

Front End Electronics for the SuperB IFR: prototype development and FE design outline

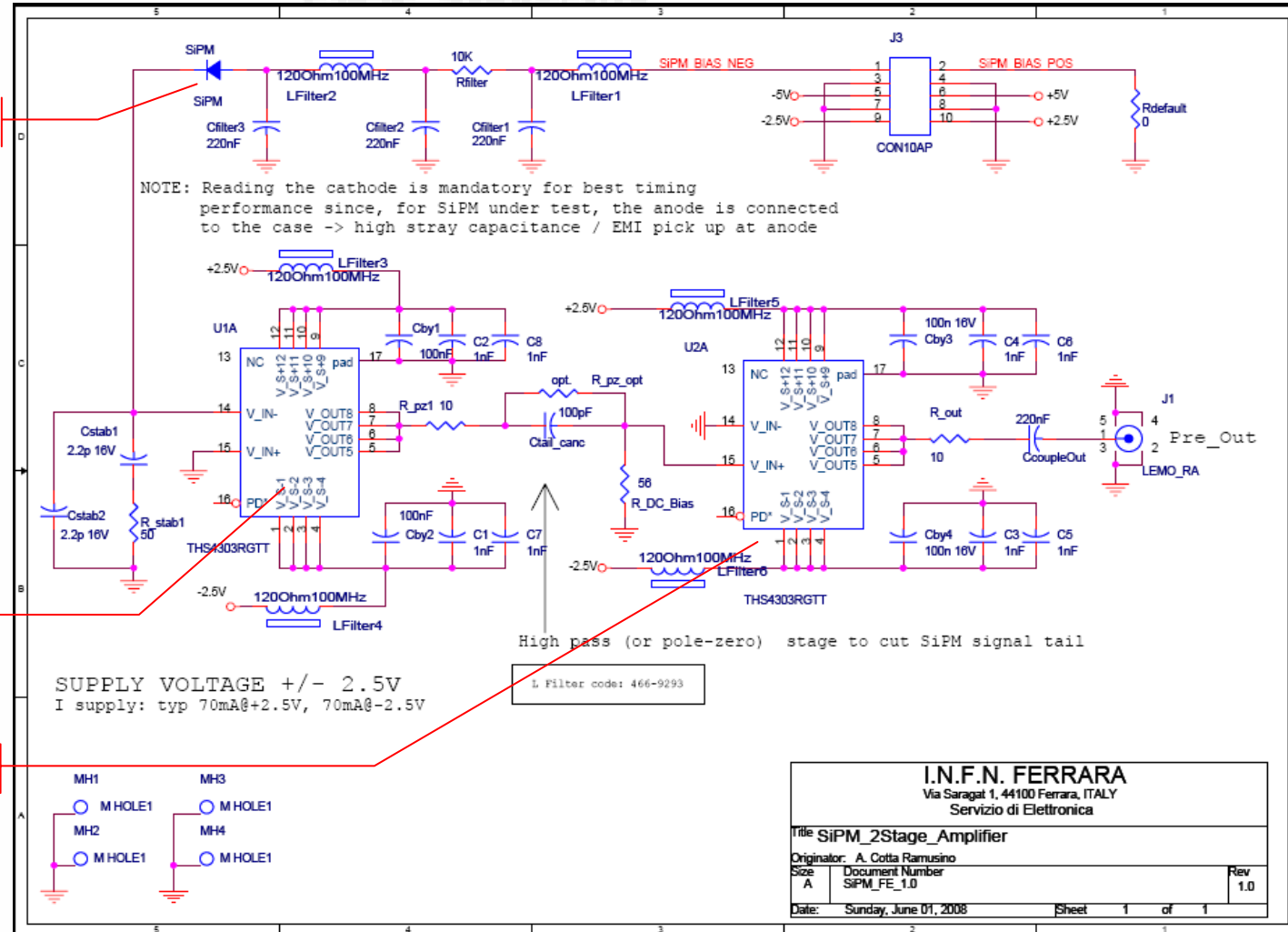
1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara

Summary

- 1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara
- 2) basic features of Front End electronics for the SuperB IFR

1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara

SiPM
(cathode readout)



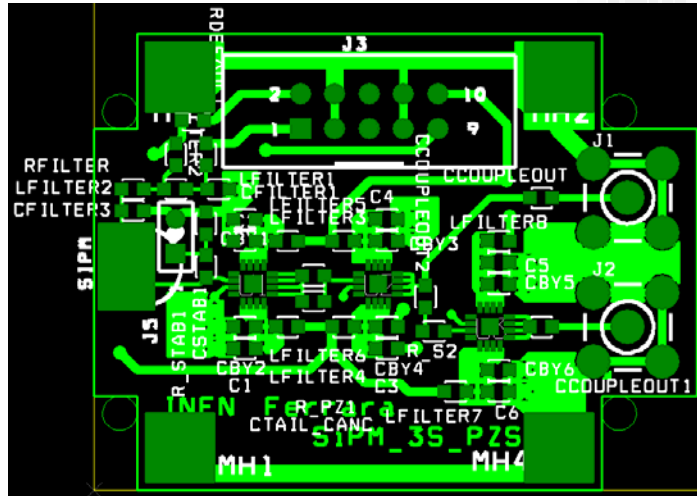
Current to Voltage
converter

Voltage amplifier
≈ 10x

Schematic diagram of the SiPM preamplifier being used in the test setup for detector R&D in Ferrara.

It is based on a THS4303RGTT amplifier with 18GHz gain bandwidth product.

