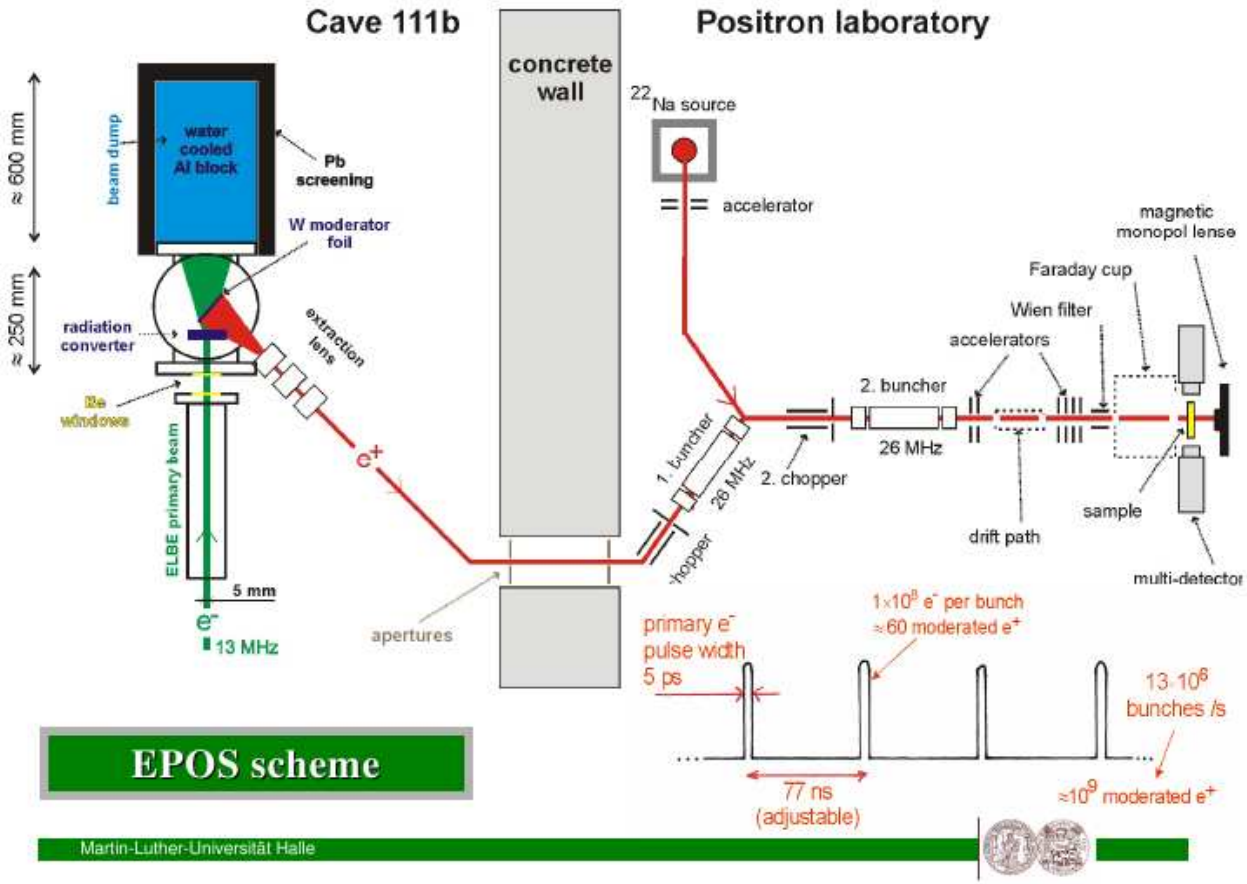
[3]Institute of Ion Beam Physics and Materials Research, Research Center Rossendorf

The application of high intensity positron beam techniques and digital lifetime positron spectroscopy in material science, Bergen (Netherlands) 2005

