14.3.2000
FTT-L3-Meeting

Jürgen Naumann
Universität Dortmund

Fast Track Trigger

Level 3

- Physical Guidelines
- General Concept & Demands
- Ideas for Solutions
- Summary

Physical Guideline

Main purpose of FTT-L3:

Triggering on resonances in particle decays

"Golden Channel": D-Meson decay

$$D^{*+} \rightarrow D^0\,\pi^+_{slow} \rightarrow (K^-\,\pi^+)\,\pi^+_{slow}$$

Typical selection:

- Cut on $m(K\pi) - m(D^0)$
  $\longrightarrow D^0$-Decay
- Cut on $\Delta m = m(K\pi\pi_{slow}) - m(K\pi)$
  $\longrightarrow D^*$-Decay

$\Rightarrow$ Test of possible track combination scales with

- n^2_{track} for D^0-Identification
- $n^2_{track} + k \cdot n_{track}$ for D^*-Identification

Possible expansions:

- Vector mesons:
  Calculate momentum transfer t from electron information
  $\rightarrow$ Electromagnetic LAr, SpaCal, (eTagger)
- $J/\Psi, Z, \ldots$:
  Search for Muons, Correlate tracks with Muon-detector informationen
  $\rightarrow$ Instrumented Iron, Forward Muon-system
- Exotics:
  Compare energy from charged and neutral particles $\rightarrow$ Hadronic LAr

$\Rightarrow$ Level1-Informationen from PQZP/QBus-System

General Concept & Demands

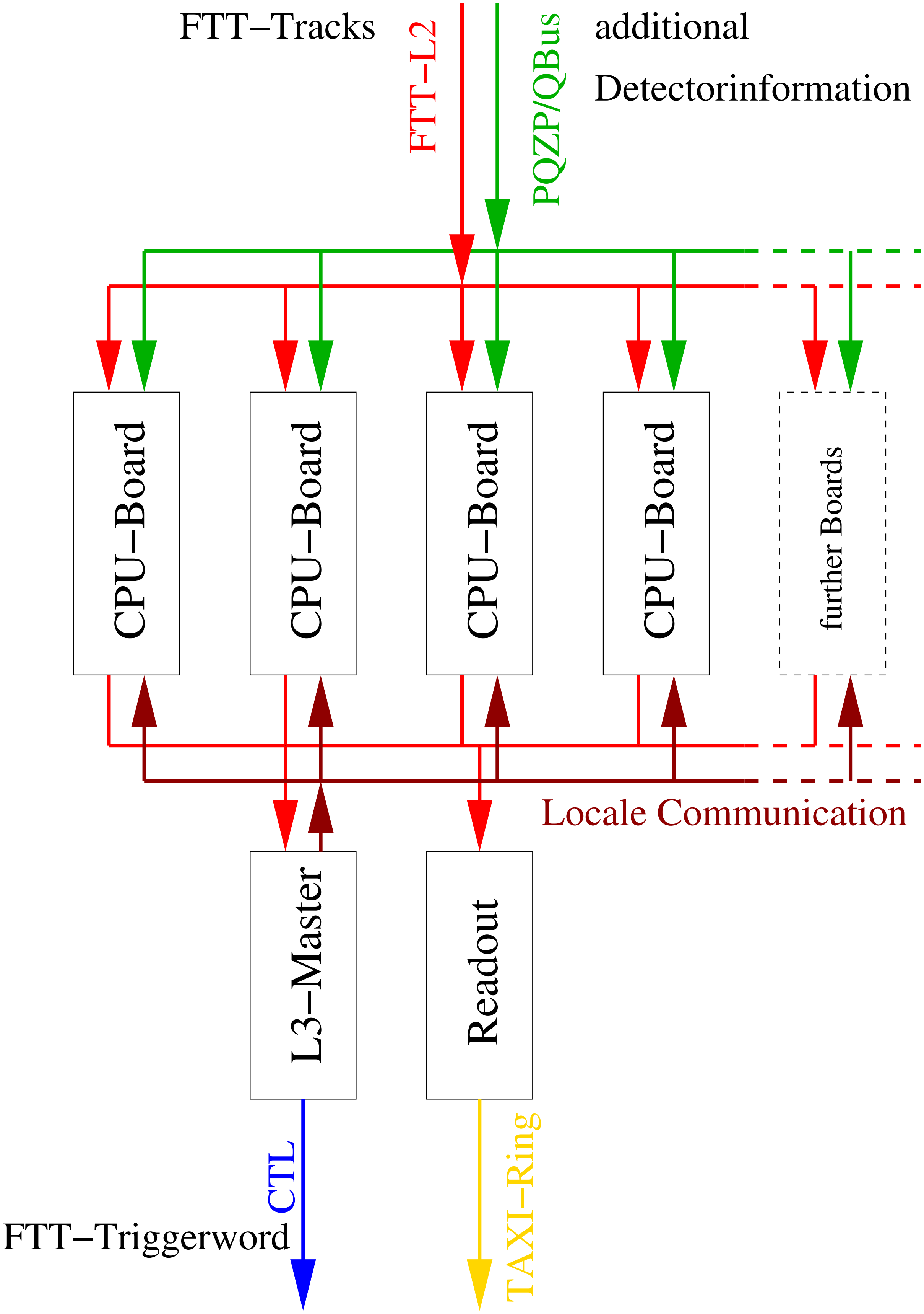
Master-Plan:

One CPU per Decay-Channel + common Infrastructure

$$\left.\begin{array}{r}\text{Programmable}\\ \text{Modular}\\ \text{Scalable}\end{array}\right\} \Rightarrow \text{Flexible}$$

Latency time for trigger decision if FTT should be a real benefit:

$$\approx 100\mu s$$



FTT–Tracks
FTT–L2
PQZP/QBus
additional
Detectorinformation
CPU–Board
CPU–Board
CPU–Board
CPU–Board
further Boards
Locale Communication
L3–Master
Readout
CTL
FTT–Triggerword
TAXI–Ring

→ Receiving FTT-L2-Data (+ PQZP/QBus)
- 50 32bit-Words from FTT-L2 (200 Byte)

⇒ FTT-L2-L3-Connection?
- Optional ∼ 2000 Byte via PQZP/QBus

  ⇒ Receiving-Units (also needed for FTT-L2)

  ↑ Transmission starts already at L1-Keep

→ Writing Data in RAM of CPUs

↓ In 10-20 μs

⇒ Locale Bus System

⇒ High Transmission rate

⇒ Parallel writing

$\rightarrow$ Performing calculations

$\downarrow$ In $t \lesssim 100\mu$s

$\Rightarrow$ Fast CPUs

$\Rightarrow$ (Real-time-) Operation-system?

$\rightarrow$ Collecting results

$\uparrow$ Only 1 bit per CPU?

