CEA - Saclay

Architecture of a Magnet Safety System using a CompactRIO and LabVIEW for the STAARQ magnet test stand in CEA Saclay

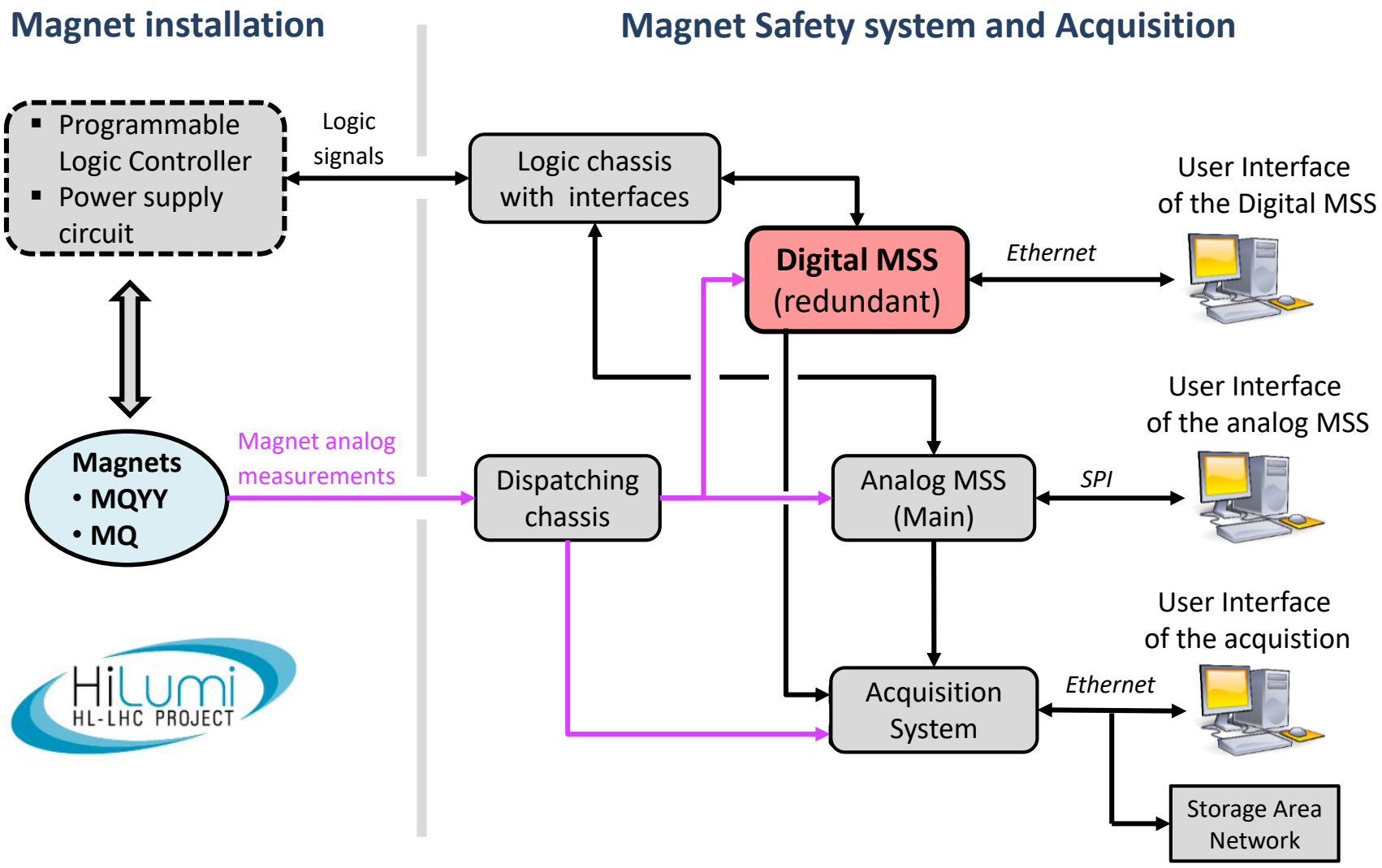
- **Philippe De Antoni,** S. Trieste, Y. Le Noa and F. Molinié
- **Irfu/DIS/LEI**

  Irfu: *Institute of Research into the Fundamental Laws of the Universe*
  DIS: *Systems Engineering Department*
  LEI: *instrumental electronics laboratory*