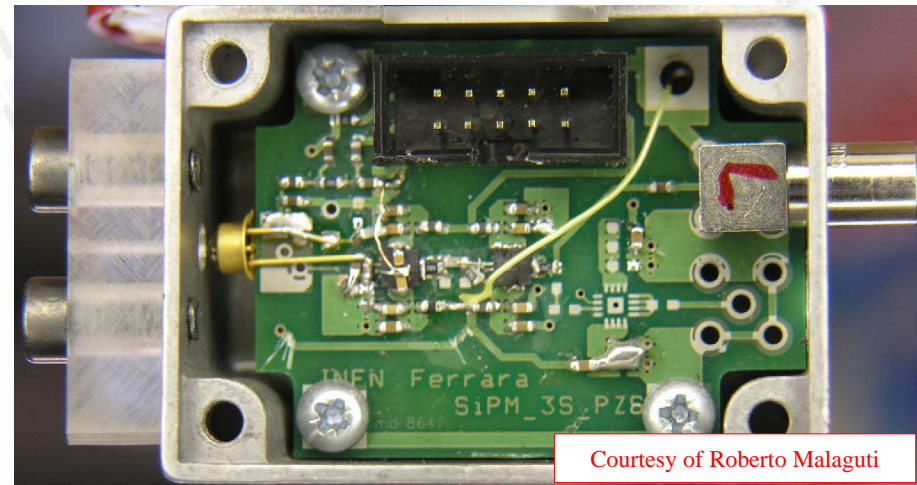
1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara



PCB design by Roberto Malaguti, INFN Ferrara.

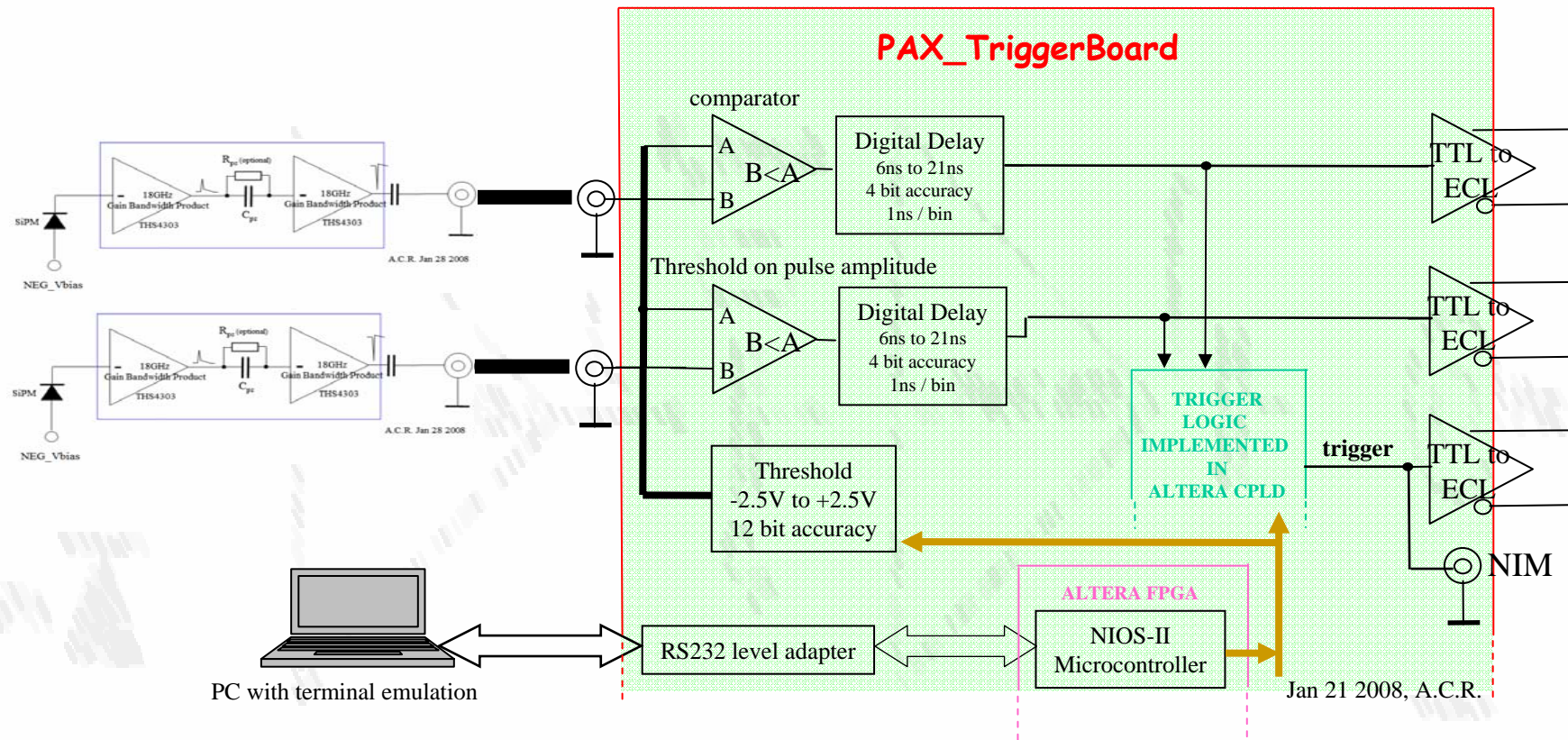
“... no battle plans survives the contact with the enemy...”

Due to the high gain-bandwidth product of the THS4303 a few modifications to the signal and power paths have been found necessary to avoid instabilities of the assembled circuit.



