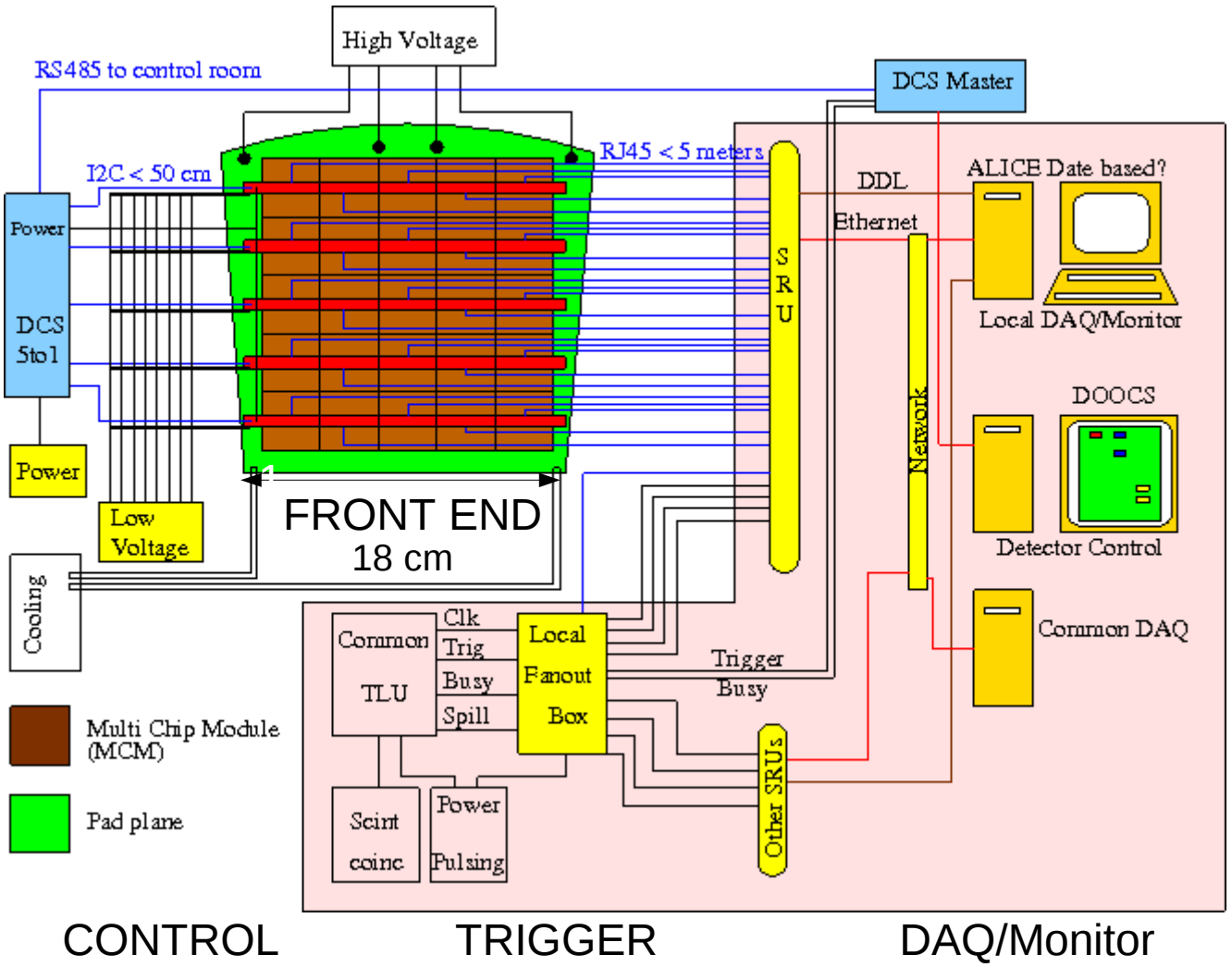
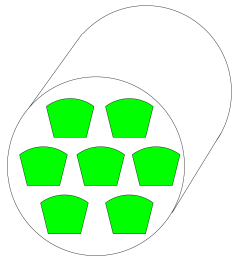
SALTRO TPC readout system

