

# *Initial thoughts on using Pulsar as an upgrade for L2*

**Ted Liu (Aug. 1<sup>st</sup>, Level 2 Review)**

→ Slides with red dot ● on right upper corner are backup slides

**Original motivations for Pulsar project (see Peter Wilson's talk):**

- (1) As teststand tool (data source and sink) for Level 2;**
- (2) if needed, as a prototype for Level 2 upgrade;**

- wasn't clear it was possible to have a design to achieve both (1) and (2),
  - not clear if we need an upgrade at all,
  - the focus has been on building the teststand tool
  - now have board design, and we are talking about Run2b upgrade
- **Is pulsar a candidate for L2 upgrade in run2b?**

**<http://hep.uchicago.edu/~thliu/projects/Pulsar/>**

## Parts Availability for Level 2 Processors

It's been Doom for every CPU  
that's ere been used in Level 2.

Both Mot & Dec were once employed;  
their product lines are now destroyed.

Will the axe next fall on Pentium, II?

→ **Bill Foster, 199? Front-End-Trigger Summary talk**

## Back to Basic:

What does Global L2 really do?



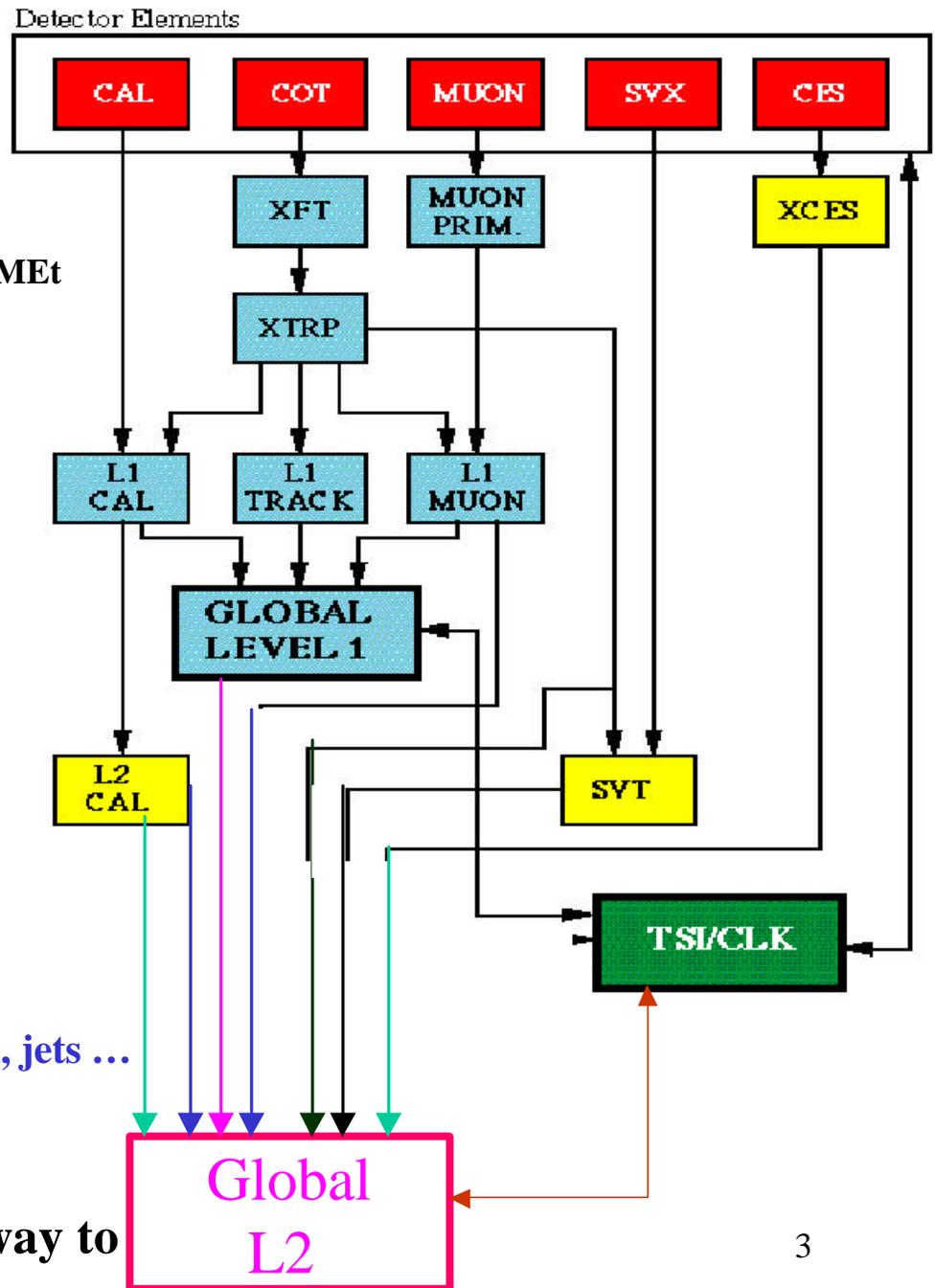
make decisions based on  
objects already found upstream

	L1	trk	svt	clist	Iso	reces	mu	SumEt,MEt
Tracks	●	●	●					
Jets	●	●	●	●				
electron	●	●	●	●	●	●		
photon	●			●	●	●		
muon	●	●	●				●	
Taus	●	●		●	●			
Met	●						●	●
SumEt	●						●	●
..								

physics object examples

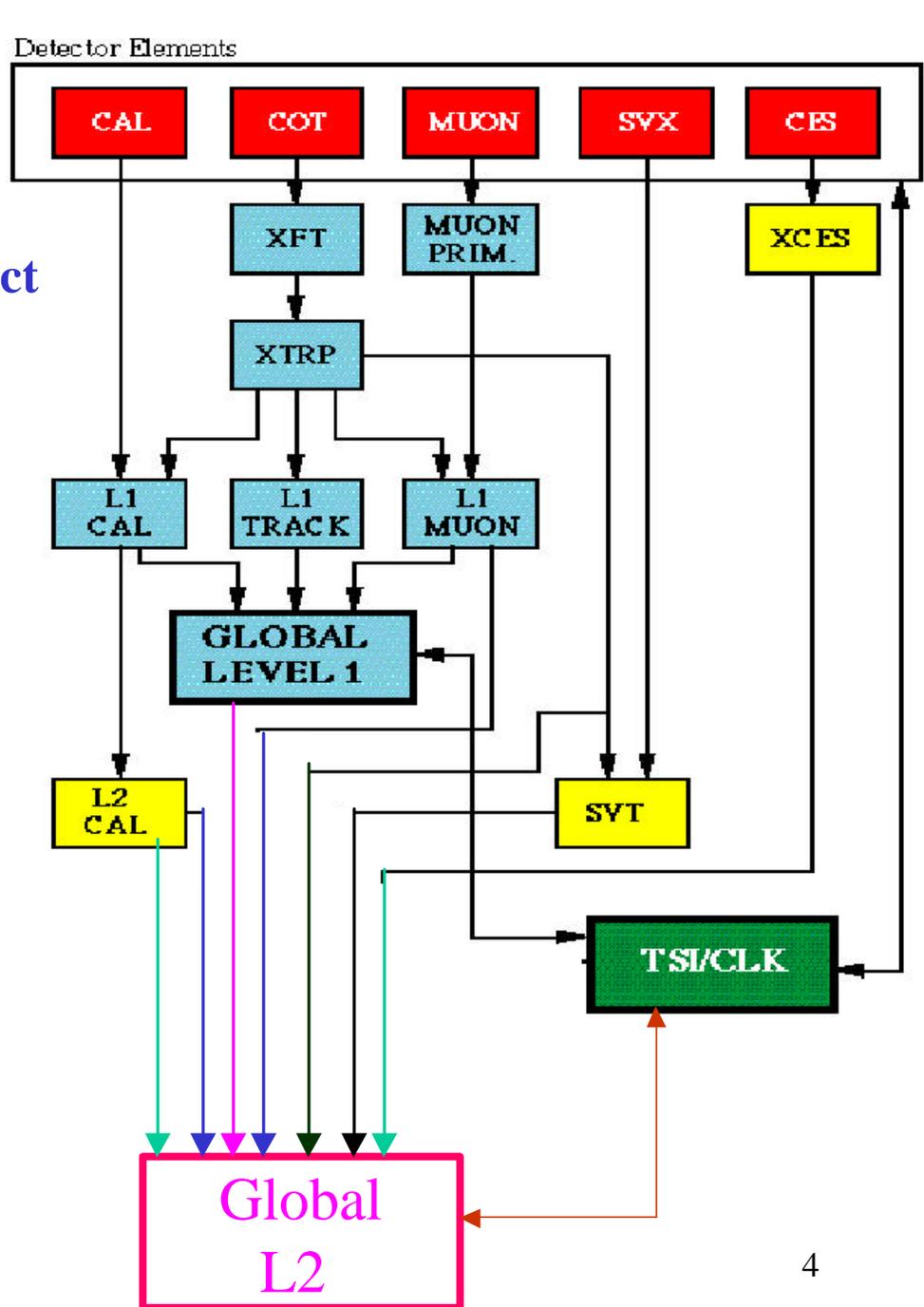
- Combines/matches trigger objects into e, muon, jets ...
- Count physics objects above thresholds, or,
- Cut on kinematics quantities

Technical requirement: need a **FAST** way to collect many data inputs...



Technical requirement: need a **FAST** way to collect many **data inputs**...

In addition, one has to deal with the fact that **each data input was implemented in a different way ...**



## Each L2 input data path was implemented differently

	SVT	XTRP	L1	CLIST	ISO	Muon	Reces
<b>Incoming data Clock rate</b>	30Mhz	7.6Mhz	7.6Mhz	20Mhz	12Mhz	30Mhz cdfclk x 4	7.6 Mhz cdfclk
<b>Interface hardware</b>	SVT cable	SVT cable	L1 cable	Hotlink+fiber	Taxi+fiber	Hotlink+fiber	Taxi+fiber
<b>data size range</b>	117bits/trk	21 bits/trk	96 bits/evt	46bits/clu	145bits/clu	~11Kbits/evt	1.5Kb/evt
<b>Latency range*</b>	~15--~50us	~1us - 10us	~132x2 ns	~a few us	~few us	~5 us	~ 6 us
<b>Fixed or variable data length?</b>	variable	variable	fixed	variable	variable	fixed	fixed
<b>Data with Buffer#?</b>	yes	yes	yes	yes	yes	no	yes
<b>EOE with data? (or from separate path?)</b>	yes	yes	-	no	no	yes	-
<b>B0 marker?</b>	BC#	BC#	no	no	no	yes	no
<b>Data gap within one event?</b>	yes	yes	no	no	yes	no	no
<b>Flow control ?</b>	Not used	not used	no	no	no	no	no <sub>5</sub>

\* Latency range also depends on L1A history, data size etc ...this is just a rough range

# Trigger Design Issues with Field-Programmable Gate Arrays

Optic Fibers and Digital Light-Links  
Permit Triggers with Digital Hi-Jinks.  
Since we have no conception  
approaching perfection  
we'll bury it all in a Xylinx.

→ Bill Foster, 199?, FE-Trigger summary talk

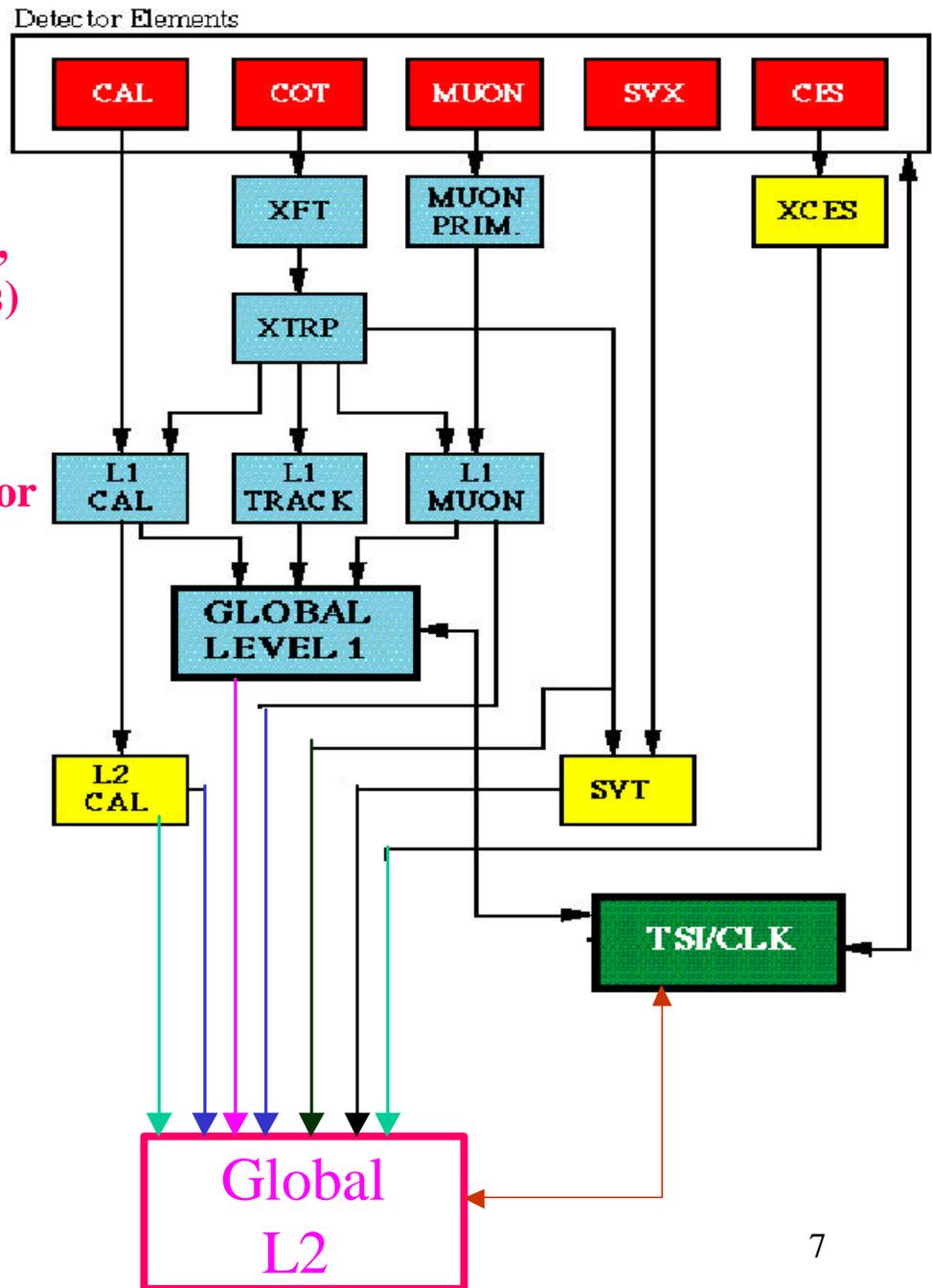
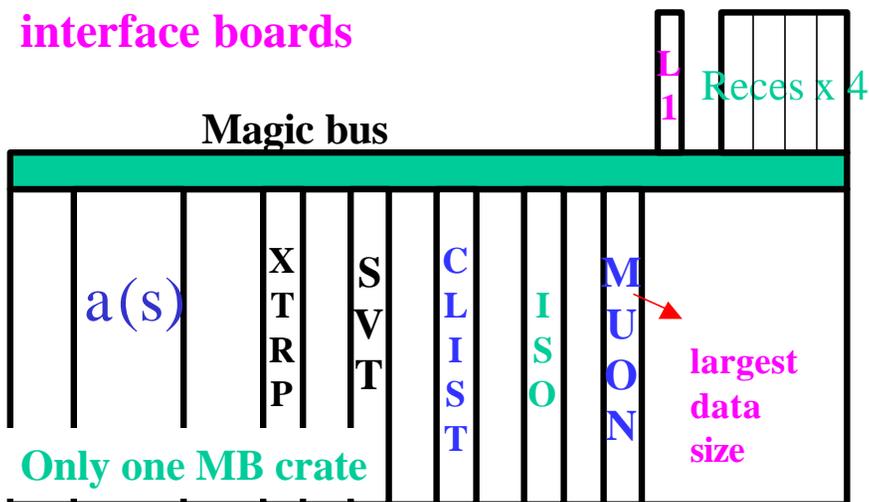
## Current L2 Global Crate:

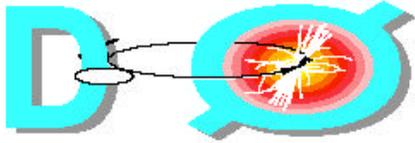
Technical requirement: need a **FAST** way to collect many **data inputs...**

→ With the technology available back then, had to design custom backplane (magicbus) and processor ...

→ data is pushed(DMA) over Mbus, some need to be pulled only when needed(PIO, for Reces and muon) to ease the bandwidth demands

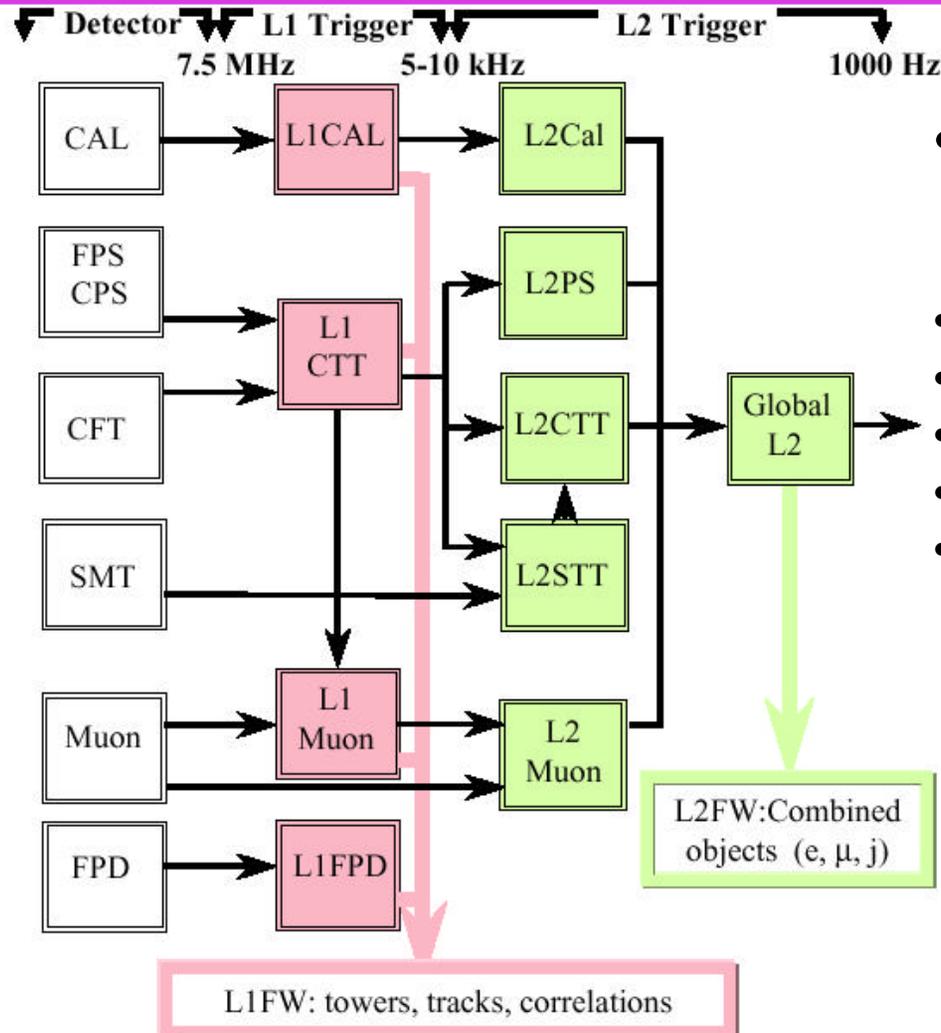
→ In addition, one has to deal with the fact that each data input was implemented in a different way ... → 6 different type of interface boards





D0 case is similar but different in many aspects:

# Trigger Organization



- system heavily depends on magicbus architecture: many magicbus crates
- L1A: 5-10KHz
- L2 time budget: 100 us
- L2A: 1KHz
- 16 FE buffers
- shorter magicbus

For Run2b:

CDF demands on L2 performance is much greater

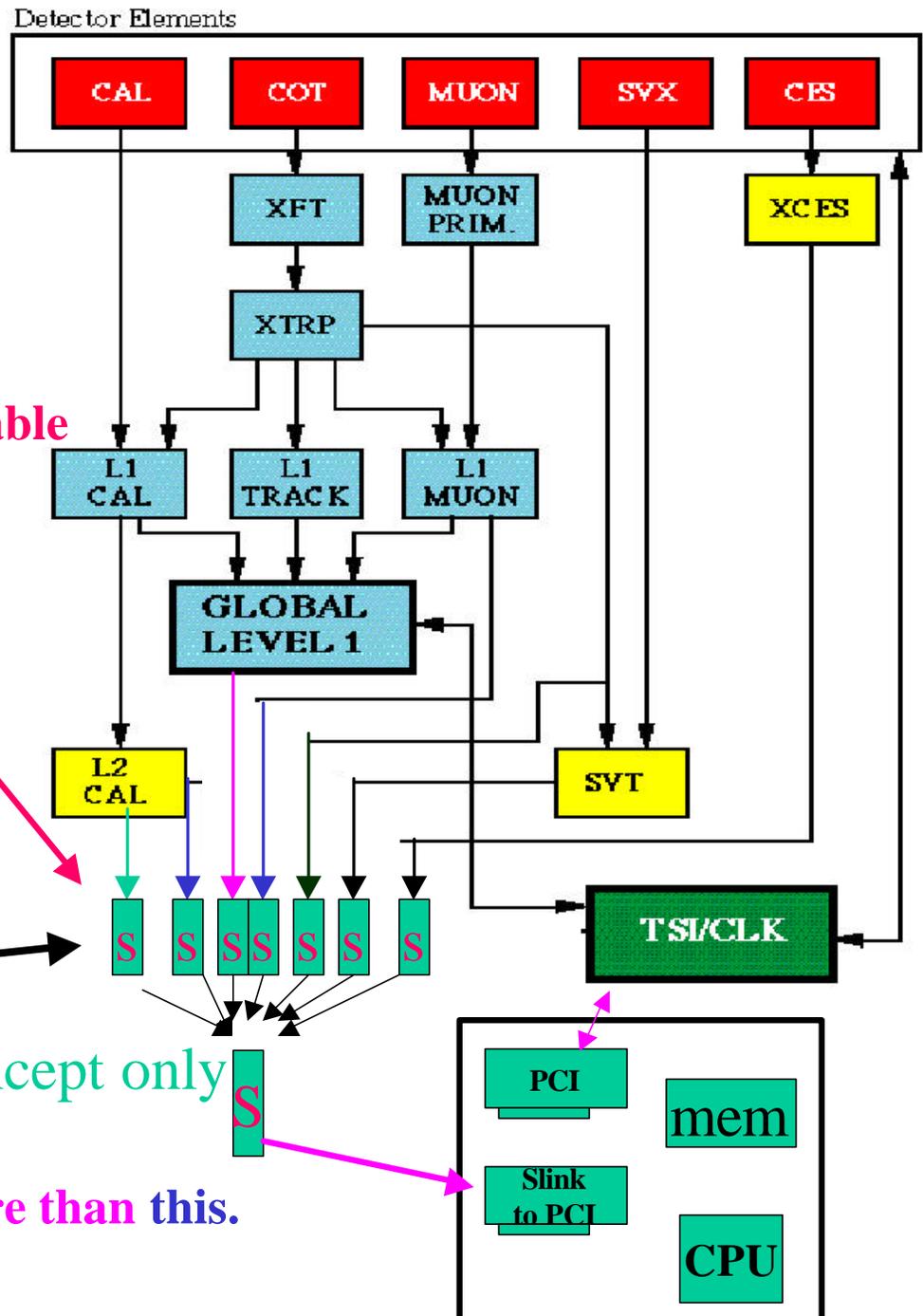
**Basic idea** on possible new approach using modern technology:

- **Convert/merge different data inputs into a standard link**
- **interface with commodity PC(s)**
- **high bandwidth, commercially available link to PCI interface cards:**

**CERN S-LINK** is designed just for this purpose for LHC and other experiments.

**Real question:** can we design an **universal** interface board (and the rest are all commercial products) ?

**Pulsar** is designed to be able to do **more than this**.





## CERN S-LINK homepage

S-LINK is a CERN specification for an easy-to-use FIFO-like test tools are all commercially available.