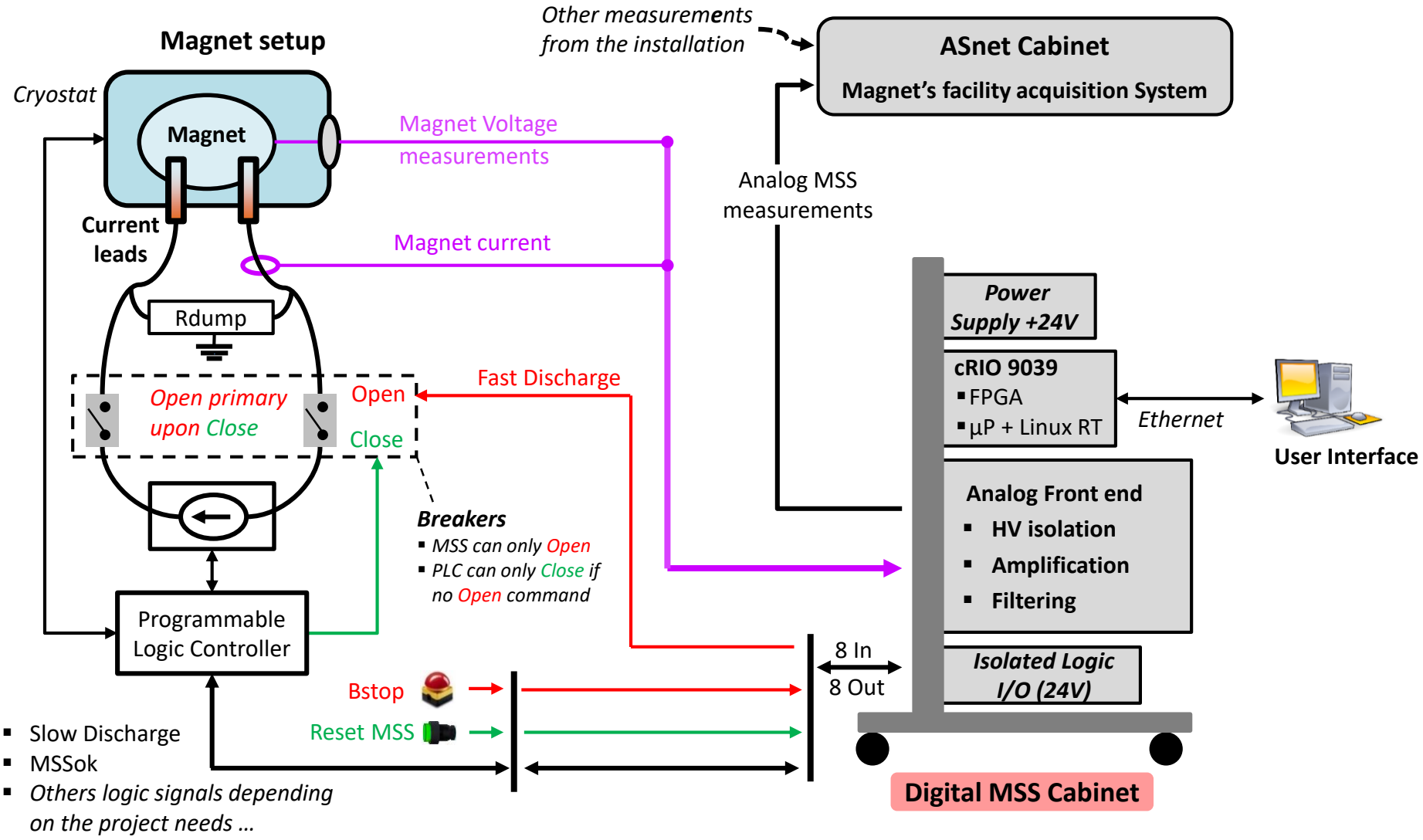
Outline

- **STAARQ magnet test stand overview**
- **MSS hardware**
  - Digital MSS cabinet architecture
  - Analog front end and Compact RIO layout
- **MSS software**
  - Main specifications and architecture
  - MSS core in the FPGA
  - Interface FPGA / µP
  - µP code functionalities and architecture
  - Compact RIO Resource usages
- **Conclusion**

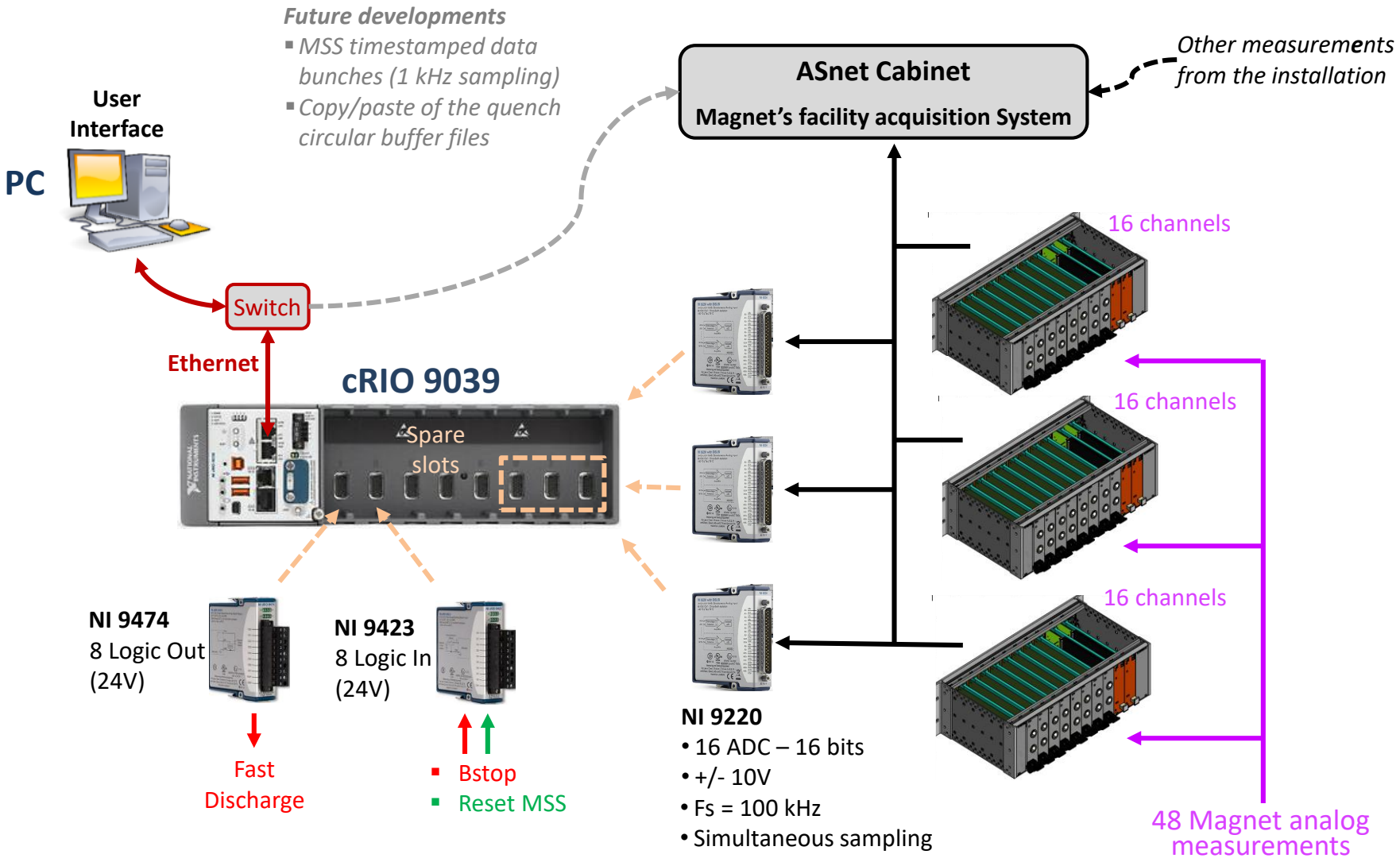
STAARQ – Magnet Test stand – Overview drawing



