Data Acquisition System of the Spin Physics Detector

A. Kulikov
DLNP, JINR

Contents:

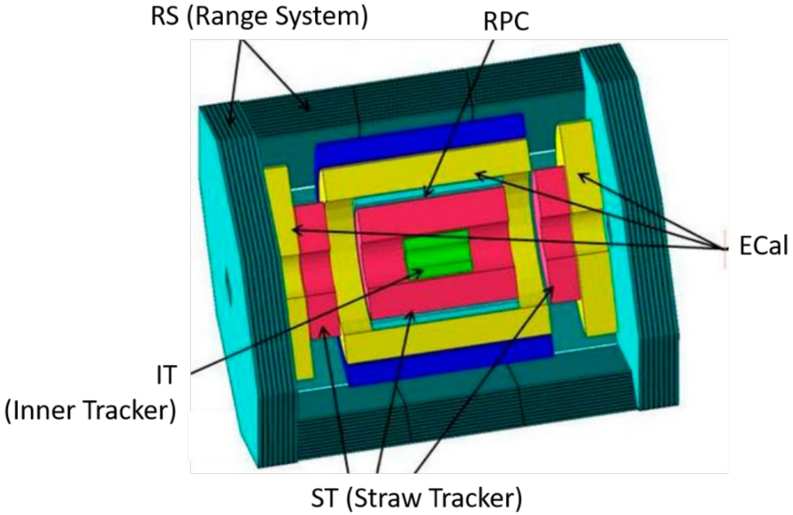
- □ Data flux
- □ Trigger-less DAQ
- □ SPD-DAQ architecture
- □ Front-end electronics
- □ Present status
- □ Plans

Estimation of raw data flow *(preliminary , very rough)*

At L = 10^{32} $cm^{-2}c^{-1}$ and √s = 27 GeV the calculated event rate in the SPD aperture is $3{\cdot}10^6$ 1/s.

Simplifications used in estimation of the data flux:

- charge particles give hits in all detectors, γ-quanta – only in ECAL
- no noise signals, no background
- zero suppression implemented
- number of channels in two end-cups is taken as 1/2 of the barrel channels
- all particles hit different cells in detectors (no double hits in a cell)
- no headers, no calibration data etc.
- when appropriate, some numbers are taken from simulations and beam tests of MPD and PANDA



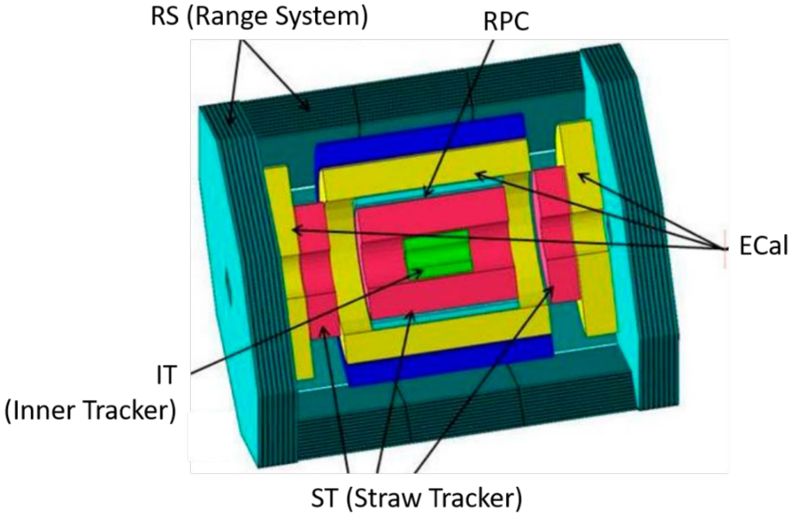
With these approximations and with some safety margin the data flux is estimated as 20 GBytes/s.

Estimation of raw data flow *(preliminary , very rough)*

At L = 10^{32} cm^{-2}c^{-1} and √s = 27 GeV the calculated event rate in the SPD aperture is $3\cdot10^{6}$ 1/s.

Simplifications used in estimation of the data flux:

- charge particles give hits in all detectors, γ-quanta – only in ECAL
- no noise signals, no background
- zero suppression implemented
- number of channels in two end-cups is taken as 1/2 of the barrel channels
- all particles hit different cells in detectors (no double hits in a cell)
- no headers, no calibration data etc.
- when appropriate, some numbers are taken from simulations and beam tests of MPD and PANDA



With these approximations and with some safety margin the data flux is estimated as 20 GBytes/s.

Main requirements to DAQ:

- ***ability to withstand high data flux***
- ***no dead time (or minimal dead time)***

Triggered or trigger-less DAQ?

Traditional hardware-based trigger system is going away in many new experiments under preparation.

Instead of triggered DAQ, free-running DAQ is used which reads the detector data with predefined frequency. This became possible due to great progress in the IT technologies and in development of high performance programmable chips FPGA.

This trigger-less approach is accepted, for example, in experiments under preparation at GSI – in PANDA, CBM, NUSTAR, in PSI (Mu3e), in future HL-LHC experiments, also is foreseen for ILC, CLIC, etc.

Is the definition “trigger-less DAQ” correct?

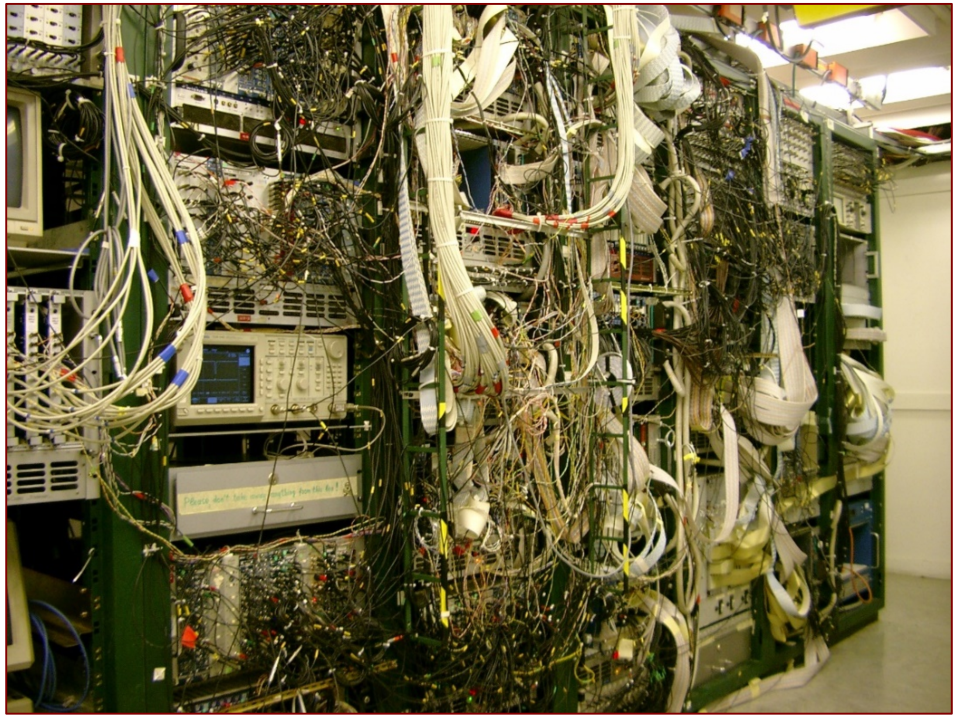
The purpose of any trigger system is on-line selection of events with specific characteristics typical for the physics process under study. The DAQ accepts only events selected by the trigger, thus greatly decreasing the data volume to be stored.