Presented by
Ulf Mjörnmark
Lund University

- Overview of current design
- The Front End Multi Chip Module (MCM)
- Low Voltage (LV) system
- Detector Control System (DCS)
- Scalable Readout Unit (SRU) & Trigger & Power Pulsing
- DAQ system
- Status & Time scale

Overview





LCTPC ENDPLATE

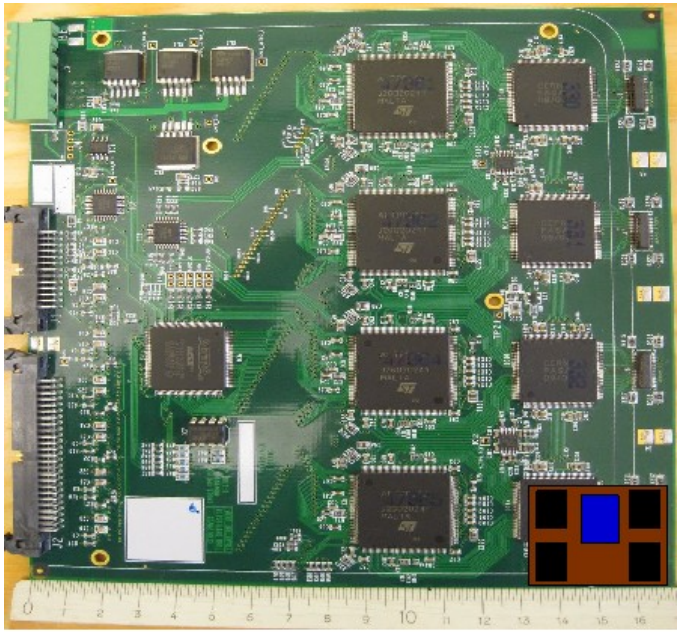
7 pad plane modules

Front End = 25 MultiChipModules connected on a pad plane = 3200 channels * 3 pad planes

SRU = Scalable Readout Unit (RD51)

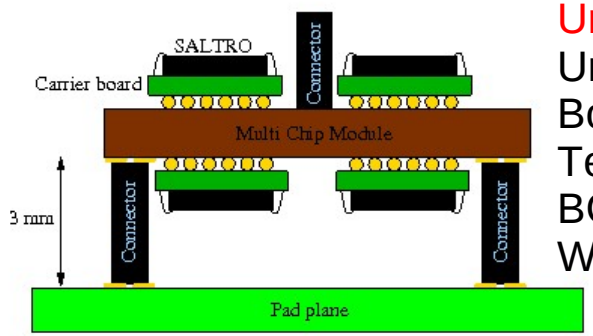
LUND UNIVERSITY

Multi Chip Module (MCM)



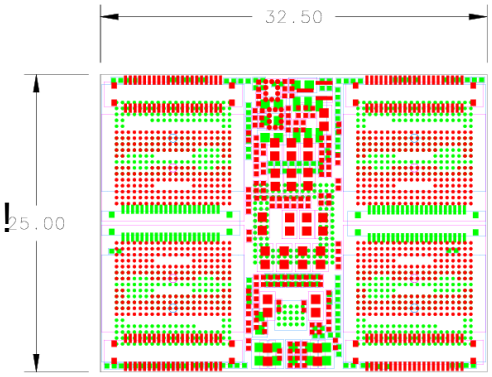
Current:
FEC = 8 ALTRO + 8 PCA16
128 channels
Parallel readout bus
17*18 cm
FPGA board controller

Multi Chip Module (MCM) = 8 SALTRO
128 channels
Serial readout
2.5*3.5 cm
CPLD (firmware Brussels)
"same" except Low Voltage