Digital MSS – Cabinet and interfaces overview



Subsystem's interconnections



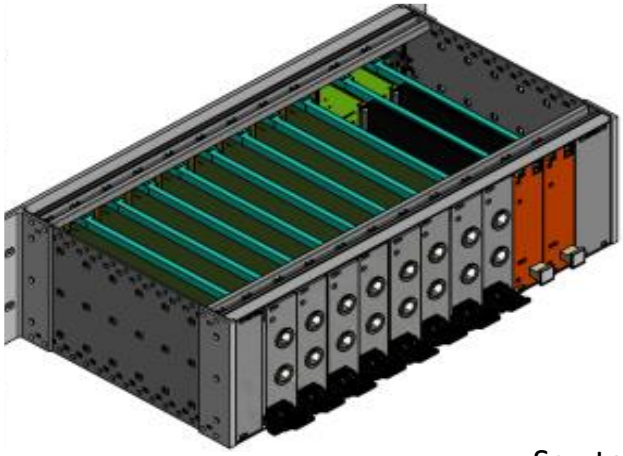
Analog Front End chassis

2 Channels

- Gain 1 to 500 - Bandwith = 1 kHz
- ± 10 V differential outputs/channel
- ± 10 V single ended outputs/channel
- Isolation working voltage = 3,5 kV

8 boards by chassis

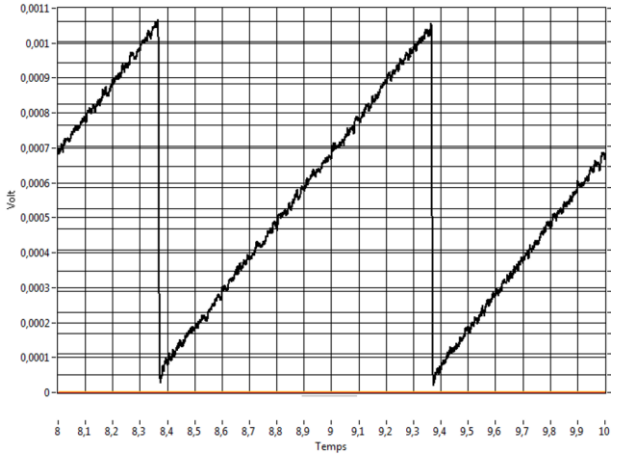
- The chassis works like an isolated voltmeter
- 16 channels



Module NI 9220 in the cRIO

- 16 ADC - 16 Bit
- Simultaneous sampling
- Fs =100 kHz
- ± 10 V differential input

Sawtooth 1 Hz, 1 mVpp, G=20.9



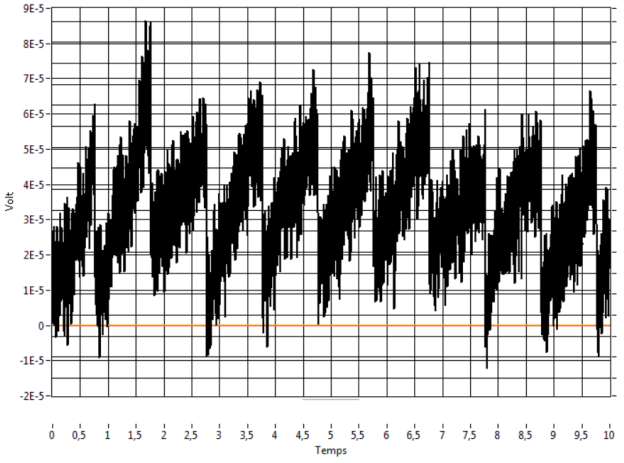
Measurements acquired with the digital MSS

- Digital filter bandwidth = 100 Hz

Worst case configuration

- HV input bridge attenuation = 10
- High input impedance to sustain 3.5kV permanently

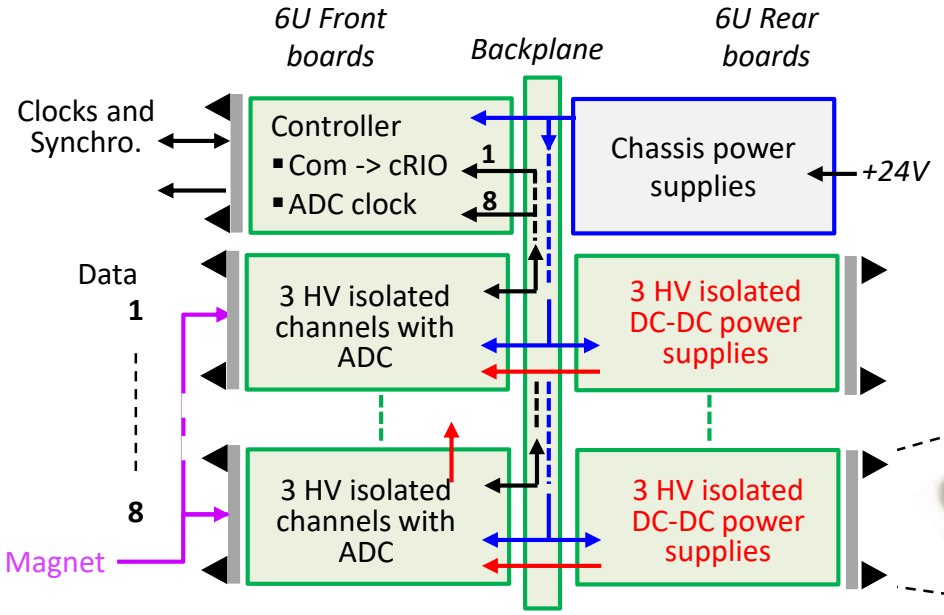
Sawtooth 1 Hz, 40 µVpp, G=20.9

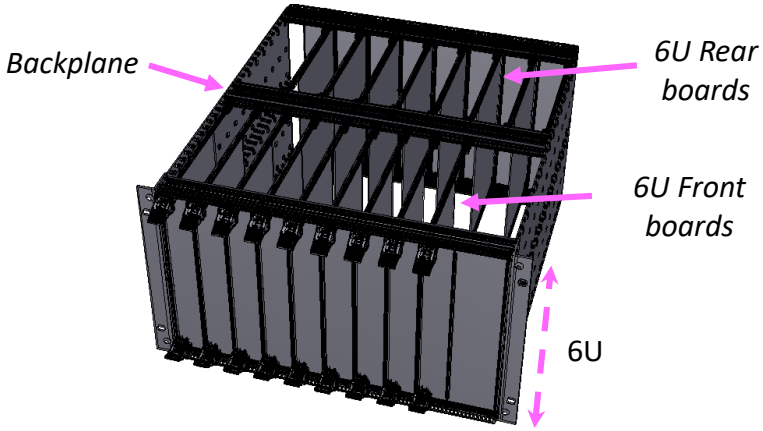


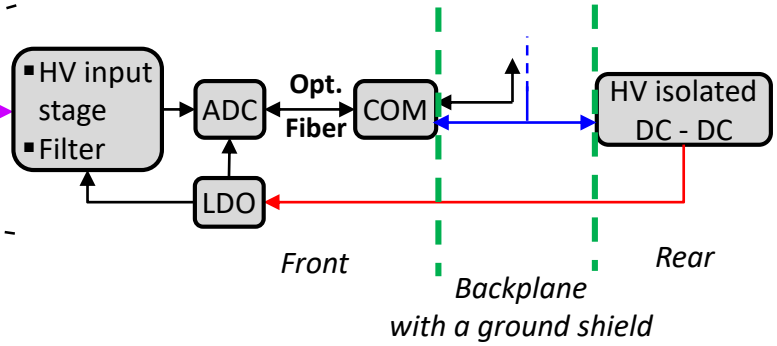
Front end chassis with ADC – Next generation - 1

