

Progress Report on **cluster-jet slow-control** activities  
of the NCBJ team

**Jerzy Tarasiuk**

NCBJ - Warsaw and Uni-Warsaw

and

**Boguslaw Zwieglinski**

NCBJ - Warsaw

# Report content

- I. A study of application of CompactRIO-9039 and LabVIEW 2018 together with other software packages purchased from National Instruments (NI) to communicate with the supervisory EPICS control system of the PANDA experiment.
- II. A purchase and tests of NI voltage-to-digital (NI-9205) and current to digital (NI-9208) converters. The first one contains 32 input channels, the second 16 channels.
- III. LabVIEW programming for the cluster-jet beam dump.

# I. A study of applications of CompactRIO-9039 and LabVIEW 2018

together with other software packages purchased from National Instruments (NI) to communicate with the supervisory EPICS control system of the PANDA experiment.

*The PANDA DCS Core team announced at CM 2/19 the plans to:*

- \* replace CS-Studio (CSS) with PHOEBUS,
- \* replace Channel Access protocol with PVaccess protocol.

*The question thus arises: How could we make our LabVIEW programs compatible with those changes? because:*

- \* PVaccess is not supported by LabVIEW
- use a protocol converter(CA<-> PVaccess)?
- \* pvaSrv was advocated [from slac. stanford.edu] for use as such a converter, but we failed to get it compiled (upon trying to do so, we have compiled a few EPICS base versions in the course of our study)

The EPICS server provided with LabVIEW has the following fields:

- \* VAL STAT SEVR for every data type,
- \* HIGH HIHI HSV HHSV LOW LOLO LSV LLSV for the double type,
- \* OSV ZSV (1/0 severity) for the boolean type.

And it has the following limitations:

- \* The LabVIEW program can access the VAL field only
  - other fields can only be accessed using the EPICS protocol (this includes access from the LabVIEW's EPICS client - so there is some access to them).
- \* The HSV HHSV LSV LLSV OSV ZSV fields are to define the alarm severities, but LabVIEW's EPICS ignores their settings - it does not signal any alarm, while other EPICS servers set STAT and SEVR when alarm conditions are met.
- \* The LabVIEW variables assigned to EPICS servers VAL fields must be put into a library, which, in order to be activated, should be deployed from LabVIEW (using its Windows IDE) - the CompactRIO cannot do it without IDE.

*Solution to these problems: use an 'external' (non-NI) EPICS 'softIoc' (from Argonne EPICS distribution), it may be compiled on a PC and run on the CompactRIO, then:*

- \* EPICS variables are defined in the server configuration, no need to use the LabVIEW IDE (and a Windows PC) to add a new one when necessary;
- \* the server handles these alarm settings;
- \* if PVaccess protocol is needed, as is suggested in the future, EPICS 7 can be used - in a way that LabVIEW program can access its variables using Channel Access protocol, while the PANDA DCS supervisory system can use PVaccess;
- \* tests have been performed with (CompactRIO 9039 and "NI Linux Real-Time 6.0"):  
EPICS 3.15.6, 3.16.2, 7.0.2.2 all compiled on Ubuntu 18.04 from Argonne sources; all libraries used by them have the same versions on both systems.

II. A purchase and tests of the NI **voltage-to-digital (NI-9205)** and **current-to-digital (NI-9208)** converters.

A justification for the NI-9205 purchase: This converter will read voltage signals directly from the vacuum sensors, bypassing the Center-Three units, thus serving a backup solution for them.

A justification for the NI-9208 purchase: This converter will read current signals from the available current sources.

Some essential numbers concerning the two C-modules from the producer:

**NI-9205**

32-Channel, +/-10V, 250kS/s, 16-Bit Analog Input Module

**NI-9208**

16-Channel, +/- 20mA, 24-Bit Current Input Module

A simple testing instrument has been designed and constructed to test both **NI-9205** and **NI-9208**. It consists of the following elements (see page 8):

- A **RaspberryPi** serving as a programming unit. It delivers a train of integers spaced by 10 between 14 and 4064 in 250 ms steps for a NI-9205 test, or a train of integers spaced by 10 between 14 and 2764 in 250 ms steps for a NI-9208 test. These trains are delivered consecutively in the ascending and the descending mode permitting to obtain a seesaw-like input signals.