Side view NOT TO SCALE

Untested &
Unpackaged chip
Bonded on Carrier
Test (unknown yield)!
BGA on main board
With industry

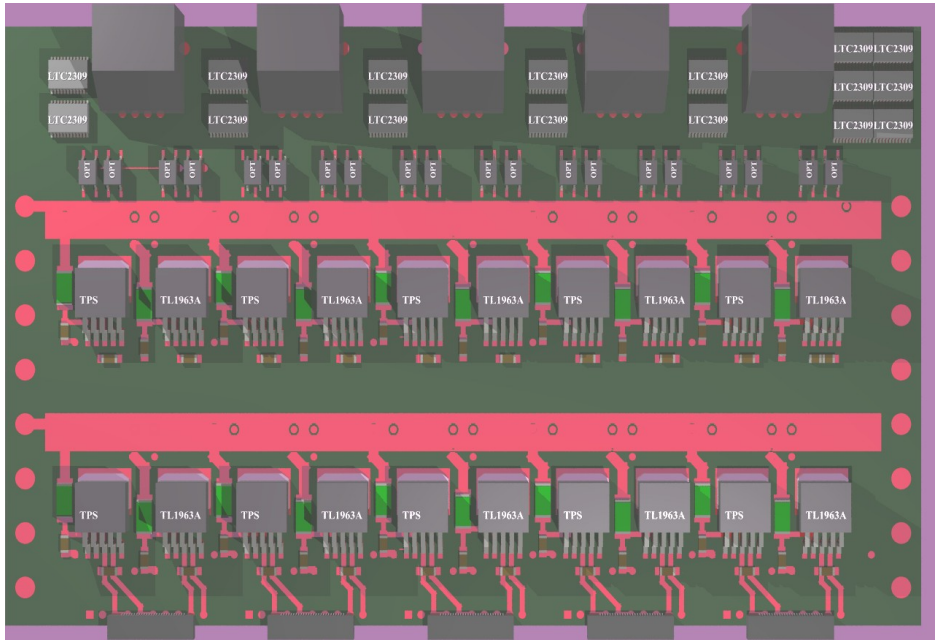


Component placement

LUND UNIVERSITY

Low Voltage Card

5 MCM * 8 Voltages = 40 regulators



MCM1 MCM2 MCM3 MCM4 MCM5

Control of individual regulator
On/off

Monitor on each Low Voltage card
48 voltages
40 currents