General specifications

- 24 channels Front end chassis with a high working voltage = 4,2 kV
- Separation of the isolated DC-DC power supplies from the ADCs
- Front-end chassis runs in stand alone – The compactRIO only reads the measurements when ready

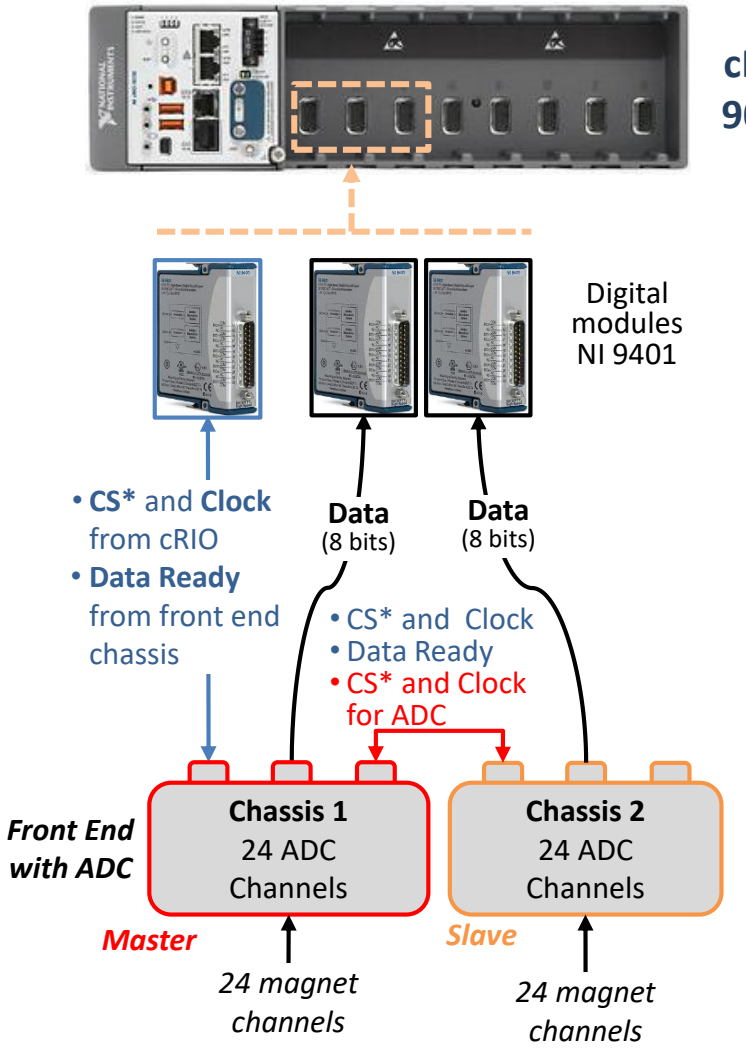






Front end chassis with ADC – Next generation - 2

- Each chassis has a control board to run the acquisition with the ADC and the reading with the cRIO
- Master Chassis
  - Provides the **CS*** and the **Clock** to trigger and read the 24 ADC of each chassis (local and internal operations)
  - Processes the **CS***, **Clock** and **Data Ready** signals for the data reading of the 2 chasssis by the compact RIO
- Slave Chassis
  - Run in parallel with the master chassis



MSS software – Main specifications

The MSS software is implemented in a cRIO 9039

- The cRIO chassis must run in stand alone mode after the power up
- The MSS core for the quench detection must be embedded in a FPGA
  - Objective of reliability
- All the others MSS functions are embedded in the FPGA or in the µP with Linux real time, depending on the architecture and the needs.

  Examples:

  - Writing of the MSS parameters in the MSS core at cRIO power-up (normal operations) or during MSS tuning (magnet commissioning)

    MSS parameters: Offset, Gain, Coefficients of the equations, voltage and time thresholds

  - Circular buffer with the pre-trigger and post-trigger functionalities to record the quench incident
  - Others …

CompactRIO – General architecture

CEA – Saclay

MSS Software

The MSS software is split in 4 layers

- UI – PC
- RT – µP with Linux RT
- HUB – FPGA
- MSS core – FPGA

PC
User Interface

Ethernet

cRIO 9039

Spare slots

NI 9474
8 Logic Out (24V)

Fast Discharge

NI 9423
8 Logic In (24V)