S-LINK is part of the [High Speed Interconnect](#) project at [CERN](#)

### S-LINK General Information

- [Overview](#)
- [News](#)
- [General information](#)
- [Specifications](#)

### S-LINK Products (now available from)

- [S-LINK Implementations](#)
- [General purpose interfaces \(hardware and software driven\)](#)
- [Testing devices and adapters](#)
- [Models](#)

### Projects using S-LINK

- [High Energy Physics](#)
  - [ATLAS](#)
  - [CDF](#)
  - [CMS](#)
  - [COMPASS](#)
  - [LHCb](#)
  - [NA48](#)
  - [NA60](#)
- [Nuclear Fusion](#)
  - [ASDEX Upgrade](#)
  - [TCV](#)
- [Astronomical Observation](#)

CERN

## S-LINK on the web [www.cern.ch/hsi/s-link](http://www.cern.ch/hsi/s-link)

### General information

- News, Specifications, Introduction

### Projects using S-LINK

- Experiments
- Institutes outside HEP

### S-LINK products

- FEMBs
- S-LINK Implementations
- ROMBs
- Testing devices
- Models



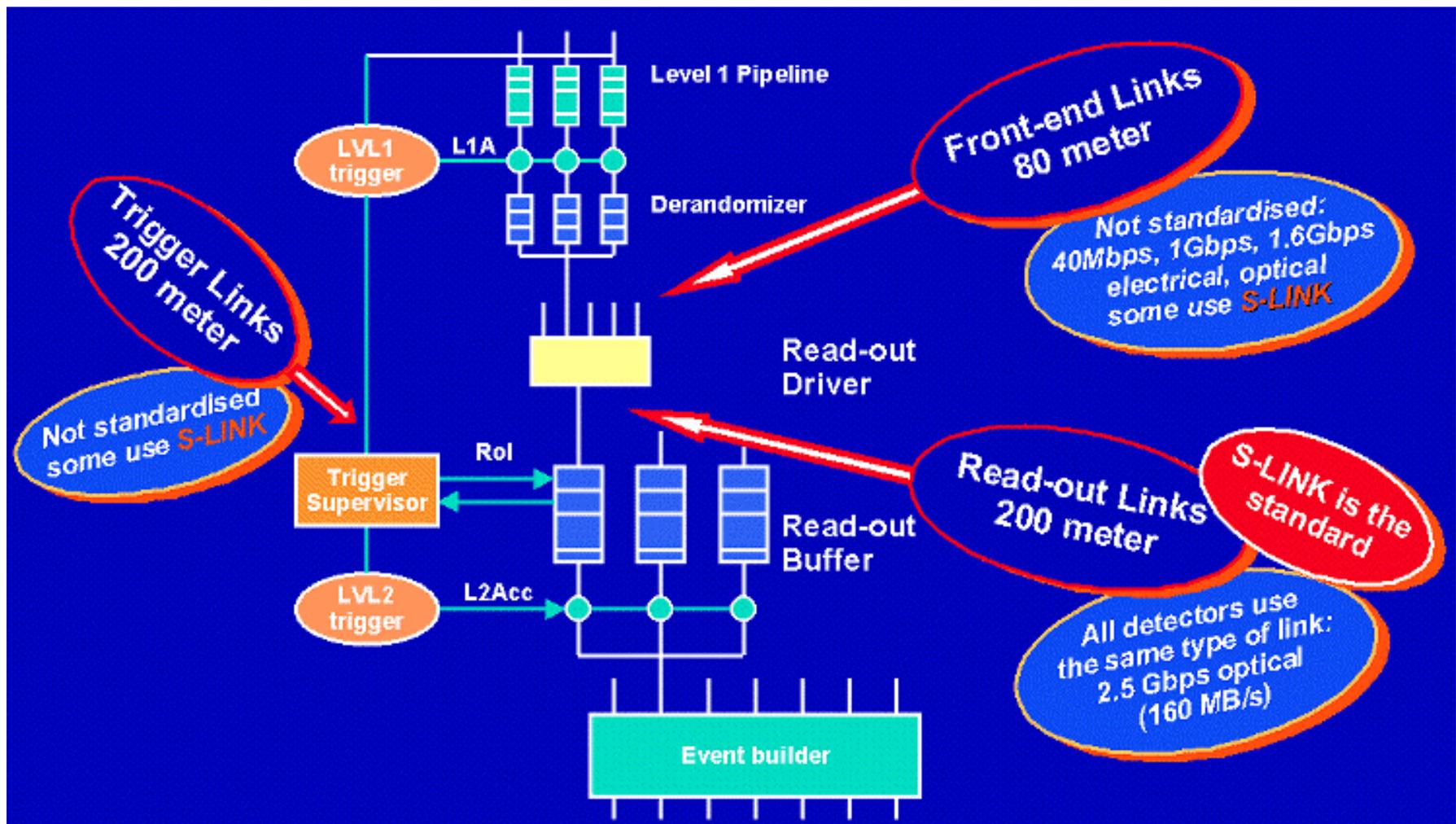
Erik van der Bij

division EP ATE/DQ



- Features not shown:
- Self test mode
  - Link down signalling
  - Return lines

ATLAS will use one standard link (S-LINK) to move the data from all Read-out Drivers (RODs) from the subdetectors from the experiment 100 meter down under the ground to the Read-out Buffers (ROBs) located in buildings on the surface. Around 1500 of those links are needed, each one moving data at 160 MByte/s.

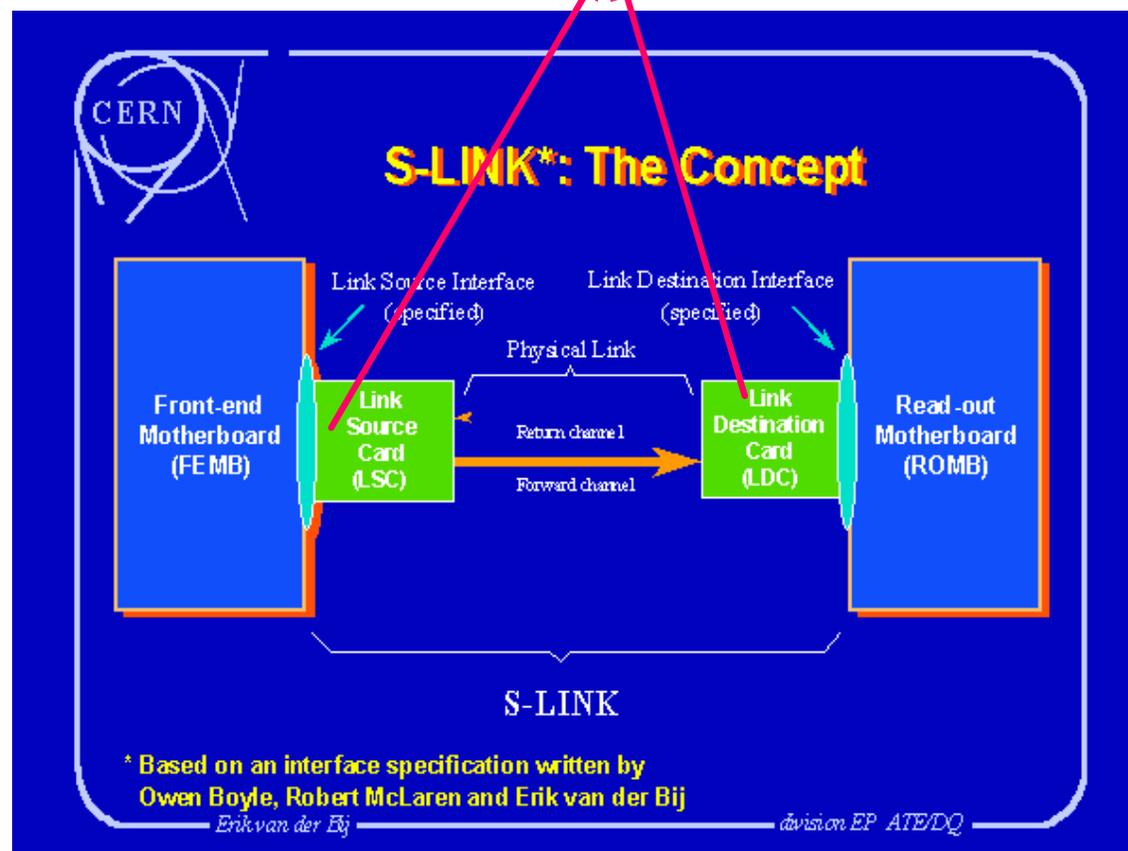
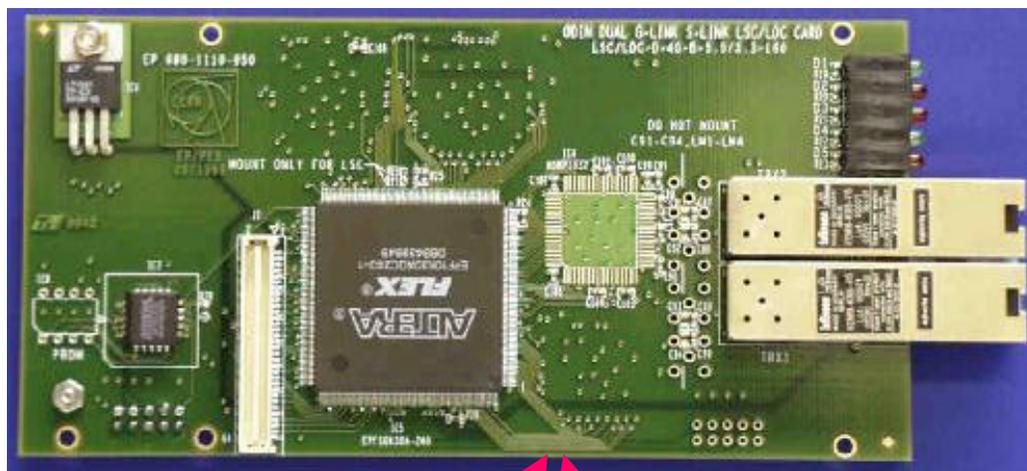


SLINK format example:

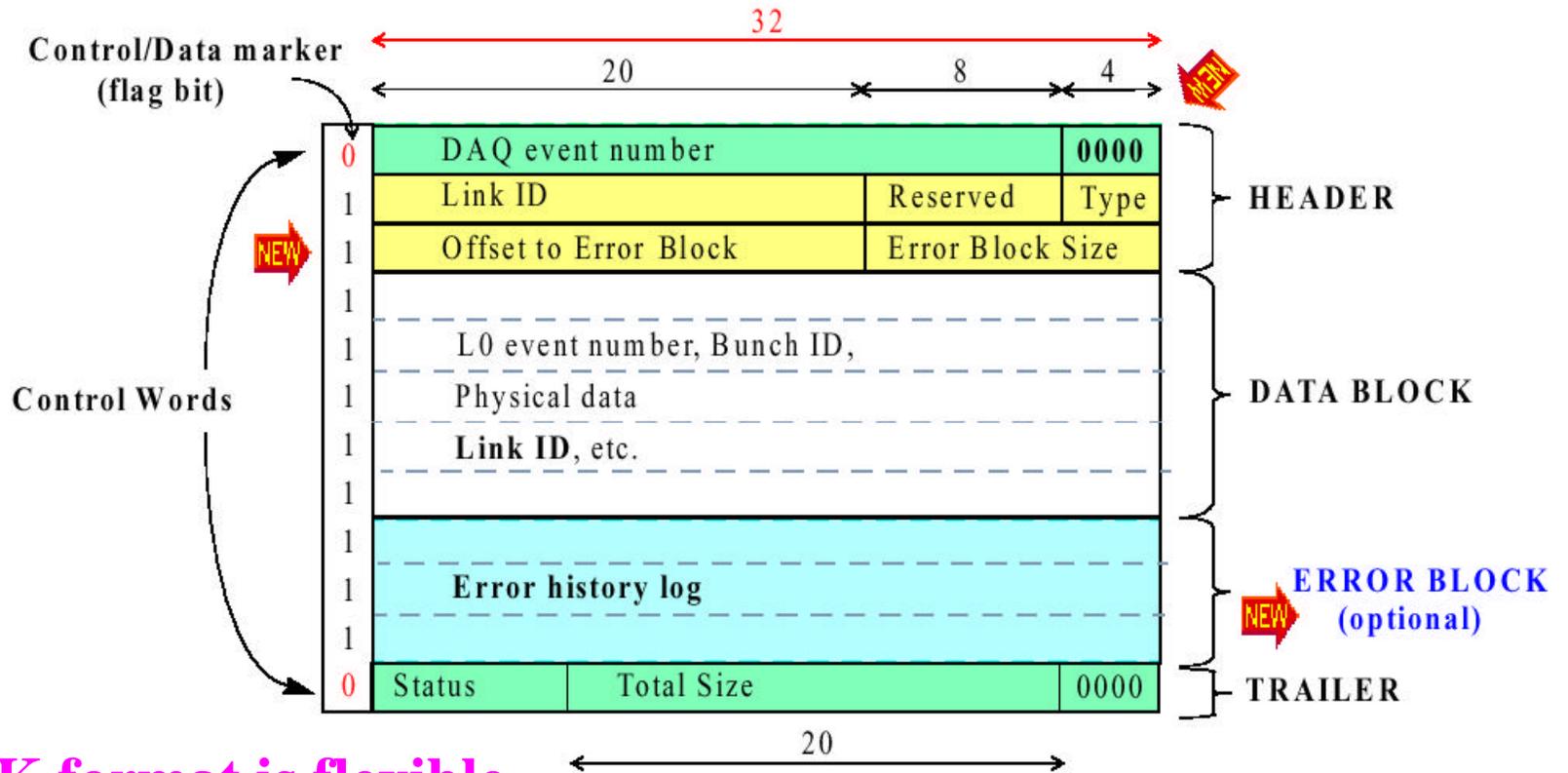
ATLAS SLINK data format

Beginning of Block control word
Start of Header Marker
Header Size
Format Version No.
Source Identifier
Level 1 ID
Bunch Crossing ID
Level 1 Trigger Type
Detector Event Type
Data or Status elements
Status or Data elements
Number of status elements
Number of data elements
Data/Status First Flag
End of Block control word

## SLINK interface mezzanine card



# Proposed frame formats (I)

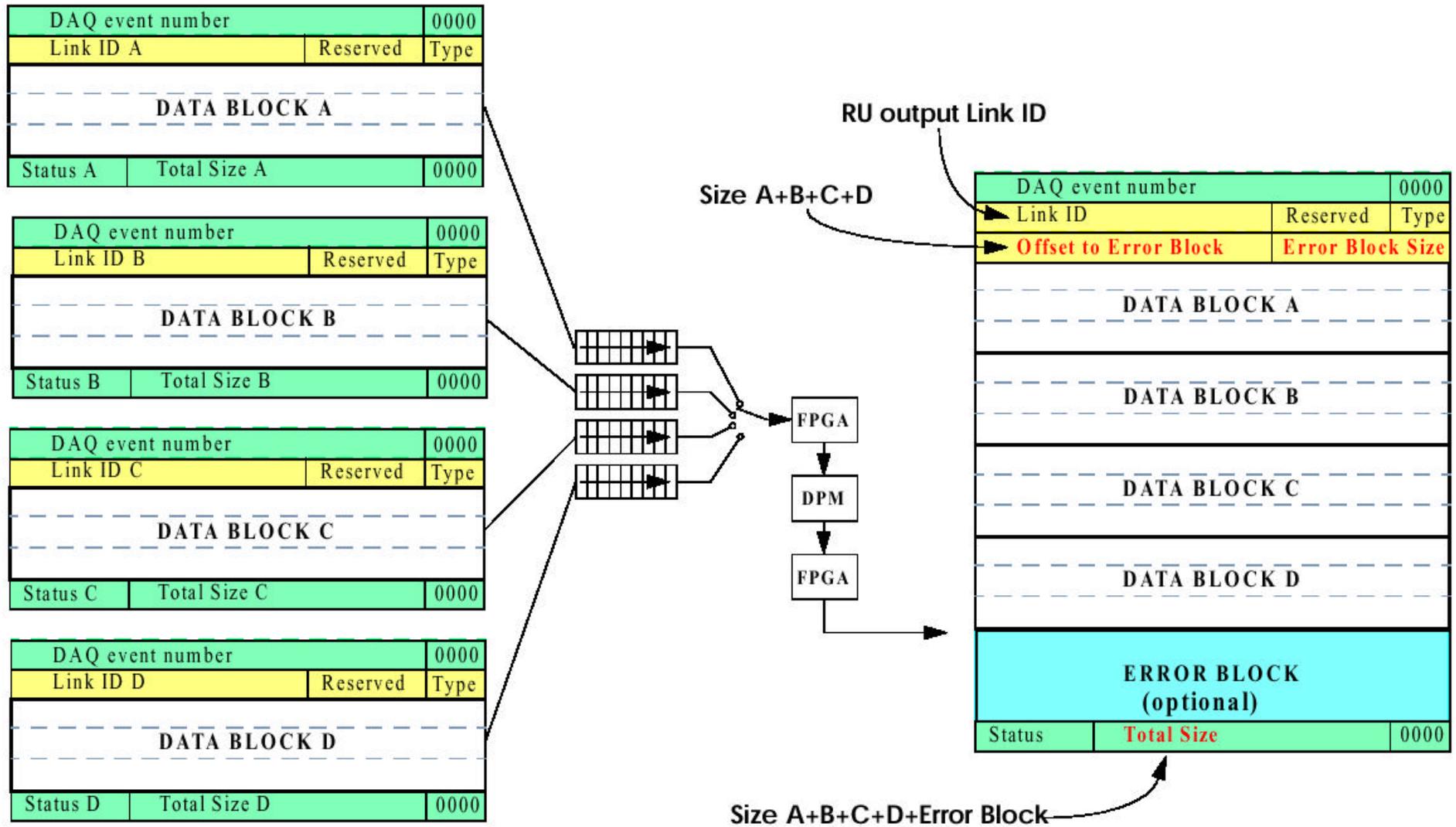


**SLINK format is flexible**

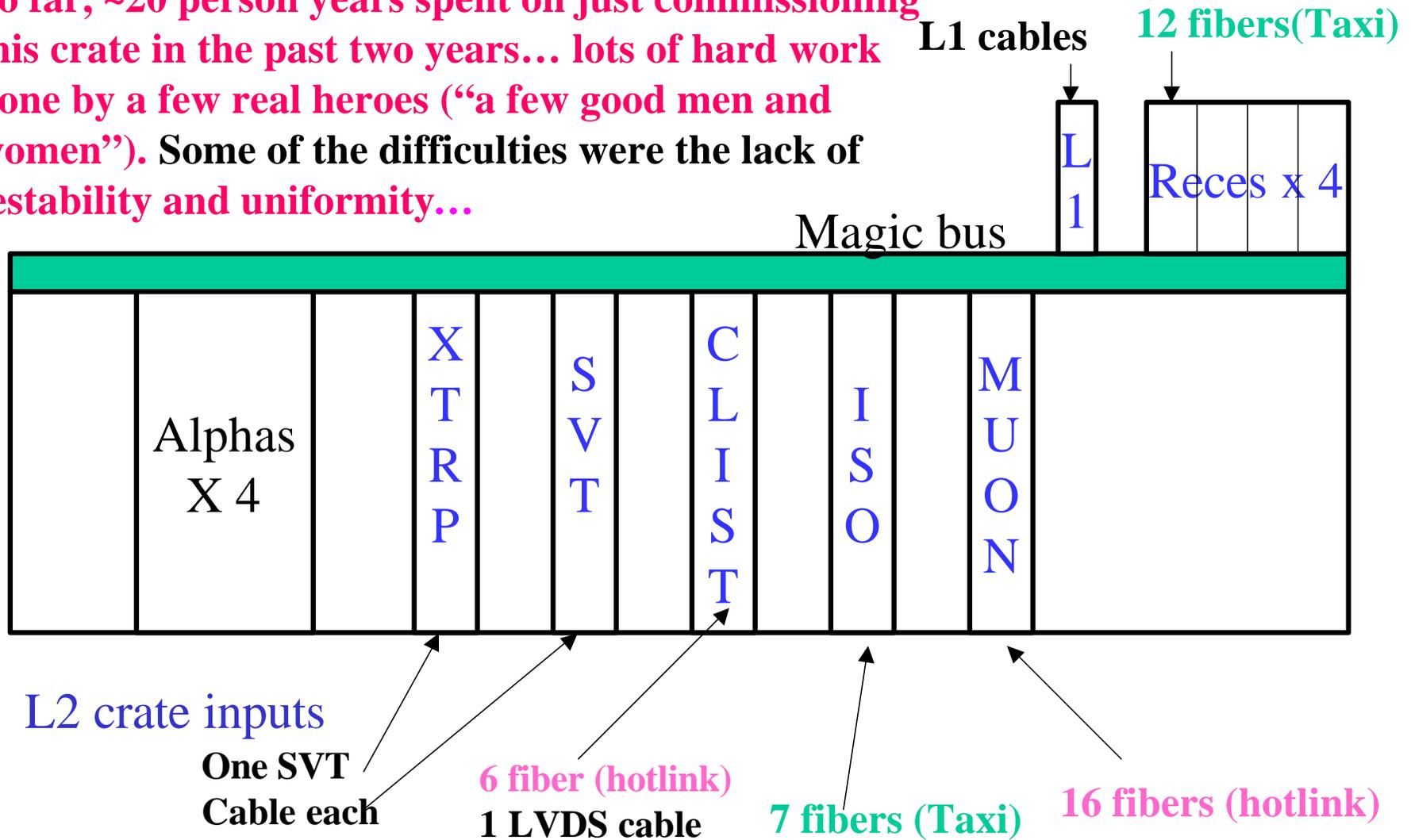
From talk by Jose Toledo (CERN/EP-ED)

<http://toledo.home.cern.ch/toledo/~toledo/daqws.pdf>

# Proposed frame formats (III)



So far, ~20 person years spent on just commissioning this crate in the past two years... lots of hard work done by a few real heroes (“a few good men and women”). Some of the difficulties were the lack of testability and uniformity...



Asked 3 questions a while ago (the answer is in Pulsar design):

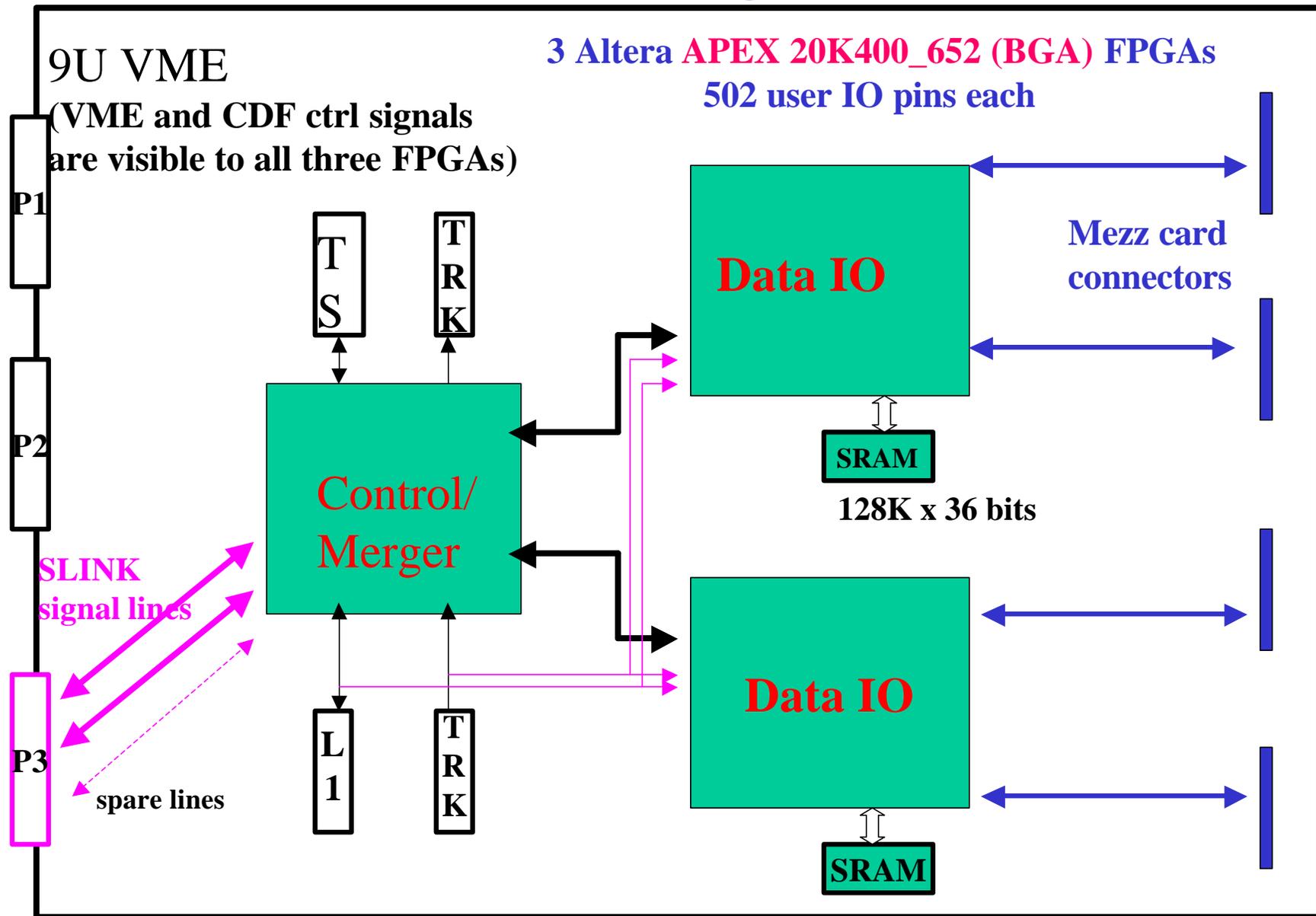
- (1) can one design an universal test (pulser)board? (testability)
- (2) can one design an universal interface board?(uniformity)
- (3) can the universal interface board be also a pre-processor?

## Commissioning Manpower Issues

For years we've promoted the talkers,  
the PAW-pounding wizards and gawkers.  
But the true worthy man,  
when the parts hit the fan,  
are glitch hunters and problem stalkers.