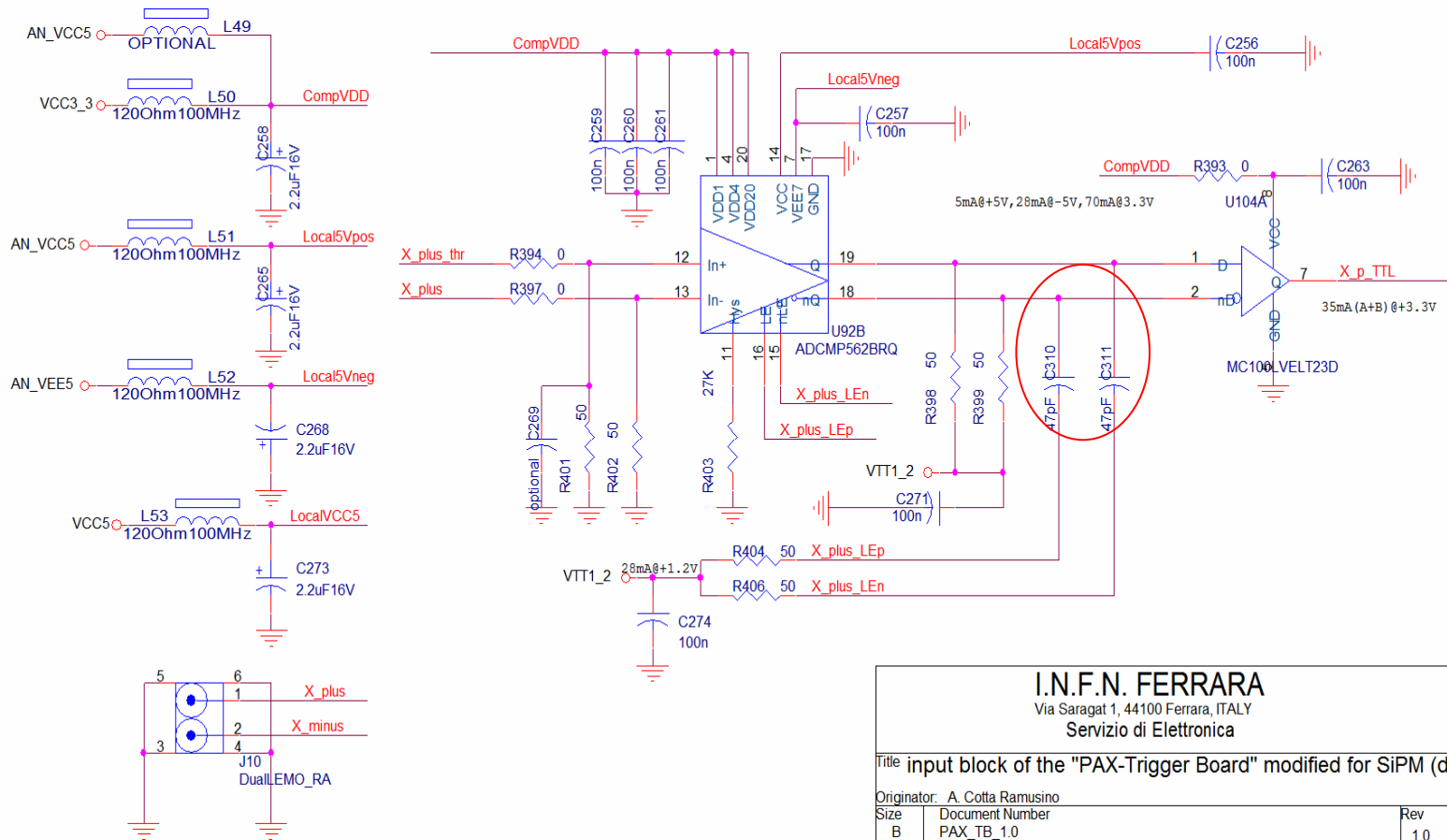
Courtesy of Roberto Malaguti

1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara



The setup for detector R&D in Ferrara is currently using a board, originally prepared for the PAX experiment, to discriminate the SiPM signals and generate a coincidence trigger. The threshold to the fast comparators and the look-up-table for the trigger generation are programmable.