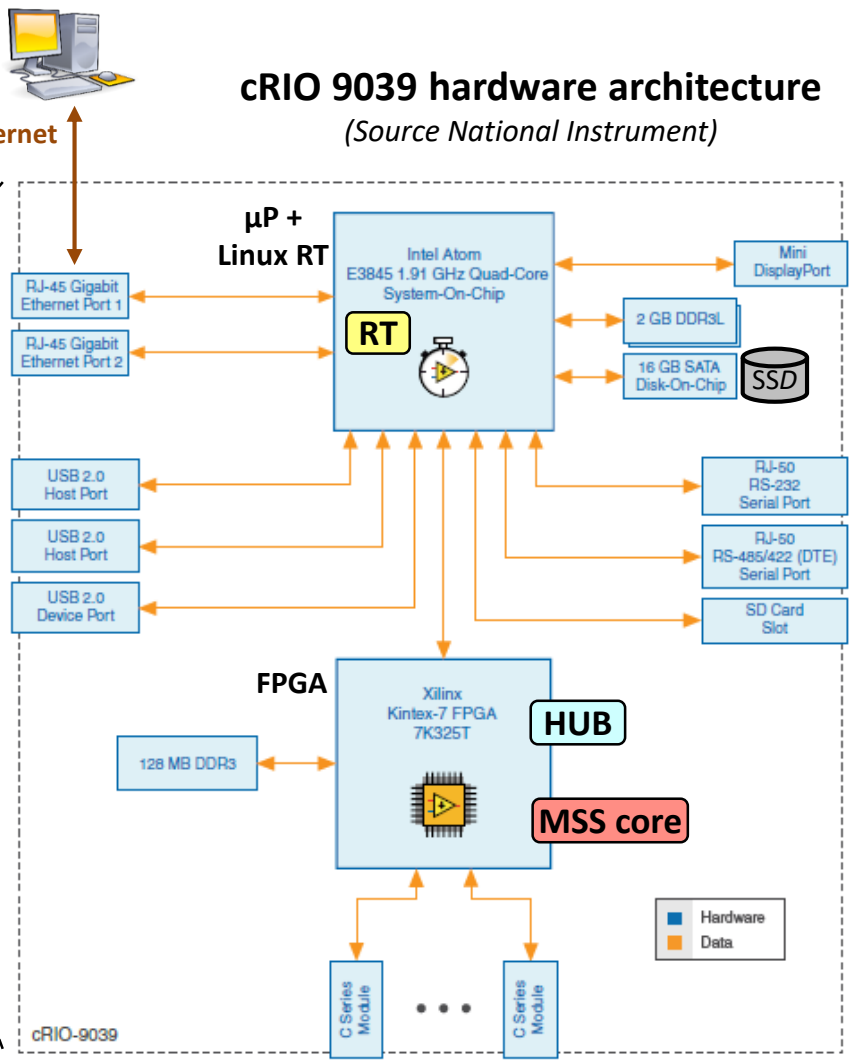
- Bstop
- Reset MSS

NI 9220
- 16 ADC – 16 bits
- +/- 10V
- Fs = 100 kHz

Magnet analog measurements

cRIO 9039 hardware architecture

(Source National Instrument)



General software architecture

- MSS software is split in 4 layers
- Each layer is independent from the upper one
- UI -> LabVIEW
- RT -> LabVIEW Real Time
- MSS core, HUB -> LabVIEW FPGA

Magnet analog measurements
- Sampling at Fe=100 kHz

Fast Discharge

- Bstop
- Reset MSS

UI *Layer 4*

MSS Supervision
- Normal running or Tuning
- Monitoring cRIO

4 * Networkstreams (FIFO mechanism using TCP/IP)

Ethernet

cRIO 9039

FPGA (Clock = 40 MHz)

MSS core

Digital processing for the quench or default detection

Layer 1

- Reset MSS
- Interlocks

Synchronous and triggered operations

HUB

Low level data interfaces and distribution

Layer 2

- **3 Controls**
- **1 Indicator**

4 * DMA FIFO

µP & Linux RT

RT

SSD

cRIO monitoring

Data streaming for UI

Circular buffer & SSD saving

Rd/Wr MSS Parameters

Monitoring datas, Synchros. & Interlocks

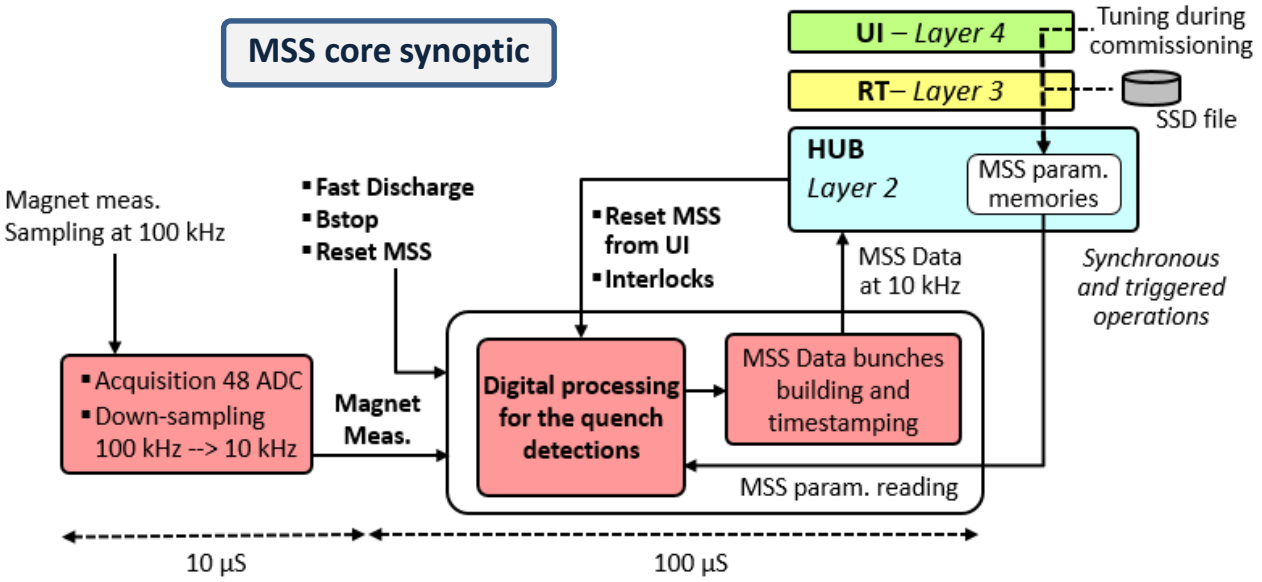
Layer 3

MSS core – General Description

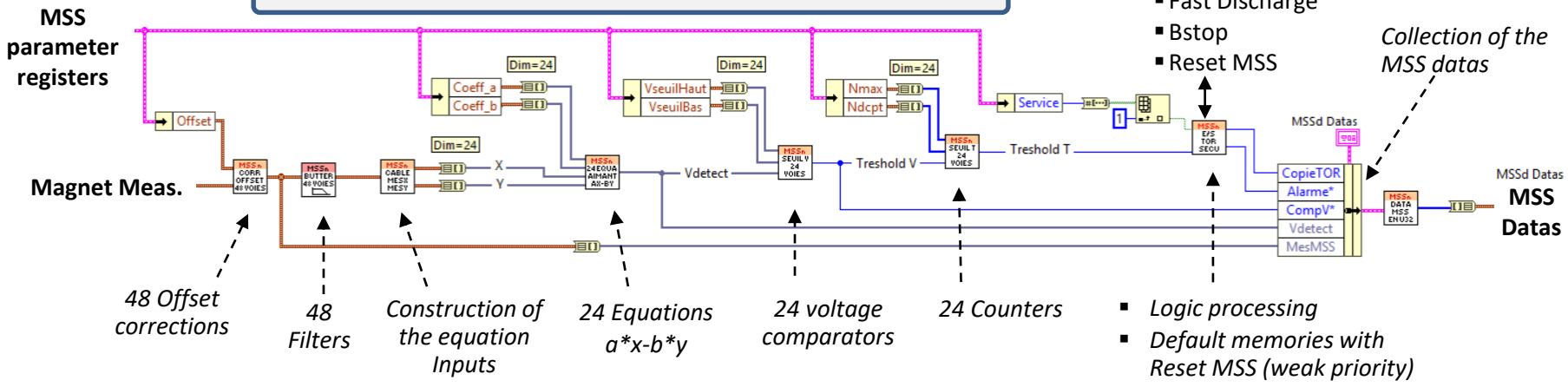
MSS core

- MSS core is the copy of an MSS analog processing
- MSS parameters
  1. Download from a SSD file during power up of the cRIO
  2. Modifications with the UI only allowed during commissioning
- MSS core runs in stand alone mode. Its state depends only of:
  1. The magnet measurements
  2. The logic Inputs