$\Rightarrow$ Locale Bus System

$\Rightarrow$ L3-Master

$\rightarrow$ Building Trigger-Word and transmission to CTL

$\Rightarrow$ L3-Master

$\rightarrow$ FTT Readout

$\uparrow$ Some 100 μs

$\Rightarrow$ VME-Backplane for R/O- and TAXI-Card

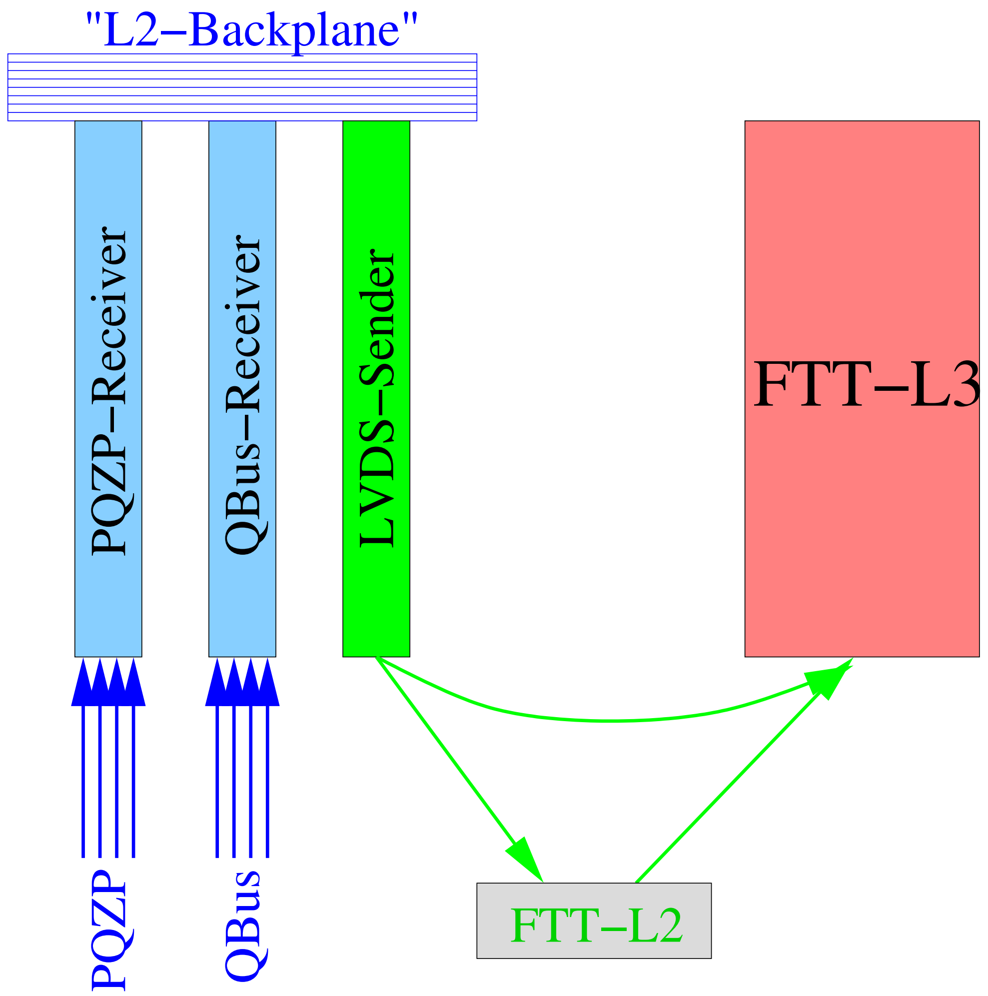
Ideas for solutions

Receiving Data:

- FTT-L2→L3:
  - Common FTT-L2/L3 Backplane
    - ∗ Decision Card of FTT-L2 in Backplane with L3-CPU-Boards
    - ↓ Requires purpose built backplane for fast loading of RAMs
  - LVDS with purpose-written Protocol
    - ∗ FTT-L2-internal Communication with LVDS
    - ∗ Width: 48 bit
    - ↑ Maximum theoretical Transmission rate: 672 MByte/s
    - ↑ Sender/Receiver on the market

- PQZP/QBus→L3:
  - Receiver-Cards in FTT-L3-Backplane
    - ↓ Existing Receiver-Cards are 9u
    - ↓ Fast Backplane / RAM loading required
    - ↓ Difficult to use industrial CPU-Boards

– Short “L2-Backplane” and LVDS-Connection to CPU-related Receiver-Cards
  - ↑ PQZP-Data (zVertex) also needed for FTT-L2 ⇒ Common source for data
  - ↑ Existing PQZP/QBus-Receiver Cards and knowledge for Backplane (L2NN/L2TT)
  - ↑ Active LVDS-Daisy-Chain allows quasi-parallel data transfer with high rates
  - ∗ LVDS-Receiver: National Semiconductor or ALTERA FPGA



Loading RAMs:

- PMC Mezzanine Cards
  - Industrial Mezzanine Standard via local PCI-Bus
  - ↑ All industrial VME- and cPCI-CPU-Boards have PMC connector(s) on Board
  - ↑ Large number of PMC cards on the market
  - ↑ Compact system
  - ↓ Limited space on card
  - ↓ Hard to monitor the system
- Short cPCI Backplanes
  - Backplanes with 3-4 connectors each (LVDS-Receiver, CPU-Board, Connector for Monitoring (1 spare))
  - ↑ 3u possible
  - ↑ Enough space on receiver cards

Backplanes:

- for loading RAMs: cPCI
  - ↑ 3u possible
  - ↓ cPCI just starts to become standard
  - ↑ Easy to monitor
  - ↓ Different solution for Readout needed
- for Readout: VME
  - ↑ Well established in H1
  - ↑ VME also needed for TAXI Card
  - ↓ Whole Backplane just to read 1 bit?
  - ↑ Master↔CPU traffic also via backplane
  - ↓ Different solution for data transfer to RAMs needed

CPU-Boards:

- PowerPC
  - $\uparrow$ Widely used in H1 (L4-Farm for example)
  - $\uparrow$ 450-500 MHz CPUs available
  - $\uparrow$ Special H1 connection to CES: 18% off
  - – Motorola $\leftrightarrow$ CES
  - $\downarrow$ Expensive (RIO3 466MHz $\sim$ 16.000DM (incl. -18% for H1))
- Intel CPU
  - $\uparrow$ Up to 800MHz CPUs available
  - $\uparrow$ Cheaper (VMIVME 450MHz $\sim$ 7000DM)

Operation System:

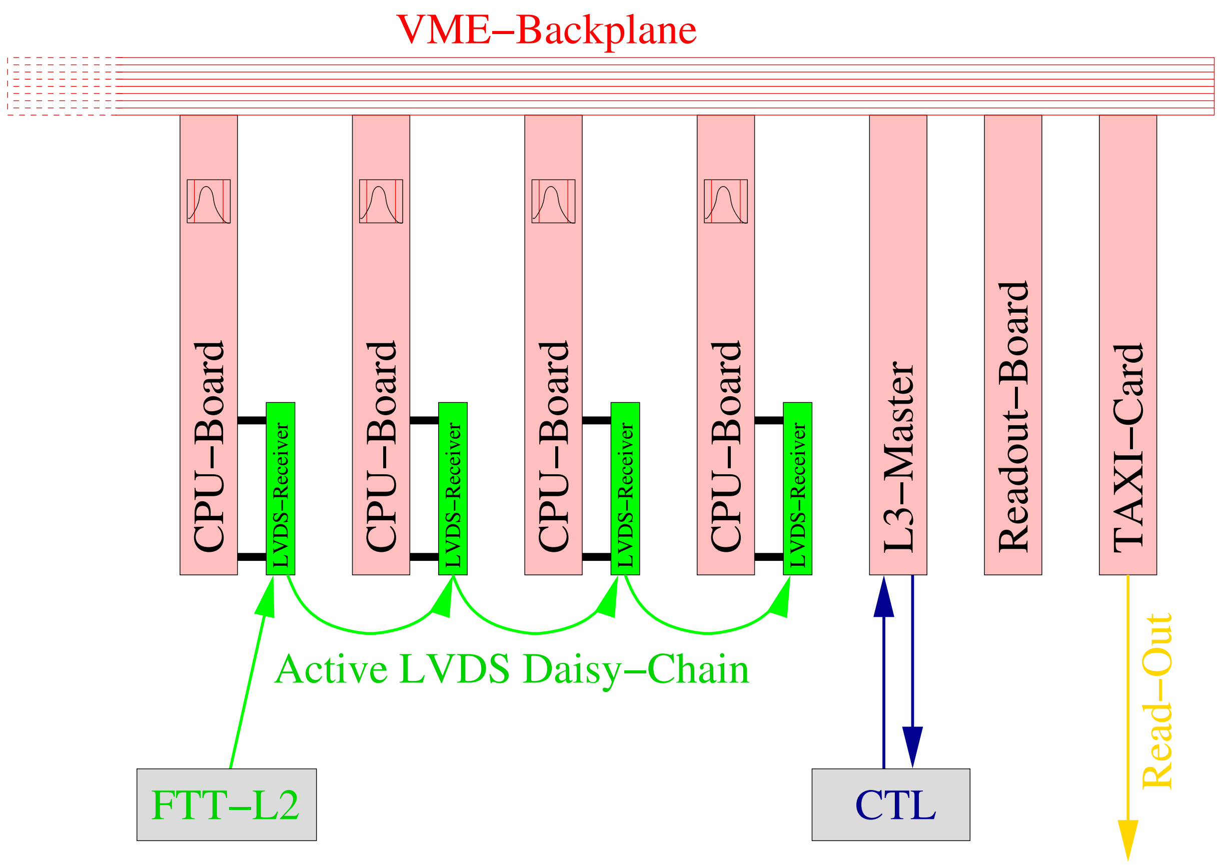
- None
  - Loading Binary from Server
  - ↑ No additional costs per Board
  - ↓ Restricted monitoring
- LynxOS
  - ↑ New “default system” for H1
  - ↓ For PowerPC only
  - Costs for licence?
- VXWorks
  - ↓ No experience
  - ↑ Runs on all platforms
  - Often found as “best solution”
- Industrial Linux = Real-time Linux
  - ↓ No experience
  - “Real-time OS running Linux in-between”
  - OS of the future?
- WindowsNT
  - Do you really want this?

Readout:

- Via L3-VME-Backplane
  - ↑ Well established in H1
  - ↓ Overdone (just 1 bit)
  - ↑ "Just plug in TAXI Card"
- Via single Wire
  - – Needed for cPCI-Solution
  - ↑ Enough for this purpose
  - ↓ Requires additional short VME Backplane
  - ↓ Requires additional local Ethernet for Master↔CPU communication