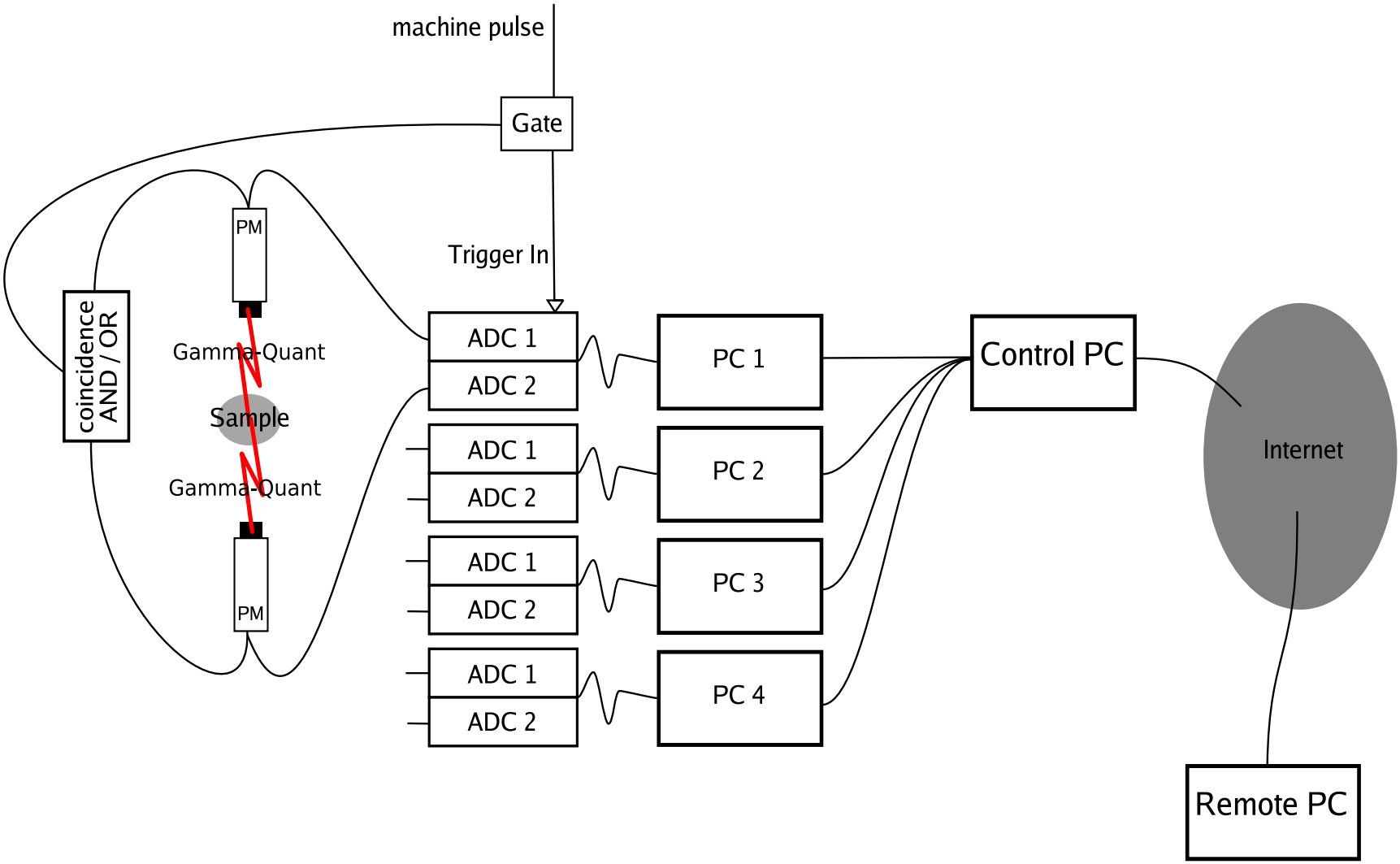
Outline

Hardware:

Starting at yesterdays talk of R. Krause-Rehberg:



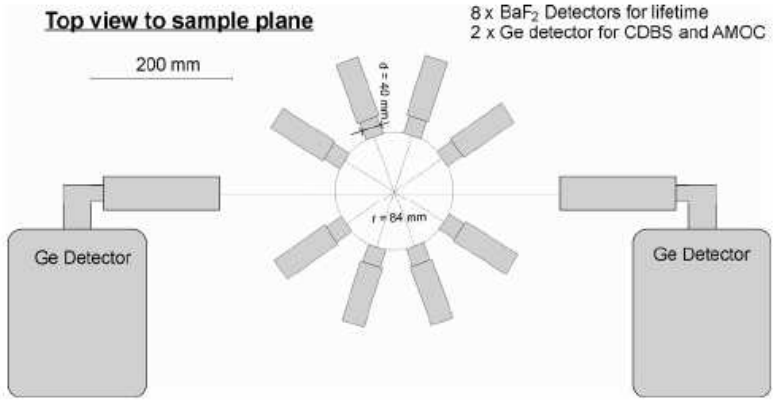
Hardware


machine pulse
Gate
Trigger In
PM
coincidence AND / OR
Gamma-Quant
Sample
Gamma-Quant
PM
ADC 1
ADC 2
ADC 1
ADC 2
ADC 1
ADC 2
ADC 1
ADC 2
PC 1
PC 2
PC 3
PC 4
Control PC
Internet
Remote PC

Hardware: Detectors

8 detectors (BaF_2-SEV + Hamamatsu PM) for lifetime measurement

- two modes of operation:
  - coincidence (AND)
    less background-noise
  - single (OR)
    faster measurement
- 2 additional Ge-detectors for Doppler-measurements



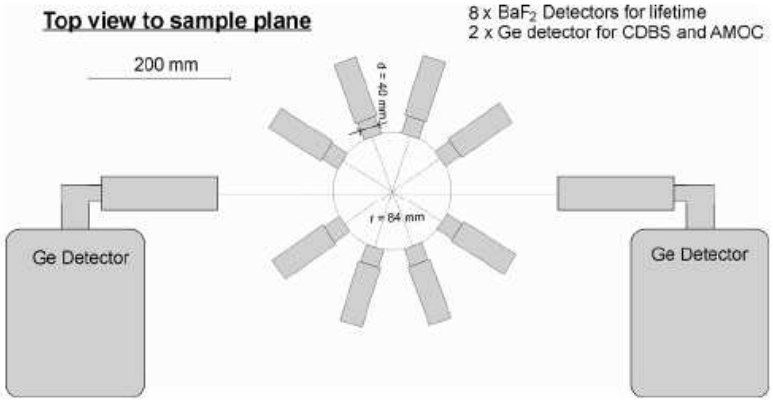
Start-signal from machine-pulse into trigger input of the digitizers (AND-coupled with coincidence).

No detection of the start-impuls → better time resolution compared to conventional systems.

Hardware: Detectors

8 detectors (BaF_2-SEV + Hamamatsu PM) for lifetime measurement

- two modes of operation:
  - coincidence (AND)
    less background-noise
  - single (OR)
    faster measurement
- 2 additional Ge-detectors for Doppler-measurements



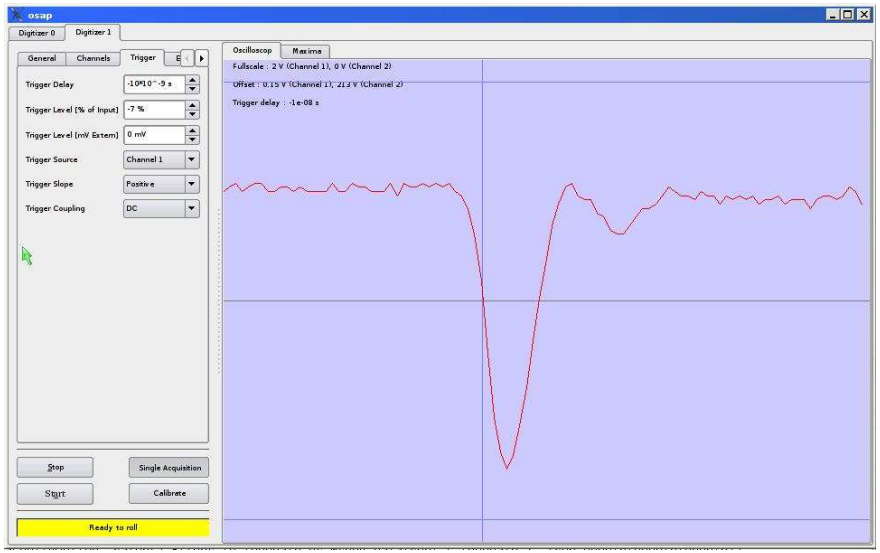
Start-signal from machine-pulse into trigger input of the digitizers (AND-coupled with coincidence).