Traditionally the logic of the trigger selection was implemented using a number of dedicated or commercial electronic modules arranged in one or several trigger layers.

Is the definition “trigger-less DAQ” correct?

The purpose of any trigger system is on-line selection of events with specific characteristics typical for the physics process under study. The DAQ accepts only events selected by the trigger, thus greatly decreasing the data volume to be stored.

Traditionally the logic of the trigger selection was implemented using a number of dedicated or commercial electronic modules arranged in one or several trigger layers.



20-years-old trigger and DAQ electronics (DIRAC)

Is the definition “trigger-less DAQ” correct?

In so-called “trigger-less” data acquisition systems the data selection takes place as well, otherwise it would be impossible to record a huge flux of raw data.

But selection here is organized in software way using a dedicated programming of FPGA chips and on-line computers.
In fact, it can be named as a “software trigger”.

Therefore, it would be more correct to use the term “free-running DAQ” instead of “trigger-less”.

What should be the readout frequency of a free-running DAQ?

In trigger-less systems the readout is performed with a fixed frequency.

$f_{readout} = ?$

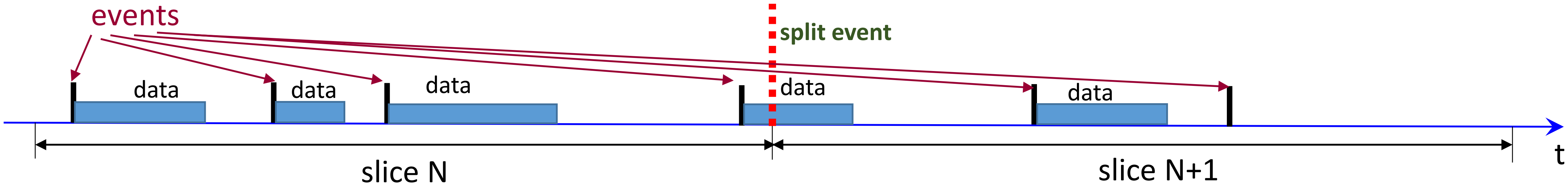
Readout frequency defines the width of the time slice between the consecutive readouts. All the data within a slice will be read out, stored in a memory and attributed to this slice number.

The choice of the time slice width depends on two factors: data rate and memory depth available in the front-end modules of the detectors.

What should be the readout frequency of a free-running DAQ?

Another factor to be taken into account while choosing the time slice width is the response time of the detectors.

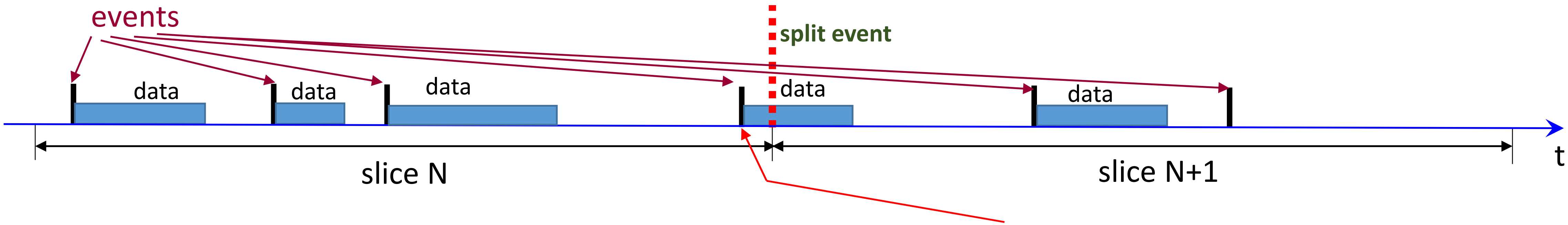
When the event arises close to the boundary between the slices, the signals may fall in different time slices. The bigger time spread of response of the detectors, the more probability of such split events. Fortunately, in SPD we have not very slow detectors, the slowest ones are gas detectors: straw tracker and range system, with a maximum drift time of ~120 ns.



What should be the readout frequency of a free-running DAQ?

Another factor to be taken into account while choosing the time slice width is the response time of the detectors.

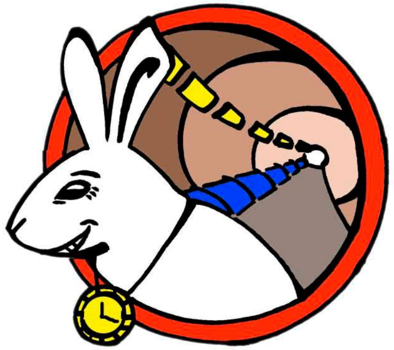
When the event arises close to the boundary between the slices, the signals may fall in different time slices. The bigger time spread of response of the detectors, the more probability of such split events. Fortunately, in SPD we have not very slow detectors, the slowest ones are gas detectors: straw tracker and range system, with a maximum drift time of ~120 ns.



The wider time slice, the lower probability of such split events. Therefore, it is reasonable to have the width of the slice much wider than the maximum time of charge collection in the detectors.

Time stamping

In order to build the events from signals accumulated in a time slice,
the data from each detector should be provided with a timestamp.
For this purpose either a dedicated clock generator or the signals from
a global time distribution system can be employed, usually with a frequency
in the 100—200 MHz range.



White Rabbit

Ethernet-based solution for sub-ns synchronization and deterministic, reliable data delivery

In NICA the White Rabbit system (developed at CERN) is used which provides synchronization for large distributed systems with timestamping of 125 MHz, sub-nanosecond accuracy and ~10 ps precision.