- A **Digital-to-Analog Converter (DAC)** MCP4725 and an **amplifier LM324N** converting the train of integers into a sequence of **voltage steps** in the -10V +10V conversion interval of NI-9205,

- A current amplifier built with the remaining part of **LM324N** to generate a train of **current steps** in the conversion range -20 mA to +20 mA of NI-9208,

- **Power supplies of +3.3V, +14.0V, and -12.0V** to supply the above elements from one of the **+24V power supplies NI PS-16** of the CompactRIO-9039,

- A **PC** to register and display the accumulated spectra.

An output from the voltage (current) amplifier is manually switched to the next converter channel after each channel measurement termination.

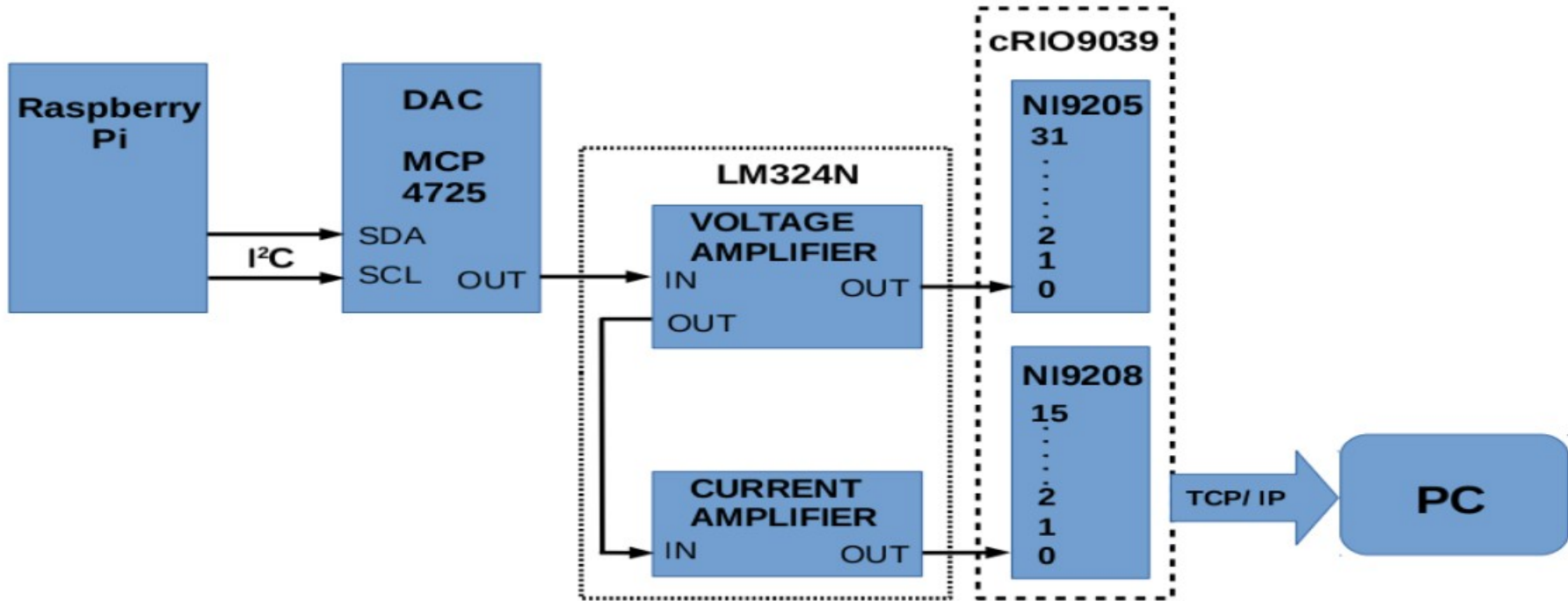


Fig. II.1 A block-scheme of the employed measuring apparatus.