1 temperature

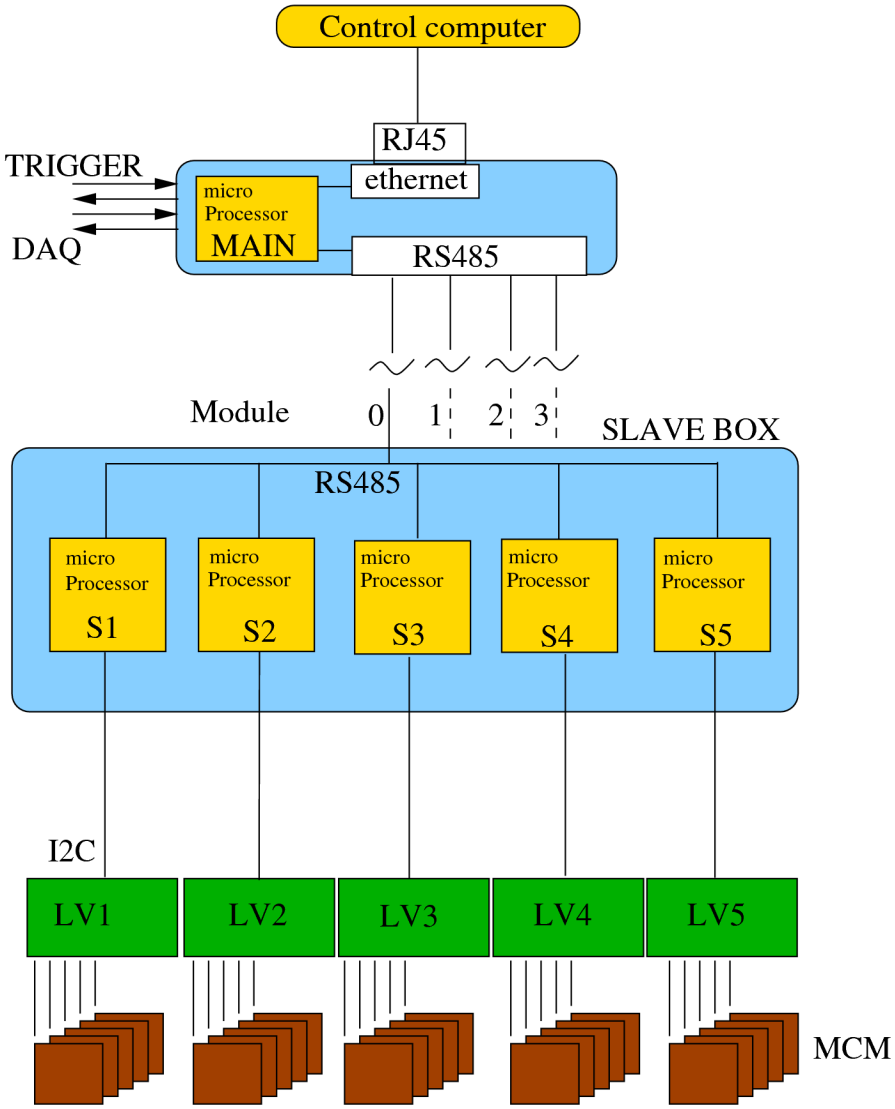
On each MCM
1 temperature
1 DAC

5 low voltage cards and
~700 settings/values per pad plane

NEED **D**etector **C**ontrol **S**ystem

LUND UNIVERSITY

Detector Control System



In control room

Display/Logging/Control:
DOOCS (used at DESY)
(Distributed Object Oriented Control System)

Disable Run if problem
STOP TRIGGER/DAQ/ALARM
How?

