DE LA RECHERCHE À L'INDUSTRIE

www.cea.fr

SALSA ASIC DSP features and interface with EPIC readout chain

Irakli Mandjavidze and Damien Neyret (CEA Saclay IRFU) for Sao Paulo University and CEA IRFU teams

EPIC DAQ/electronics WG meeting

18/04/2024

SALSA specifications

EIC and EPIC basics

SALSA architecture

DSP features

Synchronous command management

Output data

Summary of open questions

SALSA CHIP TARGET SPECIFICATIONS

Versatile front-end characteristics

- Dedicated to MPGD detectors and beyond
- 64 channels
- Large input capacitance range, optimized for 50-200 pF, reasonable gain up to 1nF
- Large range of peaking times: 50-500 ns
- Large choice of gain ranges: 0-50, 0-250, 0-500 fC or 0-5 pC
- Large range of input rates, up to 100 kHz/ch with fast CSA reset (limit assumed for EPIC: 25 kHz/ch)
- Reversible polarity
- Front-end elements can be by-passed

Digital stage

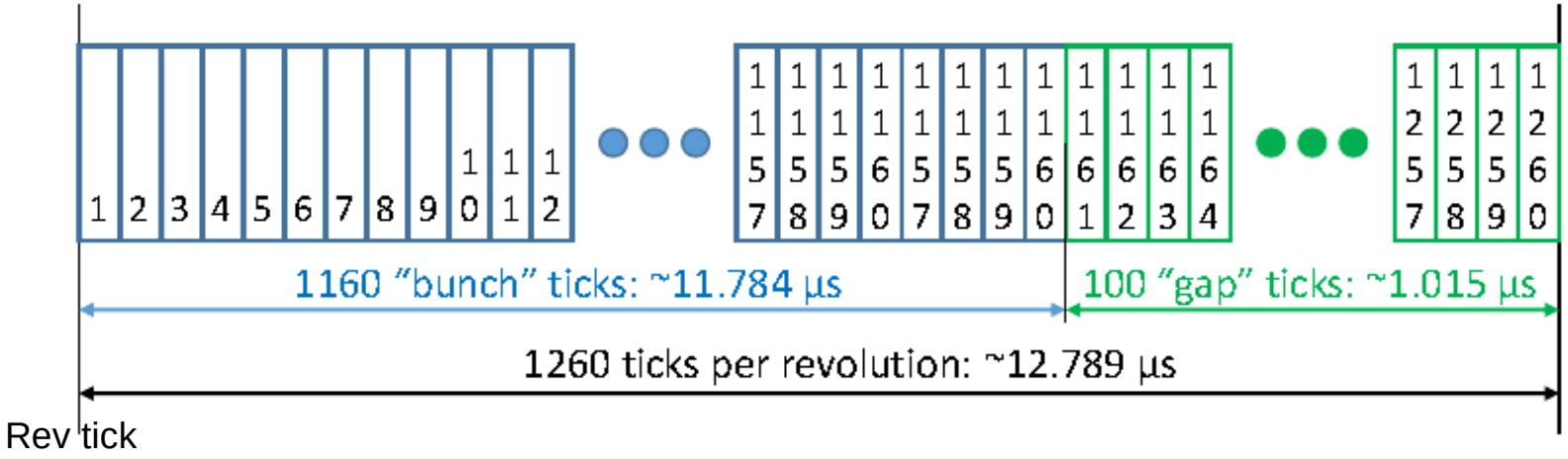
- Fast sampling ADC for each channel on 12 bits (> 10 effective bits) at up to 50 MS/s
- Possibility under study to double rates by coupling pairs of channels
- Integrated DSP for internal data processing and size reduction, treatment processes to be selected according to user needs
- Continuous readout compatible with streaming DAQ foreseen at EIC, triggered mode also available
- Several 1 Gb/s output data links

General characteristics

- ~1 cm² die size, implemented on modern TSMC 65nm technology
- Low power consumption ~ 15 mW/channel at 1.2V
- Radiation hardened (SEU, TID)