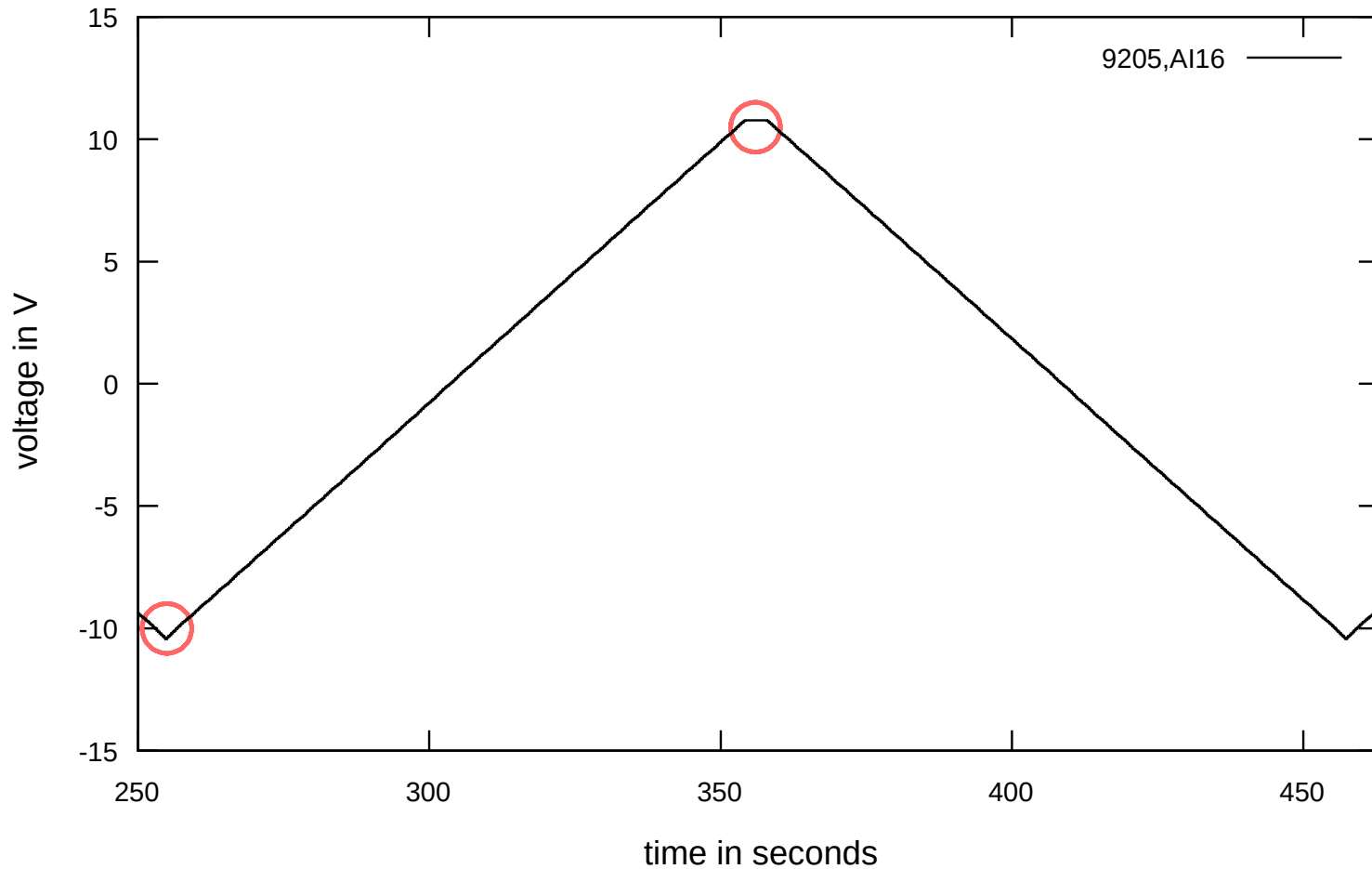


Fig.II.2 A full cycle of voltage variation for the channel #16, showing the branches with a positive and a negative slope. The regions of nonlinearities are indicated with a red circle.