MSS core synoptic



MSS core: Top level of the DSP (LabVIEW FPGA code)



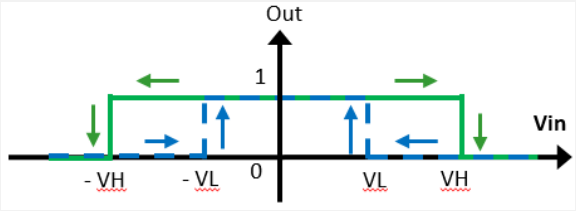
MSS core – Procedure to design the stages

For each stage
- Specifications
- VI FPGA code design
- Test bench

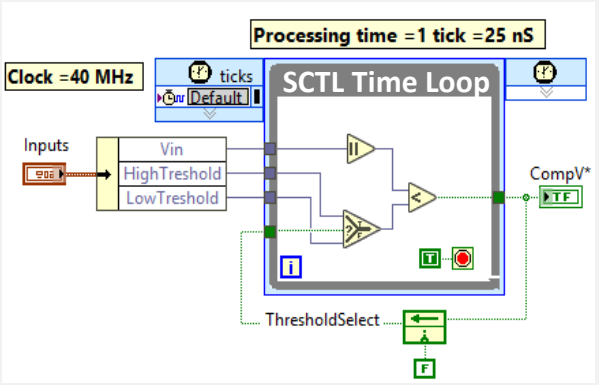
When all the stages are designed and tested
- FPGA compilation
- Reports analysis (Timing, …)

Example: Design of the one channel voltage comparator

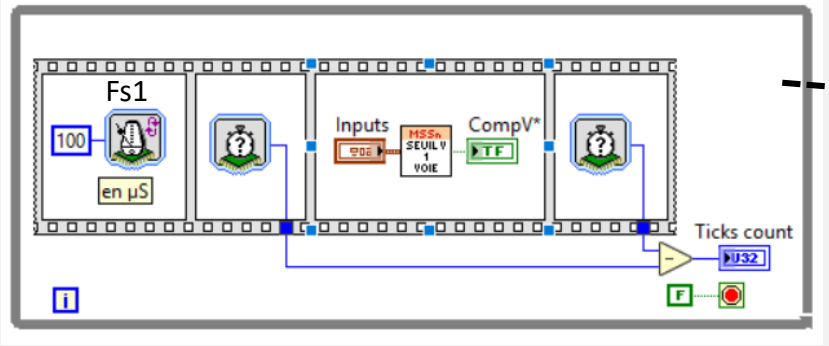
1. Specifications of the voltage comparator