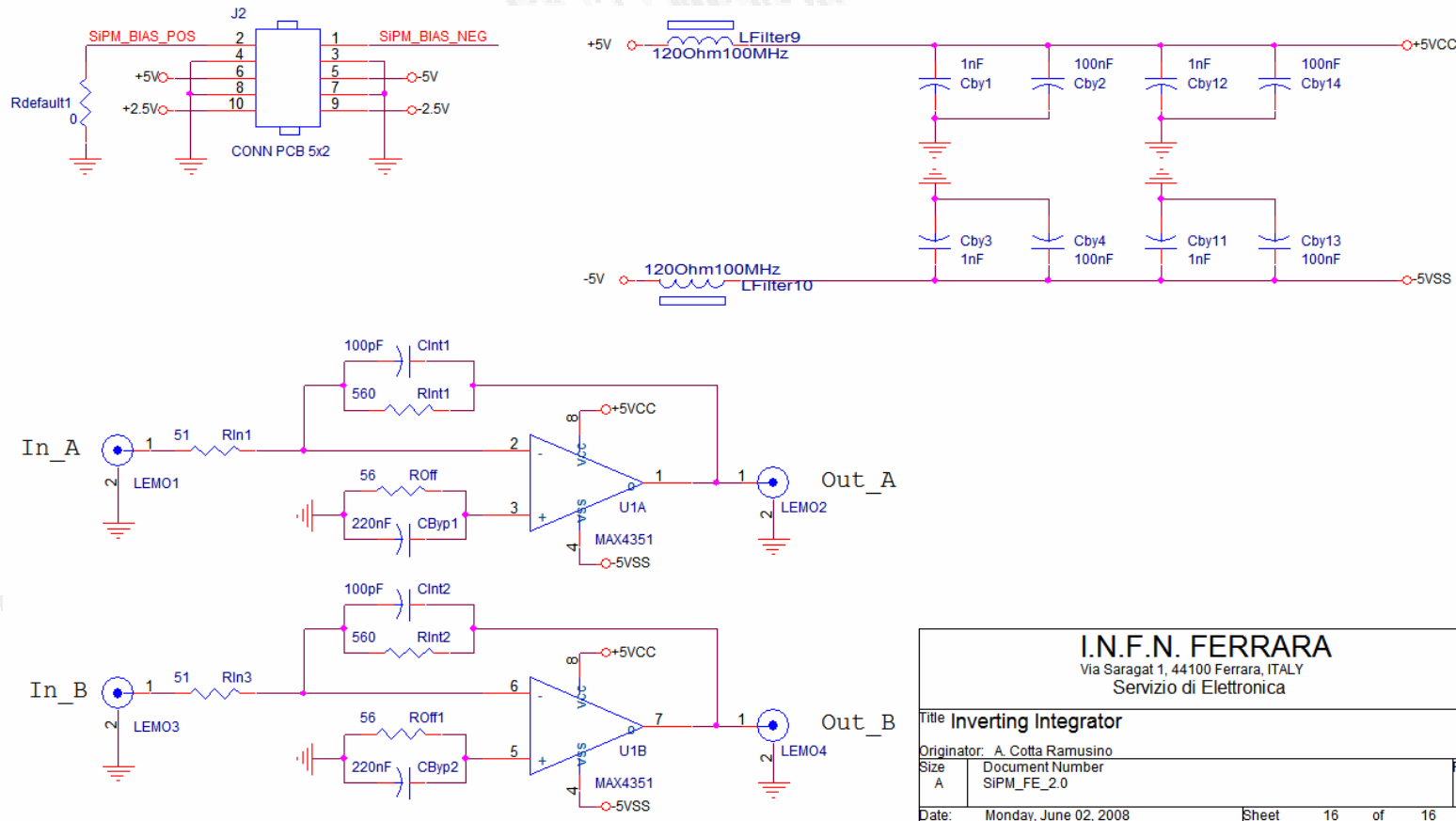
1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara



A few modifications were needed to adapt the “PAX_trigger_board” to the detector R&D application. The discriminator stage, in particular, had to be modified to stretch the fast comparator output to a minimum width.

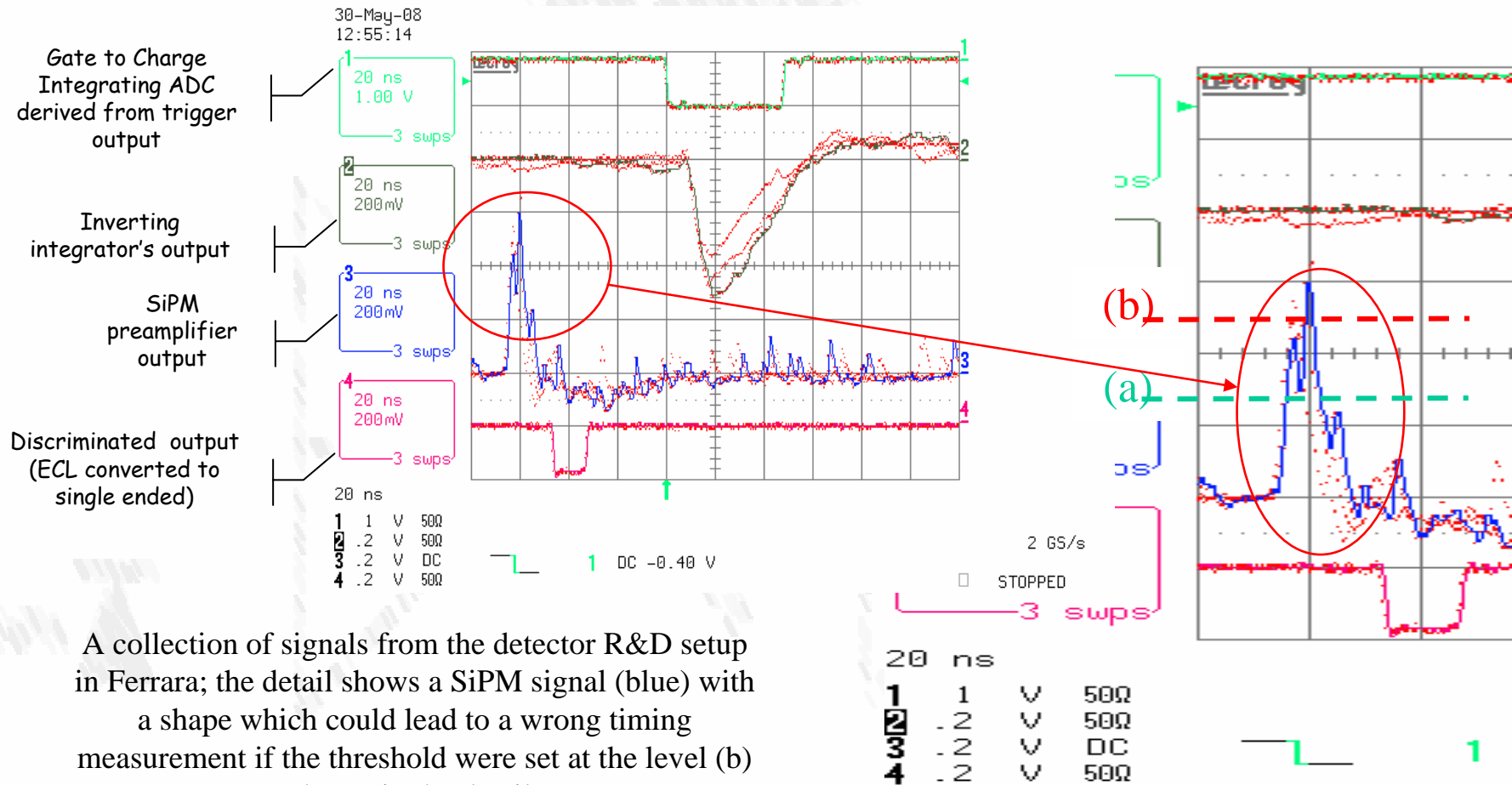
A 15ns width, determined by coupling capacitors of 47pF from the comparator outputs to the comparator “latch enable” inputs, was chosen.