→ **Bill Foster, 199? FE-Trigger summary talk**

# Pulsar design



3 APEX20K400 FPGAs on board = 3 Million system gates/80KB RAM per board<sup>17</sup>

2 128K x 36 pipelined SRAMs with No Bus Latency: 1 MB SRAM (~5ns access time)

## APEX 20K devices: System-on-a-Programmable-Chip solutions

APEX 20K Programmable Logic Device Family Datasheet

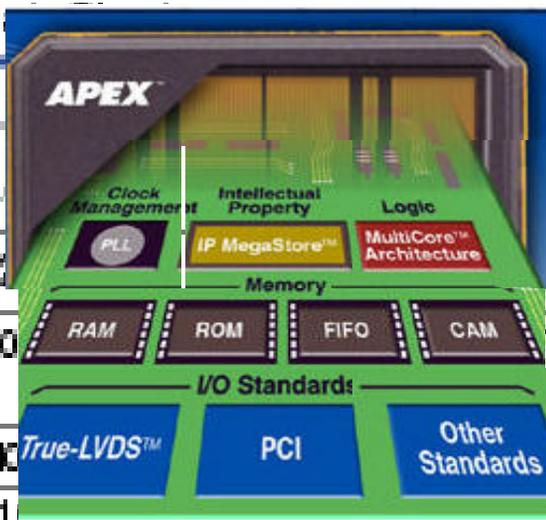


Table 2. APEX 20K Device Features Note (1)

Feature	EP20K300E	EP20K400	EP20K1000E	EP20K1000E	EP20K1500E
Maximum system gates	728,000	1,052,000	1,052,000	1,052,000	2,392,000
Typical gates	300,000	400,000	400,000	400,000	1,500,000
LEs	11,520	16,840	16,840	16,840	51,840
ESBs	72	104	104	152	216
Maximum RAM bits	147,456	212,992 (26KB)	212,992	311,296	442,368
Maximum macrocells	1,152	1,664	1,664	2,432	3,456
Maximum user I/O pins	408	502	488	598	808

Note to tables:

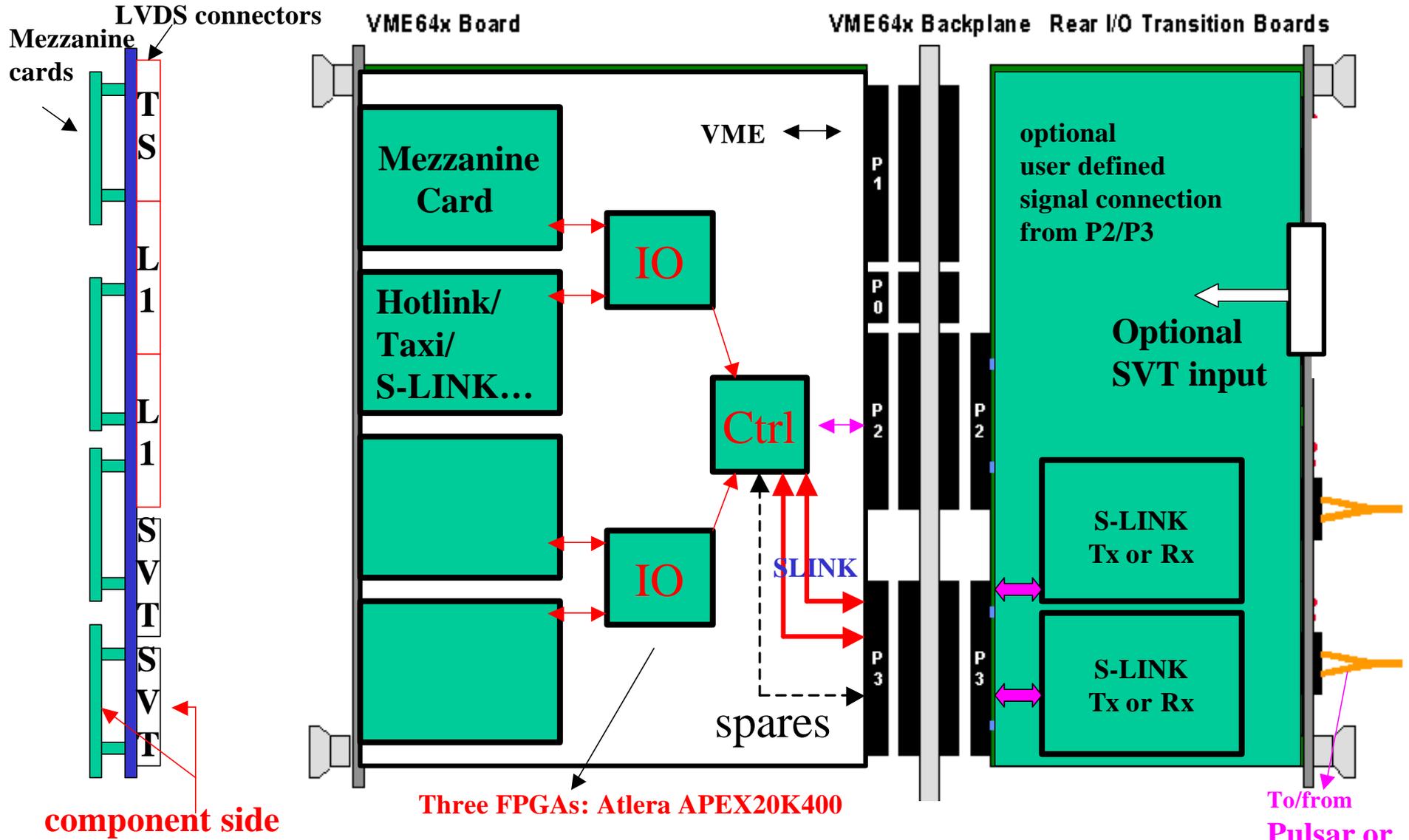
The cost of FPGA is roughly:

# 10 system gates per penny

Front-panel  
(double width)

PULSAR design

Each mezzanine card can have up to 4 (hotlink/Taxi) fiber channels



Three FPGAs: Atera APEX20K400

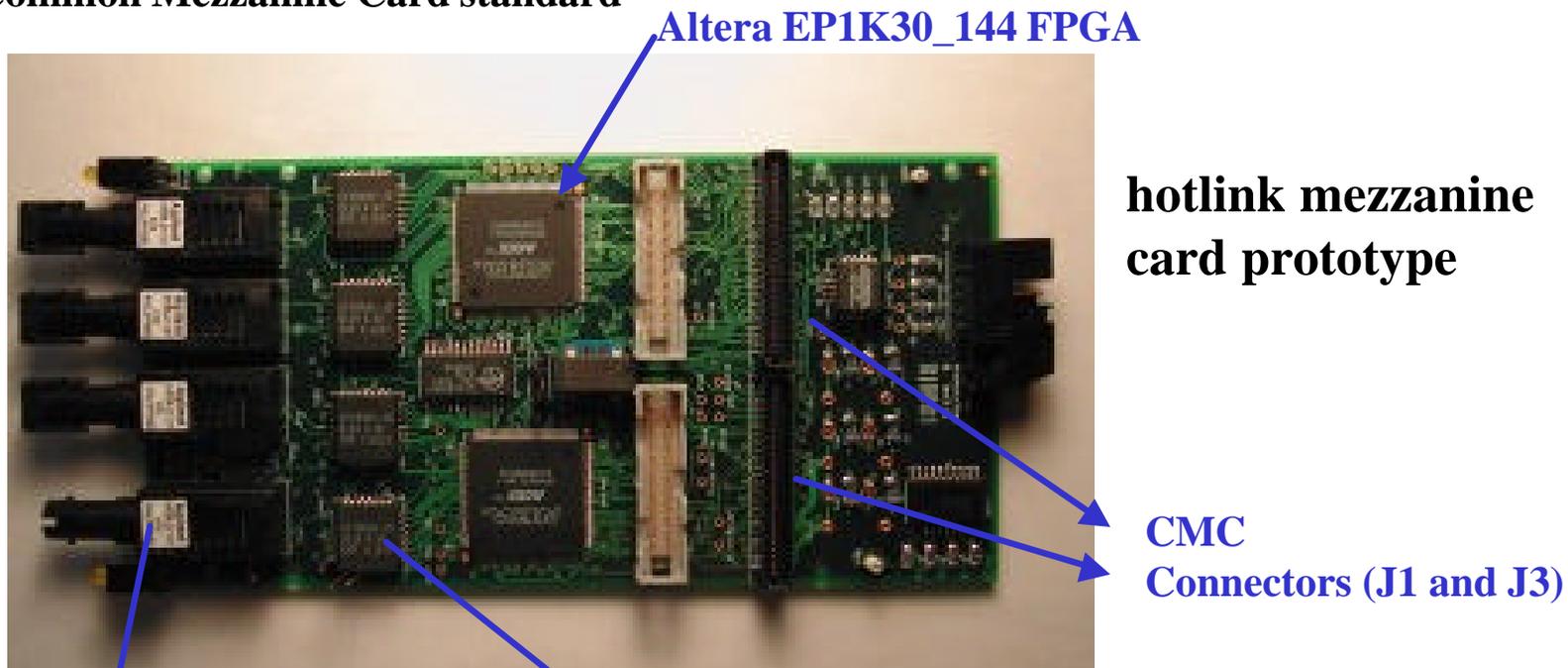
To/from  
Pulsar or  
a PC

The mezzanine card connectors can be used either for user I/O or SLINK cards

## Custom Mezzanine cards (**follow CMC standard**)

- Hotlink: Tx and Rx (CLIST, Muon data paths)
- Taxi: Tx and Rx (Iso, Reces data paths)

CMC: Common Mezzanine Card standard

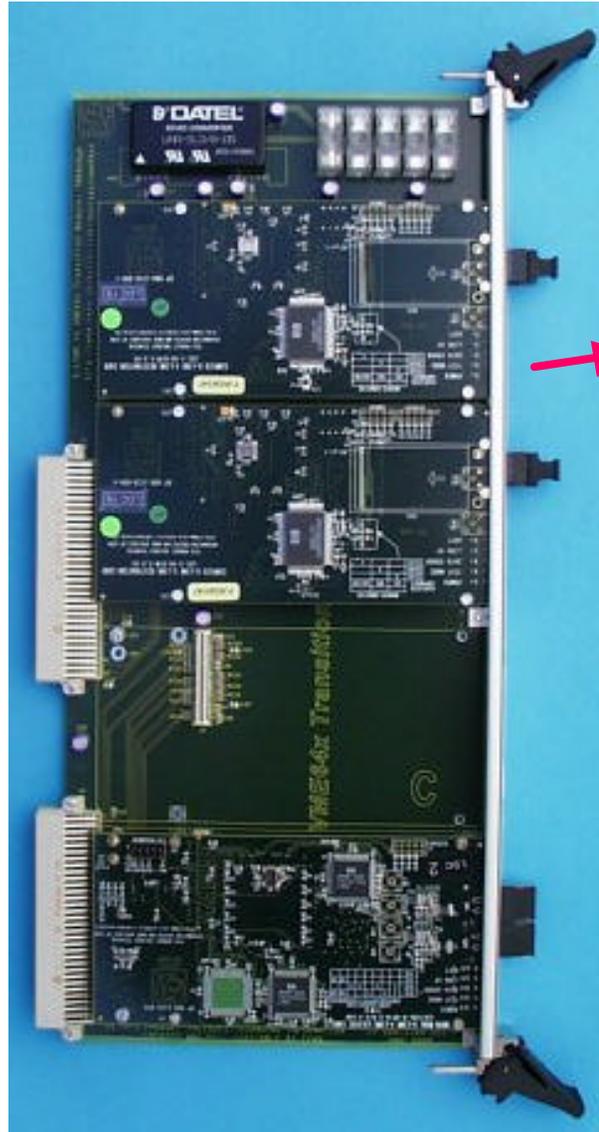
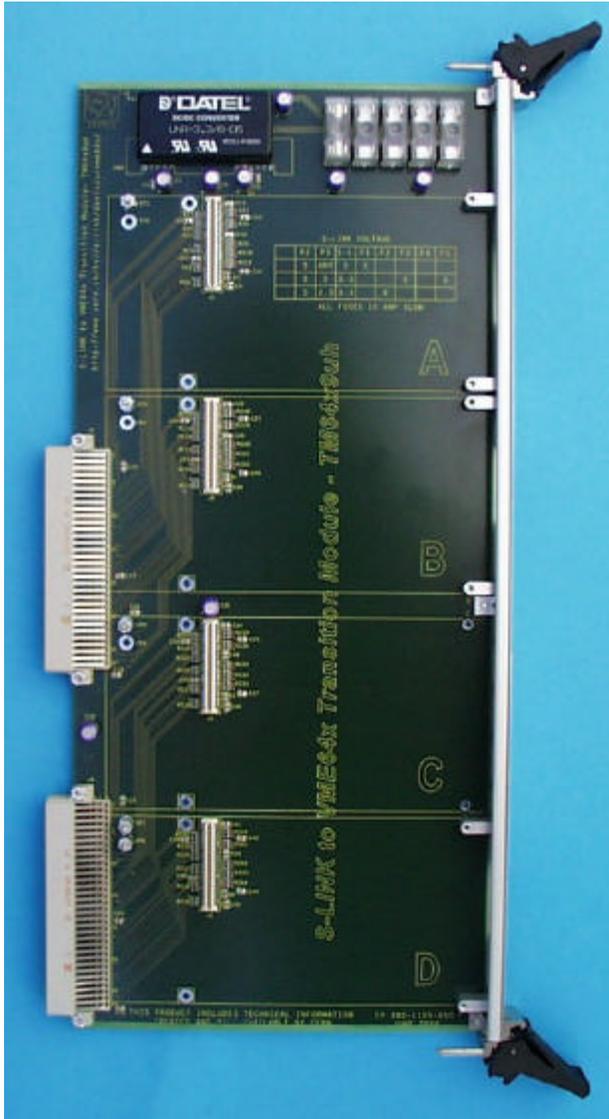


Hotlink Optical Tx/Rx: HFBR-1119T/2119T

Taxi Optical Tx/Rx: HFBR-1414T/2416T

Hotlink Tx/Rx: CY7B923JC/933JC

Taxi Tx/Rx: AM7968/7969-175JC



Loaded with SLINK Mezzanine cards

Can simply use CDF CAL backplane.

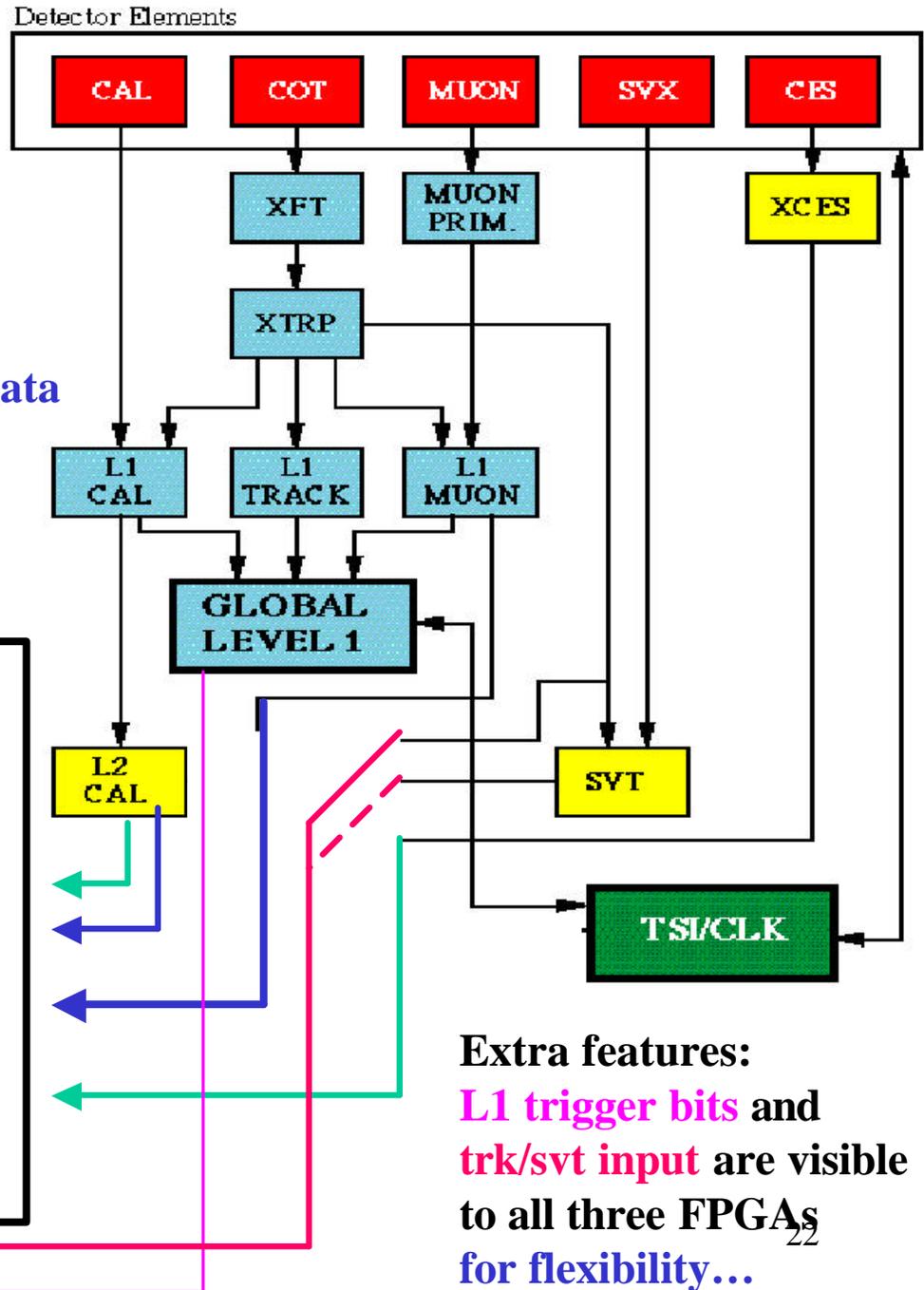
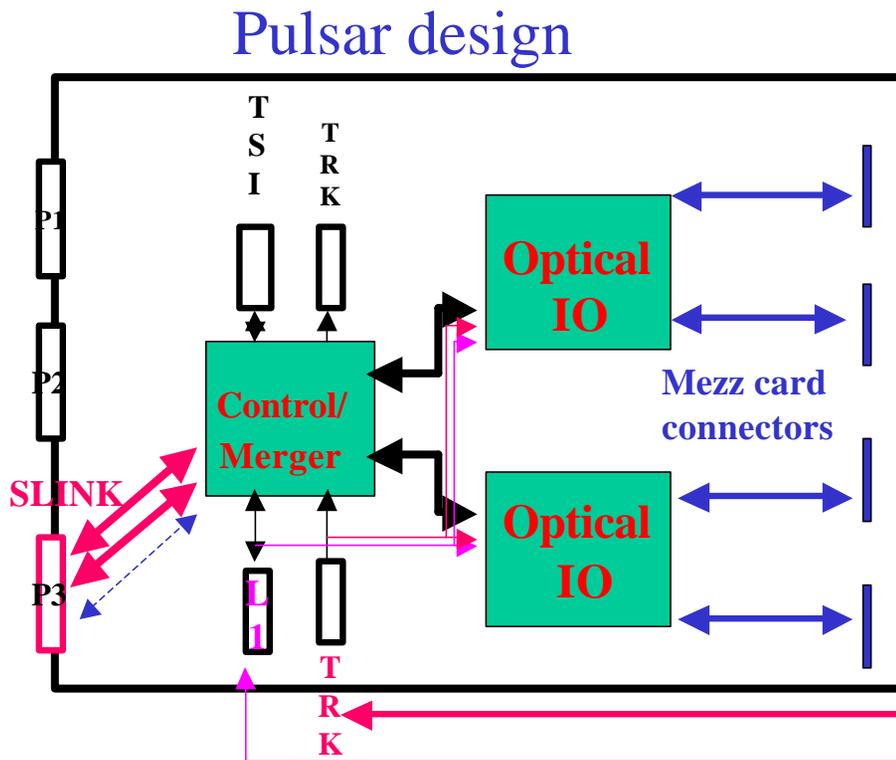
CERN sent us two transition modules.

The transition module is very simple (just a few SLINK CMC connectors). It uses P2 type connector for P3. We will only use P3 for SLINK and spare (user defined) signals.

## Pulsar approach:

**Goal:** only build **one type** of custom interface board and the rest are all commercial products.

Use mezzanine cards to take care optical data paths (Cluster, **Isolation**, Muon and **Reces**)



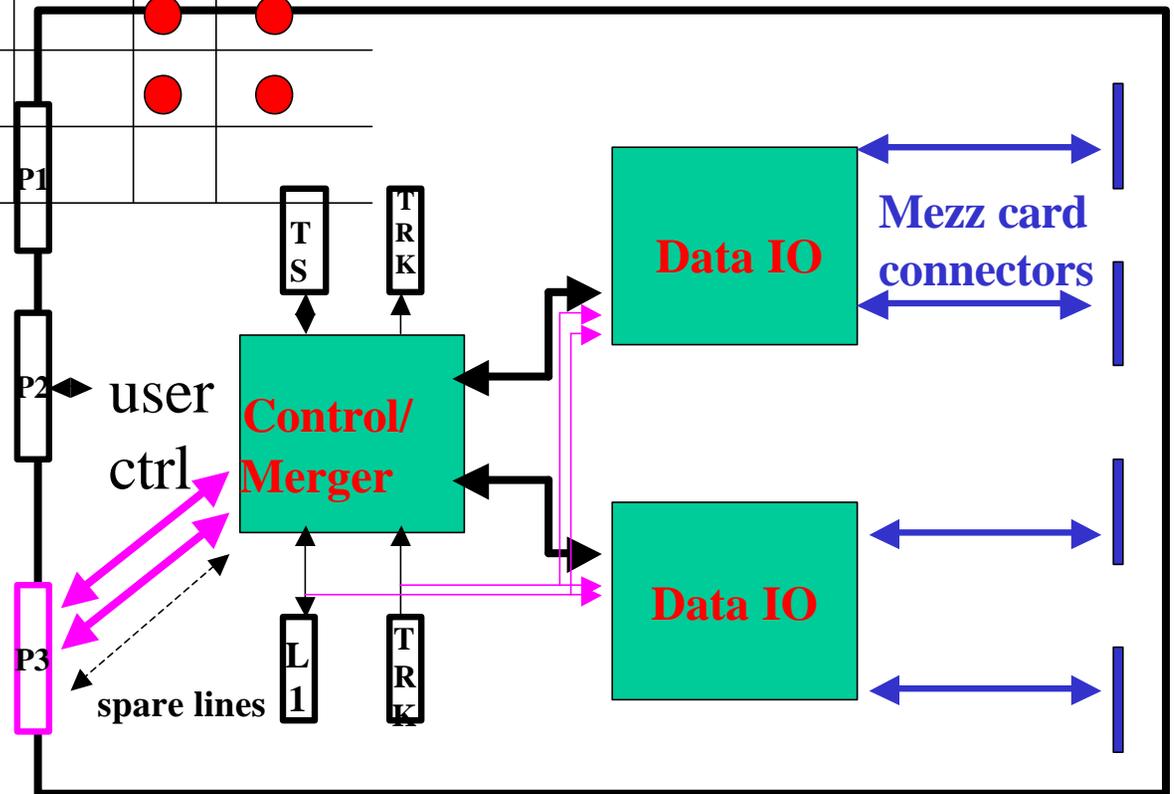
Extra features:  
**L1 trigger bits** and **trk/svt input** are visible to all three FPGAs for flexibility...

	L1	trk	svt	clist	Iso	reces	mu	SumEt,MEt
Tracks	●	●	●					
Jets	●	●	●	●				
electron	●	●	●	●	●	●		
photon	●			●	●	●		
muon	●	●	●				●	
Taus	●	●		●	●			
Met	●						●	●
SumEt	●						●	●
...								

What does Level 2 really do?

- Combine/matches trigger objects into e, muon ...
- Count objects above thresholds, or,
- Cut on kinematics quantities

Most trigger objects need L1 and track/svt trigger information, this is reflected in Pulsar design: SLINK I/O for flexibility





**Can we use Pulsar to upgrade Level 2 if needed?**

**it is possible... just some initial thoughts here...**

**for teststand: PULSAR → PULSer And Recorder**

**for upgrade: PULSAR can be used as**

**Processor “Controller”**

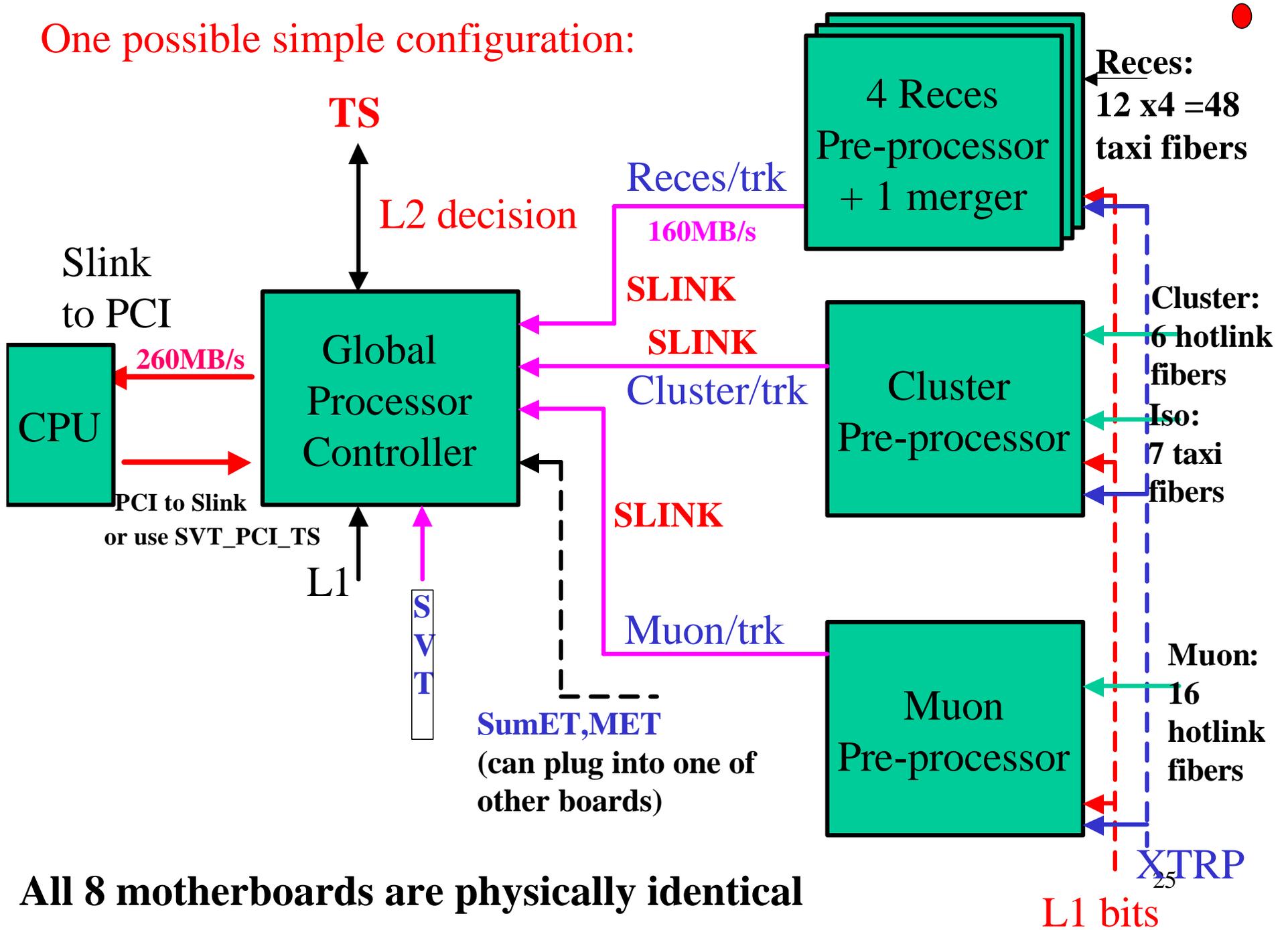
**pre-processor/merger**

**pULSer**

**And**

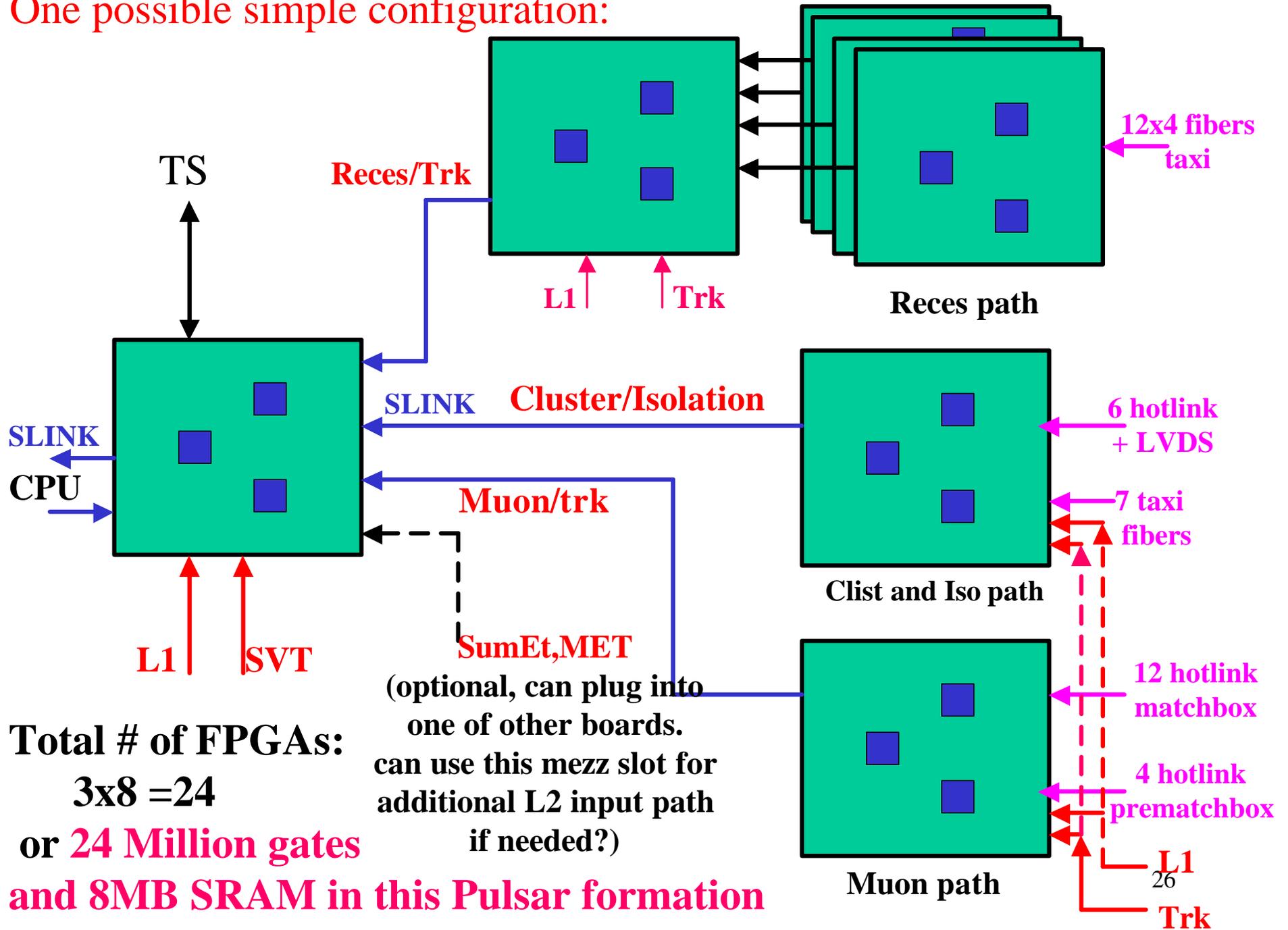
**Readout**

One possible simple configuration:



All 8 motherboards are physically identical

One possible simple configuration:

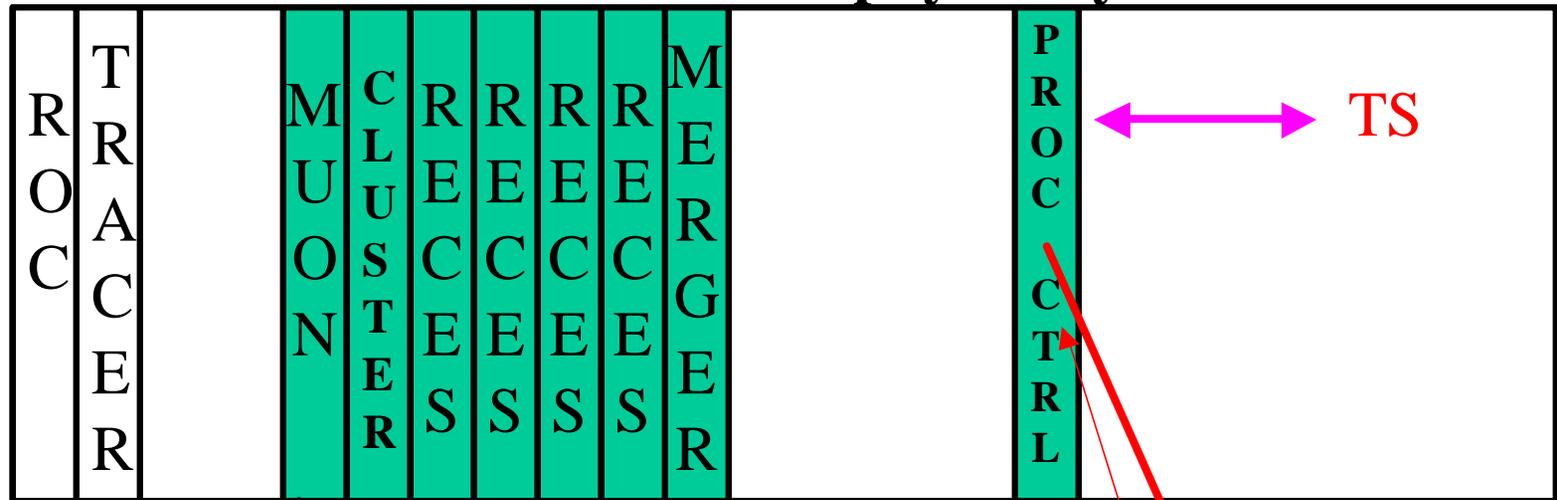


**Total # of FPGAs:**  
 $3 \times 8 = 24$   
 or **24 Million gates**  
 and **8MB SRAM in this Pulsar formation**

(optional, can plug into one of other boards. can use this mezz slot for additional L2 input path if needed?)

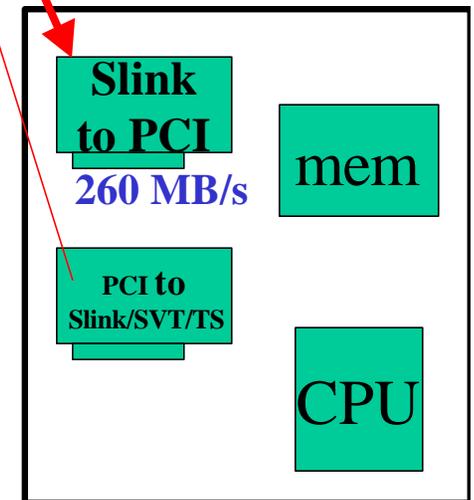
## All 8 motherboards are physically identical

Possible  
New L2  
Decision  
Crate



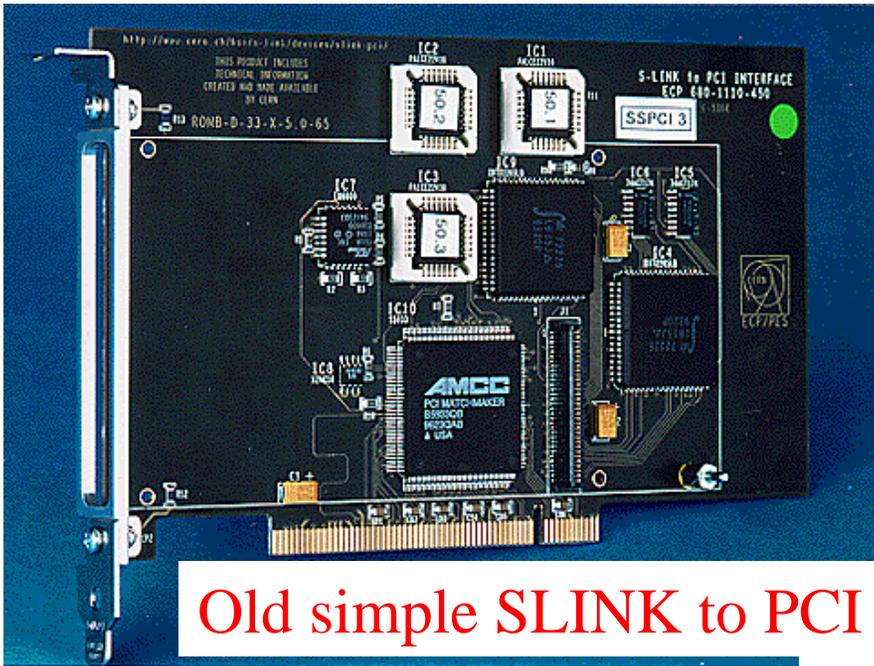
Each Pulsar board take two slots (due to mezzanine cards)  
Total: 8 pulsar boards = 16 slots

**Baseline design:** use pre-processors to simply suppress/organize data, use processor controller to simply pass data to CPU via Slink to PCI and also handshake with TS. All trigger algorithm will be handled by CPU.



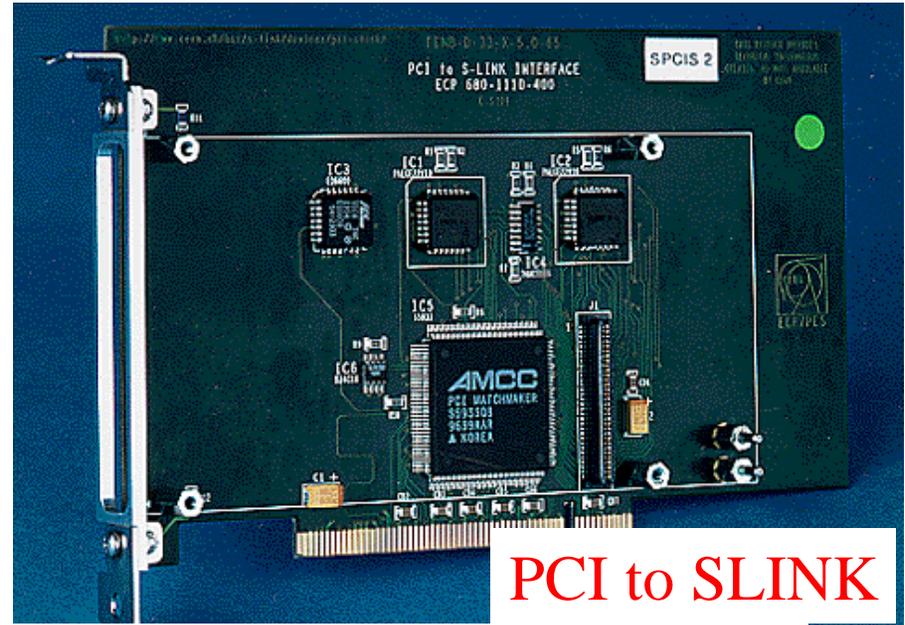
GHz PC or  
VME processor

Proven technology, has been used by a few experiments to take hundreds of TB data in the past few years



Old simple SLINK to PCI

120MB/s



PCI to SLINK

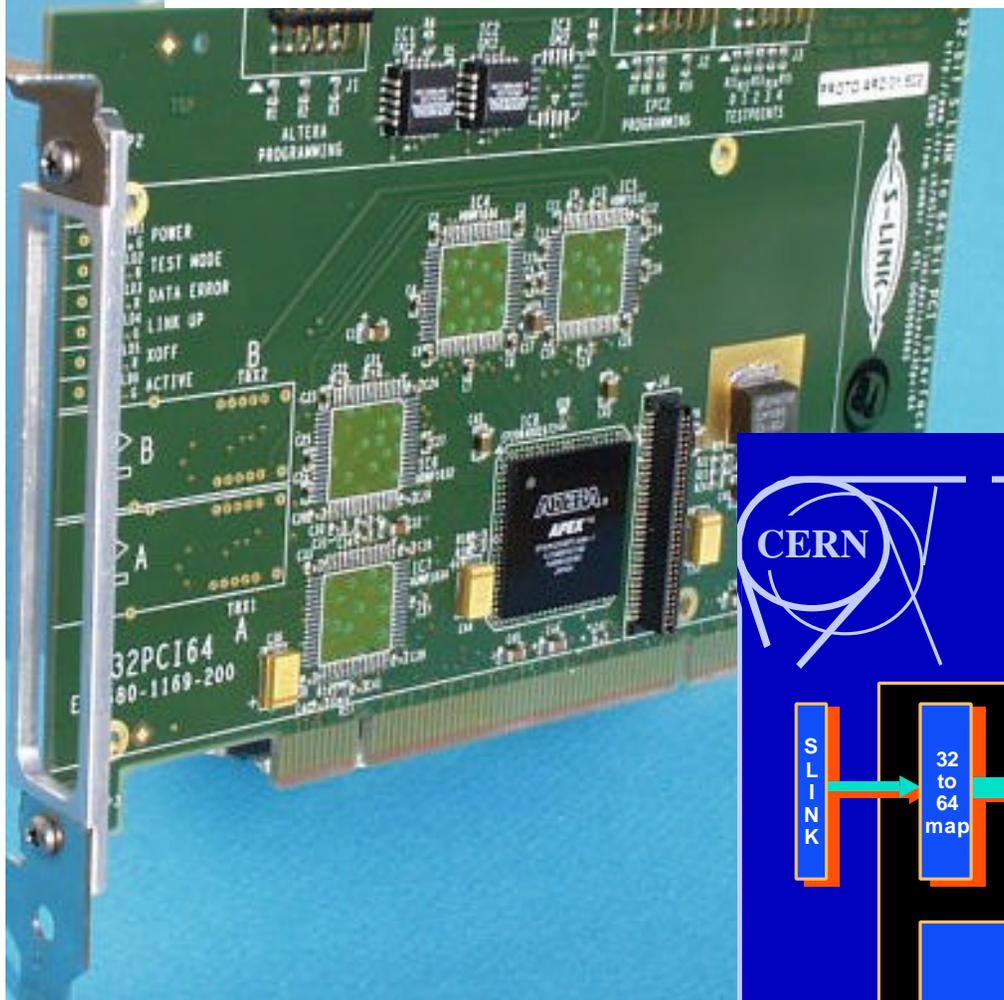
New SLINK interface mezzanine card:



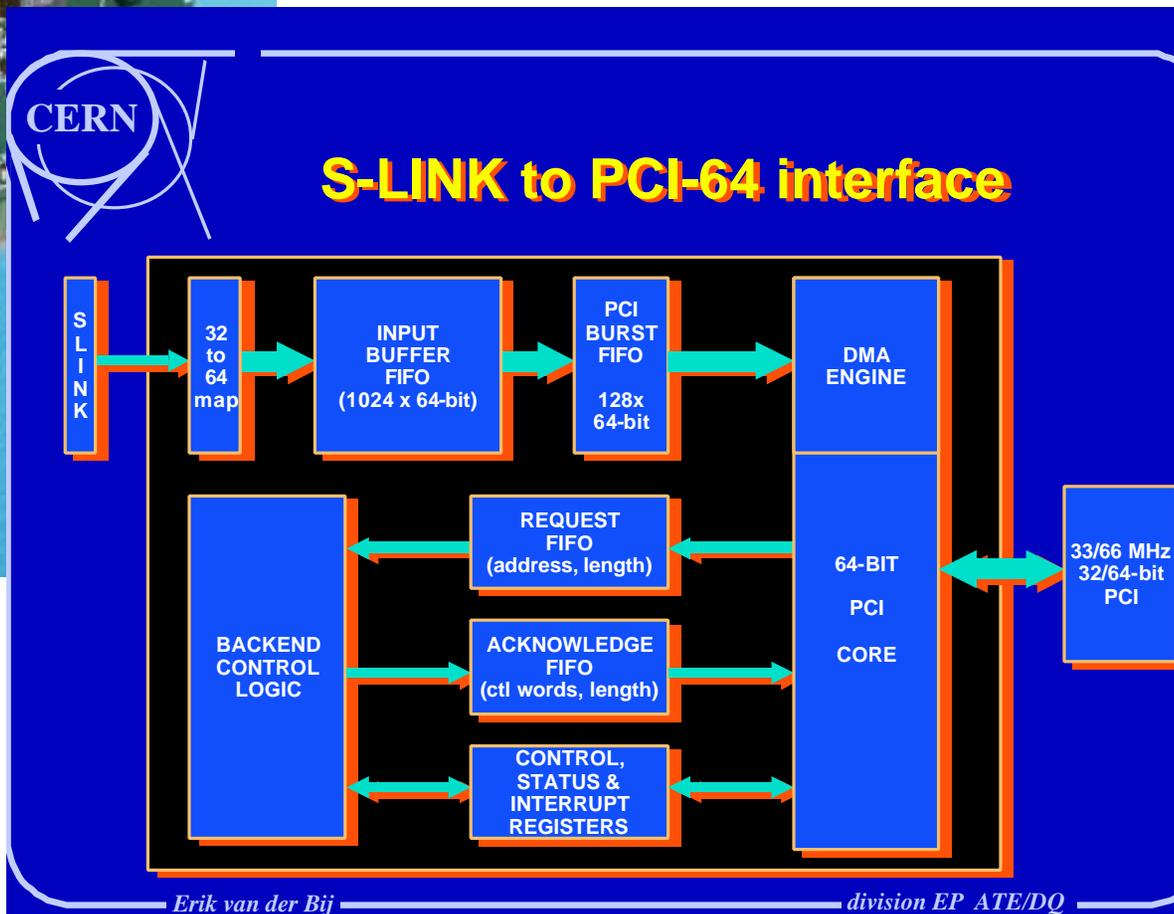
160MB/s



# New 32-bit SLINK to 64 bit PCI interface card: S32PCI64

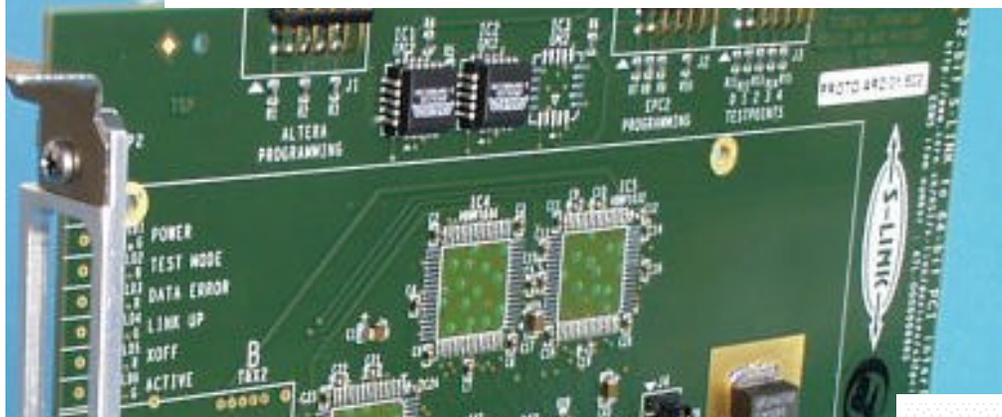


- highly autonomous data reception
- 32-bit SLINK, 64-bit PCI bus
- 33MHz and 66 MHz PCI clock speed
- up to **260MByte/s** raw bandwidth



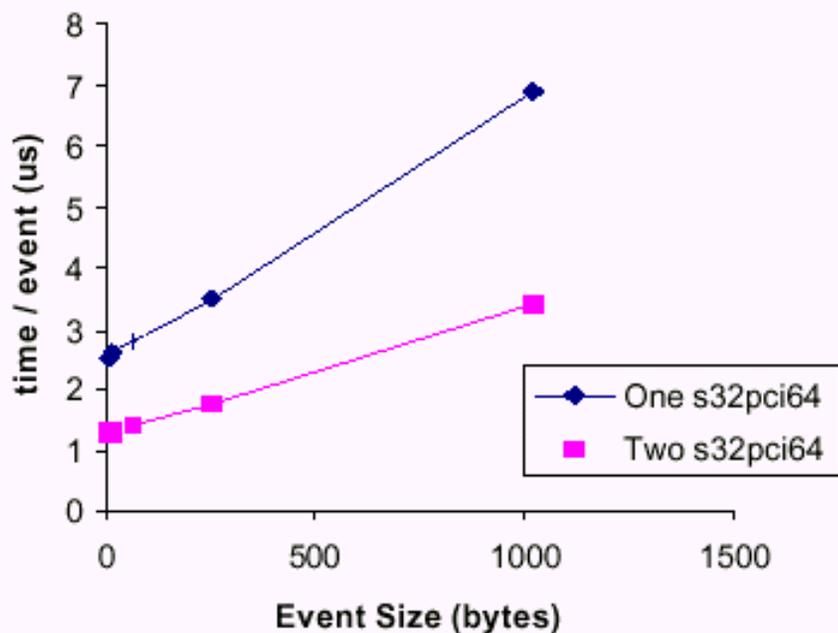
High-speed follow up of the Simple SLINK to PCI interface card

## New 32-bit SLINK to 64 bit PCI interface card: S32PCI64

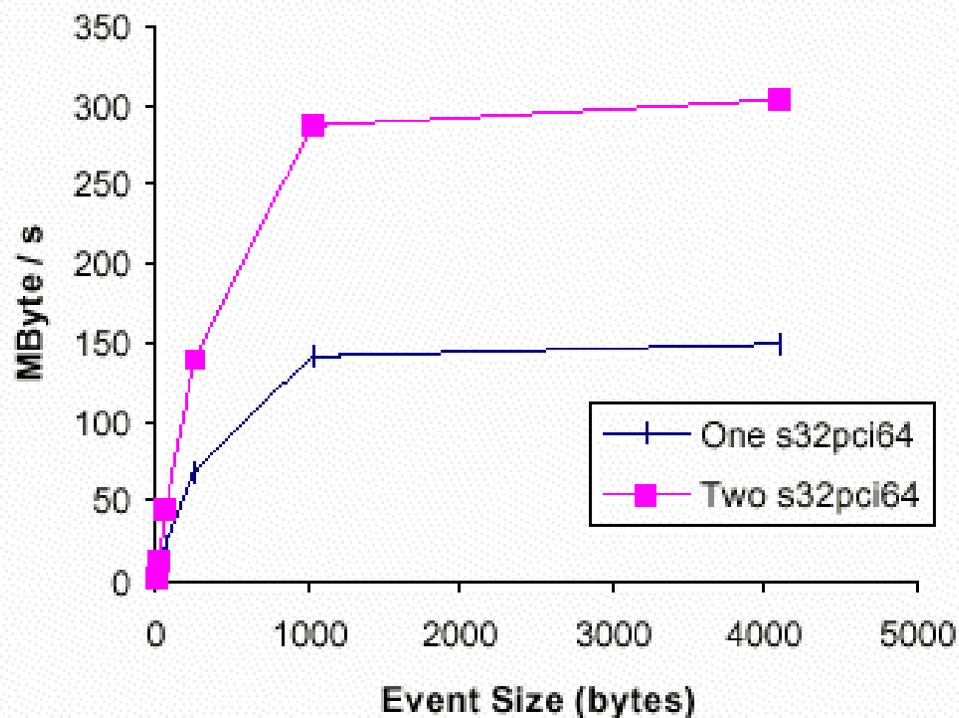


- highly autonomous data reception
- 32-bit SLINK, 64-bit PCI bus
- 33MHz and 66 MHz PCI clock speed
- up to **260MByte/s** bandwidth

S32PCI64 overheads



S32PCI64 throughput



<http://hsi.web.cern.ch/HSI/s-link/devices/s32pci64/>

## Baseline Design (with the simple configuration) could be:



- **PreProcessor/Interface versions** of the board gather data from each subsystem and package the data. This can include sparsification based on L1 trigger bits, tracking, or the data itself.
- **Processor controller/Merger version** of the board merges data from interface boards and packages the data for transfer to a CPU. The data can be further sparsified at this stage. This board also provide the interface between L2 and the TS.
- **Both types of PreProcessor board can be used to readout data for TL2D**

The default operation would have the final decisions made in code in the CPU.

---

**Pulsar is designed to be able to do more if necessary.  
Designed to be quite flexible... will come back to this later**

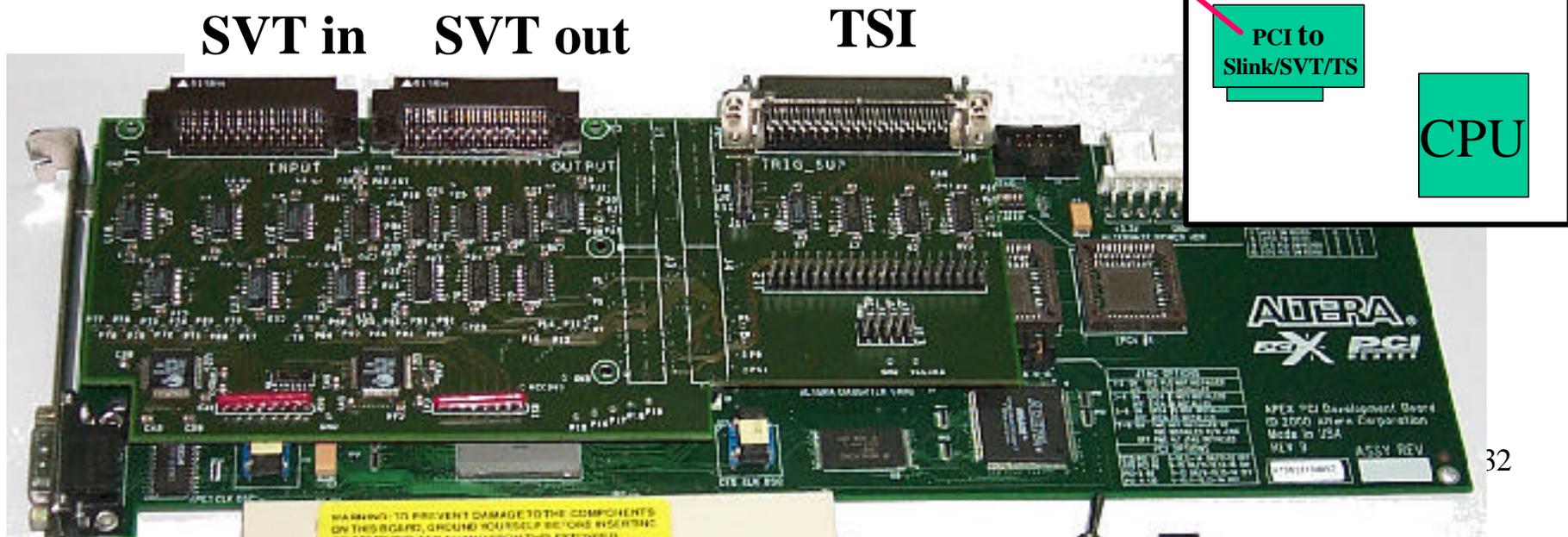
## L2 decision from CPU: how to handshake with TS?