White Rabbit allows one to time-tag measured data and hence to group the hits to events on the timing base.

SPD-DAQ architecture

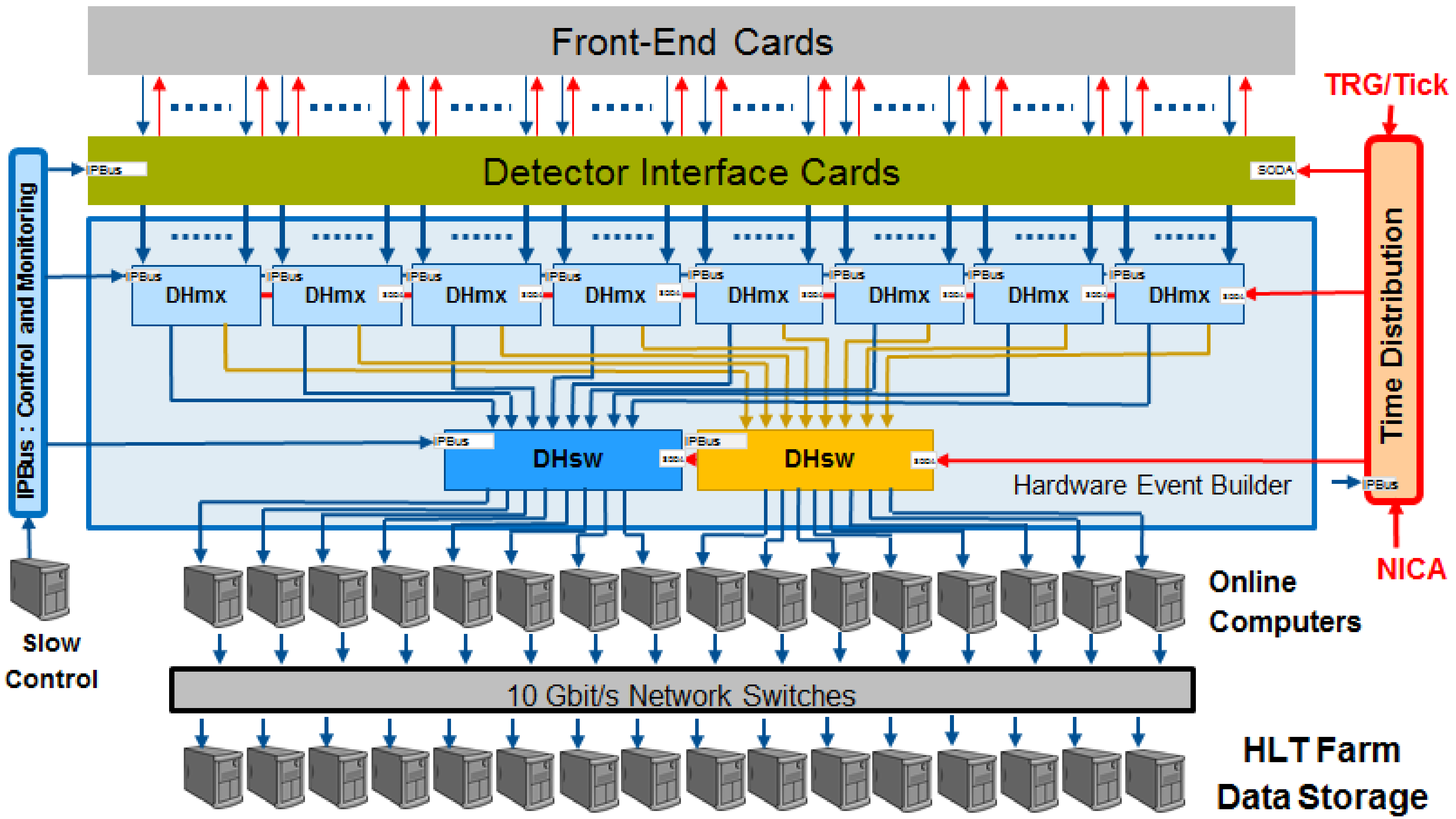
In DAQ of SPD we are planning to employ the ideas developed for the modernized DAQ of COMPASS, with FPGA chips in the core of the system.

We are working in collaboration with Igor Konorov from the Technische Universität of München (TUM) who is the coordinator of the DAQ-COMPASS and who has developed a number of the DAQ electronic modules.

In October 2018 he has sent us for discussion his proposal on DAQ for SPD:
“Data Acquisition System for the Spin Physics Detector”

which can be taken as a base for building of the SPD-DAQ.