WANTED

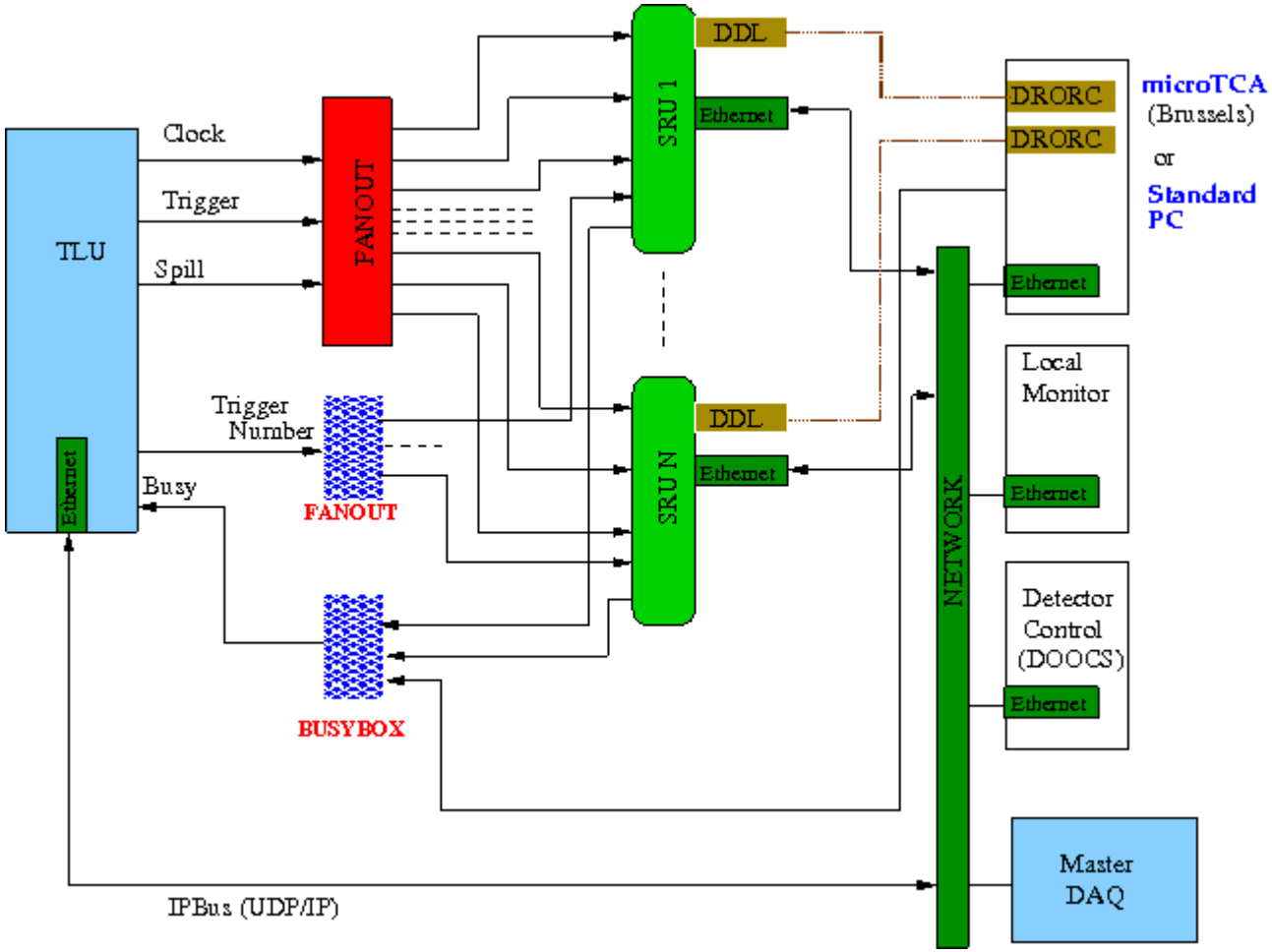
Near electronics

Standalone Monitoring
Automatic power off!