There could be a few ways to achieve this:

(1) could use PCI to SLINK card: send a SLINK message back to Pulsar Processor Controller, then Pulsar handshakes with TS;

(2) Or use SVT\_PCI\_TS daughter card built by Franco Spinella:



Franco Spinella built a PCI daughter card: SVT-PCI-TS which can be plugged into Altera

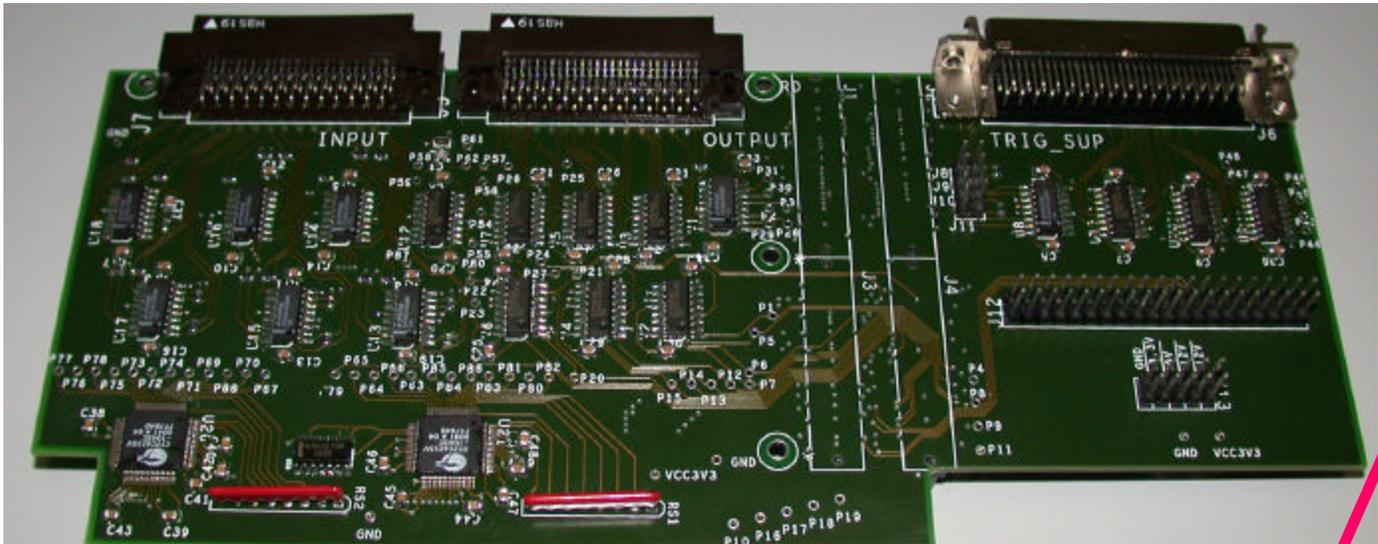
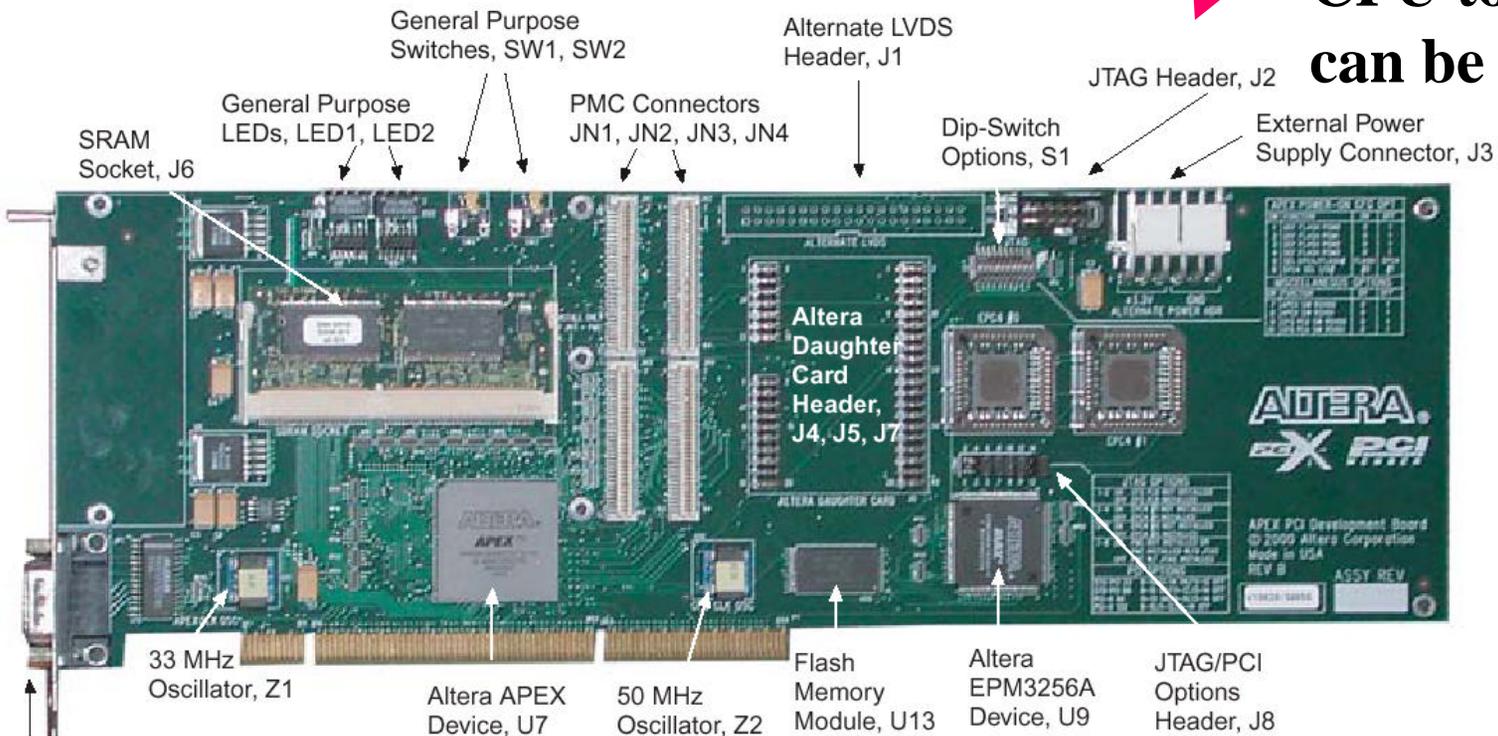


Figure 1. APEX PCI Development Board

PCI board

The time to send a decision from CPU to Pulsar/TS can be short:

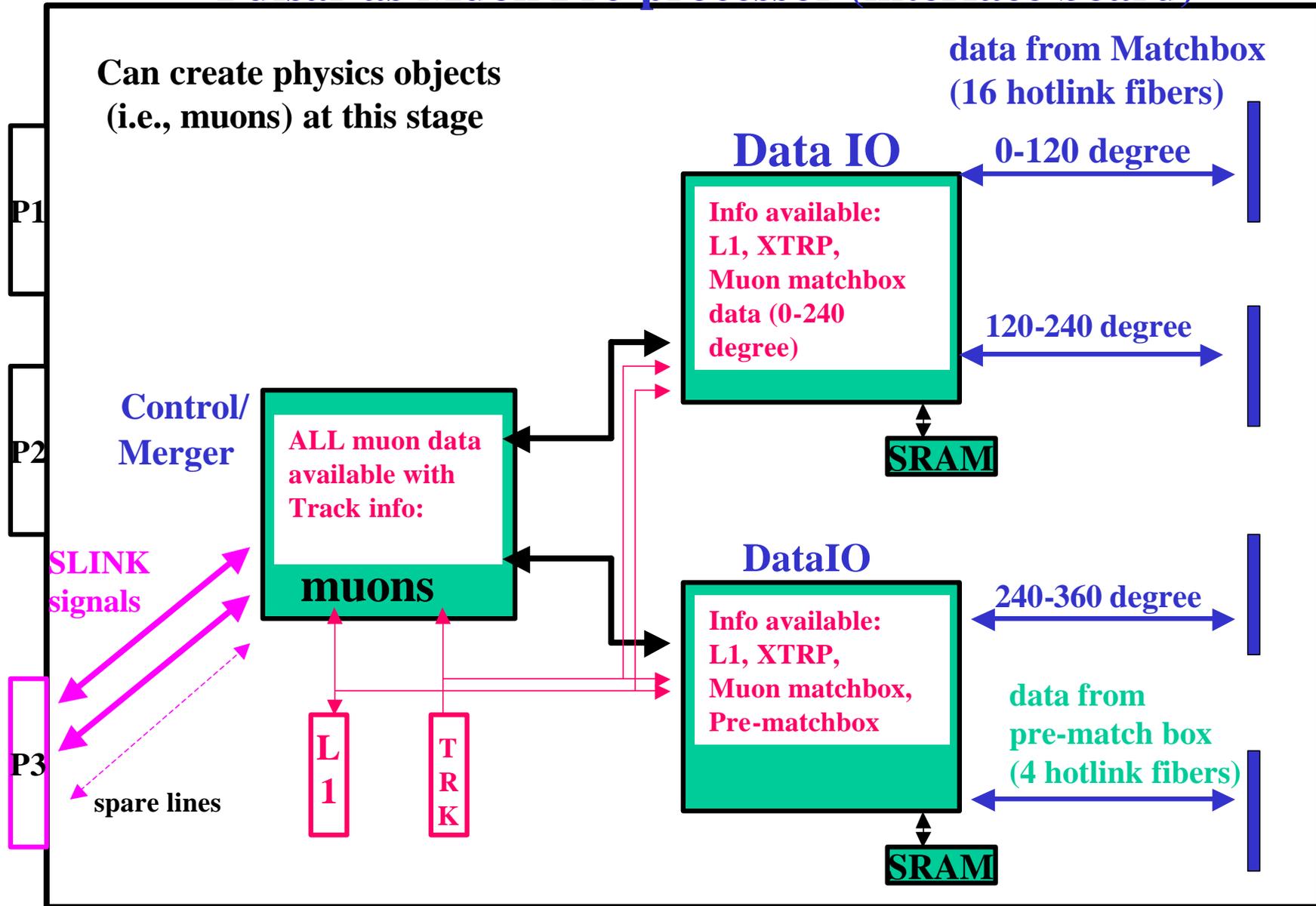


~ 1us measured value

# Board Level Pulsar Flexibility

- **Powerful modern FPGAs (3M gates) and SRAMs (1MB)**
- **Have L1 trigger bits and XTRP info available**
- **Each Pulsar could be used as pre-processor**
  - create physics objects such as muon and electrons, etc using SRAM as LUTs ...
  - suppress data size (only pass ROI downstream)
  - Reduce processing time
  - **Optional: if CPU is really fast enough, no need to create physics objects inside FPGAs.**
- **see some examples next.**

# Pulsar as Muon Pre-processor (interface board)



## Muon data path:



**Muon path has 16 fiber (hotlink) inputs, large data size (11Kbits/evt).  
12 of them from matchbox, each fiber cover 30 degree in phi.  
4 from pre-matchbox.**

## Designed to have many options:

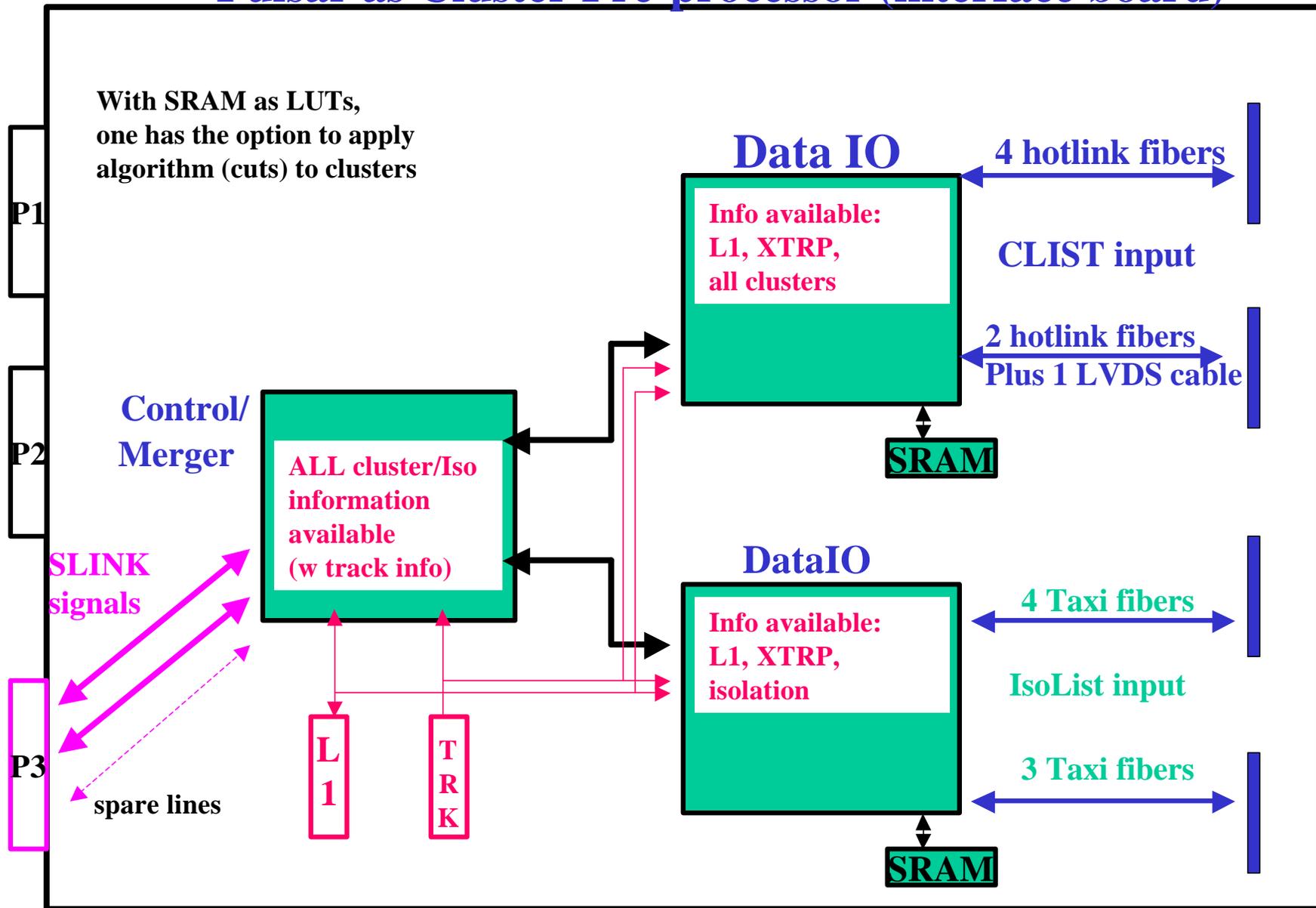
### Simple ways to suppress muon data (examples):

- **first check L1 bits: if no muon info is needed, send out empty data package with only header and trailer, where header will be stamped with buffer number, bunch counter etc.**
- **if muon info is needed, then check XTRP(track) bits, may only pass the ROI (Region-Of-Interest) phi data downstream;**
- **or zero suppress the data based on muon data itself**

### Sophisticated way to suppress muon data:

- **use XTRP LUT (SRAM) to pre-match tracks with muon data;**

# Pulsar as Cluster Pre-processor (interface board)

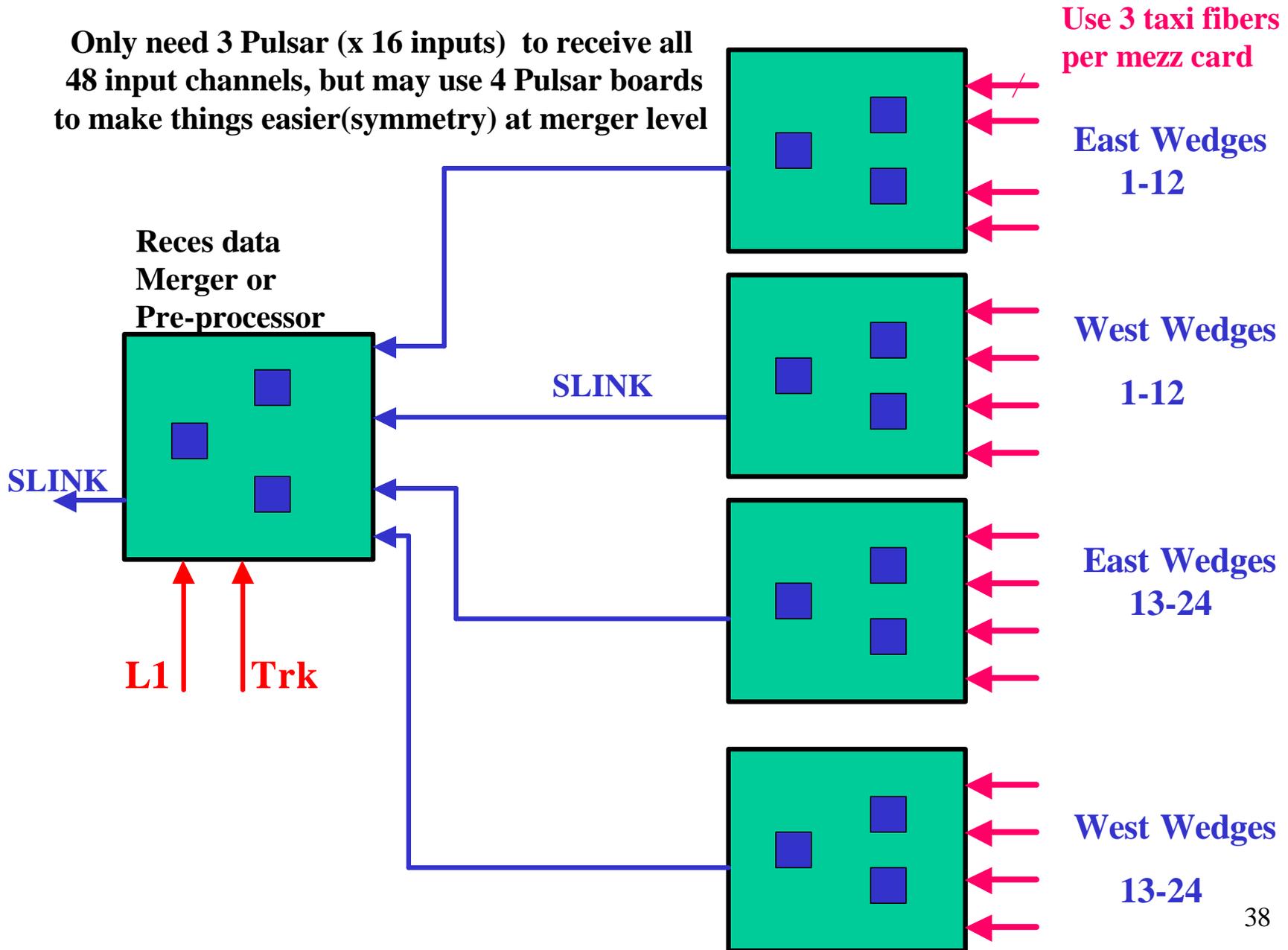


Can already create trigger objects at this stage (optional)

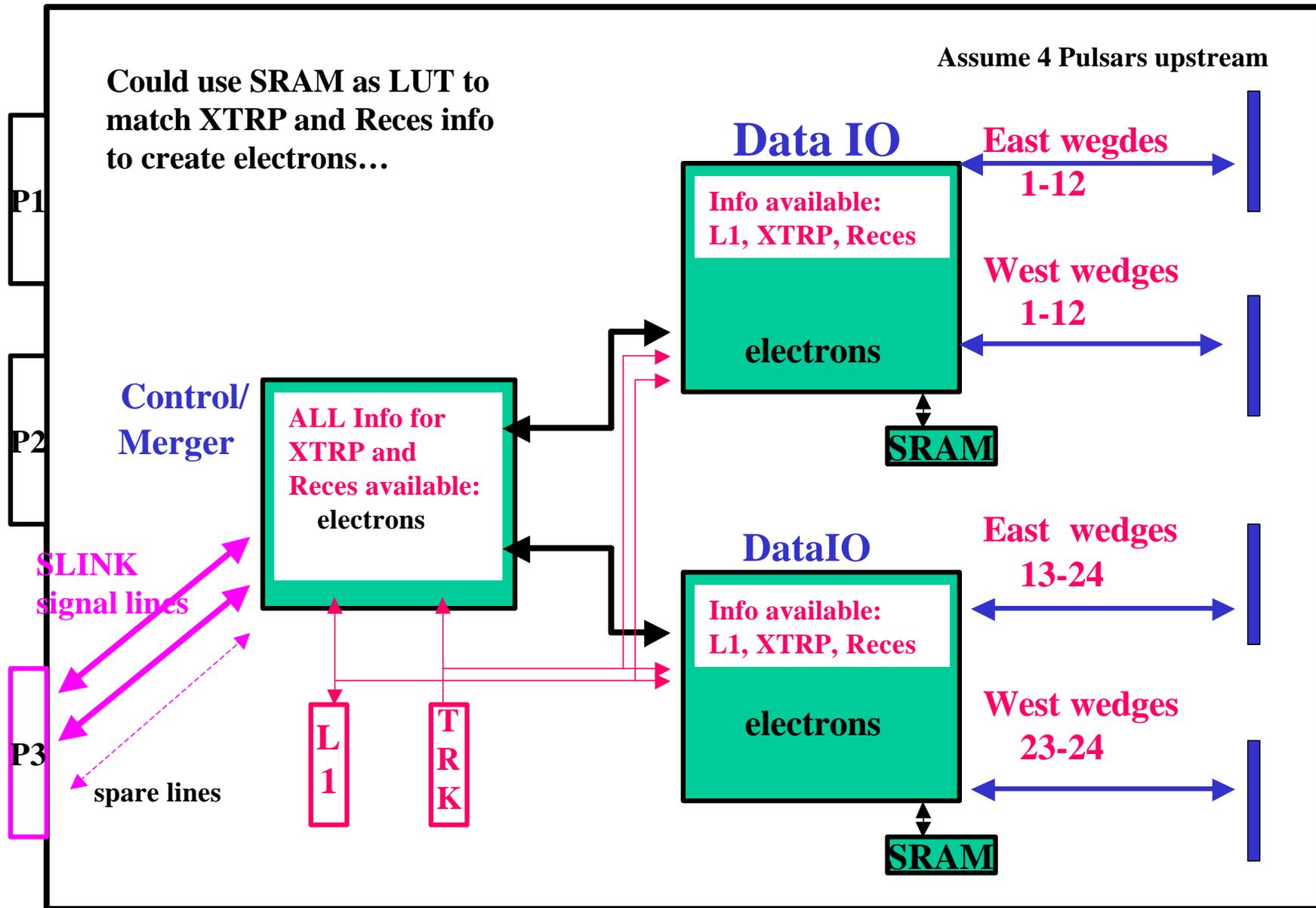
# One possible way to sink Reces data: 48 fibers (wedges):

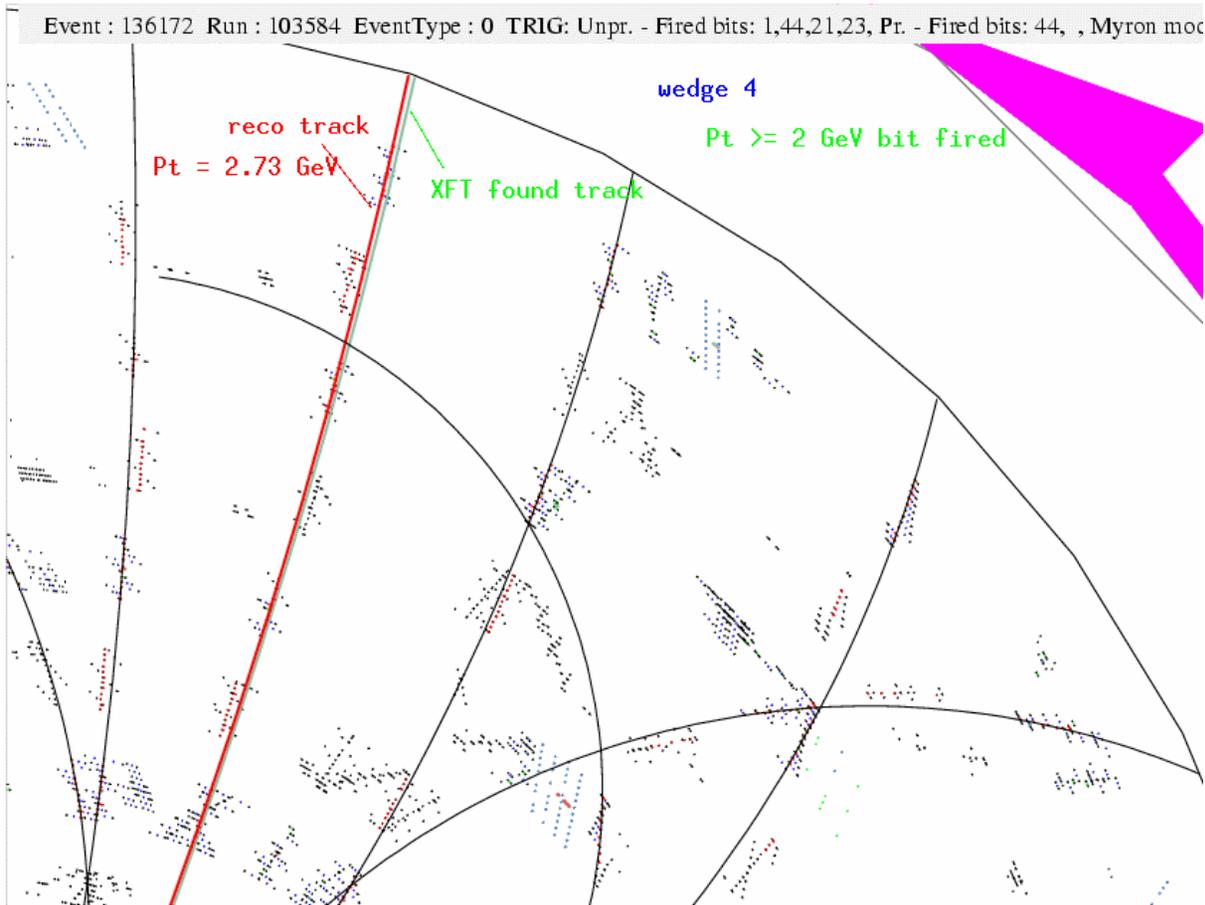


Only need 3 Pulsar (x 16 inputs) to receive all 48 input channels, but may use 4 Pulsar boards to make things easier(symmetry) at merger level