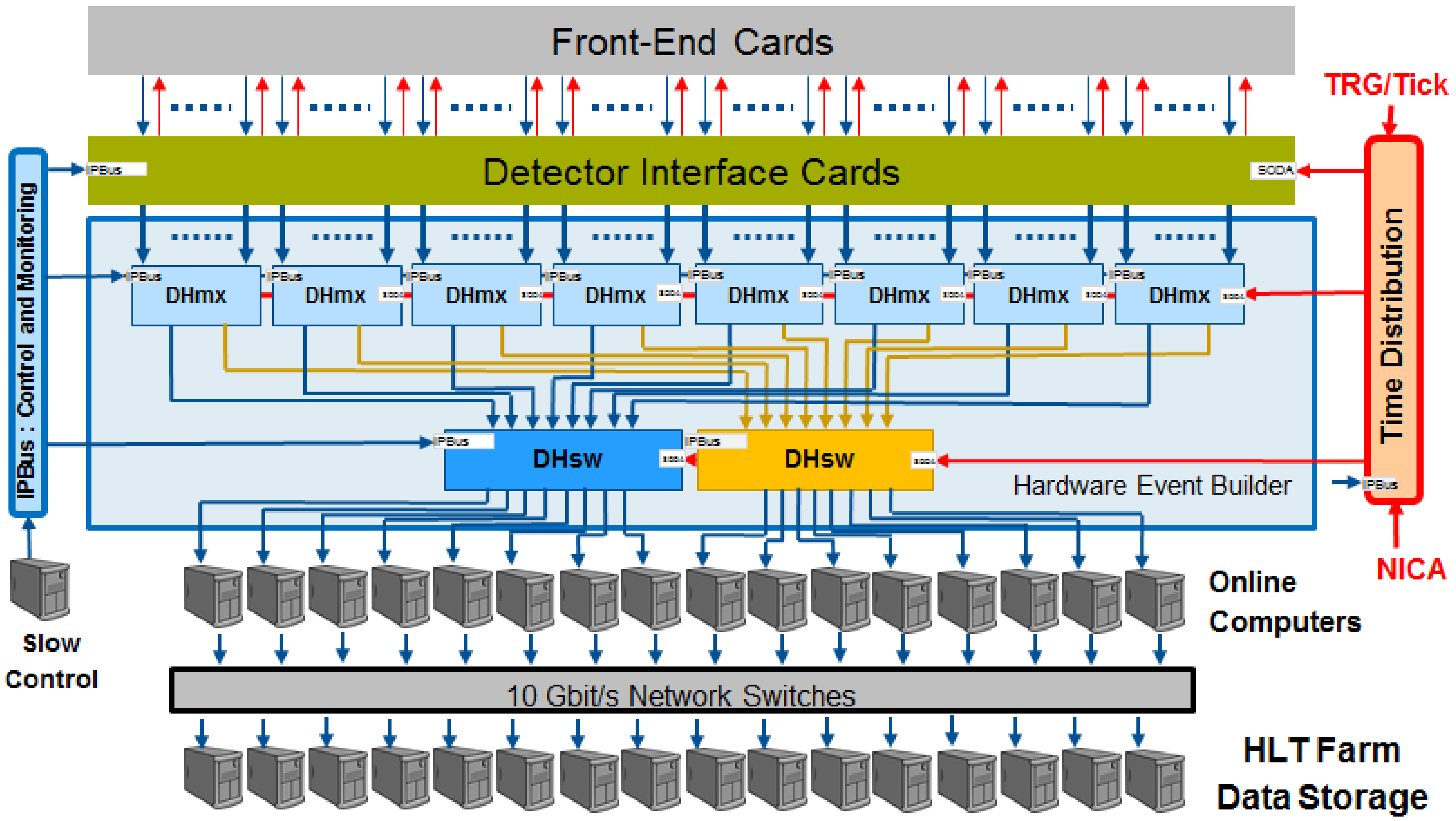
SPD-DAQ architecture *by Konorov*



Slow control accesses FEE via the detector interface cards (DIC) using UDP-based IPBus protocol.

DIC retransmit clock signals to FEE and convert detector information to a high speed serial interface running over an optical link.
UCF (*Unified Communication Framework*) protocol will be a standard high speed link protocol within the DAQ.

SPD-DAQ architecture *by Konorov*

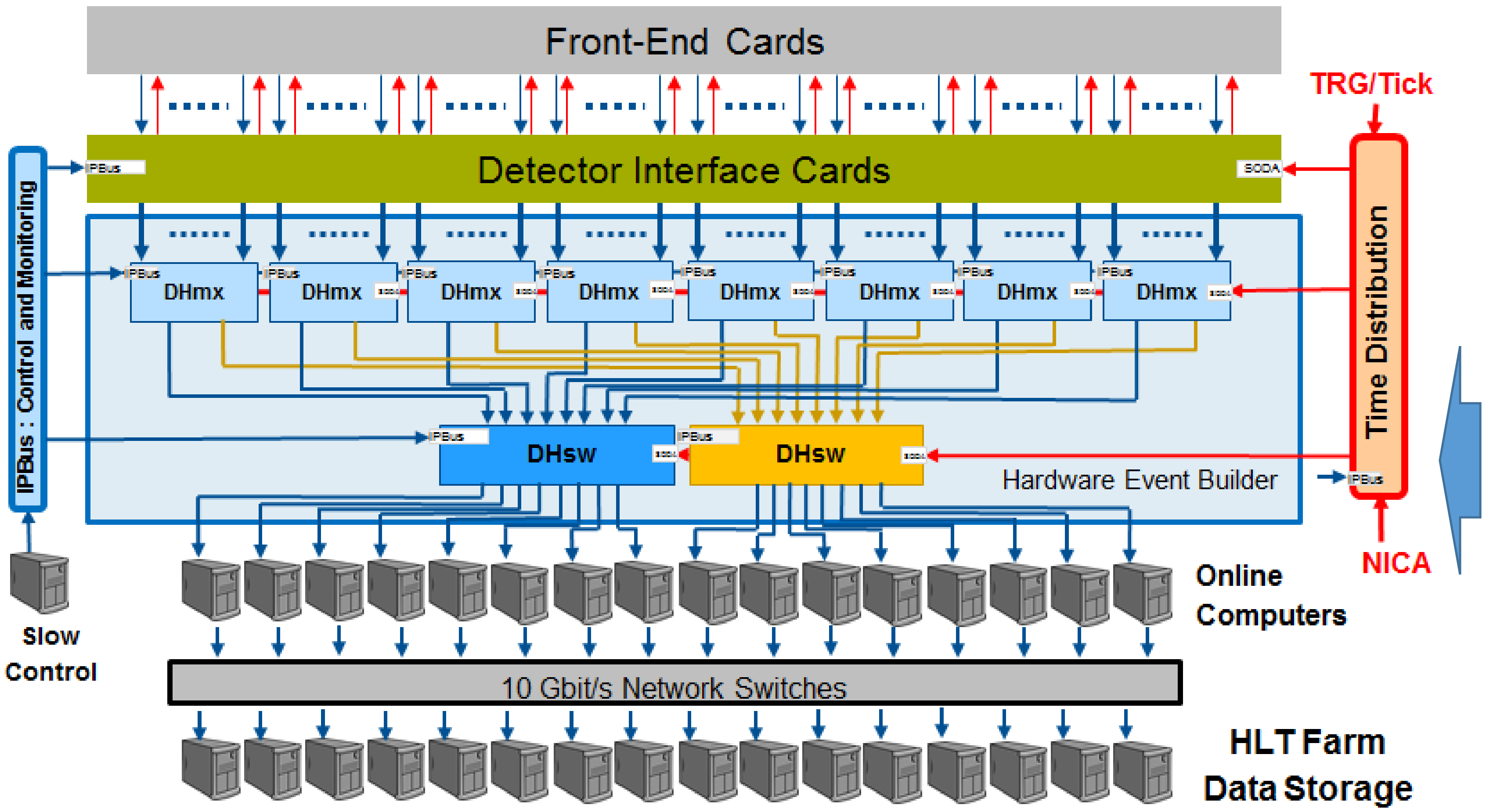


The multiplexer (DHmx) modules receive detector information via serial links, verify consistency of data, and store them in DDR memories.