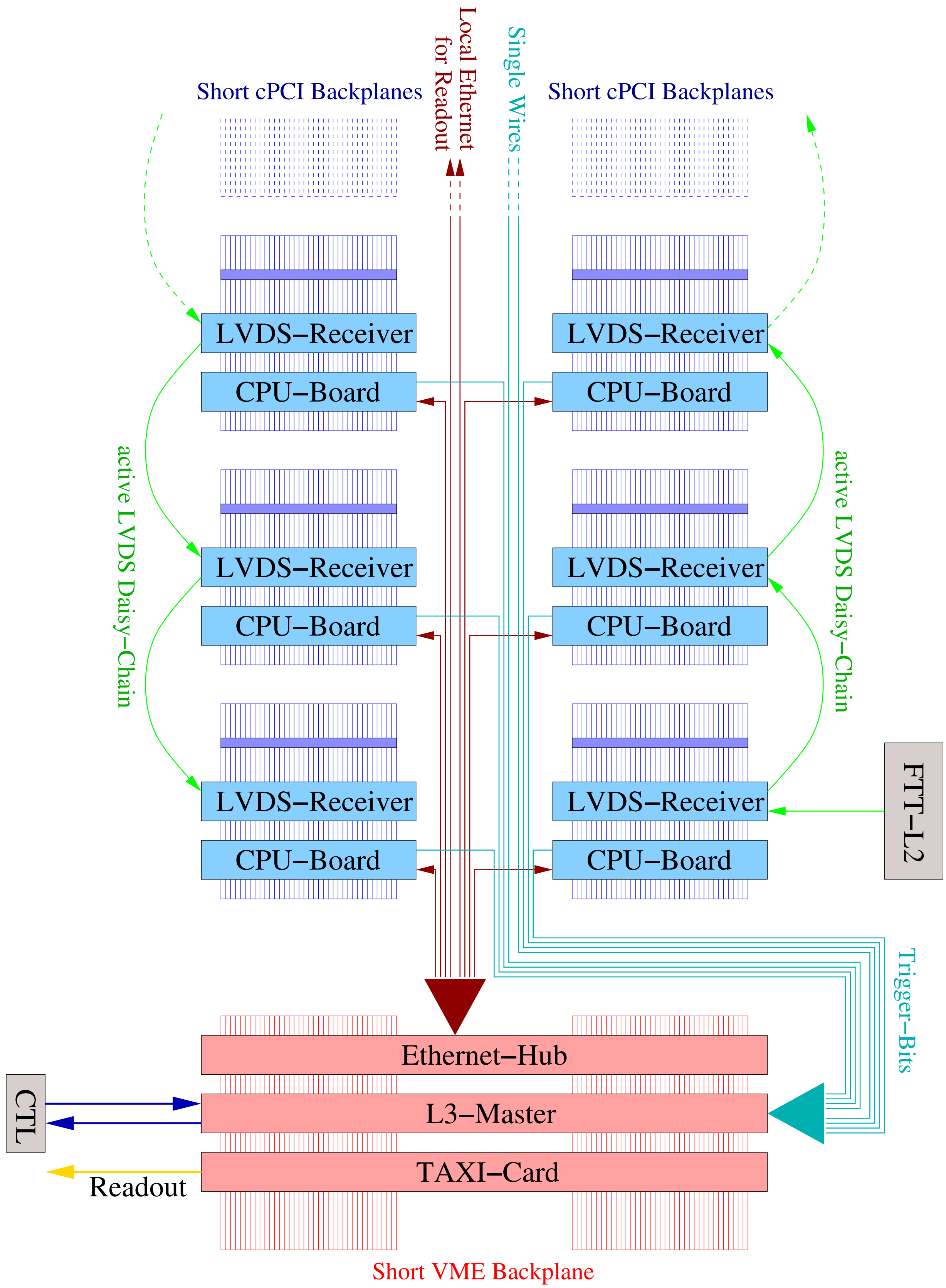
"VME-Solution":

- Data-Input via active LVDS Daisy-Chain
- PMC-Board for connection to RAM
- PowerPC-Boards in VME-Backplane
- Readout via VME-Backplane



“cPCI-Solution”:

- Data-Input via active LVDS Daisy-Chain
- cPCI-Board for connection to RAM
- Intel-CPU-Boards in local cPCI-Backplane
- Readout via single wires and short VME Backplane
- Local Ethernet for FTT-L3 internal communication



Local Ethernet
for Readout
Single Wires
Short cPCI Backplanes
Short cPCI Backplanes
LVDS–Receiver
CPU–Board
LVDS–Receiver
CPU–Board
LVDS–Receiver
CPU–Board
LVDS–Receiver
CPU–Board
LVDS–Receiver
CPU–Board
LVDS–Receiver
CPU–Board
active LVDS Daisy–Chain
active LVDS Daisy–Chain
FTT–L2
Trigger–Bits
Ethernet–Hub
CTL
L3–Master
TAXI–Card
Readout
Short VME Backplane

Summary

Main topics:

- Intel CPU $\leftrightarrow$ PowerPC
- VME $\leftrightarrow$ cPCI
- PMC $\leftrightarrow$ local Bus