2. VI Code of the Volt. Comp.

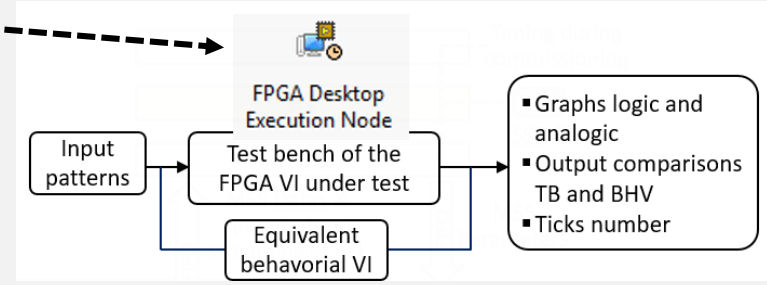


3. VI FPGA test bench with the calculation of the ticks number



Building of an Execution Node

4. Test bench

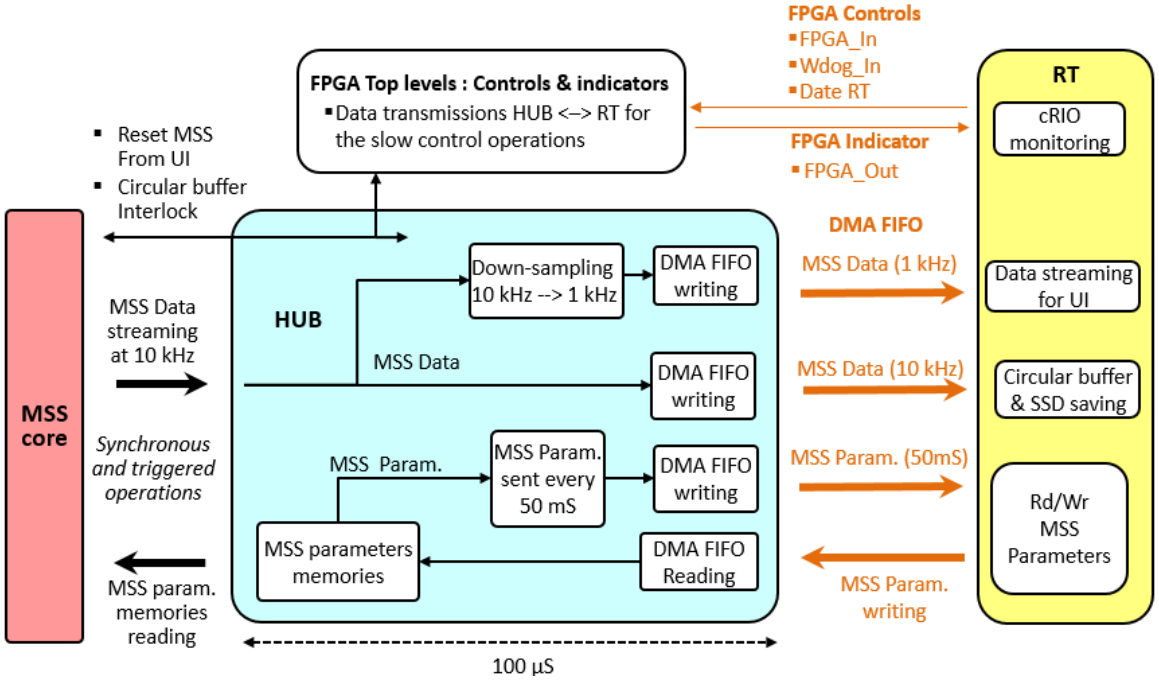


HUB – General description

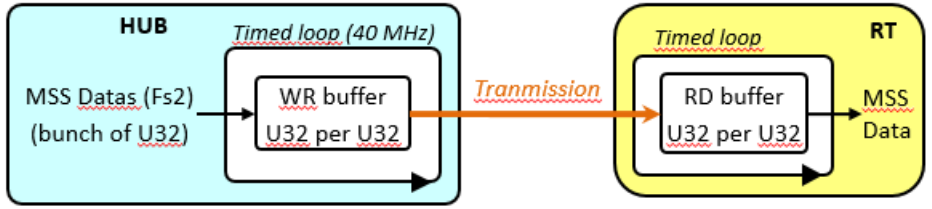
HUB

- Interface between MSS core and RT
  1. Data stream of MSS datas (Fs1) toward the circular buffer
  2. Data stream of MSS datas (Fs2) toward the UI
  3. Data stream of MSS parameters (50 mS) toward the UI and MSS parameters writing
  4. Slow control operations
     - Monitoring DMA oveflows and Reset of DMA monitoring
     - Interlock between the FPGA and RT when the circular is recording
     - Out logic toward the PLC of the RT watchdog state
     - Reset MSS from the UI
     - - - -
- The number of FPGA top levels Controls & indicators are minimized to save FPGA resources
- Design methodology comparable to the MSS core

Controls & Indicators work like asynchronous registers



Example of a **DMA FIFO** mechanism for the data stream



RT – General description

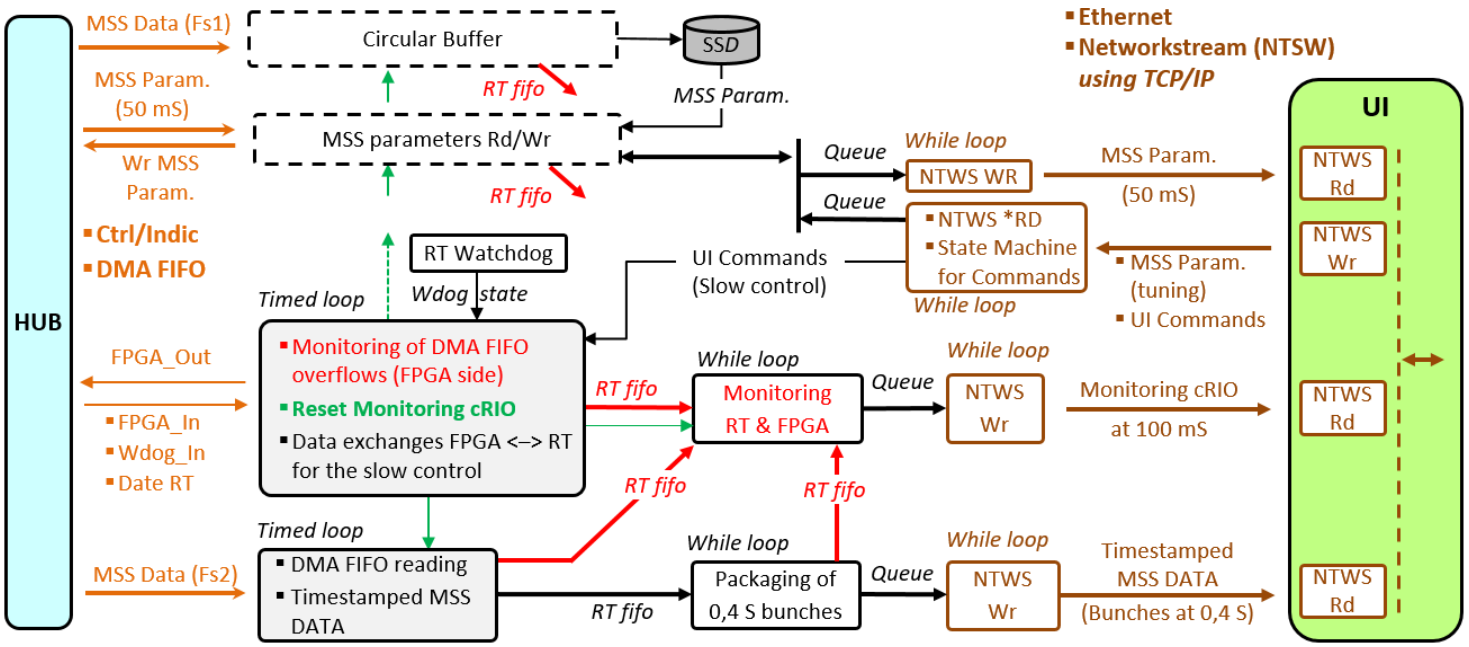
CEA – Saclay

Real time MSS functions

- Circular buffer using the 2 GB (DDR3) and the SSD disk
- Reading (for verification) and writing of the MSS parameters (from SSD or UI)
- Interface with between the HUB and the "high level" devices (Ethernet, USB …) for data streaming, asynchronous single Rd/Wr …

Real time monitoring of the cRIO (FPGA and RT)

- CPU et memory usages
- Time loops: Finish late, actual execution times
- All loops except NTWs: error code, Overflows (DMA, RT fifo, Queue …)
- RT watchdog



- UI manages the connections and the disconnections of the NTWS
- cRIO acts like a slave

cRIO – FPGA used ressources

CEA – Saclay

Table 6: **Kintex-7 FPGA Feature Summary by Device**

Device	Logic Cells	Configurable Logic Blocks (CLBs)		DSP Slices[2]	Block RAM Blocks[3]			CMTs[4]	PCIe[5]	GTXs	XADC Blocks	Total I/O Banks[6]	Max User I/O[7]
		Slices[1]	Max Distributed RAM (Kb)		18 Kb	36 Kb	Max (Kb)						
XC7K70T	65,600	10,250	838	240	270	135	4,860	6	1	8	1	6	300
XC7K160T	162,240	25,350	2,188	600	650	325	11,700	8	1	8	1	8	400
XC7K325T	326,080	50,950	4,000	840	890	445	16,020	10	1	16	1	10	500
XC7K355T	356,160	55,650	5,088	1,440	1,430	715	25,740	6	1	24	1	6	300

- 48 measurement channels
- 24 equations : a*x-b*y
- *Down-sampling filters 10kHz to 1kHz not implemented in the HUB*

Reports

Summary

Compilation completed successfully.