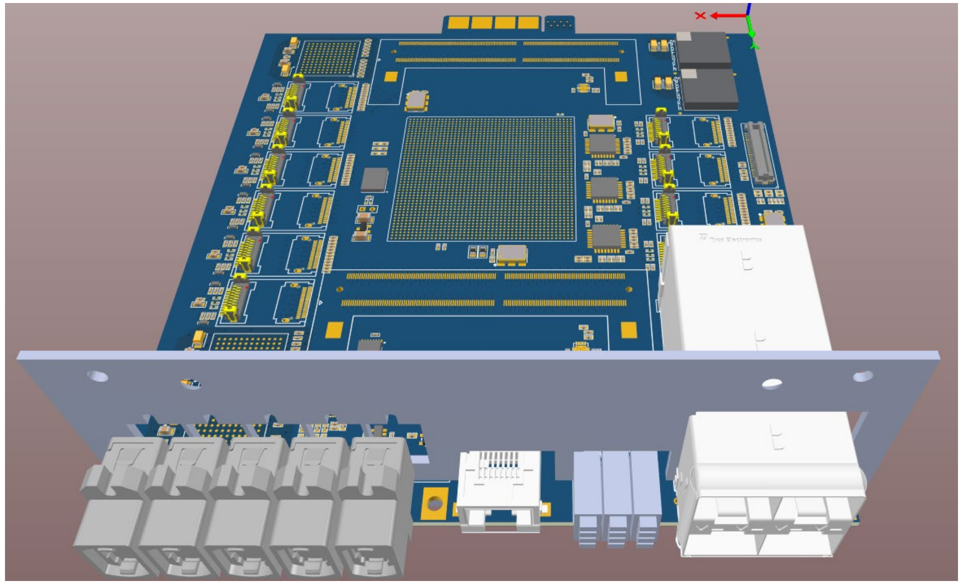
The multiplexer is equipped with 32 GBytes of memory.

All accepted data are assembled in sub-events and distributed to two switches.
Each multiplexer has a bandwidth of 2 GBytes/s.

SPD-DAQ architecture *by Konorov*



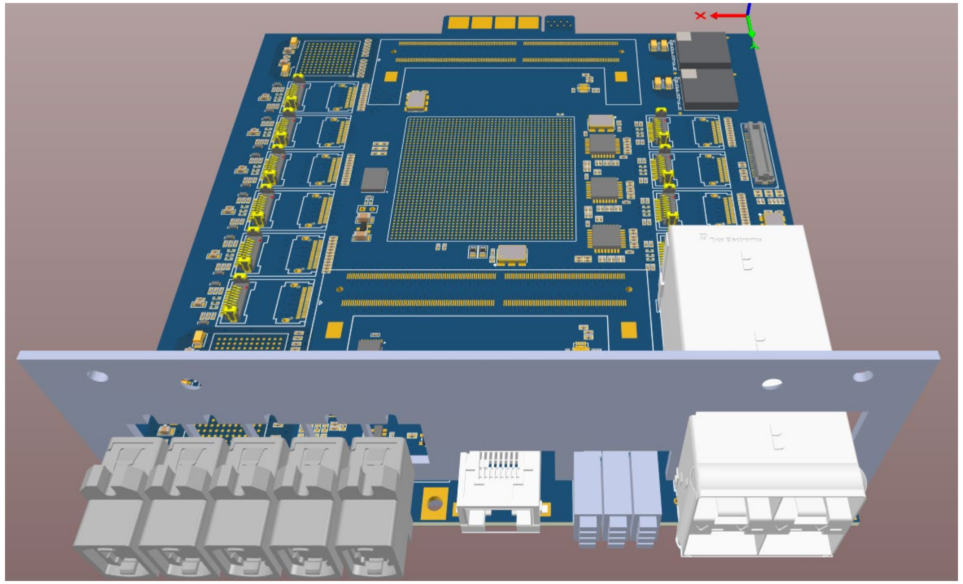
The switches (DHsw) perform the final level of event building and distribute the assembled events to 20 (?) on-line computers.



One and the same hardware FPGA-based module developed in TUM can be programmed either as a multiplexer or as a switch.

The multiplexer *DHmx* has 48 high speed input and up to 8 output interfaces.
A bandwidth of incoming interface can be programmed from 2 to 10 Gbps.
Outgoing interface runs at a fixed speed of 10 Gbps.

DHsw firmware functions as 10x10 switch and performs event building with a maximum throughput rate of 10 GBytes/second.

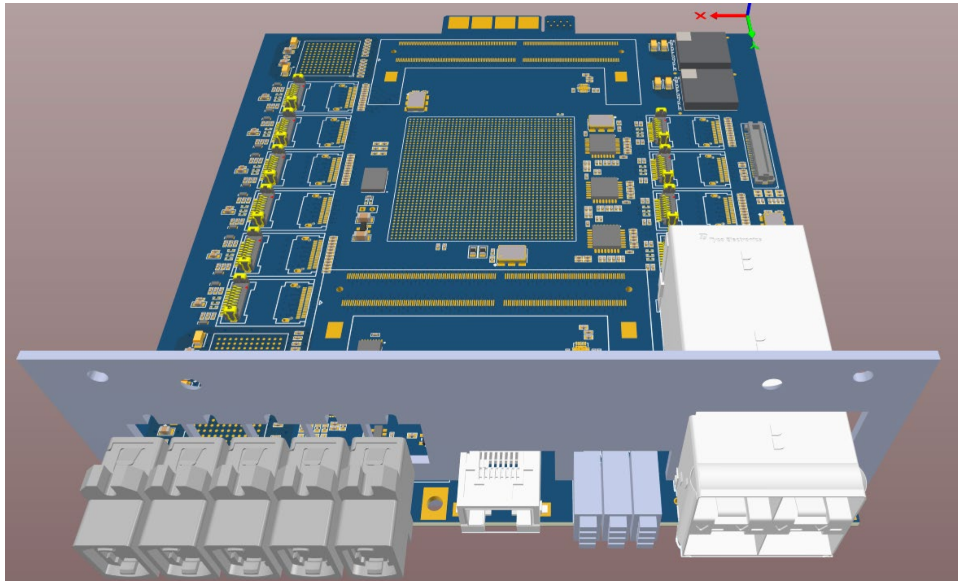


One and the same hardware FPGA-based module developed in TUM can be programmed either as a multiplexer or as a switch.

The multiplexer *DHmx* has 48 high speed input and up to 8 output interfaces.
A bandwidth of incoming interface can be programmed from 2 to 10 Gbps.
Outgoing interface runs at a fixed speed of 10 Gbps.