No detection of the start-impuls → better time resolution compared to conventional systems.

Hardware: Digitizer

- 8 digitizers: Acqiris DC211 - 4GS/s, max. 1GHz input frequency, 8bit vertical resolution
- always two coupled together in one crate to form one 2-channel digitizer

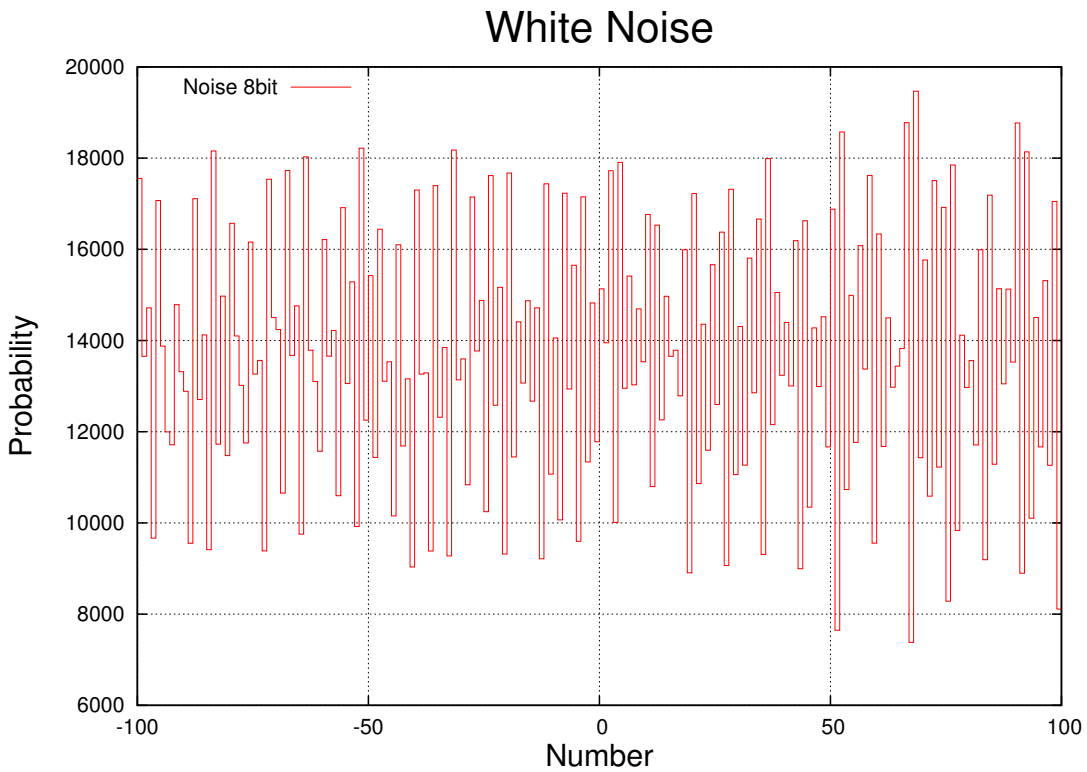
One of four digitizers



Anode-pulse of SEV + PM (of $Na^{22} - \gamma$)