200 MSS parameters (U32) using logic resources

Device Utilization

Total Slices: 58,1% (29605 out of 50950)
Slice Registers: 31,0% (126291 out of 407600)
Slice LUTs: 32,0% (65306 out of 203800)
Block RAMs: 4,3% (19 out of 445)
DSP48s: 13,5% (113 out of 840)

Timing

80MHz (Used by non-diagram components): 80,00 MHz (Met MHz maximum)
40 MHz Onboard Clock: 40,00 MHz (Met MHz maximum)

Reports

Summary

Compilation completed successfully.

400 MSS parameters (U32) using RAM Blocks

Device Utilization

Total Slices: 34,1% (17358 out of 50950)
Slice Registers: 11,6% (47092 out of 407600)
Slice LUTs: 21,9% (44654 out of 203800)
Block RAMs: 8,1% (36 out of 445)
DSP48s: 2,3% (19 out of 840)

✓ **No timing violation in the design**

Timing

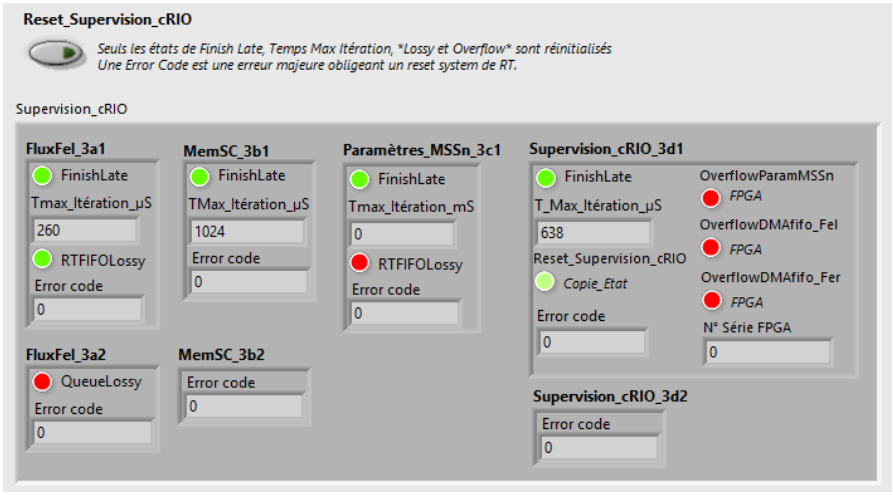
40 MHz Onboard Clock: 40,00 MHz (Met MHz maximum)
80MHz (Used by non-diagram components): 80,00 MHz (Met MHz maximum)

cRIO – RT used ressources - Monitoring

Intel Atom E3845 – 4 cores - 1.91 GHz

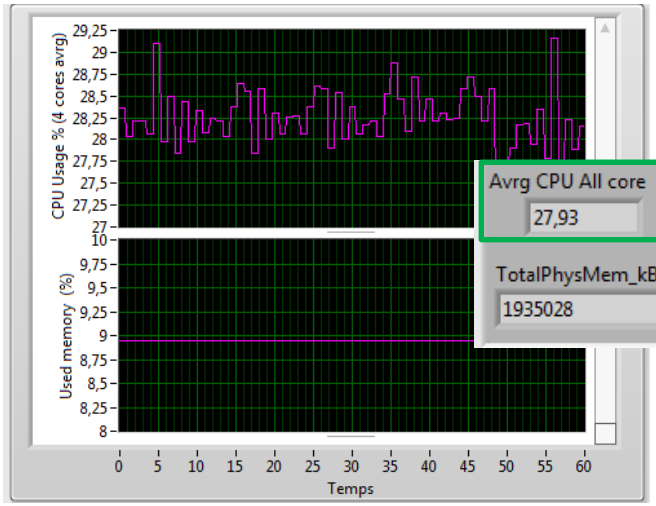
- More than 1,5 GB to build the circular buffer at 10 kHz
- CPU and memory usages are low and flat
- The views are part of the cRIO monitoring panels

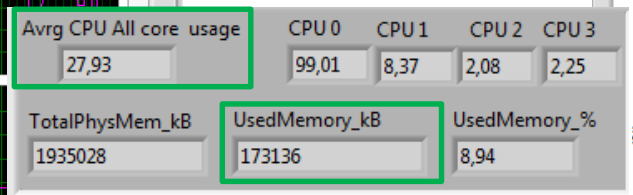
RT loops and DMA fifo monitoring



Memory and CPU usages

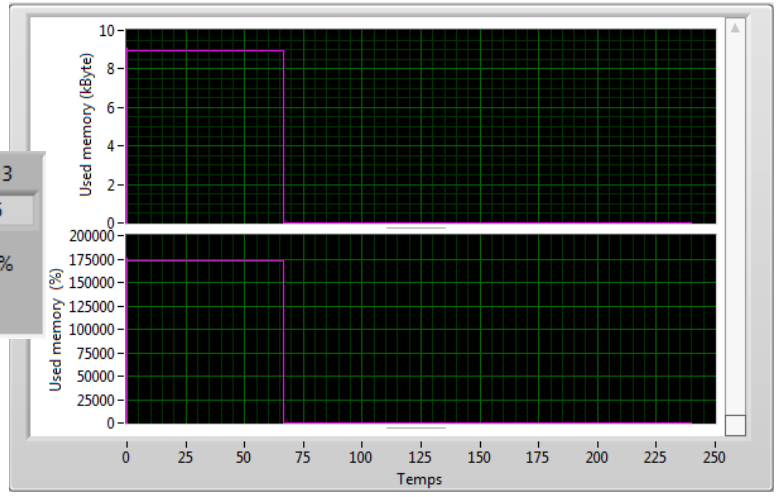
Fast variations (sampled every 0,1S)





Time in S

memory usage - Long term variations



Time in hour (10 days)

Conclusion

- **MSS hardware architecture is modular for the scalability**
- **MSS software is split in 4 layers from the FPGA to the UI**
  - Each layer is independent from the upper one
  - Each layer follows a modular organization to facilitate the modifications and the adding of new functions
  - The MSS core in the FPGA runs in stand alone mode
- **Next stages of the digital MSS project**
  - Tests with simulated quench signals
  - Test with the quadripole Q4 short (Hi-LUMI)
    - Measurements only
    - No connection with the breakers
    - October 2019
  - Deployment for the STAARQ test facility
    - As a redundant MSS of an analog MSS
    - Exploitation april 2021