DHsw firmware functions as 10x10 switch and performs event building with a maximum throughput rate of 10 GBytes/second.

The whole system can be upgraded to much higher flux than 20 GB/s by implementing additional hardware.



One and the same hardware FPGA-based module developed in TUM can be programmed either as a multiplexer or as a switch.

The multiplexer *DHmx* has 48 high speed input and up to 8 output interfaces.
A bandwidth of incoming interface can be programmed from 2 to 10 Gbps.
Outgoing interface runs at a fixed speed of 10 Gbps.

We consider the proposal of Konorov as a good solution for the SPD-DAQ.
At the same time we are not obliged to follow it literally, and do not throw away the option of combining it with more traditional approach (not 100% FPGA-based).

DHsw firmware functions as 10x10 switch and performs event building with a maximum throughput rate of 10 GBytes/second.

Front-end electronics for the free-running DAQ-SPD

General FEE requirements from the DAQ system *(preliminary, to be specified later)*:

- Self-triggered (*trigger-less*) FEE operation
- Digitizing on-board
- Zero suppression
- Large memory to store data accumulated in a time slice
- Timestamp included in the output format

Front-end electronics for the free-running DAQ-SPD

General FEE requirements from the DAQ system *(preliminary, to be specified later)*:

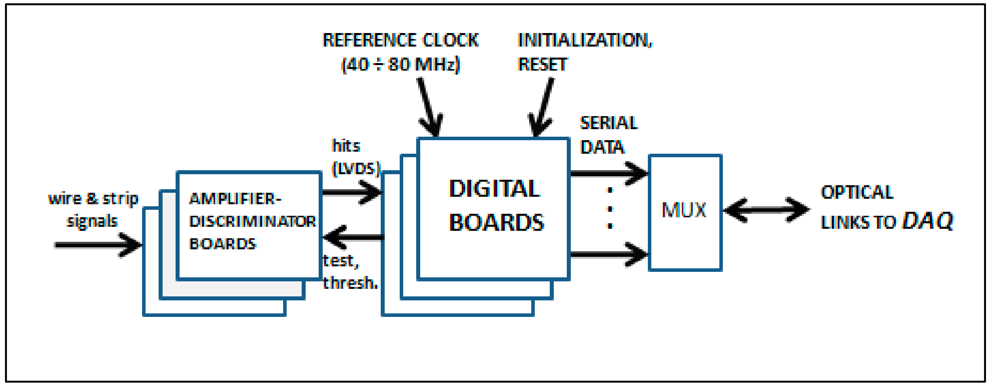
- Self-triggered (*trigger-less*) FEE operation
- Digitizing on-board
- Zero suppression
- Large memory to store data accumulated in a time slice
- Timestamp included in the output format

Let us look how front-end electronics of the SPD detectors meets these requirements: what is available or what is under development somewhere.

Range system

The SPD range system closely follows the design of the range system of PANDA, which is in a well-advanced state. This concerns, in particular, the analog front-end electronics. There were developed amplifier-discriminator boards, preamplifier-invertor for strips allowing to use the same digital electronics both for wires and strips.

The digital part of the PANDA front-end electronics is still in the process of development. Its properties are similar to what we want for the SPD-DAQ:



Block diagram of the data stream in PANDA

no trigger: readout by clock signals
timestamp attached to data
buffer memory
use of FPGA chips
optical links to DAQ

But digital part is in VME standard!!

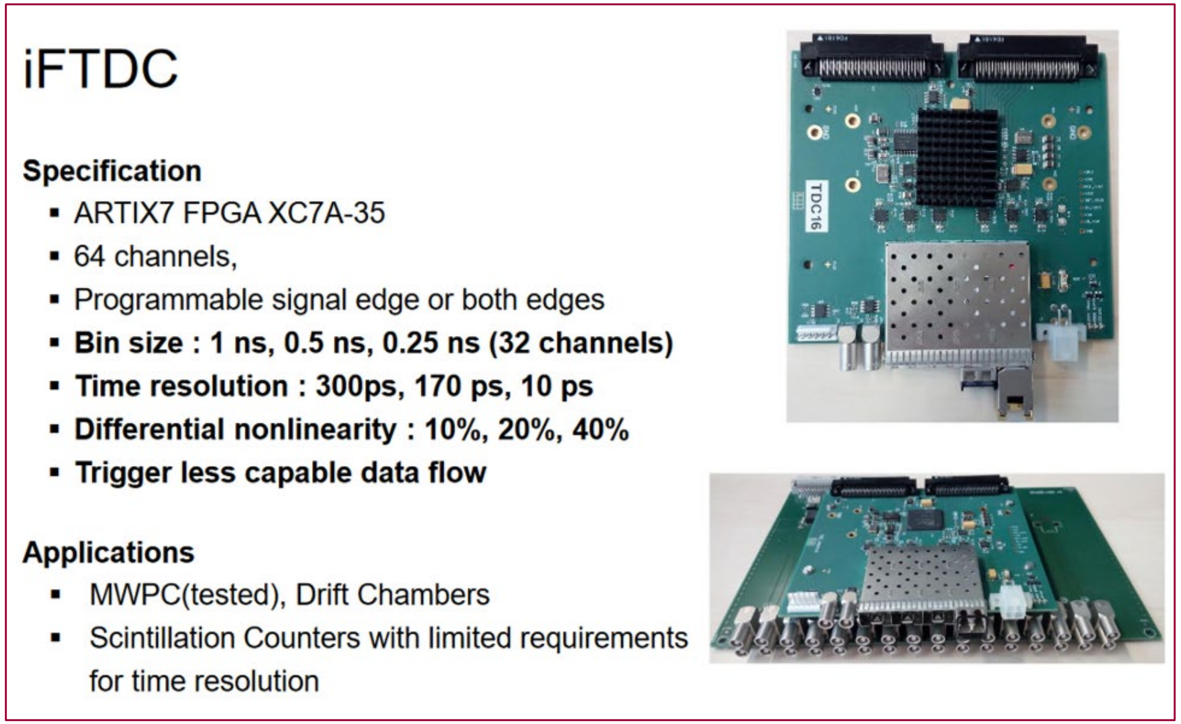
Straw tracker

I.Konorov developed so-called “iFTDC” which is a TDC module built using FPGA chip.

iFTDC can work both in triggered or trigger-less mode (this has been tested and confirmed), has precision down to 150 ps: well above the requirements of the straw tracker.