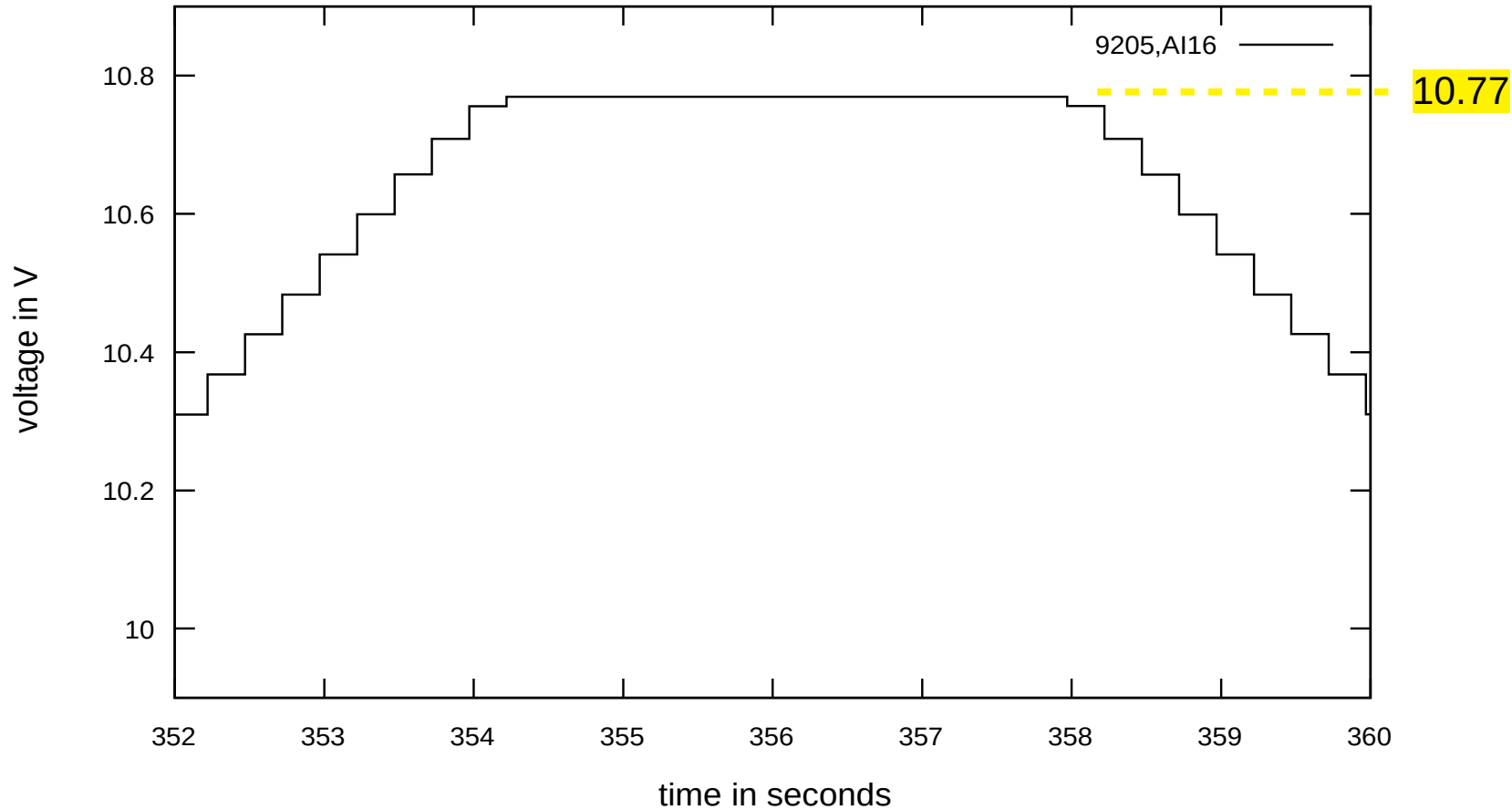


Fig. II.3 An upper nonlinearity region for the channel #16 expanded. The saturation of converter readings is caused by exceeding the permissible range of the input for NI-9205.