1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara



Then a simple inverting integrator stage was prepared to provide a charge output to the charge integrating ADC used in the DAQ setup at INFN –Ferrara

1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara



A collection of signals from the detector R&D setup in Ferrara; the detail shows a SiPM signal (blue) with a shape which could lead to a wrong timing measurement if the threshold were set at the level (b) shown in the detail.

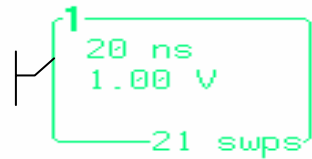
The proper timing would be given by a threshold set at level (a).

Registering the threshold crossing times at different thresholds (at a cost of an increased number of TDC channels needed) would allow one to assign the proper time measurement to the event

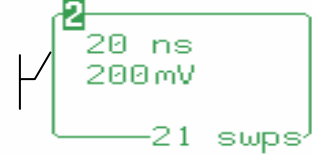
1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara

30-May-08
13:00:00

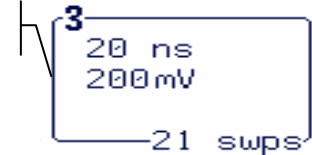
Gate to Charge
Integrating ADC
derived from trigger
output



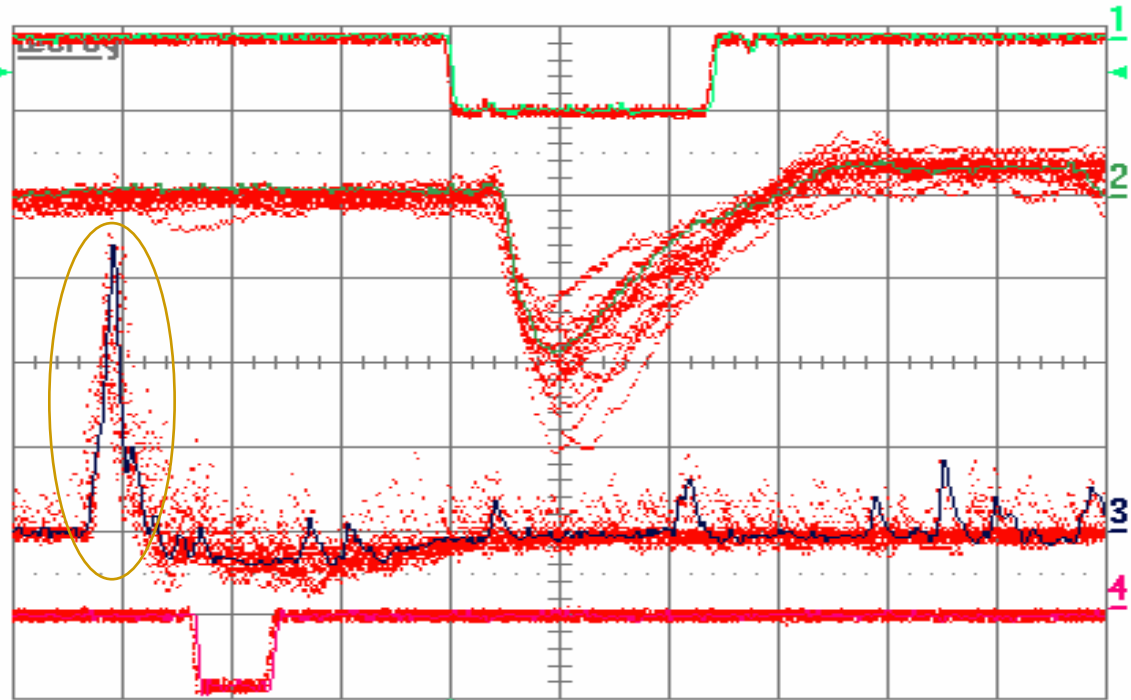
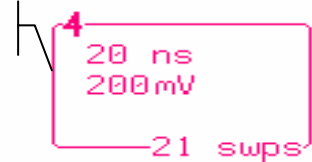
Inverting
integrator's output



SiPM
preamplifier
output



Discriminated output
(ECL converted to
single ended)



20 ns

1	1	V	50Ω
2	.2	V	50Ω
3	.2	V	DC
4	.2	V	50Ω

1 DC -0.40 V

Further investigation is necessary to find some type of “on-board” (as opposed to off-line) signal processing which could spare us from to take timing measurements at different thresholds on the same SiPM signal.