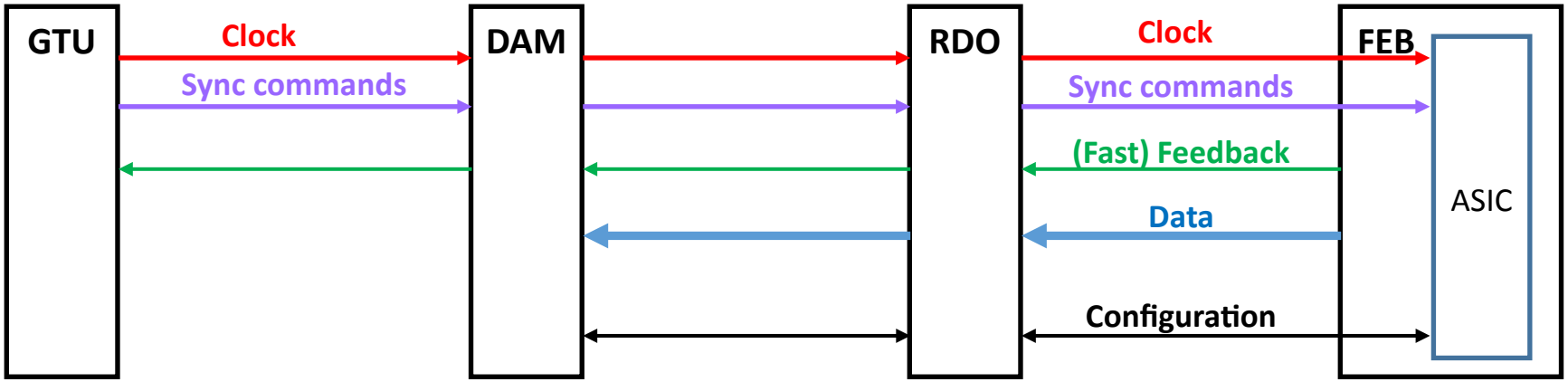
EIC beam structure and bunch crossing timing

- Beam structure repeats every ~12.7886 μs
  - → Revolution frequency: ~78.195 kHz
- 1260 clock ticks in each revolution
  - → Clock period: ~10.14968 ns, frequency 98.52525 MHz
  - → 1160 filled “bunch” ticks, may be not all filled with particles
  - → 100 “gap” ticks without particles



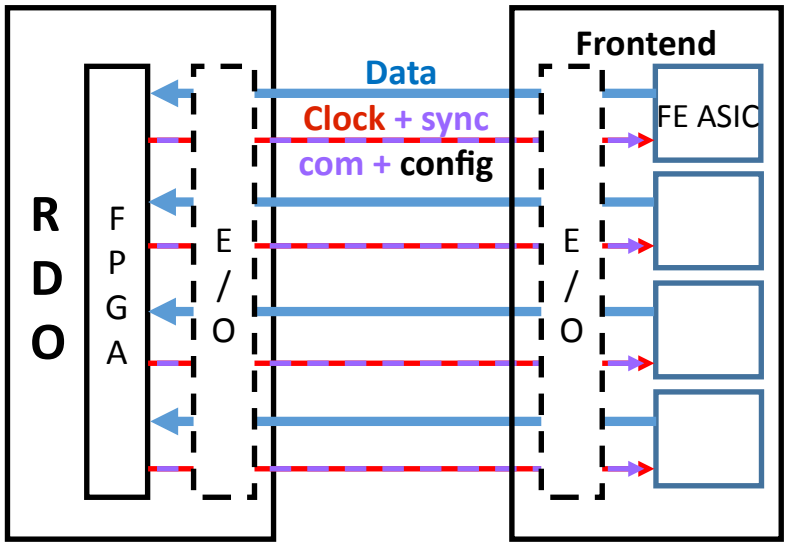
- Clock delivered to front-end ASICs as System clock
  - → Sampling clock (50 MHz) to be generated in ASIC from system clock

Communications chain in EPIC DAQ

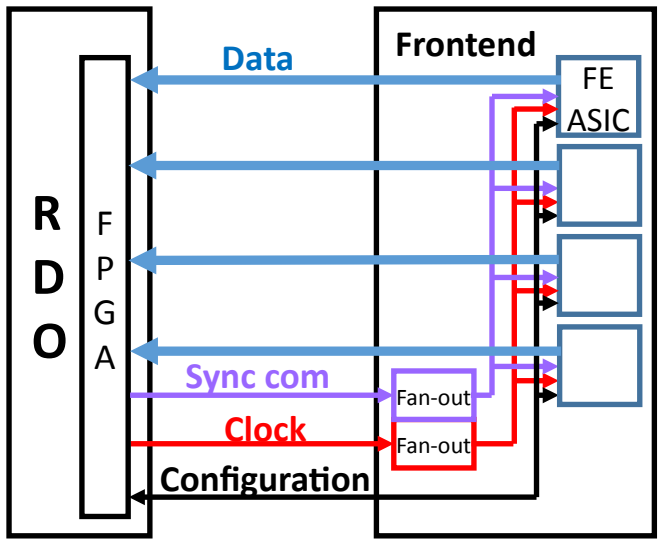


Two options for RDO to frontend board communications

Embedded signals (baseline option)