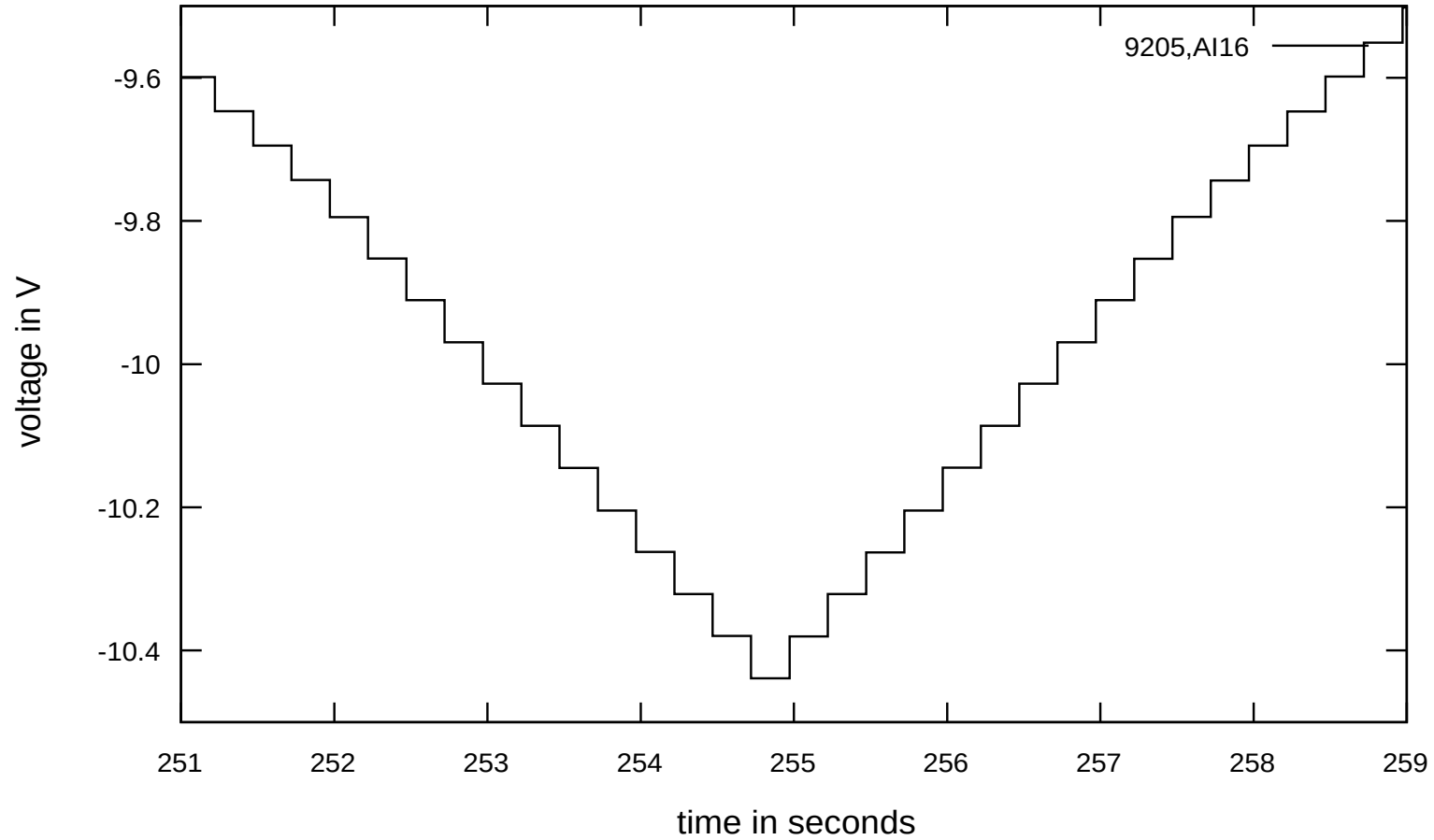


Fig. II.4 A lower nonlinearity region for the channel #16 expanded.

**The results of measurements** for each converter input channel have been registered to disk for further display and analysis. The analysis consisted in a least-squares fitting of a linear function of time to each of the increasing and decreasing branches of the full cycle. The above mentioned nonlinearity regions have been excluded from the fit.

In spite of the fact that individual measurements have a rather small dispersion around the mean (ca. +/- 0.025% see Fig.II.6), a more detailed inspection of the results demonstrates imperfections of the applied scheme of measurements.

For example, a difference between the converter readings and the fitted linear trend displayed in Fig. II.5, demonstrates oscillations on the millivolt scale, which are ascribed to nonlinearities in DAC conversion.