For now let's assume that each SiPM signal is processed by 2 comparators+TDC channels (with different thresholds)

1) prototype of SiPM signal processing channel being used in detector R&D in Ferrara

Data Acquisition System used in Ferrara for detector R&D was setup
(hardware and on-line software)

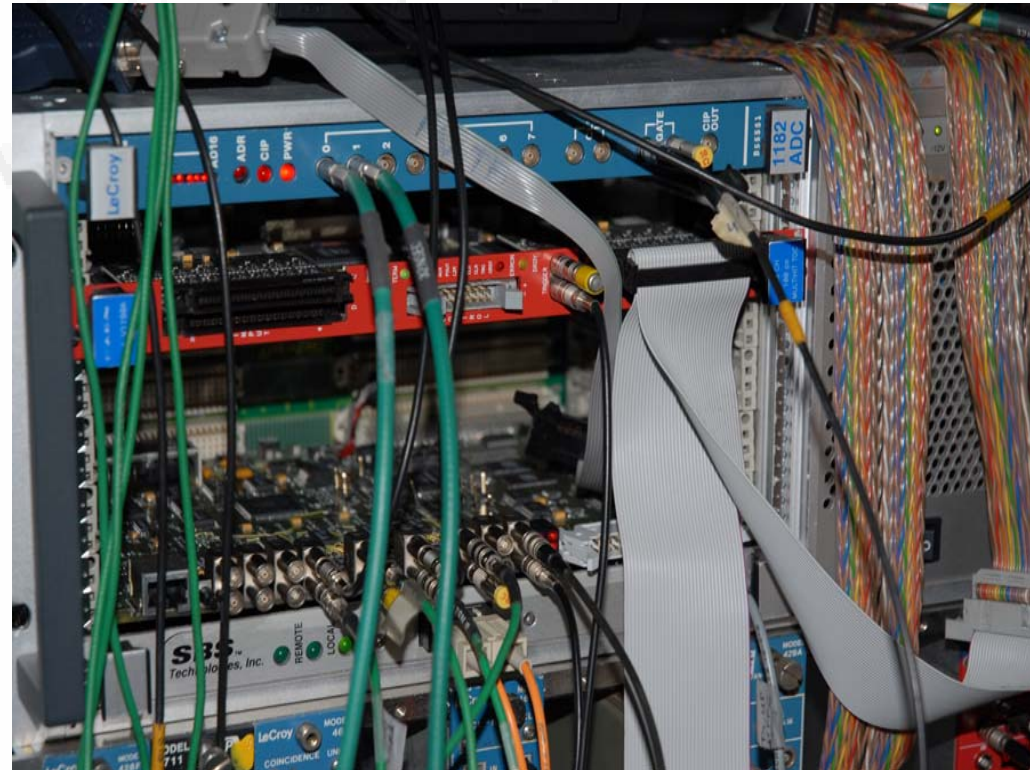
by

Stefano Chiozzi, INFN – Ferrara

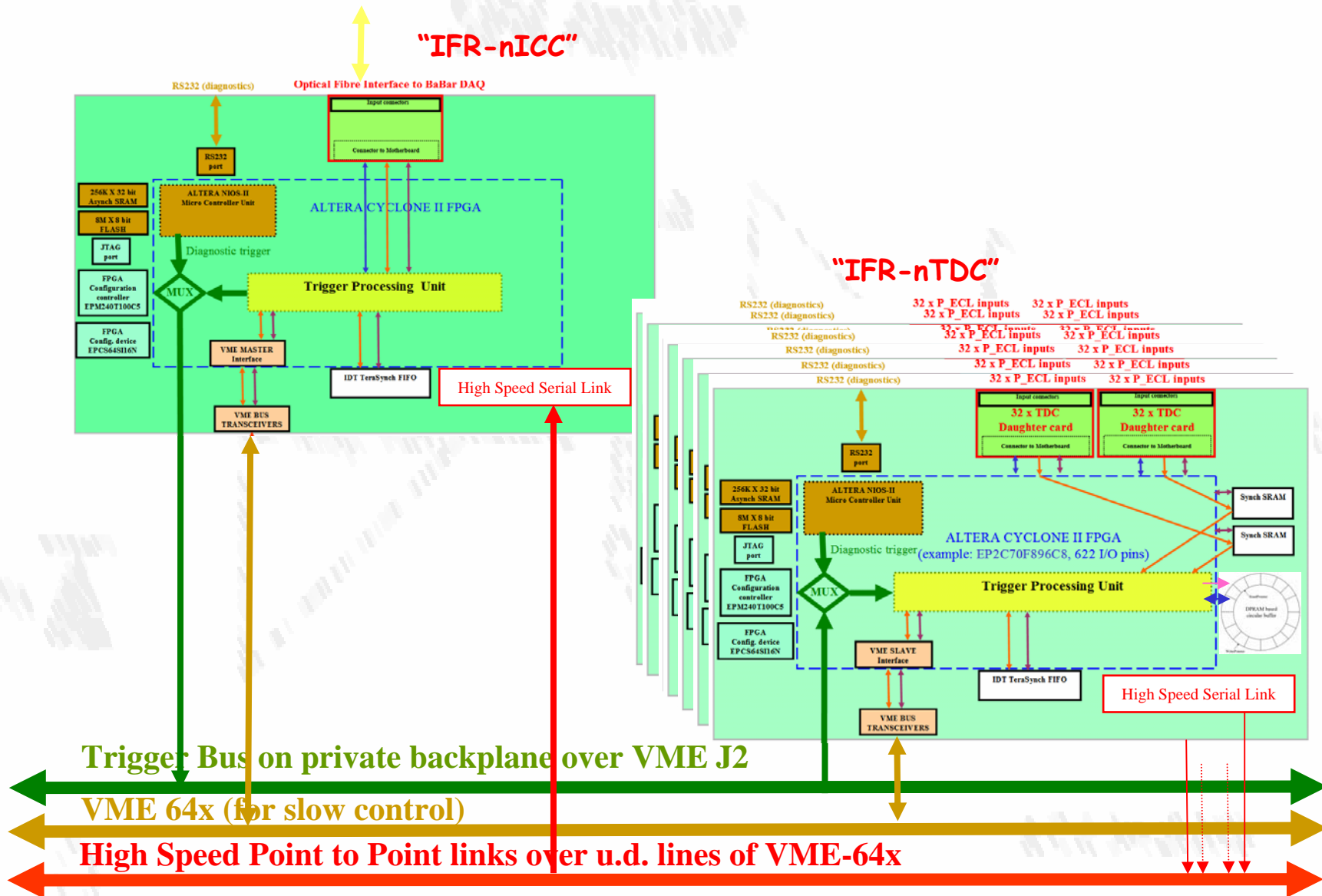
and it is based on a VME crate connected to a host PC by means of a VME–PCI bridge (on optical fiber) SBS model 618

The crate hosts:

- a CAEN V1190A TDC, 100ps resolution, 128 ECL inputs
- a LeCroy 1182 8 input, 12 bit, 50fC resolution charge integrating ADC
- the “PAX_trigger board” (only for powering it)



Results from data taking are shown in specific talks by Gigi Cibinetto and Wander Baldini, INFN-Ferrara.