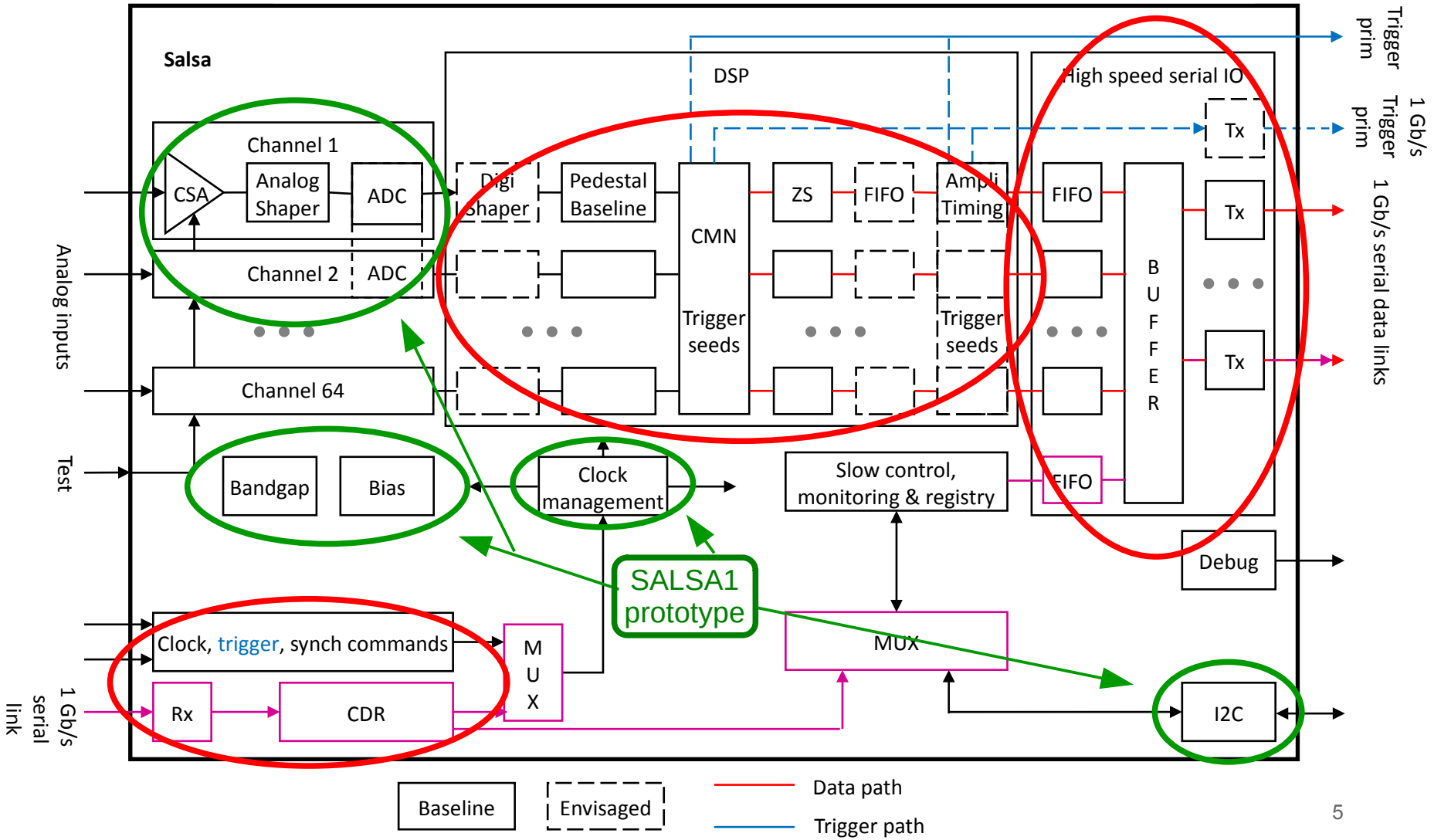


Split signals (backup option)



PRELIMINARY CHIP ARCHITECTURE

DSP, input and output interfaces discussed today to be implemented in SALSA2 prototype



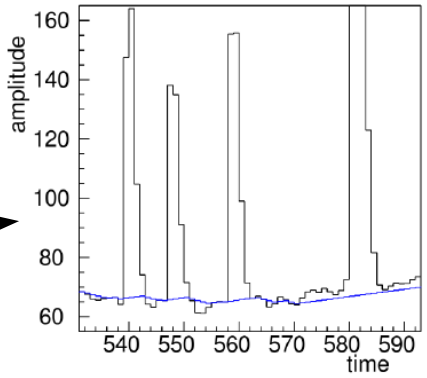
DSP DATA PROCESSING, PRELIMINARY VERSION

General remarks

- Data processing, reduction and formatting from ADC values to output links
- Each process can be deactivated individually by user
- Process parameters through ASIC registers
- Part of codes from SAMPA chip
- Most of DSP features determined, details still under study, suggestions welcome !