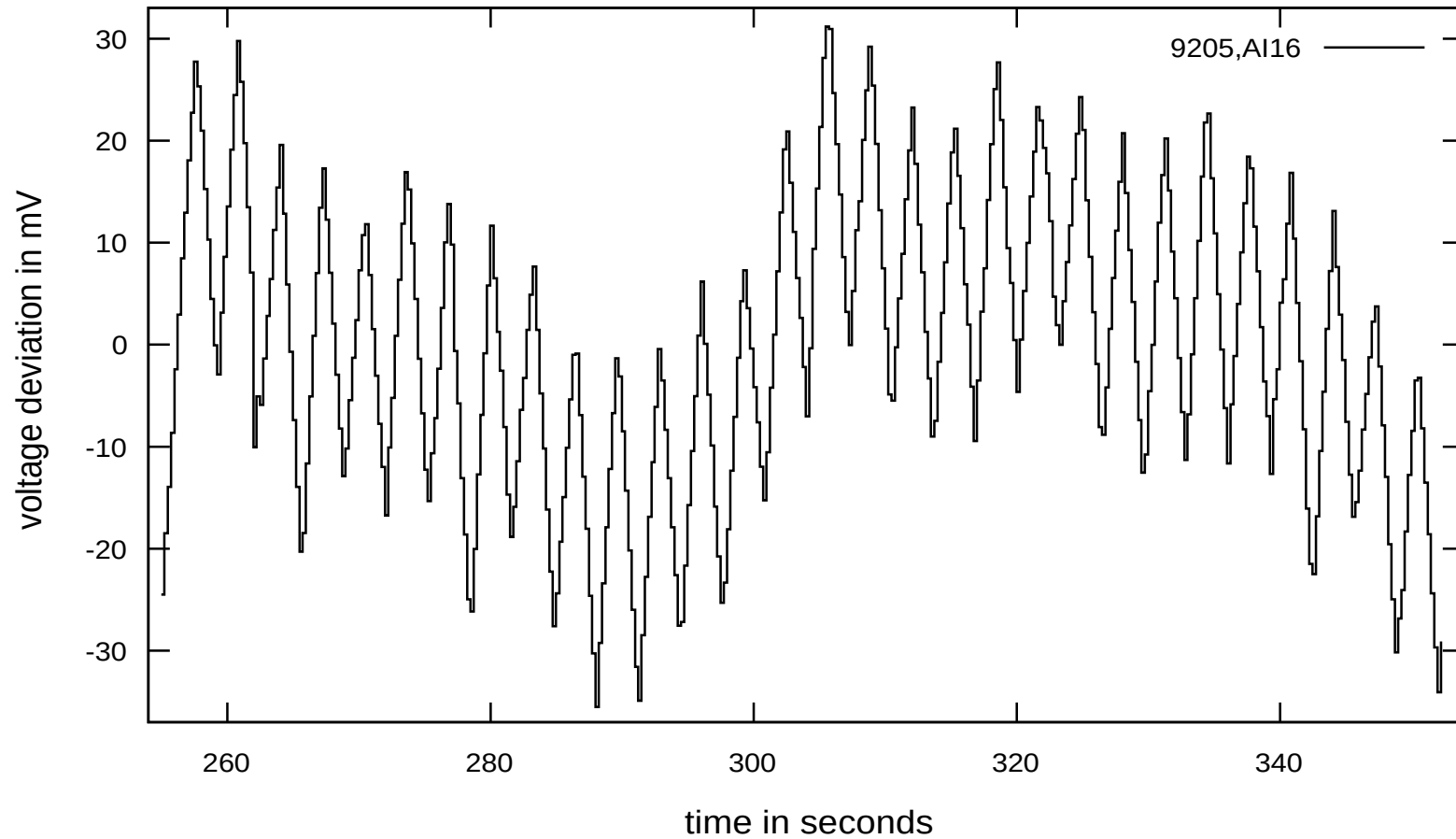


Fig. II.5 Voltage readings minus linear fit (deviation from the fit) for the channel #16.

A collection of all extracted slopes [V/s] is presented in Fig. II.6. Negative slopes (in blue) are displayed with an inverted sign. The measurements contain as a rule two cycles (two red and two blue points), which are grouped under an appropriately numbered red segment (above the displayed points). However, there are some segments with an apparently longer length (as #15) containing the results of a few cycles. The results for channel #0, collected at significantly different times are displayed both at the beginning and at the end of the horizontal axis.

The results of measurements with the 16 channel current-to-digital converter NI-9208 will be mentioned only briefly.

The measured slopes (in mA/s) are presented in Fig. II.7 using the same convention as in Fig. II.6. Most of the results have a well defined mean value of the slope  $438.08 \mu\text{A/s}$  with a small dispersion around the mean value. However, for quite a few of them the deviations are significant, which indicates, as we think, random instabilities of some of our power supplies.