Preliminary outline of the Front End Electronics for the SuperB IFR

2) basic features of Front End electronics for the SuperB IFR

Features (preliminary) of n(ew)On-Detector Electronics (nODE) for SuperB's IFR

According to latest estimates I got from R.Calabrese, W.Baldini, G.Cibinetto :

- Number of detector channels: ≈ 20.000 (need to read fibers at both end of sensitive elements)
 - ≈ 20 front-end crates with ≈ 16 boards/crate, assuming 64 inputs/board
 - 12 more DLINKs from 12 more "IFR-nICC", with respect to present IFR
- "Hit" rate : 500kHz/channel, in the hottest region, arising from:
 - particle rate : $O(100\text{Hz}) / \text{cm}^2$ (including background)
 - dimensions of a detector element : $< 400\text{cm} \times 4\text{cm}$ (thickness 20mm)

According to the latest results from detector tests in Ferrara:

(R.Calabrese, W.Baldini, G.Cibinetto, S. Chiozzi, A.Cotta Ramusino, R.Malaguti)

it seems likely that we will need to take at least 2 time measurements per channel (crossing of a lower and an upper threshold). Chances are that we might even need 4 threshold crossing measurements, to better reconstruct the sensor signal shape and improve the timing of the event

On the other end, if a reliable algorithm can be implemented in FPGA to identify, from the 2-4 measurements, the time of arrival of the first photons of the event, we could send to DAQ just one timing measurement per channel:

IT WOULDN'T SAVE TDC CHANNELS BUT AT LEAST IT WOULD SAVE BANDWIDTH

2) basic features of Front End electronics for the SuperB IFR

Let's assume an L1-accept extraction window (trigger jitter) of 1us.

I get (from estimated rates): probability of no hits in 1us $\approx 60\%$

→ detector occupancy is about 40% (mainly due to background)

Note: $P(1\text{hit}/1\text{us}) \approx 30\%$, $P(2\text{hit}/1\text{us}) \approx 7\%$, $P(3\text{hit}/1\text{us}) \approx 1\%$

The resulting \langle event size \rangle for each IFR Front End Board would be :

16 byte for header and trailer + 4 byte * 64 * 0.4 = 118 $\approx 2^7$ byte

→ \langle event size $\rangle \approx 2^7$ byte if 2 measurements per channel are sent to DAQ

The new IFR Crate Controller Card ("IFR-nICC") will have to collect and send an average of 2^7 byte * 16 * 8 = 2^{14} bit per event over the D-LINK

→ @1.2Gbps this will take: $\approx 16192 * 5/4 * 0.83\text{ns} = 16.86\text{us}$ 😞

→ @2.5Gbps this will take: $\approx 16192 * 5/4 * 0.4\text{ns} = 8.096\text{us}$ 😊

We do need a faster D-link to comply with a 100kHz \langle trigger rate \rangle

2) basic features of Front End electronics for the SuperB IFR

The new IFR Crate Controller Card ("IFR-nICC") will have to collect:
an average of $2^7 \text{ byte} * 16 = 2^{11} \text{ byte}$ per event in $\approx 8 \mu\text{s}$ →
256MB/s throughput for the crate backplane

TO ACHIEVE such a bandwidth is not trivial with just one VME controller in the crate even if using 2e-sst block transfers to read from each VME slave module.

One solution could be to use **a custom backplane** (could be a VME-64x backplane with an add-on) defining private **"Point to Point" differential connections** for high speed links between the new TDC Front End cards ("IFR-nTDC") and the "IFR-nICC": the backplane would also feature a standard VME bus for house-keeping tasks (configuration, debug etc.)

For the front-end card we need to design a custom module based on either the ACAM "TDC-F1" or the CERN "HP-TDC" ASICs, both with trigger matching units.

(The maximum latency allowed by the TDC-F1 is smaller than present BaBar L1 accept latency, but this is not an issue if the "IFR-nTDC" cards are designed to comply with option 2): "Variable latency with addressing by time, and queuing of triggers" described in the document

"FCTS protocol and related DAQ issues" Gregory Dubois-Felsmann & Steffen Luitz, 24 April 2008)



ELSEVIER

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Nuclear Instruments and Methods in Physics Research A 518 (2004) 493–494

**NUCLEAR
INSTRUMENTS
& METHODS
IN PHYSICS
RESEARCH**
Section A

www.elsevier.com/locate/nima

A new drift chamber TDC readout for the high intensity program of the NA48 experiment

R. Arcidiacono^{a,1}, N. Cartiglia^{a,b}, S. Chiozzi^c, M. Clemencic^{a,b}, A. Cotta Ramusino^c, C. Damiani^{c,d,*}, A. Gianoli^c, L. Milano^{c,d}, R. Malaguti^c, F. Petrucci^{c,d}, M. Scarpa^{c,d}