Baseline corrections

- Pedestal subtraction with fixed value per channel
- Common mode correction to reduce common noise impact, based on median value of samples of all channels for each sample time
- Baseline slope following algorithm
- Other baseline corrections suggered ?



Plot from SAMPA doc

Digital shaping

- Cancellation of signal tail or peaking time correction with cascade of 4 first order IIR filters
- Algorithm from SAMPA, 2 x 4 parameters

$$y[n] = a_1 y[n-1] + b_0 x[n]$$

DSP DATA PROCESSING, PRELIMINARY VERSION

Zero suppression

- Keep samples above fixed thresholds, possibly neighbor ones in time and in channel number
- Possibility to drop too short set of samples above threshold
- Possibility to keep 1 sample over N, or to limit the number of samples to keep
- Possibility to keep raw data from time to time (to be defined) to monitor reconstruction

Feature reconstruction

- To further reduce data flux by extracting reconstructed data
- For instance peak finding algorithm, with extraction of amplitude + time + width
- Algorithm to be determined, must be simple enough to be compatible with ASIC constraints

Trigger management (not at EPIC)

- Samples kept when trigger signals received, with configurable latency
- Followed or not by zero suppression, feature reconstruction, etc...

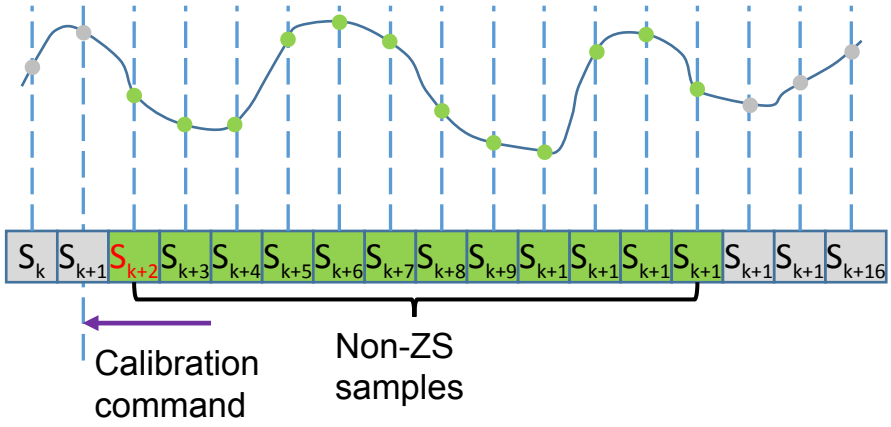
Trigger generation

- Trigger primitives generated when samples above threshold, with conditions on number of samples, multiplicity, etc...
- Possibility to reduce latency by placing trigger generation early in the processing chain
- Nature of trigger primitives to be defined (logic signal, data on specific fast link, etc…)
- Is this feature needed for EPIC ? What requirement on latency ? On trigger primitives ?

DSP DATA PROCESSING, PRELIMINARY VERSION

Calibration data

- Generated on demand with specific synchronous commands
- Generation of calibration data of several types
  - non-ZS raw or corrected data
  - test pulses injected at front-end on one or several channels
  - what else is needed ?
- Taken during empty bunch time gap ? During specific runs ?



Information data

- Monitoring data like chip configuration, internal chip status (currents, voltages), environmental data (temperature, radiation, etc...)
- Used to transmit slow-control responses
- Software scaler histogram to evaluate occupancy per channel evaluation, interesting for detector monitoring ? Other suggestion ?
- Generated on demand with specific synchronous commands and/or slow-control

STRUCTURE OF DATA OUTPUT

Output data stream based on packets

- Different types of packets, see below
- Packet identification by type, numbering and optionally timestamps
- Each packet buffered and transmitted through one of the Gbit/s links (1 only at EPIC)
- Size of data words to be defined (24 bits ? 32 ? 48 ?). Constraints from EPIC DAQ chain ? Maximum size ?