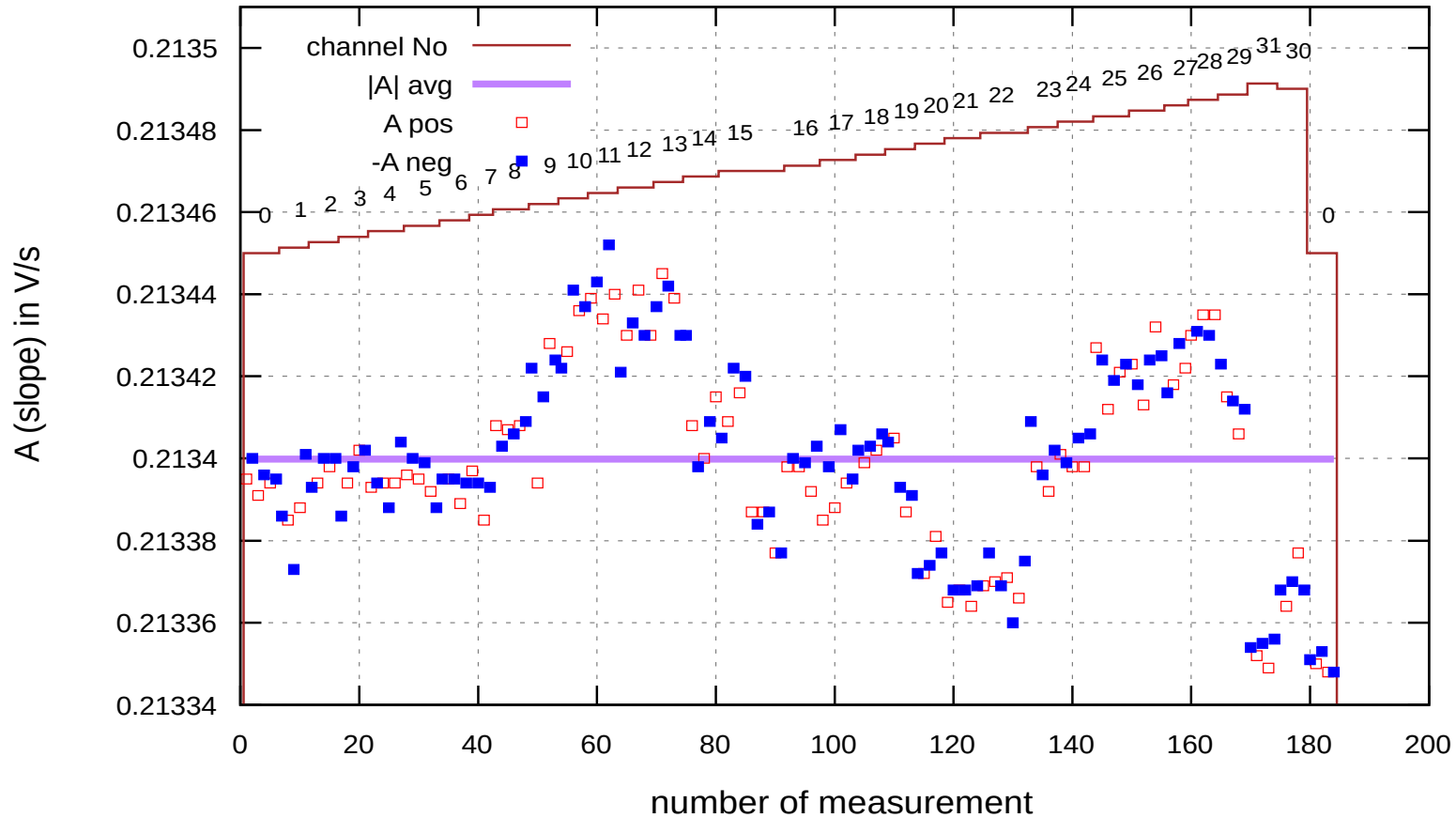


Fig. II.6 A collection of slopes fitted to the measurements for each channel of NI-9205. The mean value 0.2134 V/s is displayed in purple.