It is planned to use iFTDC in COMPASS, NA64 and is reasonable to employ it in SPD.

Straw tracker group considers also other options for FEE.



iFTDC

Specification

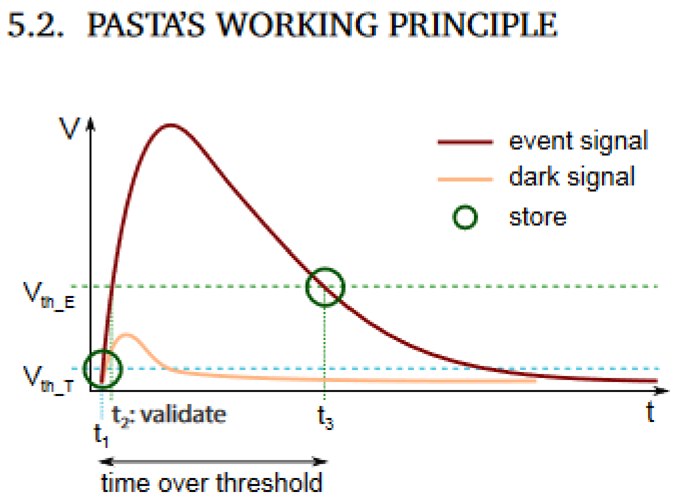
- ARTIX7 FPGA XC7A-35
- 64 channels,
- Programmable signal edge or both edges
- **Bin size : 1 ns, 0.5 ns, 0.25 ns (32 channels)**
- **Time resolution : 300ps, 170 ps, 10 ps**
- **Differential nonlinearity : 10%, 20%, 40%**
- **Trigger less capable data flow**

Applications

- MWPC(tested), Drift Chambers
- Scintillation Counters with limited requirements for time resolution

Vertex detector

For the microvertex detector of PANDA a dedicated chip **PASTA** *(PAnda STrip Asic)* is being developed to be used in a free-running DAQ.



5.2. PASTA'S WORKING PRINCIPLE

PASTA provides measurement of time and charge of the signal from *microstrip* sensors.

PASTA uses 160 MHz clock giving a coarse timestamp with precision of 6.25 ns. Special schematics implemented in the chip provides fine timing with resolution of ≈50 ps.
A charge is measured by the *TOT* method.

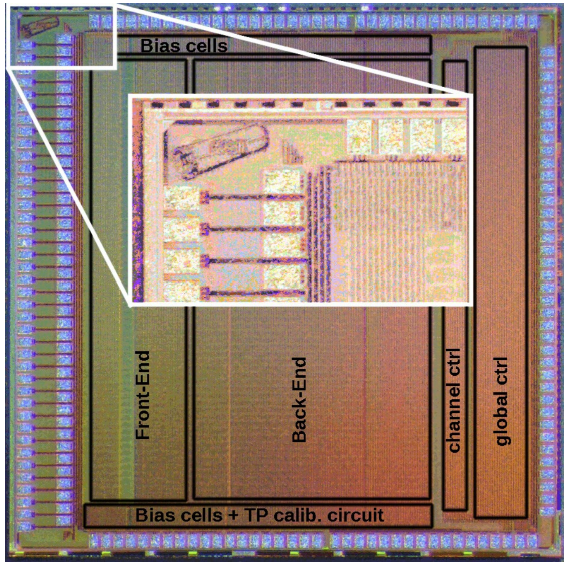
For the *pixel* detector of PANDA a trigger-less readout electronics based on ToPiX chip is under development.

Vertex detector

In Turin the electronics based on **TIGER** chip (Turin Integrated Gem Electronics for Readout) is developed for trigger-less readout *of the GEM detectors*. It provides charge and time measurements.

TIGER parameters

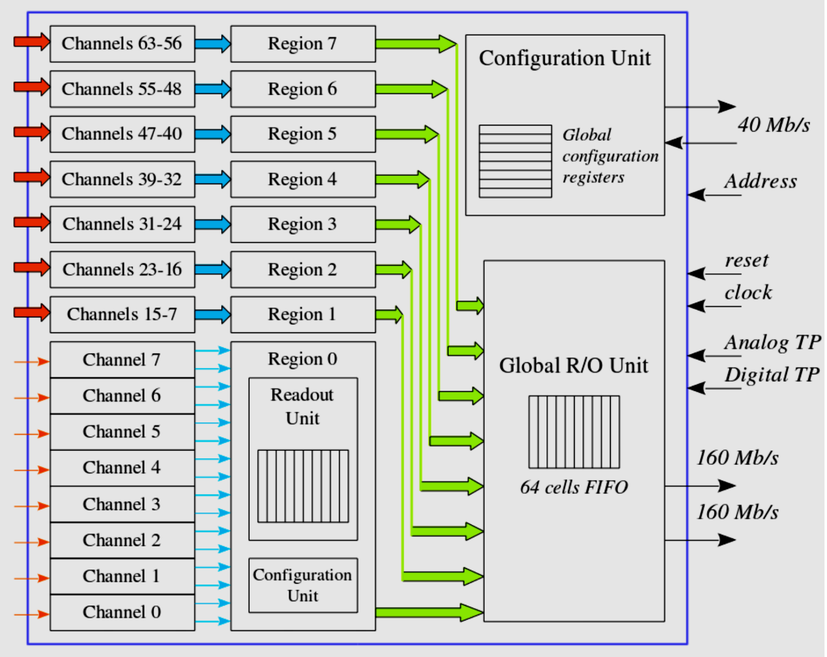
5 x 5 mm^2 110nm CMOS technology
64 channels: preAmp, shapers, TDC/ADC, local controller
Digital backend inherited from TOFPET2 ASIC (SEU protected)
On-chip bias and power management
On-chip calibration circuitry
Fully digital output
4 TX SDR/DDR LVDS links, 8B/10B encoding
Nominal 160 MHz system clock
10 MHz SPI configuration link
Sustained event rate > 100 kHz/ch



Vertex detector

Another project in Turin: **ToASt** (Torino Amplifier for Strip detectors)

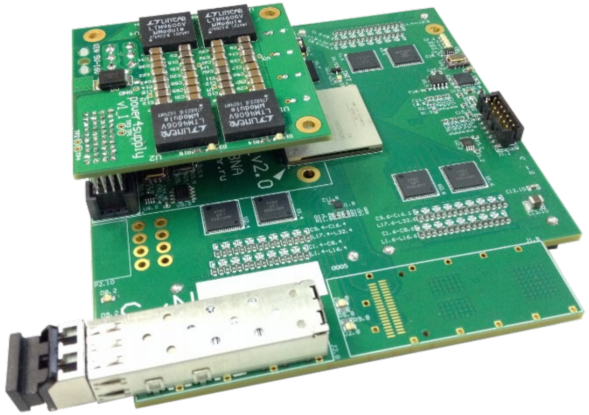
developed for the PANDA MVD strip detector readout
analog front-end from PASTA ASIC
digital back-end derived from ToPiX ASIC