Physics data packets

- Header + ADC sample values + reconstructed values
- Includes timestamps, chip address, channel numbers, possibly flags
- Sample data structure channel by channel
- Detail of format under discussion (cf next slide)

Calibration data packets

- Same format as physics packets + type of calibration data

Information packets

- Carry information data: ASIC configuration, slow-control feedback, environmental informations, channel counting rates, etc...

Error packets

- Information packet generated when error or warning encountered in ASIC

DATA FORMAT, UNDER DISCUSSION

Example of packets with 24-bit words

Bits	23							16	15				11			8	7							0
Packet header line 1 Common for all types of packets	1	1	Type of packet 3b					Chip address 3b	Frame number 4b				Packet number 12 LSB bits (4096)											
Packet header line 2 Optional	1	1	0	System clock timestamp of packet 21 LSB bits (2.1e6 ~ 21ms with 100 MHz clock)																				
Sample subheader	0	0	Number of sample words 4b (16)					Channel number 6b (64)					Frame number 2 LSB			Sampling time stamp 10b (0-1024)								
Sample word	Sample value of the 1st sample time 12b												Sample value of the 2nd sample time 12b											
Sample word	Sample value of the 3rd sample time 12b												Sample value of the 4th sample time 12b											
	⋮												⋮											
Sample word	Sample value of the Nth sample time 12b												Sample value of the N+1th sample time 12b											
Reconstructed data subheader	0	1	Type of reconstructed data 4b					Channel number 6b (64)					Reconstructed value 1 (for instance: time)											
Reconstructed data values	Reconstructed value 2 (for instance: amplitude)												Reconstructed value 3 (for instance: TOT)											
Packet trailer	1	0	0	Flags 3b				Checksum of packet					Total size of packet in number of 24-bits words 12b (4096)											

SALSA SYNCHRONOUS COMMANDS, PRELIMINARY VERSION

Context

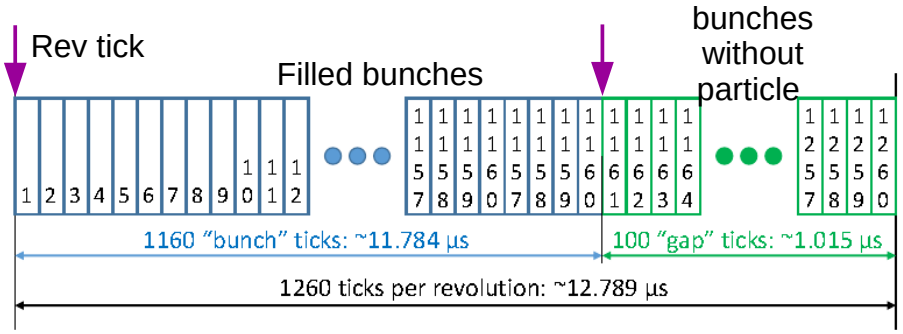
- Commands received from DAQ in synchronization with system clock (98.5 MHz) on 6 bits
- Can be received at each clock with embedded signals, but not in case of split signals

Data taking management commands (those useful for EPIC)

- **T0SYNC**: new time frame → reset packet and clock counters, realign clock phases, and make chip ready to read data in a new time frame
- **STARTREAD**: activate sample data generation in DSP
- **ENDREAD**: deactivate sample data generation in DSP, finish to process remaining samples in FIFOs, then send a specific packet when no more sample is remaining
- **CALIB0...N**: generate calibration data of type N
- **INFO0...N**: generate information packet of type N

Correspondence with EIC time structure ?

- When do we do T0SYNC ?
  - Each Rev tick ? One over N ?
  - Or do we follow DAQ time frames ?
- Do we read bunches without particle ?
  - We can always get calibration data from that gap even if not reading



OPEN QUESTIONS TO EPIC MPGD AND DAQ GROUPS

DSP processing

- Most of DSP features determined, but still room for adjustments
- Opinions about baseline corrections, digital shaping, and zero suppression algorithms ?
- Suggestions about peak finding algorithm ?
- Trigger generation feature interesting in the EPIC DAQ logic ? With which latency ?
- Is there a strategy on MPGD detector calibrations and monitoring based from data ? Is there an interest on counting rates per channel measured in SALSA ?

Data format

- Any constraints on output format ? Word size ? Maximum size of packets ?
- Additional information needed for data analysis ? For detector calibrations ?

Synchronization with EPIC DAQ and EIC accelerator

- Maximum time between two synchronous commands ?
- Compatibility with EPIC DAQ time frames ? What if not correlated with Rev ticks ?
- Time frame length matters on size of internal memories and clock counters, what capacity in SALSA ?
- Compatibility of SALSA synchronous commands with the EPIC DAQ to be checked