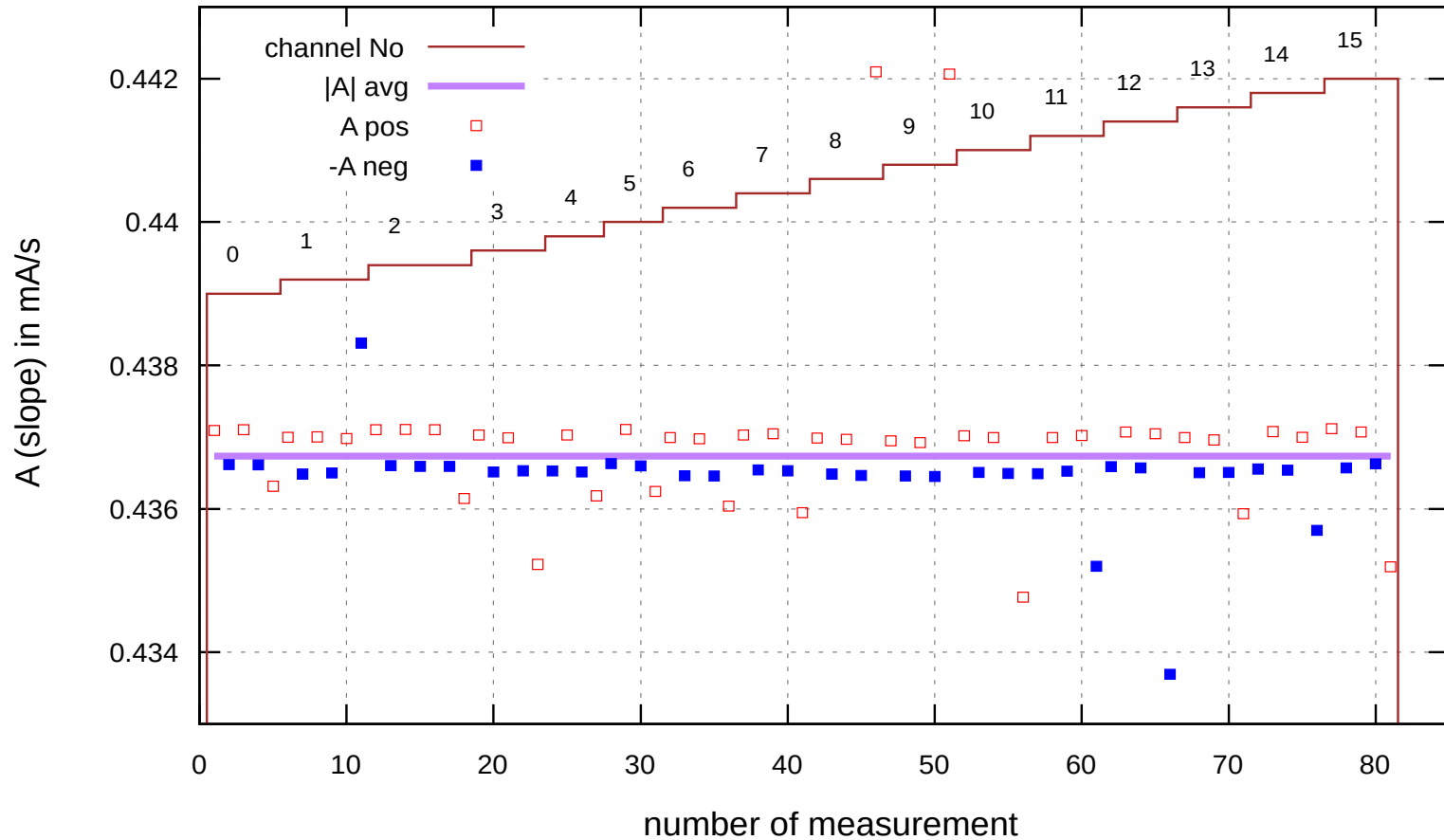


Fig. II.7 A collection of slopes fitted to the measurements for each channel of NI-9208.



## Conclusions

A simple system has been designed and mounted, permitting to deliver a staircase voltage (current) to an input of each of The 32(16) channels of NI-9205 (NI-9208). The tests performed indicate (with a moderate accuracy) that both these NI C-modules behave in accordance with the producer's characteristics. The designed system is planned to be incorporated into the cluster-jet generator (CJG) and the cluster-jet beam-dump (CJBD) emulators *e.g.* to deliver the signals of vacuum sensors.

### **III. Programming in LabVIEW of the cluster-jet beam-dump (BD) operation.**

Two scenarios for the BD vacuum infrastructure exist at present. In our programming we follow the one described in the CJ-target TDR, i.e. containing both forevacuum and Roots pumps. At a later stage the pumping of hydrogen through a large diameter tube by a forevacuum system of the cluster-jet generator (CJG) will also be considered.

**THANK YOU FOR YOUR ATTENTION !!!**