Hardware: Digitizer: Problem of Noise

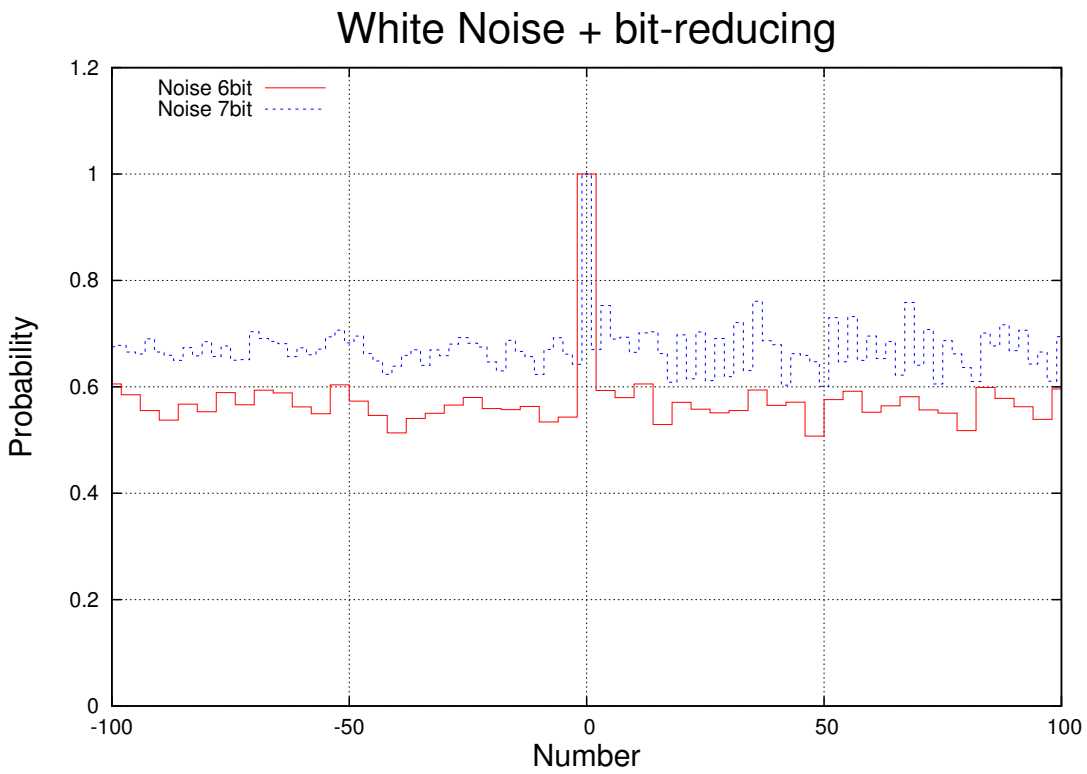
Testing the linearity of the digitizers with White Noise:



Hardware: Digitizer: Problem of Noise

The Problem of Noise

The last 2 bits are noisy but the last bit isn't randomly distributed.



Hardware: Computers

Hardware all the pc's have:

- Dual 2.8GHz Intel© Xeon™
- 4GB RAM
- Gigabit Ethernet

The clients will boot their OS over network.

- Easier system maintainance (Example: Software update)
- Less noise, heat and trouble with hardware

Special hardware of the server:

- Graphics
- 200GB HD (mirrored by RAID1 for data-integrity)

Special hardware of the clients:

- Acquiris Crade with 2 digitizers
- no HD
- no graphics at all

Hardware: Computers

Hardware all the pc's have:

- Dual 2.8GHz Intel© Xeon™
- 4GB RAM
- Gigabit Ethernet

The clients will boot their OS over network.

- Easier system maintainance (Example: Software update)
- Less noise, heat and trouble with hardware

Special hardware of the server:

- Graphics
- 200GB HD (mirrored by RAID1 for data-integrity)

Special hardware of the clients:

- Acquiris Crade with 2 digitizers
- no HD
- no graphics at all

Hardware: Computers: Operating System

Linux is choosen for the OS - Distribution: “Gentoo”

- The programmer knows Linux far better than Windows
- Windows had problems with hyperthreading and $>$ 2GB RAM and is less optimised
- Better suitable for booting the clients over net
- No expenses for licenses

Acquisation-Software

A homebrewn Laboratory-Software-Suite is needed.
How hard can it be?