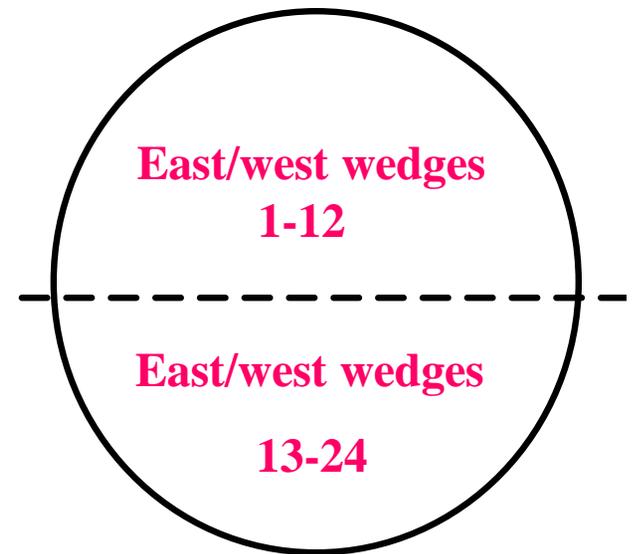


# Pulsar as Reces data PreProcessor (merger)



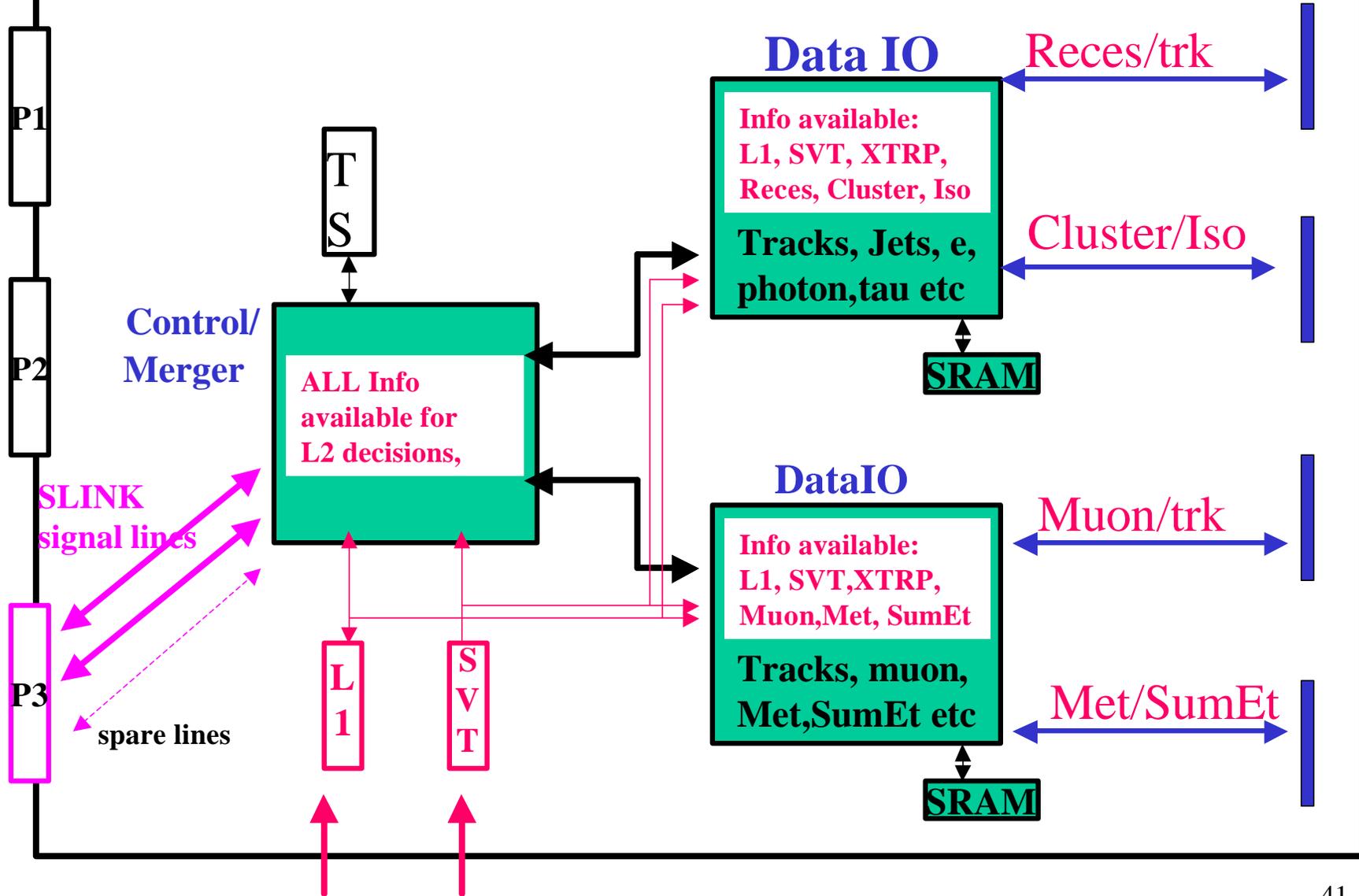


**XTRP is in r-phi**



# Pulsar as Processor Controller

Pulsar design is driven by physics needs ...



SVT data arrives

	L1	trk	svt	clist	Iso	reces	mu	SumEt,MEt
Tracks	●	●	●					
Jets	●	●	●	●				
electrons	●	●	●	●	●	●		
photons	●			●	●	●		
muons	●	●	●				●	
Taus	●	●		●	●			
Met	●						●	●
SumEt	●						●	●
...								

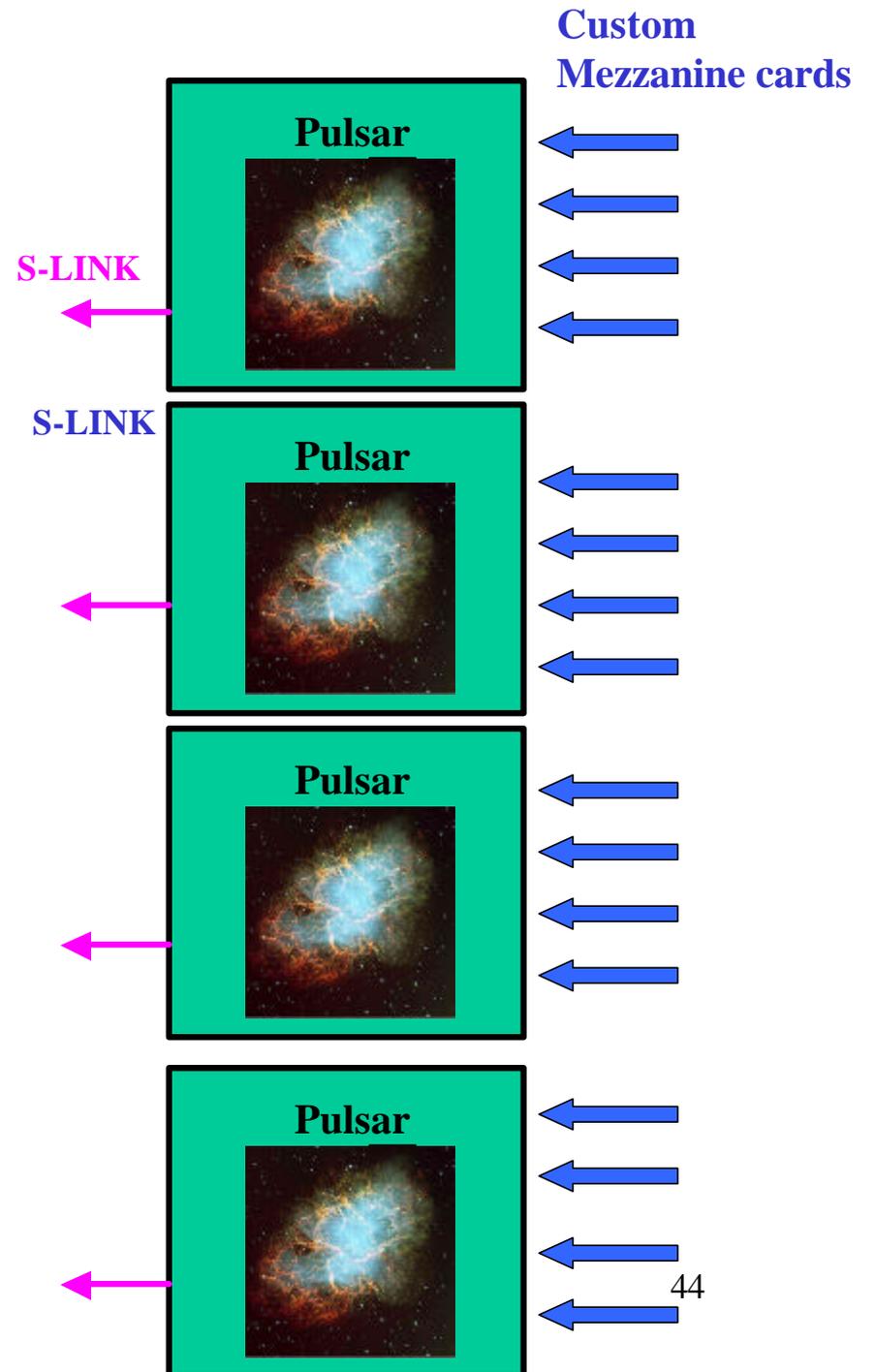
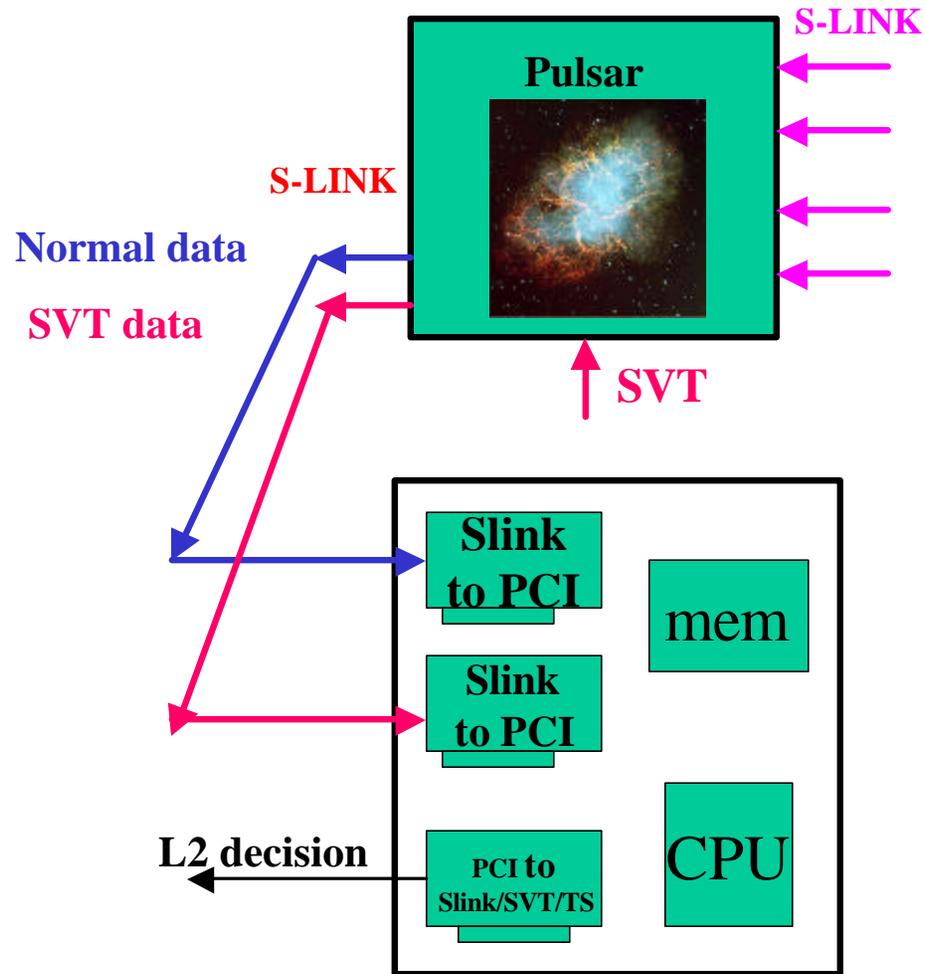
**Back to Basic again:** What does Level 2 really do?

- Create/match trigger objects into e,muon,jets... etc
- Count objects above thresholds, or,
- Cut on kinematics quantities

# System Level Pulsar flexibility

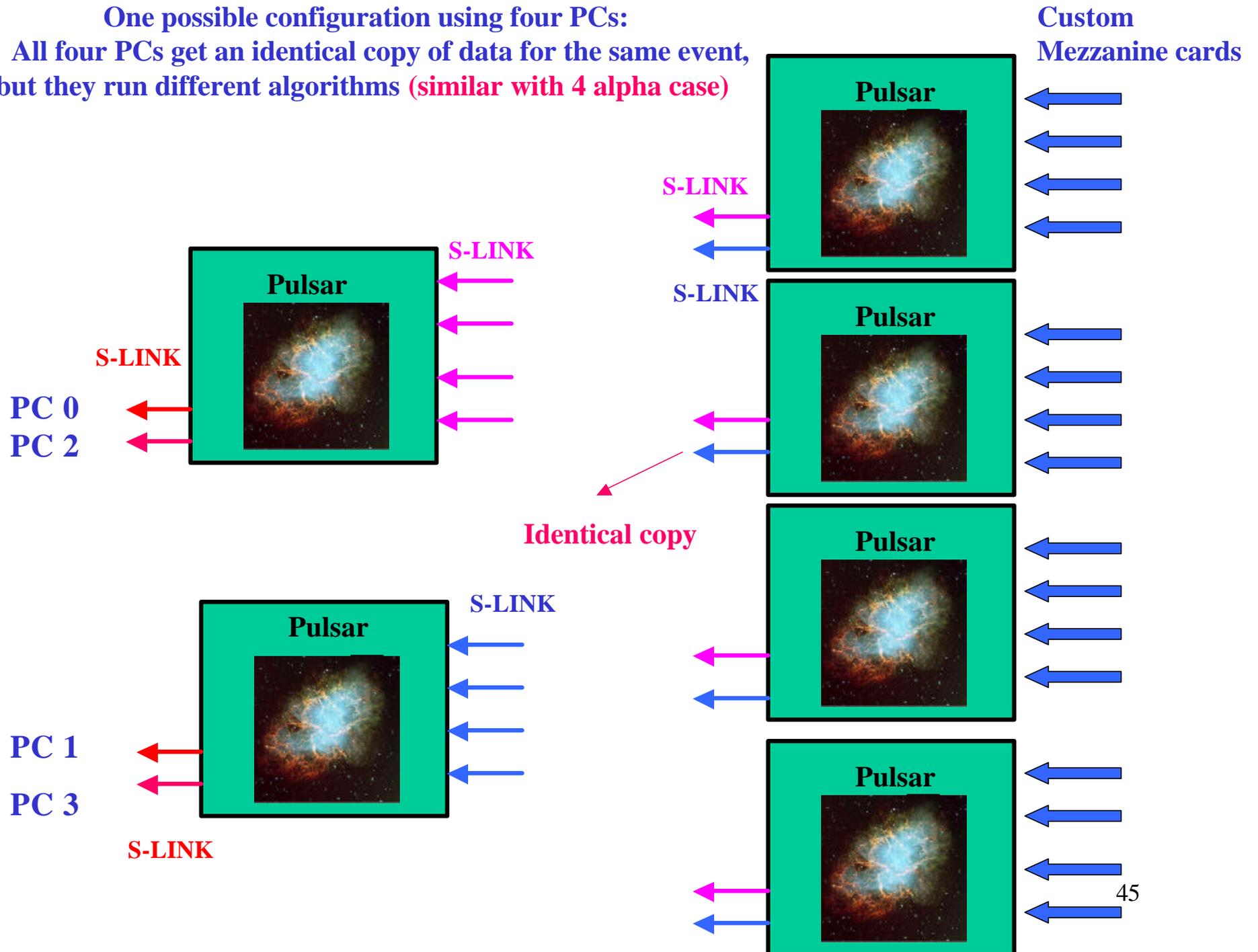
- **Flexible “lego style” design**
- **Many different configuration possible to meet Run2b trigger needs**
  - **Allows multiple PCs to be used for trigger decision**
    - **different trigger algorithm for the same event**
    - **same trigger algorithm but each PC deal with different buffer**
- **see a few examples next ...just to show the flexibility**

One possible configuration  
 one PC use two PCI slots to receive SLINK data:  
 One receives normal trigger data,  
 the other is dedicated for SVT data (arrives late).  
 CPU can start working on the early arrival data first



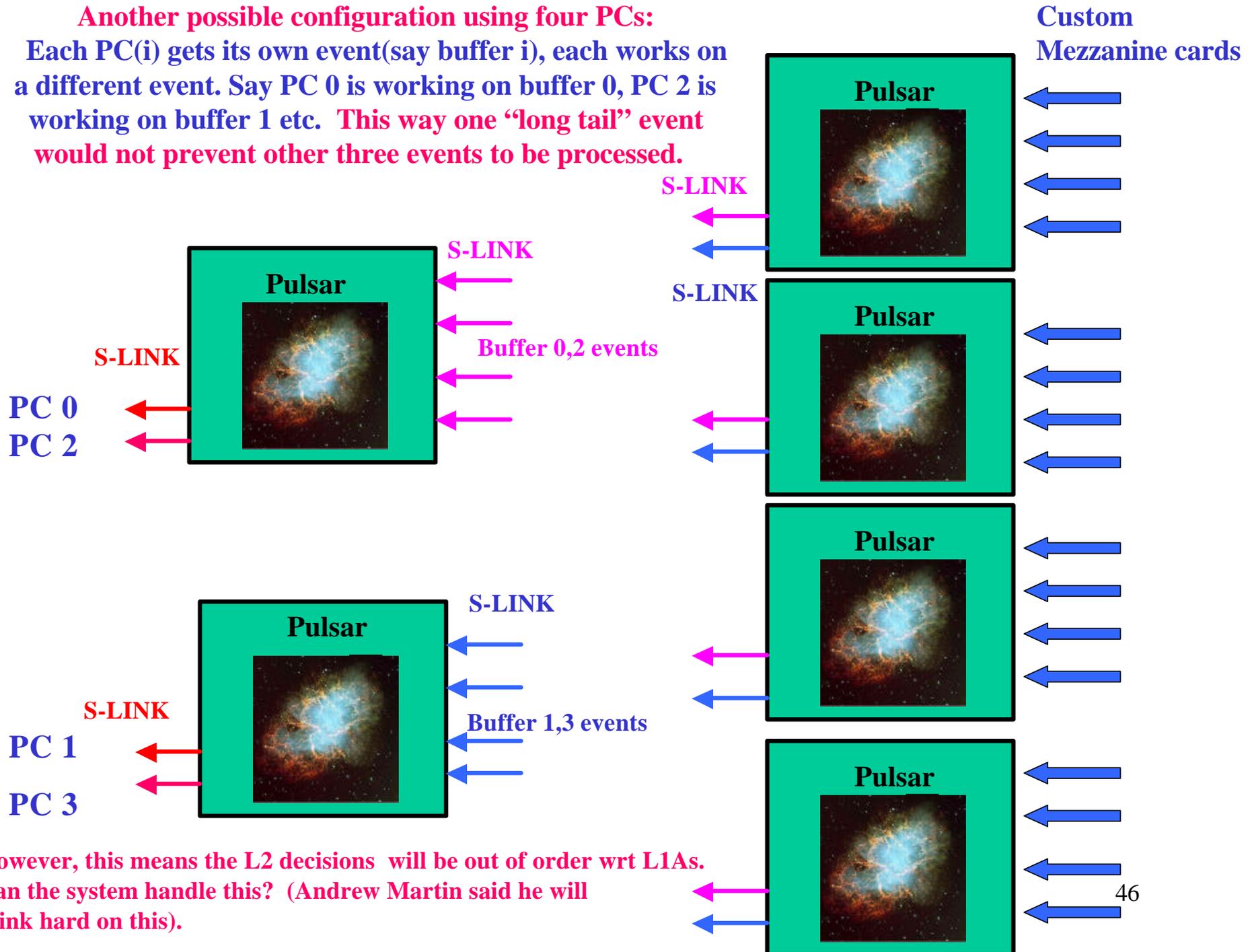
**One possible configuration using four PCs:**

**All four PCs get an identical copy of data for the same event, but they run different algorithms (similar with 4 alpha case)**

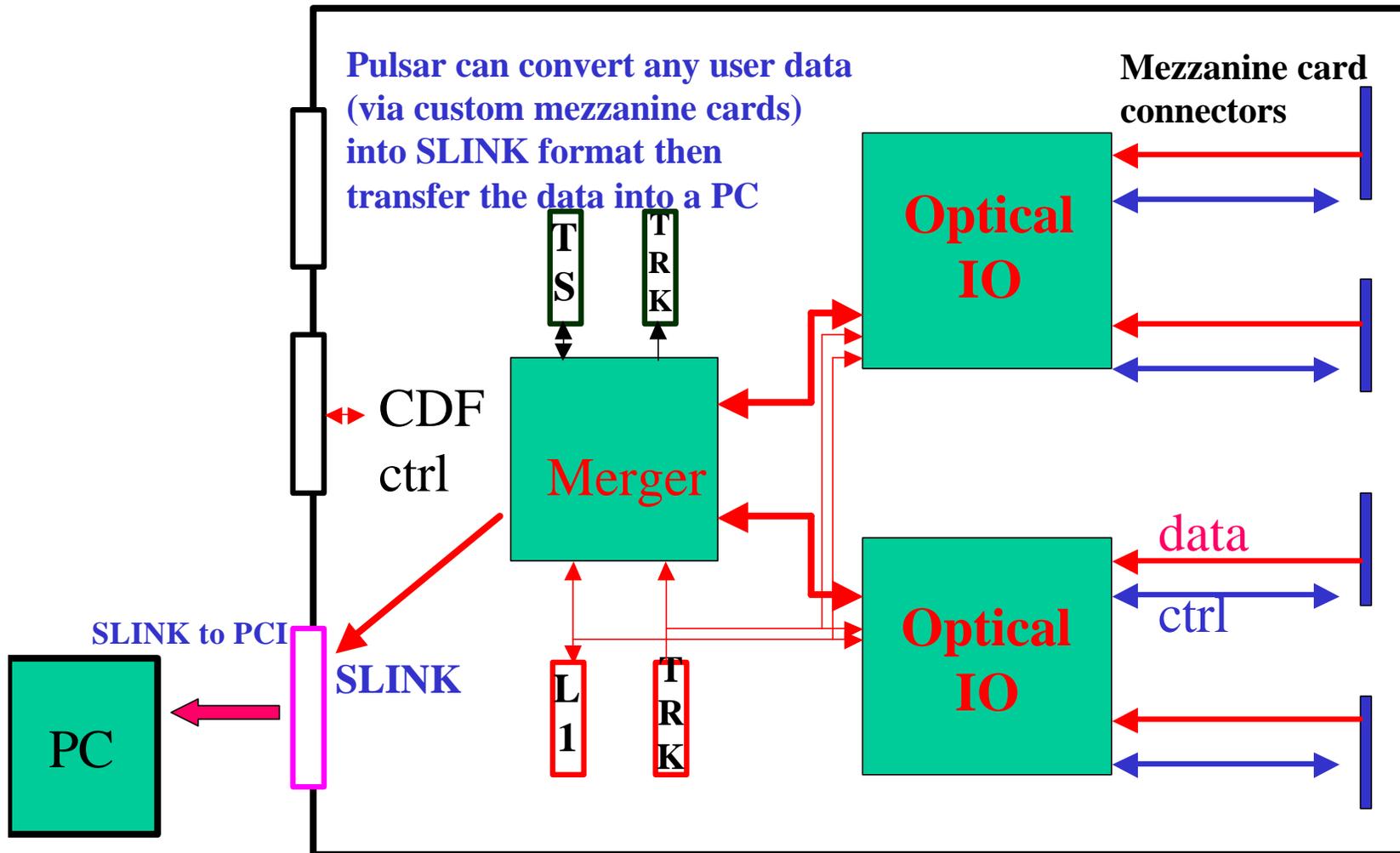


**Another possible configuration using four PCs:**

Each PC(i) gets its own event(say buffer i), each works on a different event. Say PC 0 is working on buffer 0, PC 2 is working on buffer 1 etc. **This way one “long tail” event would not prevent other three events to be processed.**

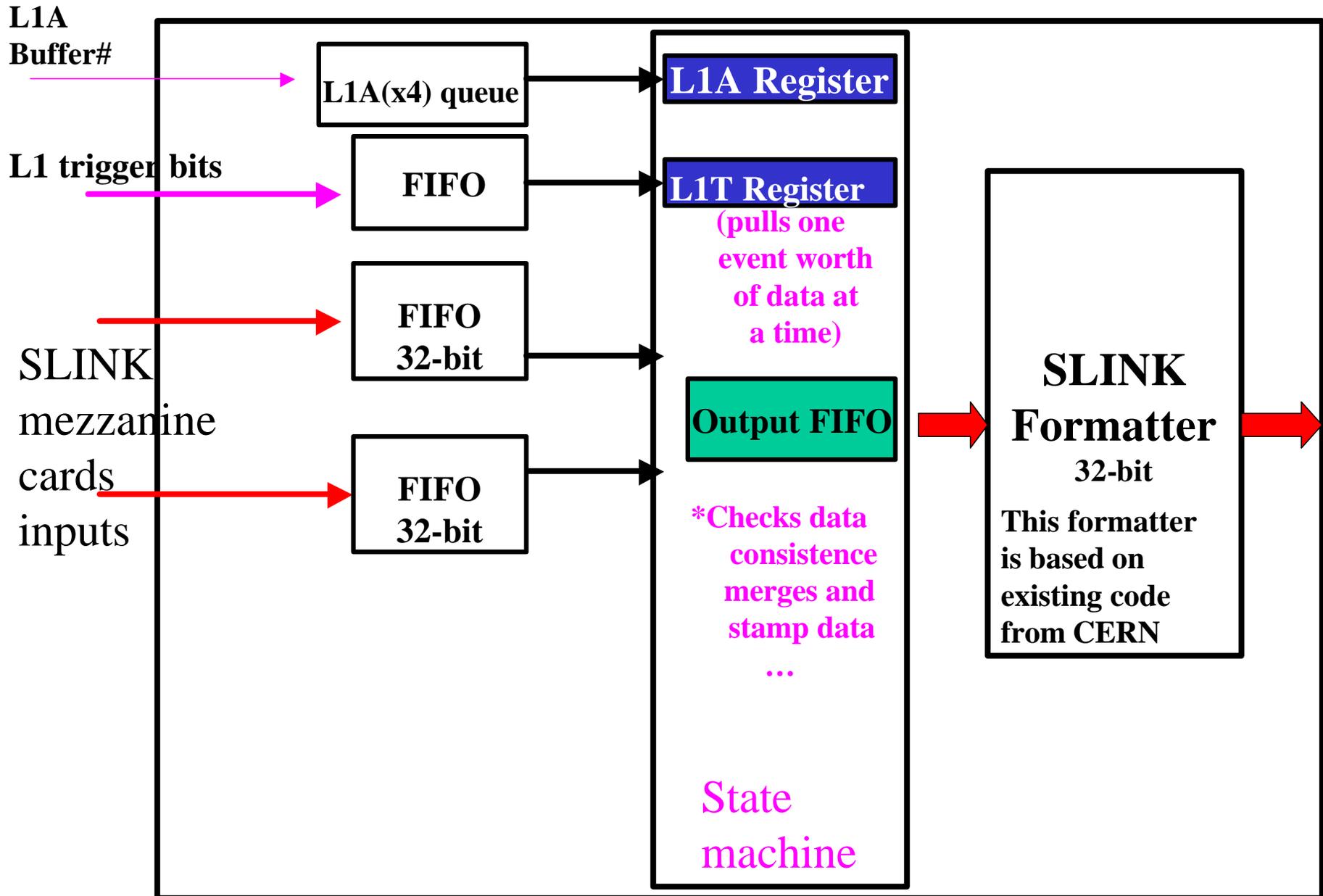


**Current firmware status: we have written firmware to record data into a PC for the teststand. Work applicable to upgrade needs.**