Some parameters:

- 64 input channels
- Time of Arrival (ToA) and Time over Threshold (ToT) measured
- master clock frequency 160 MHz
- 2 output serial links at 160 Mb/s
- charge resolution 8 bit
- time resolution (r.m.s.) 1.8 ns

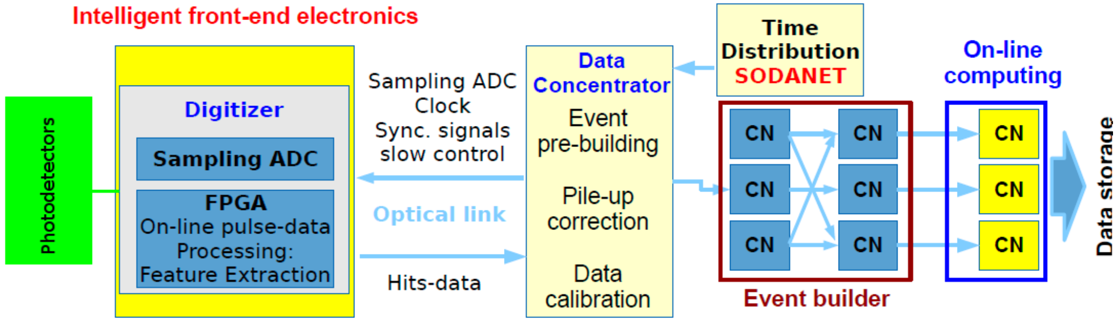
Electromagnetic calorimeter



ADC64s2 *(designed for MPD)* is a 64-channel 12-bit 62.5 MS/s ADC with signal processing core and Ethernet interface. It has dedicated serial links for clock synchronization and data readout. *Was it tested in a trigger-less mode?*

Rather high power consumption: 13W/6W (active/idle)

In PANDA the ADC trigger-less chain is under realization, *but on the μTCA platform …*



The trigger-less readout chain of the PANDA Forward Spectrometer.

Feature extraction algorithm which can be embedded in FPGA allows to greatly decrease the volume of data to be transferred. Instead of transmitting all the ADC samples, only time and charge of the hit are delivered to the next stage of the DAQ chain.
This algorithm is under development in different projects (PANDA, COMPASS, …).

Summary for front-end electronics

1. Front-end electronics suitable for free-running DAQ is being created in different new experiments, some of these developments are close to the final stage.

2. Some modifications could be required for use of this electronics in SPD.

3. The SPD detector groups have to take care of front-end electronics of their detectors capable to run with a trigger-less DAQ.

Common discussions of the detector and DAQ teams are necessary!

Present status

JINR participants:

		FTE
Afanas’ev Leonid	LNP	0.3 ↗
Duginov Victor	LNP	0.2
Frolov Vladimir	LNP	0.5
Gritsay Konstantin	LNP	0.8
Isupov Alexandr	LHEP	0.3
Kulikov Anatoly	LNP	0.7
Tereschenko Vyacheslav	LNP	0.2 ↗

Collaborating partners:

I. Konorov (Munich)

Expressed their interest:

SINP MSU (they proposed to be fully responsible for slow control)

Czech University groups (DAQ development)

Warsaw University of Technology (readout electronics, data compression)

Present status

In 2018 we prepared a room with antistatic floor coating and antistatic furniture in DLNP for DAQ-SPD development, and a lot of equipment has been purchased:

- electronics rack
- VME and NIM crates
- CAEN electronic modules VME and NIM
- high performance oscilloscope and generator
- other devices and tools
- 3 server platforms: two INTEL-based and one AMD-based
- computer components for the servers: hard discs, memories, switches, etc.

Done in 2019:

- two servers physically assembled on the basis of the server platforms Supermicro;
- their individual operation was tuned up, as well as their interconnection via Ethernet switch;
- the server with an AMD processor assembled;
- speed of read/write operations was estimated for the Supermicro platform for different operation systems (Lynux, FreeBSD) and different configurations;
- new order for the components was prepared and submitted, including those for optical links. Mostly obtained.

To do before June:

- to put in operation and test the server with the AMD processor
- to continue the tests of different configurations
- to measure the speed of read/write processes for different platforms and configurations
- to make an order to TUM for the modules (developed by Konorov)

To do by 30.09.2019 for CDR:

to develop preliminary version of general structure of DAQ-SPD

to agree with detector groups a common vision of front-end electronics

to agree with the group responsible for the SPD test zone the composition of the DAQ/Trigger equipment

to prepare the text on DAQ for CDR

Finances

In 2018 the approved budget for DAQ was 100 000 $. In fact, we spent even ~15% more for arranging the DAQ test bench from scratch.

In 2019 the planned budget is 20 000 $. We’ll probably need more (~+20 000 $) to buy some commercial equipment and components, and to pay TUM for the modules.

Finances

In 2018 the approved budget for DAQ was 100 000 $. In fact, we spent even ~15% more for arranging the DAQ test bench from scratch.

In 2019 the planned budget is 20 000 $. We’ll probably need more (~+20 000 $) to buy some commercial equipment and components, and to pay TUM for the modules.

Manpower

It seems, at the present stage we have sufficient group for development of the main DAQ architecture.
But we are short of people to develop and realize different computing services: data bases,
data logger, monitoring, slow control etc.
We also lack for experts for the DAQ hardware, especially those familiar with FPGA programming.

Finances

In 2018 the approved budget for DAQ was 100 000 $. In fact, we spent even ~15% more for arranging the DAQ test bench from scratch.

In 2019 the planned budget is 20 000 $. We'll probably need more (~+20 000 $) to buy some commercial equipment and components, and to pay TUM for the modules.

Manpower

It seems, at the present stage we have sufficient group for development of the main DAQ architecture.
But we are short of people to develop and realize different computing services: data bases,
data logger, monitoring, slow control etc.
We also lack for experts for the DAQ hardware, especially those familiar with FPGA programming.

Other needs

We urgently need to arrange an air conditioning in the DAQ room in DLNP.
The contract for this job was signed last week.

Thank you for attention