Overall needs

- network-transparent
- highly customizable and extendible
- (in parts) accessible via internet

Acquisation-Software

A homebrewn Laboratory-Software-Suite is needed.
How hard can it be?

Overall needs

- network-transparent
- highly customizable and extendible
- (in parts) accessible via internet

Acquisation-Software

Several steps to make life easier:

1. divide the apps into server- and client-part
2. invent a simple, yet fast and sophisticated network-communications-protocol
3. use existing tools and apps where possible (mysql for the database and apache+PHP for the webinterface)
4. think about standardized interfaces to replace parts easily

Acquisation-Software

Several steps to make life easier:

1. divide the apps into server- and client-part
2. invent a simple, yet fast and sophisticated network-communications-protocol
3. use existing tools and apps where possible (mysql for the database and apache+PHP for the webinterface)
4. think about standardized interfaces to replace parts easily

Acquisation-Software

Several steps to make life easier:

1. divide the apps into server- and client-part
2. invent a simple, yet fast and sophisticated network-communications-protocol
3. use existing tools and apps where possible (mysql for the database and apache+PHP for the webinterface)
4. think about standardized interfaces to replace parts easily

Acquisation-Software

Several steps to make life easier:

1. divide the apps into server- and client-part
2. invent a simple, yet fast and sophisticated network-communications-protocol
3. use existing tools and apps where possible (mysql for the database and apache+PHP for the webinterface)
4. think about standardized interfaces to replace parts easily

Acquisation-Software: Clients

Tasks:

- Fetching the data from the digitizers
- Evaluating the data (through exchangable plugins)
- Sending the results over the network to the server

Just simple needs?

- some advanced concepts for plugins
  (more on the analyzer-plugins later on)
- need for total network-transparency
  to remote-control the systems

Acquisation-Software: Clients

Tasks:

- Fetching the data from the digitizers
- Evaluating the data (through exchangable plugins)
- Sending the results over the network to the server

Just simple needs?

- some advanced concepts for plugins
  (more on the analyzer-plugins later on)
- need for total network-transparency
  to remote-control the systems

Acquisation-Software: Server

Tasks:

- controlling the clients
  - state (on/off, measuring/pausing, etc.)
  - analyzers / filters
  - parameters
- controlling the laboratory
  - vacuum
  - temperature
  - magnetic fields
  - beam
- controlling the measurement
  - sequence / loops / conditions

Acquisation-Software

Fulfilling these tasks gets easier by using complicated plugin-concepts.

Plugins will be used for: Analyzers, Filters, Modifiers, data acquisation, data-display and -export, etc...

Acquisation-Software: Analyzers

What is the aim of the analyzer-plugin-concept?

- An interface to plug different methods of analysation into the chain of data-acquisation.
- and to replug them without recompiling the apps
- ...and even without restarting the apps
- analyzers can also make use of filters

Acquisation-Software: Analyzers

Currently planned analyzer-plugins:

- Lifetime - Implementing constant fraction or integral constant fraction to measure positron-lifetime
  (as described by [Poggi, 2003] or [Becvar, 2004])
- Doppler - To measure the doppler-broadening
- other debugging analyzers like:
  - noise
  - oscilloscope
  - first sample
  - etc...

Conclusion

Contact us if you are interested in joining the development.
arnold.krille@gmail.com or a.krille@fz-rossendorf.de

Get the slides at http://positron.physik.uni-halle.de/

Literature

F. Becvar, J. Cizek, I.Prochazka, J. Janotova
The asset of ultra-fast digitizers for positron-lifetime spectroscopy
NIM A 539 (2005) 372-385

L. Bardelli, **G. Poggi**, M. Bini, G. Pasquali, N. Taccetti
Time measurements by means of digital sampling techniques: a study case of 100 ps FWHM time resolution with a 100 MSample/s, 12 bit digitizer
NIM A 521 (2004) 480-492