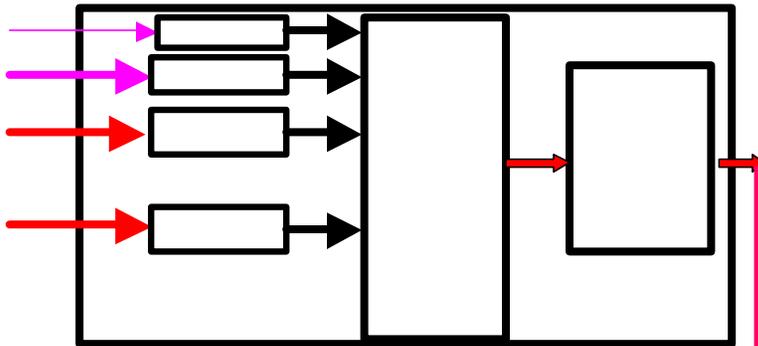
Pulsar in (**general purpose**) recorder mode (directly into a PC)

## Firmware for Pulsar in recorder mode(into a PC):

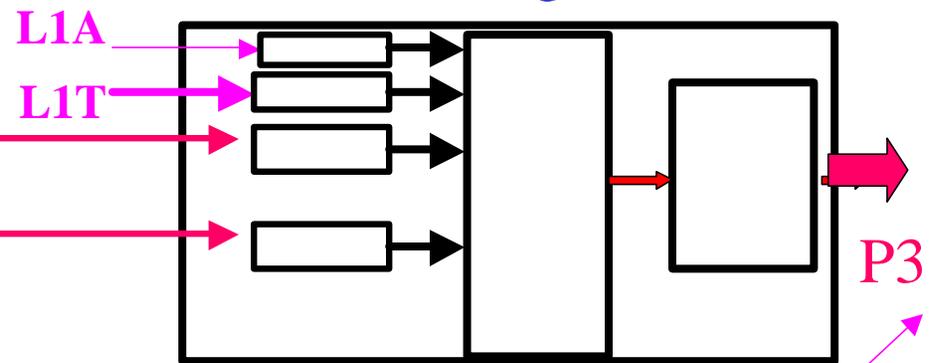


# Firmware design is similar in all three FPGA's on all versions of the Pulsar

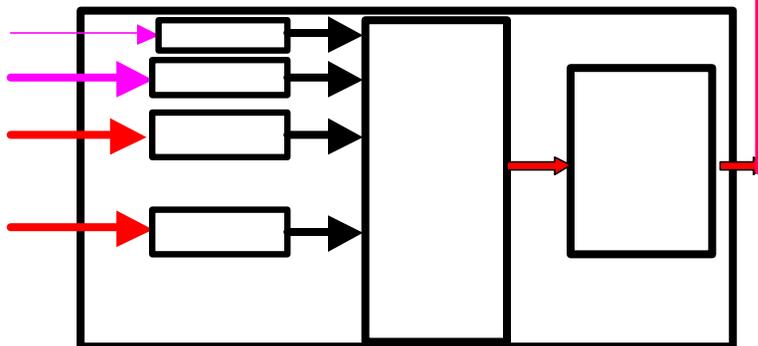
DataIO FPGA



Control/merger FPGA



DataIO FPGA



Data package (per L1A)  
in SLINK format sent to PC...

## SLINK message format



Trailer:

Data size

Error checking bits

etc ...

Header:

Source ID

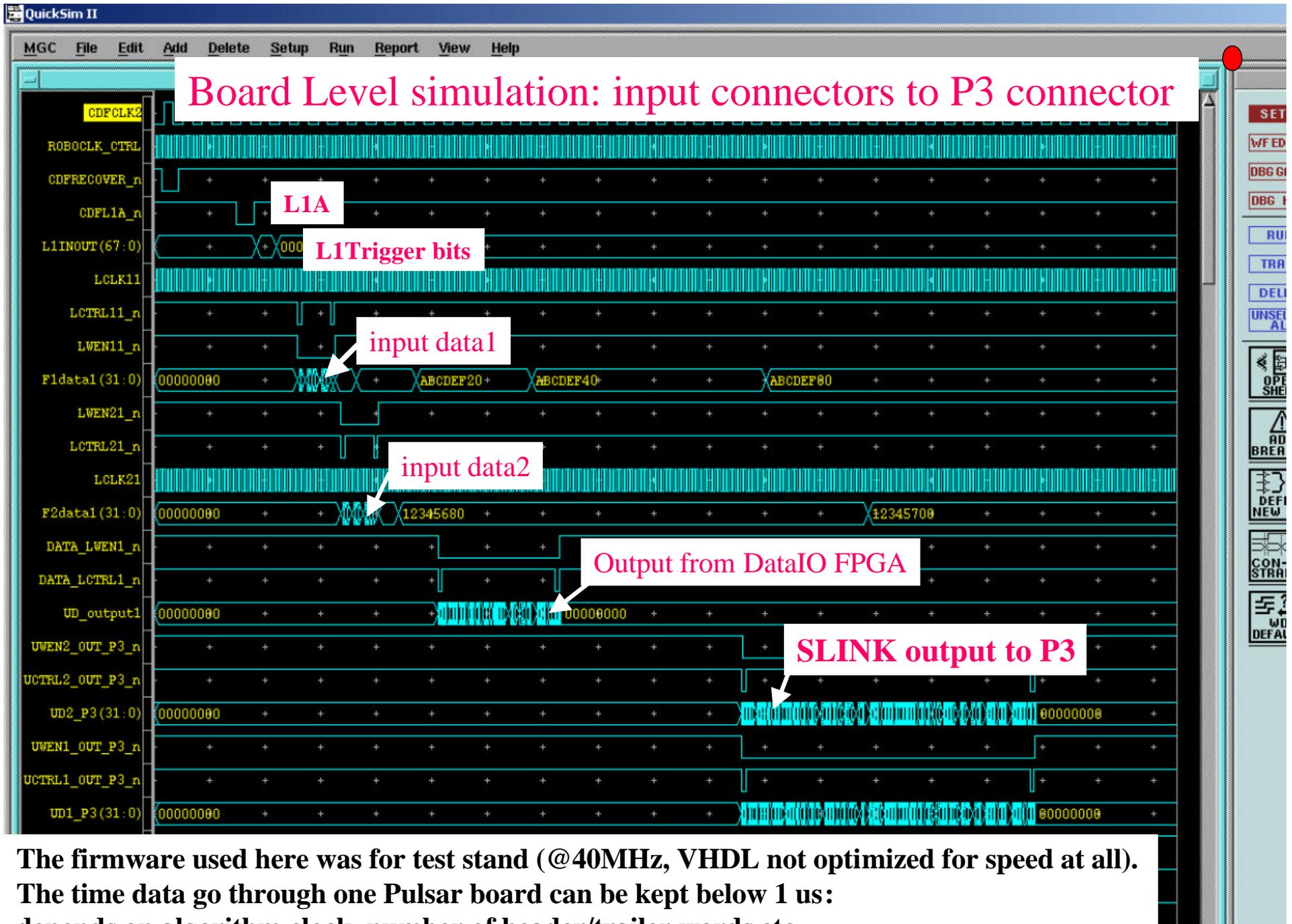
L1A buffer #

L1T bits

Format version #

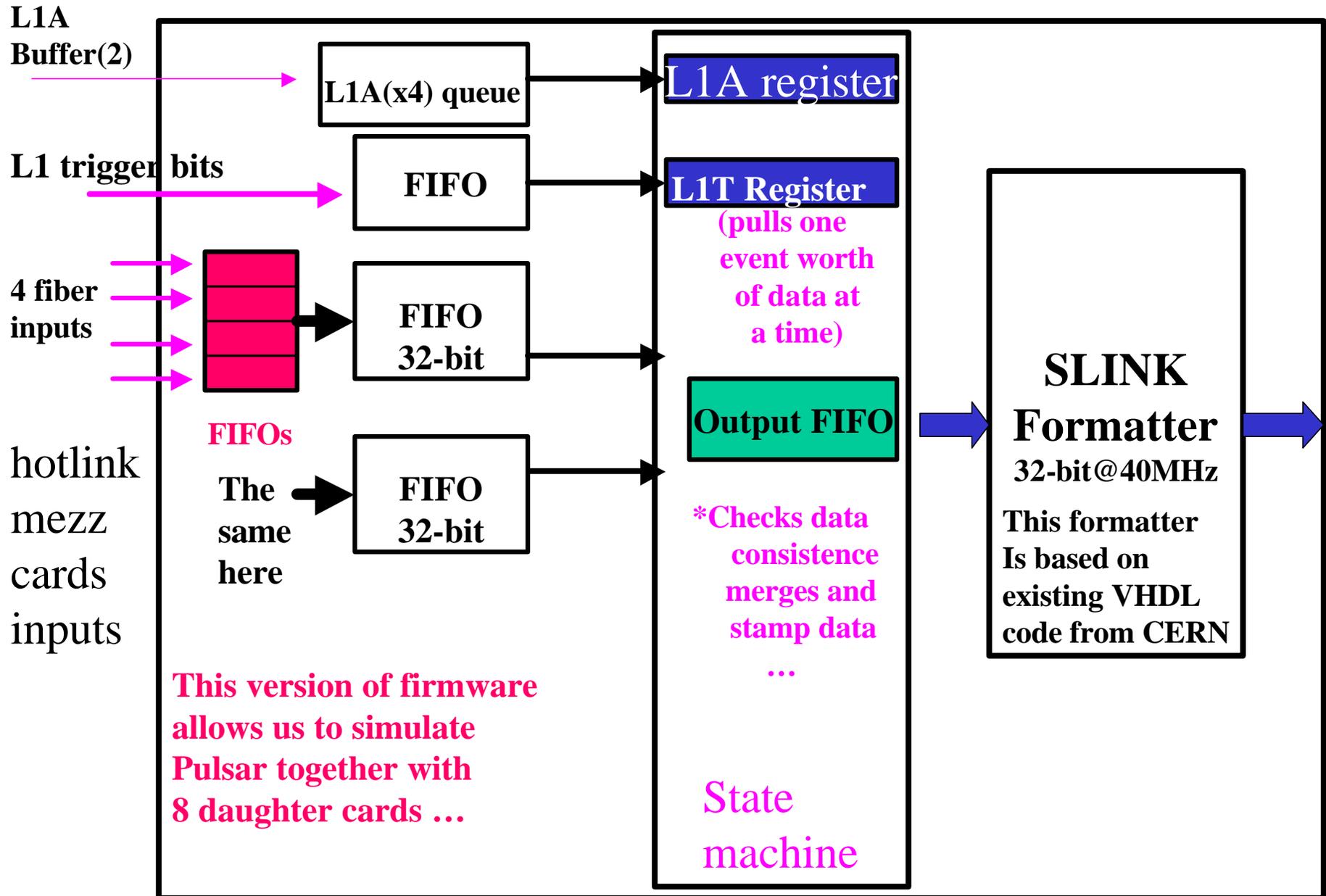
header size etc...

**Self-describing data bank format ...**



The firmware used here was for test stand (@40MHz, VHDL not optimized for speed at all).  
 The time data go through one Pulsar board can be kept below 1 us:  
 depends on algorithm clock, number of header/trailer words etc.

added hotlink mezzanine card inputs (the rest is the same)



# multi-board simulation

Rx mezzanine

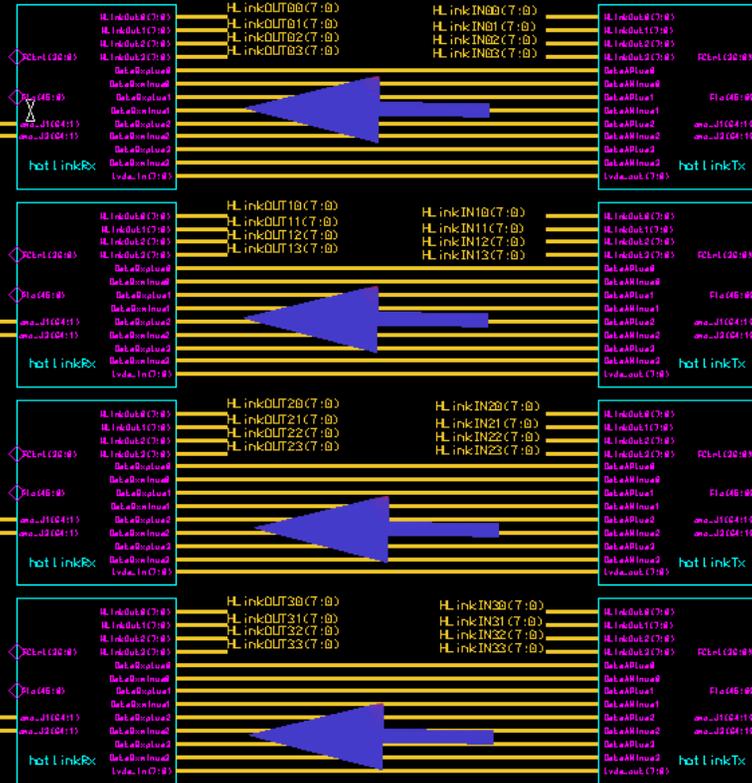
Tx mezzanine

inputs  
forced  
from  
here

Pulsar  
motherboard

VME\_P1<159:0>  
SO\_SPARE<24:0>  
SEINK2<10:46:0>

SLINK  
Output  
to P3

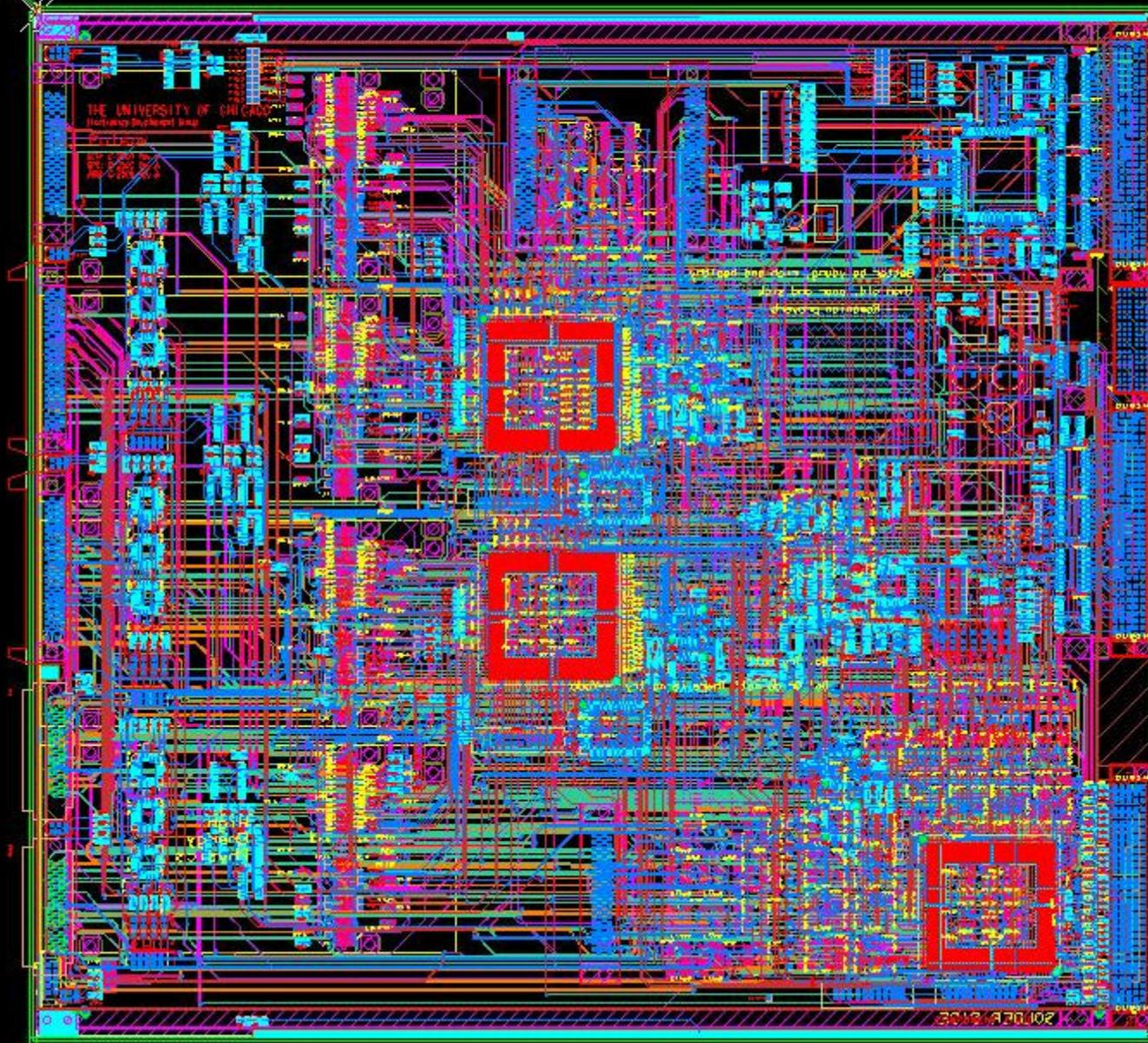


## Pulsar and his eight daughters

It takes about 1.5 GB memory to run the simulation on 2GHz/2GB PC.

Intensive board level simulation has been done:

[http://hep.uchicago.edu/~thliu/projects/Pulsar/Pulsar\\_hardware.html](http://hep.uchicago.edu/~thliu/projects/Pulsar/Pulsar_hardware.html)

Selected  
Delta: 13[http://hep.uchicago.edu/~thliu/projects/Pulsar/Pulsar\\_doc.html](http://hep.uchicago.edu/~thliu/projects/Pulsar/Pulsar_doc.html)

After Pulsar  
Mini-review:

Finished  
layout for  
all layers

Layout  
optimized  
with  
careful  
trace analysis

Order PCB  
Next week

## Firmware work needed for Pulsar:

assume **new people involved**

### **For Teststand:**

- core Tx code written for hotlink case and simulated in Quartus II. Can be modified to take care of other cases. Need 4 months (1FTE) more work to fully develop the Tx firmware for all cases;
- core Rx code written (Hotlink to SLINK, SLINK merger) with VME accesses, simulated at board level as well as at multi-board level. Can be modified easily to take care of Taxi cases as the core firmware is the same. Need 4 months (1FTE) more work to fully develop Rx (into a PC) firmware for all cases; need to define SLINK format for L2 (~ 1month).

### **For upgrade (baseline design):**

- Rx firmware in teststand case as good starting point. Need to implement Level 2 buffers for VME readout (~ 1 month work).
- simple muon data suppression code (~ 1 month)
- muon-track matching firmware (~2month)

## **If Pulsar is chosen for Run2b upgrade, what will the schedule look like **if we have 4 FTE?****

- **Sept-Dec 2002 (if we have 4 FTE):**  
finish all mezz/Aux cards, Pulsar prototype testing, Rev B if needed  
SLINK to PCI software work, teststand software,  
additional firmware work for testing ALL basic functionalities of prototypes
- **Dec-July 2003 (if we have 4 FTE):**  
commission L2 teststand for each data path and for existing system,  
able to both source and sink (to a PC) for each data path,  
advanced teststand software/firmware, high speed SLINK to PCI software,  
define CDF L2 SLINK format for all data pathes;prove of principle tests
- **Aug.-Dec.2003 (if we have 4 FTE):**  
use teststand to fine tune receiver firmware for each data path; system integration  
at crate level with test stand; L2 code testing for new system.
- **Jan.-July 2004 (if we have 4 FTE):**  
use teststand to drive the new system in standalone mode, study/optimize the  
performance, request test runs with actual beam...

## Initial Pulsar test stand cost estimate (without contingency)

Items	Prototype + Pre-production
Pulsar	\$50K
Mezz cards Aux cards	\$30K
Engineer	\$50K
SLINK	\$20K
Total	\$150K

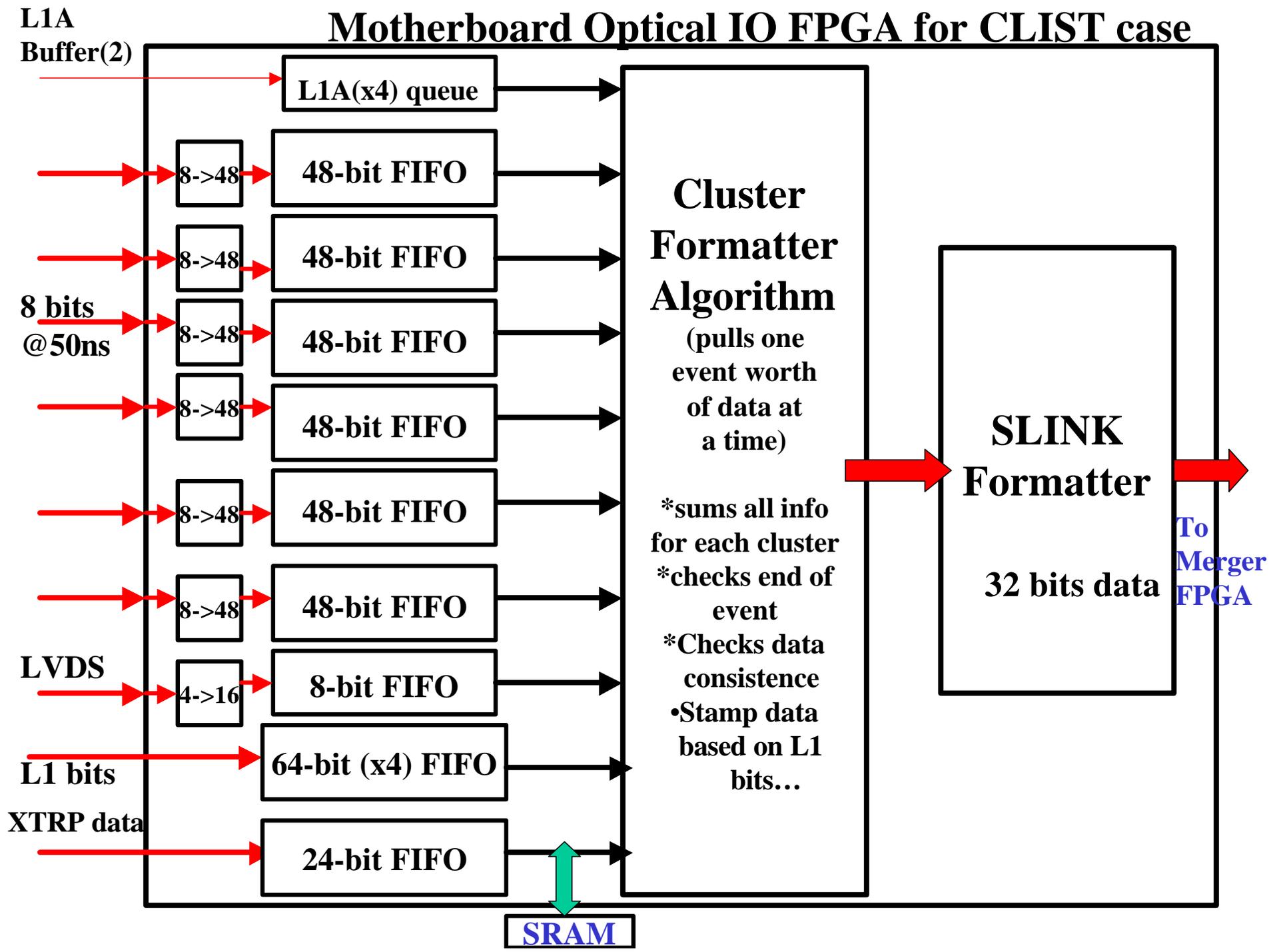
**The numbers are rough estimates, will have more accurate numbers soon. we are now getting quotes for PCB and assembly for Pulsar prototypes ...**