^a *INFN Torino, via P. Giuria 1, 10125 Torino, Italy*

^b *Department of Physics, University of Torino, via P. Giuria 1, 10125 Torino, Italy*

^c *INFN Ferrara, via Paradiso 12, 44100 Ferrara, Italy*

^d *Department of Physics, University of Ferrara, via Paradiso 12, 44100 Ferrara, Italy*

Abstract

A new read-out for the drift chambers (DCH) (8192 channels) of the NA48 experiment at CERN has been developed and realized by the Ferrara and Torino INFN sites and has taken data during the 2002 run. The core of the system is a set of 32 VME-9U Time-to-Digital-Converter boards (NA48-TDC). The NA48-TDCs record the time of arrival of signals from the DCH and store them in 40 MHz pipelined ring memories pending the trigger supervisor's decision. Dual memories and data extraction resources allow independent and simultaneous processing of level-1 and level-2 trigger requests. Time measurements are performed by the TDC-F1 commercial ASICs, having an intrinsic time resolution of 120 ps and multi-hit capabilities. The NA48-TDC board features a maximum sustained rate of 500 kHz per channel.

© 2003 Elsevier B.V. All rights reserved.

2) basic features of Front End electronics for the SuperB IFR

On page 7 of the referenced document two alternatives are proposed for achieving “deadtime-less” L1 accept processing:

<< Basic requirement is no intrinsic per-L1Accept deadtime, and deadtime due to “full” designed to be at most ~1% at the nominal 100 kTps trigger rate.

This can be achieved in two ways...

1- BaBar-like fixed-latency model

Works (roughly) if it is possible to deliver L1Accepts at a minimum spacing equal to the shortest time interval by which the (assumed fully pipelined) trigger can distinguish consecutive events.

Essentially this means that there can't be a meaningful limit on the minimum command spacing (100 ns ~ 1% deadtime)

You don't have to be able to do this for an unlimited number of events, of course, just for bursts long enough that statistically you don't get significant deadtime due to “full”. We will do some modeling of this.

In almost any scenario this does require being able to handle overlapping readout windows.

2- Variable latency with addressing by time, and queuing of triggers

In general requires additional pre-L1Accept “ring buffer” space, as closely-spaced triggers will effectively be delayed in transit.

>>

Feedback: the “IFR-nTDC” could be designed to operate in either way. In a previous experience (of myself and colleagues of the electronic shop in INFN-Ferrara) in designing a TDC card we complied to a requirement similar to the one proposed at point (2) and therefore we have no objection to it.

2) basic features of Front End electronics for the SuperB IFR

Quoting page 8 on the subject of buffering:

<< We assume two levels of buffering in the FEEs, in both models:

- A continuously-running ring buffer upstream of the L1Accepts, long enough for the maximum trigger latency, in either model.
 - In Model 1, its length can be essential equal to the trigger latency (plus some constant offset)
 - In Model 2, it needs to be longer than this by enough to handle 99% of anticipated trigger bursts. We need to do modeling to be quantitative about this, but we anticipate that the answer will be $O(10\mu s)$ of additional capacity (i.e., roughly a doubling).
- A post-L1Accept buffer. This would likely be constructed as a number of fixed-size slots as in BaBar's design. The number of L1Accept slots required needs to be determined by modeling, but is very likely to be substantially more than the four in the BaBar design.
 - The driving parameters are that events must be able to be acquired about ten times faster than in BaBar, but the actual readout probably cannot be comparably faster (link speeds are only 2x faster, and we probably cannot afford to have significantly more ROMs).
 - The amount of such buffering needed would be somewhat larger in Model 1, by an amount we'll need to determine by modeling.

>>

Feedback: I don't see any problem in accomodating pre-Level1Accept data in a dual port memory organized as a circular buffer: from the channel rate estimate we can expect, at the board level, roughly 26hits / μs . \rightarrow we must store ≈ 128 byte / μs

If we organize the circular buffer with pages of 256 byte (headroom included) to store all "hits" for 1 μs interval and we keep 32 such pages (32 μs worth of history) we end up with a size of 8kB for the dual port memory, which is reasonable.

No problem either when considering the post-L1Accept buffer. I would like to make this a simple FIFO, to hold the event data until the link to the "IFR-nICC" is free from eventually transmitting a previous event data block.

2) basic features of Front End electronics for the SuperB IFR

Quoting page 9 again on the subject of buffering-II:

<< Overlapping readout windows need to be supported

This has implications for the copy of event data from the ring buffer to the post-L1Accept buffer.

We propose that the protocol support copy by reference when windows overlap. This reduces the internal bandwidth required in the FEEs.

This requires the system to model an event as composed potentially of one or more by-reference segments followed by a by-value segment. At a minimum, the by-value segment of an event must be retained somewhere in the system until enough time has passed that a future event cannot need any part of it. This could be done either in the FEE or in the ROM, trading off complexity in the FEE against complexity in the FCTS protocol and the ROM.

We expect to propose one possible implementation as a reference point but describe alternatives as well.

>>

Feedback: I would propose to transfer the burden of referencing previously transmitted data to the "IFR-nICC" and to make it relative to only just the data extracted by the trigger immediately preceding.

The "IFR-nICC" receives 16 links from the "IFR-nTDC"s and buffers the data into internal buffers prior to merging them and sending the "crate-wide" packet over the D-LINK to the ROM.

During this process the "IFR-nICC" may copy the input "receive buffers" onto "input reference buffers" and determine, for each of the 16 sections concurrently, up to what point the data from previous event and data for current event do overlap.

The "IFR-nICC" would send the pointer to the last "hit" of the previous event which is also part of the "new" and then appends the "new" hit information.

This scheme works if "hit" information in the event packet are in strict temporal order, rather than, for instance, order for channel address.