Low Voltage Board

370 W >>> power pulsing
110 W (regulators) cooling?
260 W CO2 cooling (Japan)

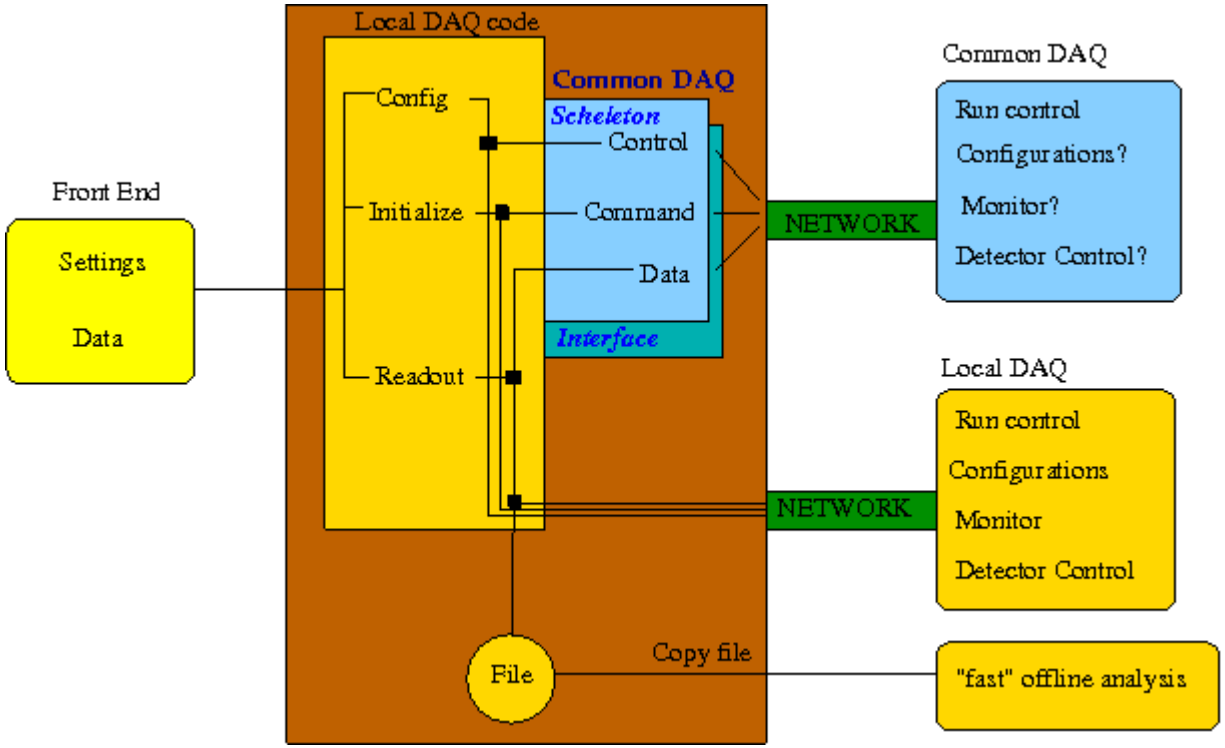
LUND UNIVERSITY



***DAQ system* – preliminary!**

LUND UNIVERSITY

Local - Common DAQ interface



Scheleton: provided by common DAQ, calls to user routines

Interface: well defined protocol, user implements code

WANTED

LUND UNIVERSITY

Trigger signal usage

Clock — Synchronous. Stability, jitter...? Frequency?
20/40 Mhz for sampling. 160 Mhz for data transfer

Trigger — Prompt, fanout to
→ SRU (and DAQ computer?), time stamp of event
→ sampled by Front End, for precise time when "particle passed the detector"

Busy — from SRUs and DAQ computer. SRUs can be daisy chained

Spill — for Power Pulsing (power off SALTRO outside spill)
Takes 100 microsec to power on! No triggers during that time!
Do one need a signal (hw/sw?) telling if system is in power pulsing mode, powered up/down?

Trigger number — distribute to SRUs. Broadcast 

LUND UNIVERSITY

Aim of a common Trigger & DAQ system

for AIDA and what comes after (AfterAIDA)

	Trigger	Run Control	Data files Common	Data files Merge	Monitor	Detector control
AIDA						
Common DAQ	yes	yes	(yes)	yes	-	-
Local DAQ						
Common	yes	yes	-	-	-	-
Local	-	yes	yes	-	yes	yes
AfterAIDA						
Common DAQ	yes	yes	yes	yes	yes	yes
Local DAQ						
Common	yes	yes	yes	-	yes	yes
Local	-	-	-	-	-	-

AfterAIDA => “standardized modular open source” Trigger & DAQ for all ILD activities (i.e. easy to compile/implement/understand/add code on all systems ….)

STATUS

Multi Chip Module

SALTRO - 600 untested naked chips exist
Carrier board - design ready and PCBs delivered
Industry collaboration for bonding/mounting
Prototype MCM to be produced (with one packaged SALTRO)
Firmware started to be developed in Brussels

Low Voltage board

Design ready
Prototype board is being tested

Detector Control System

Hardware - design ready and hardware made, tested and working
Firmware – in progress, mostly done
DOOCS – not yet done anything

Trigger/DAQ

Bought one SRU – sent to Brussel for tests and development of firmware
Trigger/Power pulsing/DAQ hard/software – adopt ALTRO EUDET DAQ: ***replace ALICE DDL with ethernet readout***. Issues: transfer rate, data size, trigger rate.
MicroTCA? Still to be defined and done

Time

2014: emphasis on electronics, software as needed for this
2015: full setup