# In Summary...

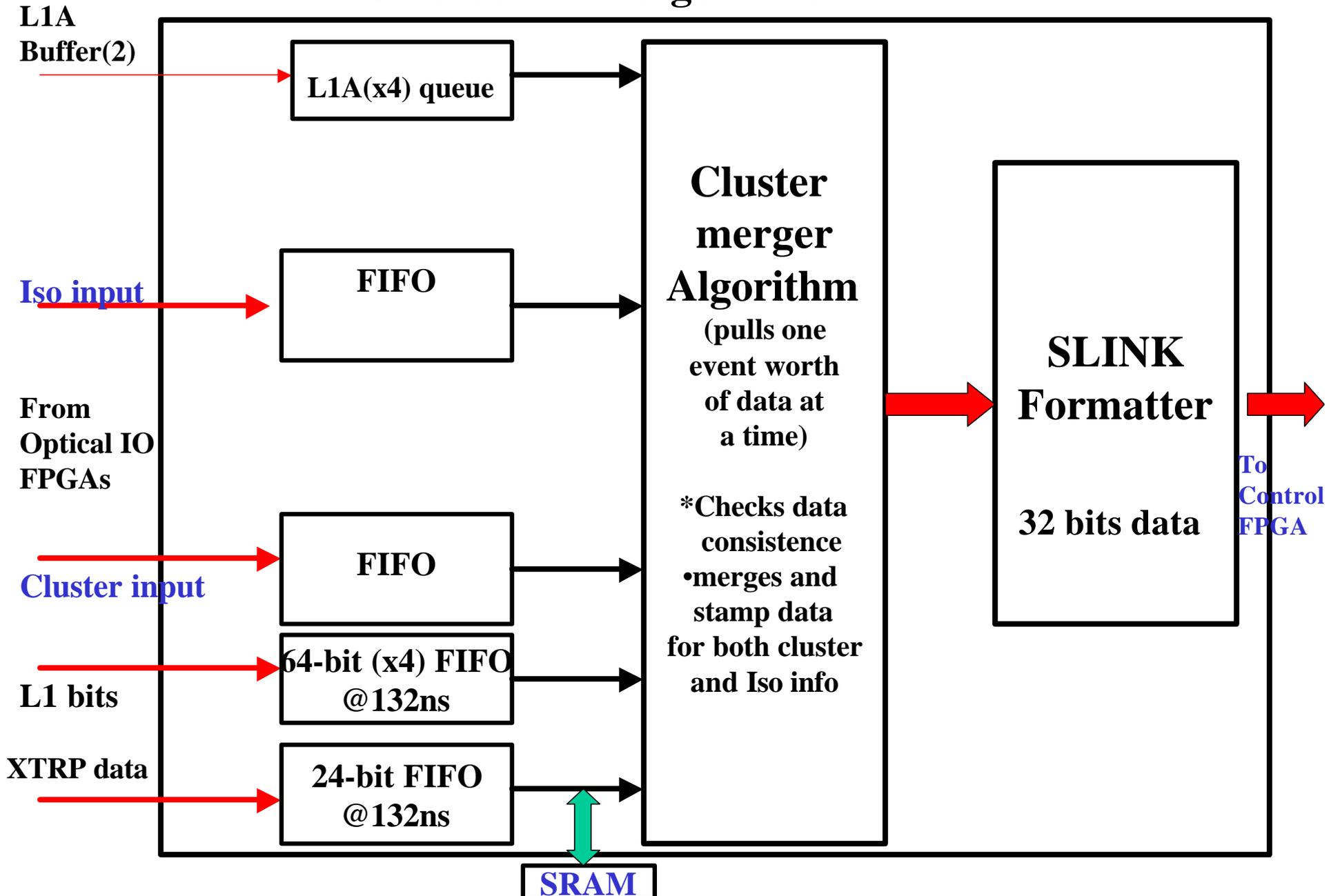
- **Pulsar design is powerful, modular and universal**
  - **Distributed processing motivated by physics**
    - **Sufficient safety margin in raw bandwidth and flexibility at both board- and system-level to handle unexpected Run II B challenges**
  - **Can be used as a test-stand as well as an upgrade path**
    - **Suitable to develop and tune an upgrade in stand-alone mode**
      - **Reduces impact on running experiment during commissioning phase**
      - **Pulsar is useful for Run II A maintenance and *any L2 upgrade***
    - **Built-in maintenance capability**
- **System relies heavily on commercial resources**
  - **Only one custom board**
    - **Firmware design is similar in all FPGA's on all incarnations of the Pulsar**
    - **Easier long-term maintenance**
  - **The rest is commercially available**
    - **Easily upgradeable CPU, PCI boards ...**
  - **Widely used link – well documented LHC standard**
    - **Knowledge transferable to *and from* LHC community**

**Some backup slides ...**

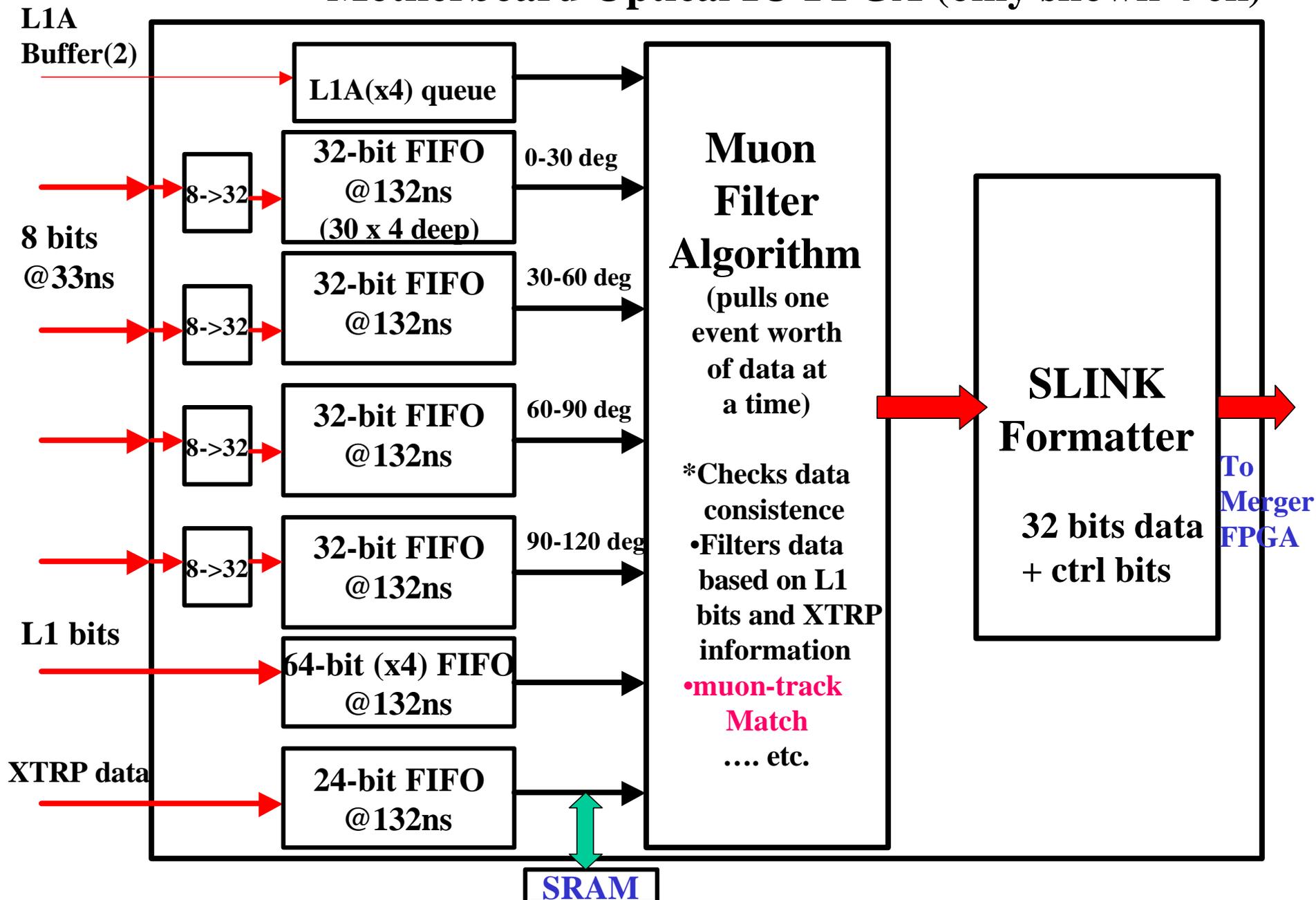
# Motherboard Optical IO FPGA for CLIST case



# Motherboard Merger FPGA



# Motherboard Optical IO FPGA (only shown 4 ch)



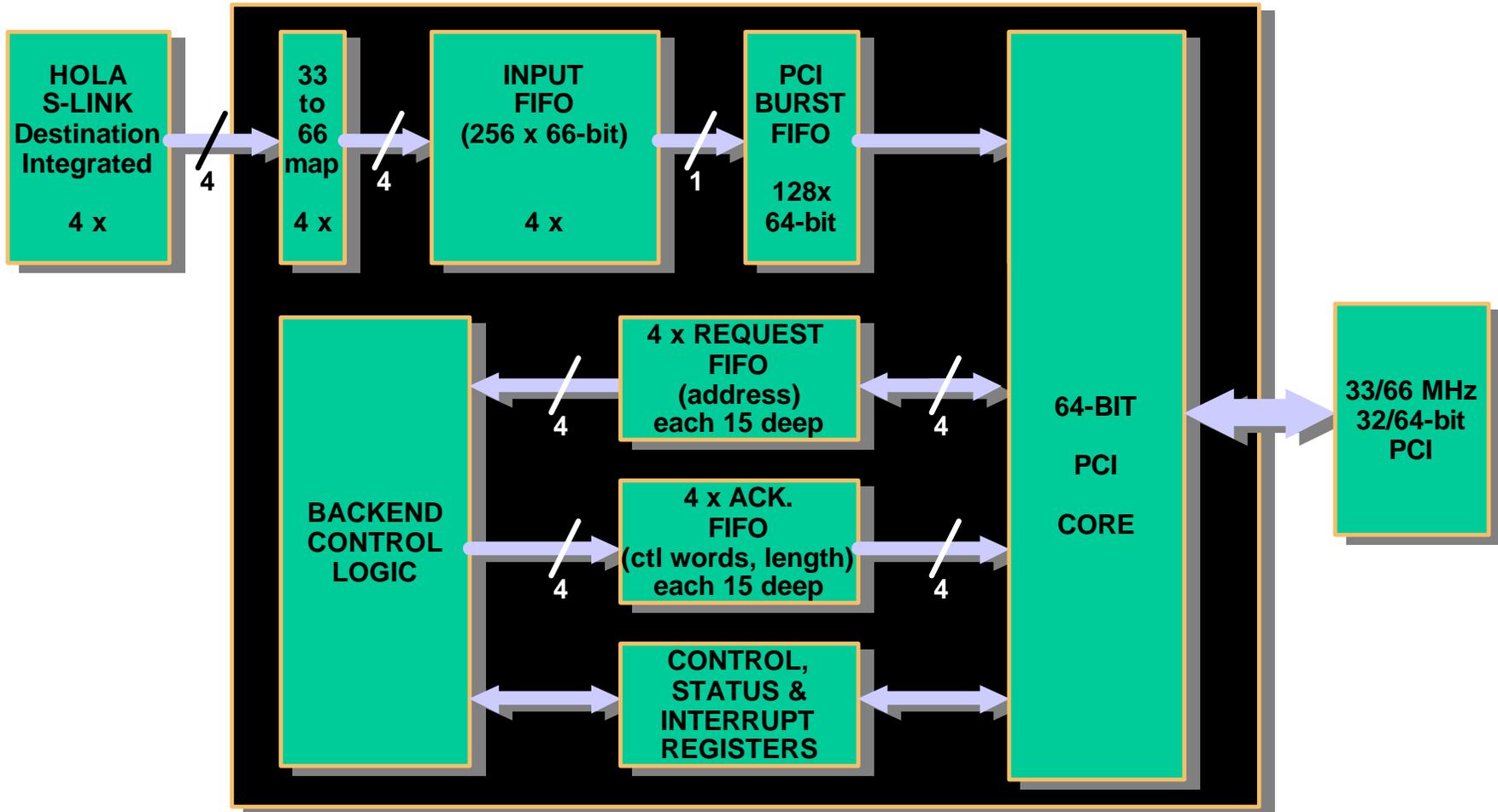
Displacement	Data Description
P0	Header Word
P0+1	FPGA Version Numbers
P0+2	Matchbox Output Bits
P0+3	CMU East - High Pt Bits
P0+4	CMU East - Low Pt Bits
P0+5	CMU West - High Pt Bits
P0+6	CMU West - Low Pt Bits
P0+7	CMX East - High Pt Bits
P0+8	CMX East - Low Pt Bits
P0+9	CMX West - High Pt Bits
P0+10	CMX West - Low Pt Bits
P0+11	CSX and HAD East Bits
P0+12	CSX and HAD West Bits
P0+13	BMU East - High Pt Bits
P0+14	BMU East - Low Pt Bits
P0+15	BMU West - High Pt Bits
P0+16	BMU West - Low Pt Bits
P0+17	BSU East Bits
P0+18	BSU West Bits
P0+19	TSU and HAD East Bits
P0+20	TSU and HAD West Bits
P0+21	Spare Word A
P0+22	Spare Word B
P0+23	TOF Bits
P0+24	CMU East Diagnostic Bits
P0+25	CMU West Diagnostic Bits
P0+26	Eta Gap Diagnostic Bits
P0+27	CMX East Diagnostic Bits
P0+28	CMX West Diagnostic Bits
P0+29	IMA East Diagnostic Bits
P0+30	IMA West Diagnostic Bits
P0+31	IMB East Diagnostic Bits
P0+32	IMB West Diagnostic Bits
P0+33	XTRP CMU Map Bits
P0+34	XTRP CMX Map Bits
P0+35	XTRP BMU Map Bits

Muon data as an example.

Each word is 24 bits (sent as 4 8-bit words over hotlink).

•  
can pack them in 24 bits similar to muon bank format, and use the other 8 bits to stamp other information ...

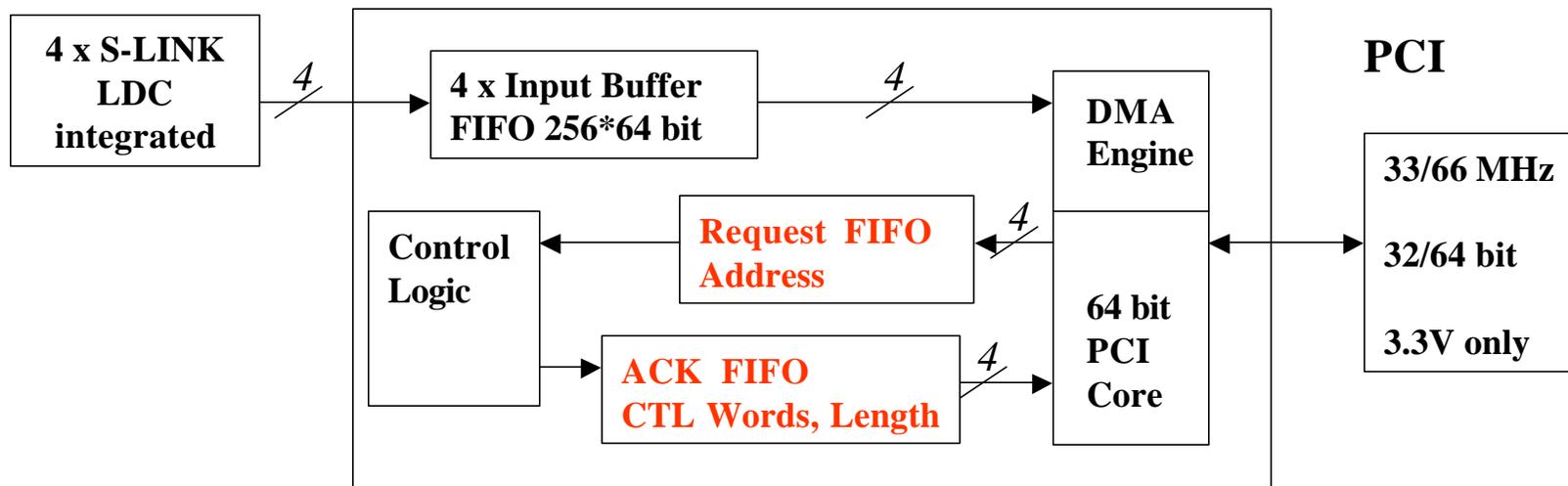
Next generation: **FILAR** (Quad HOLA S-LINK to 64-bit/66 Mhz PCI interface)  
Prototype will be built early 2003.



# 4S32PCI64 QuadIn

- Software and hardware design effort minimal
  - Same programming model as S32PCI64
    - optimised even more: only 4 instead of 6 PCI accesses needed per event (fixed buffer size, read ACK FIFO gives also empty status)
  - all hardware components exist, just integration

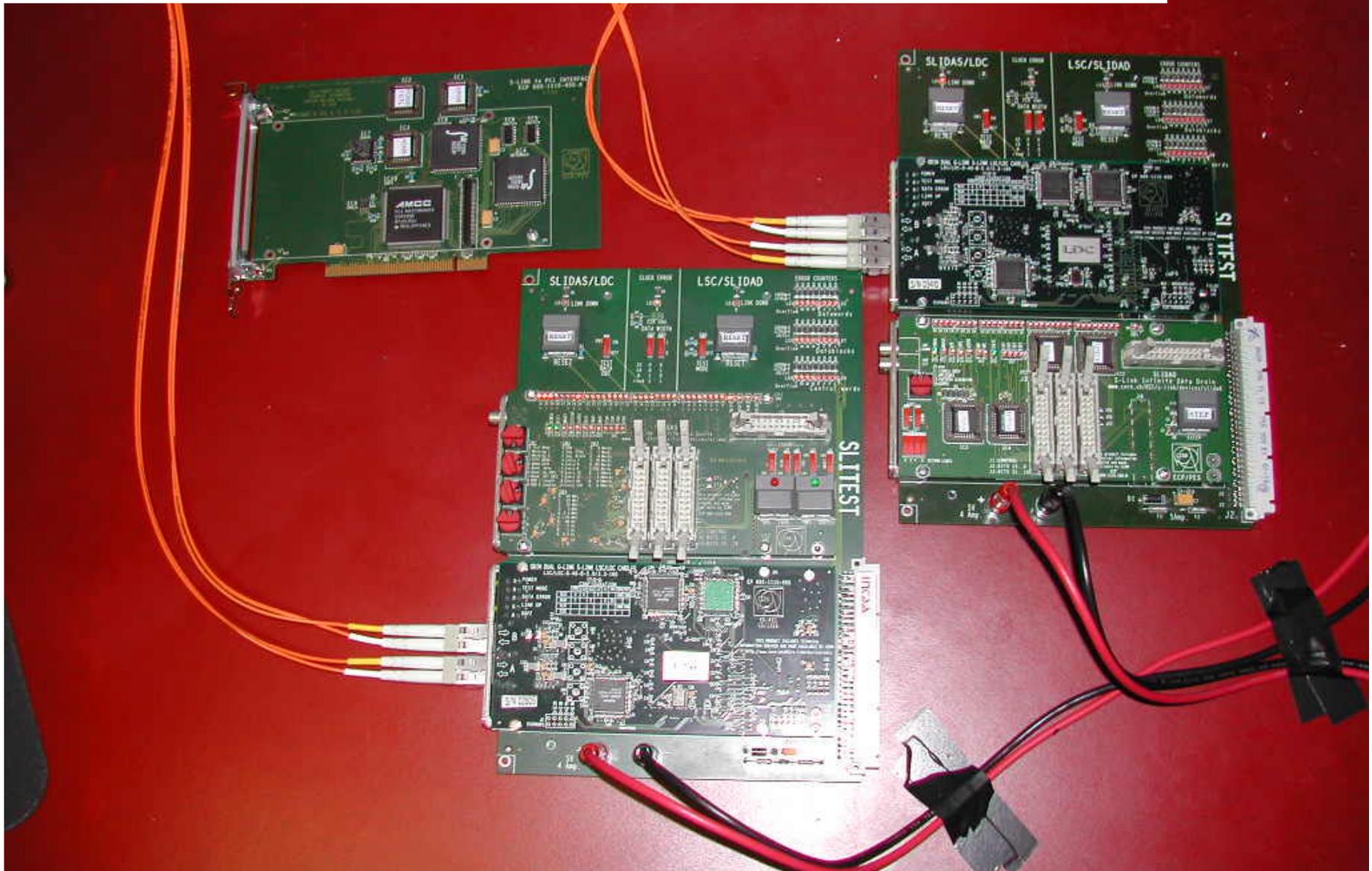
## 4S32PCI64 block diagram



**SLINK teststand in 163-C: picture taken by Peter Wittich**



**SLINK teststand in 163-C: used by summer student Derek Kingrey**



## Summary:

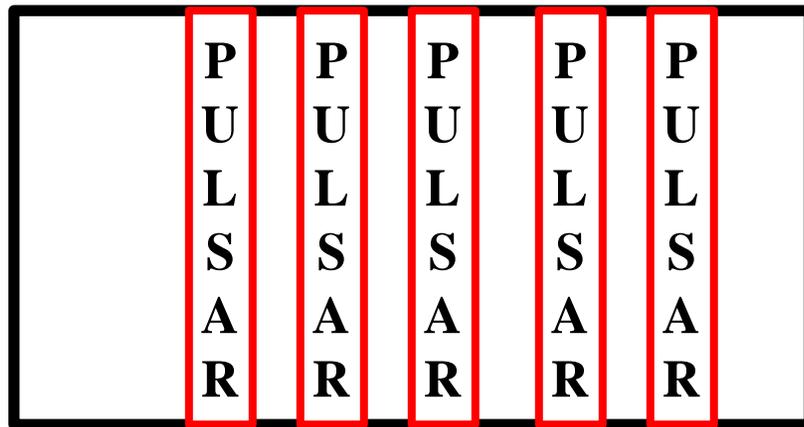
- Pulsar is designed primarily as a teststand tool for CDF Level 2 trigger system;
- The design is general enough that it can be used in many applications:
  - (1) as teststand tool for CDF L2 decision crate and SVT system;
  - (2) as spare interface board for CDF L2 decision crate;
  - (3) as possible upgrade path for L2 decision crate;
  - (4) as general purpose DAQ/trigger diagnostics tool:  
can source/sink/spy LVDS/Hotlink/Taxi/G-LINK etc.
  - (5) as simple DAQ system (such as in a test beam environment)

The flexible design (“lego style”) makes it possible to use Pulsar as a general purpose tool (within and outside CDF)

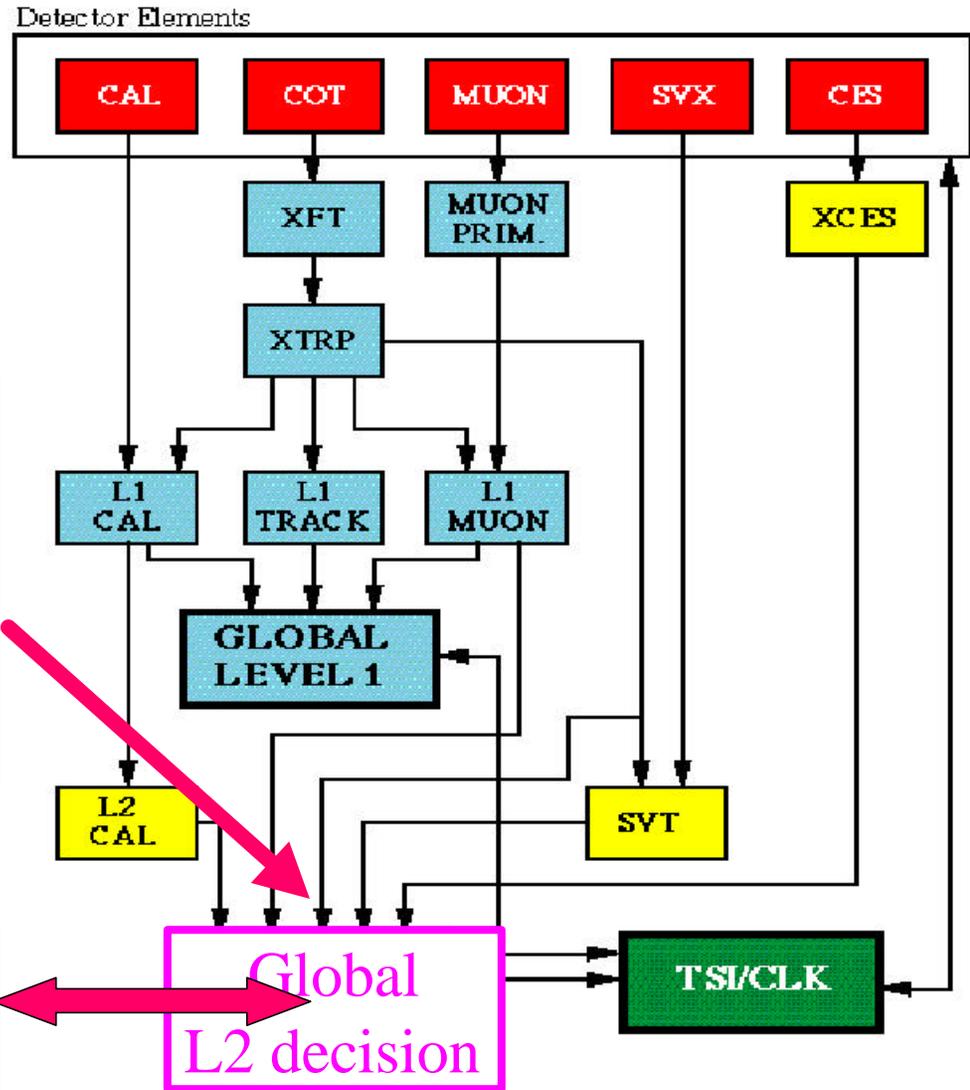
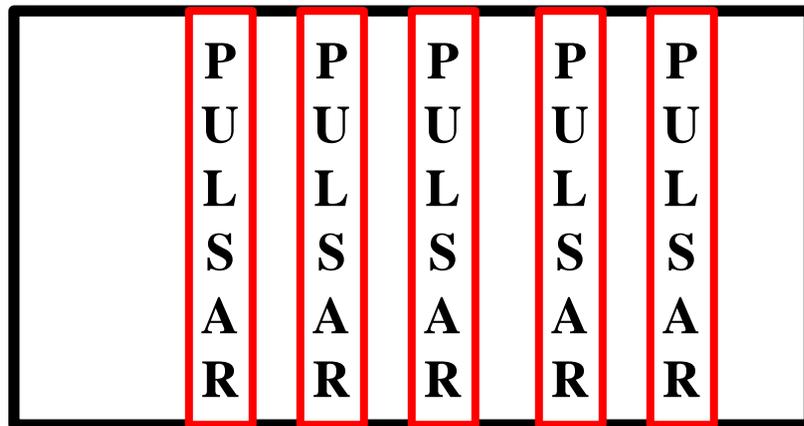
Future expansion is as cheap/fast/easy as designing a mezzanine card,  
the rest are commercially available(SLINK + PC)

# What Pulsar can do for L2?

(1) As Pulser and Recorder



(2) As a candidate for Run2b



<http://fizzie.uchicago.edu/~thliu>

**/Pulsar/Pulsar\_L2review\_Aug01.ppt**

<http://fizzie.uchicago.edu/~thliu>

**/Pulsar/Pulsar\_L2review\_Aug01.prn.pdf**