

---

# SSD offline status

---

Enrico Fragiacomò

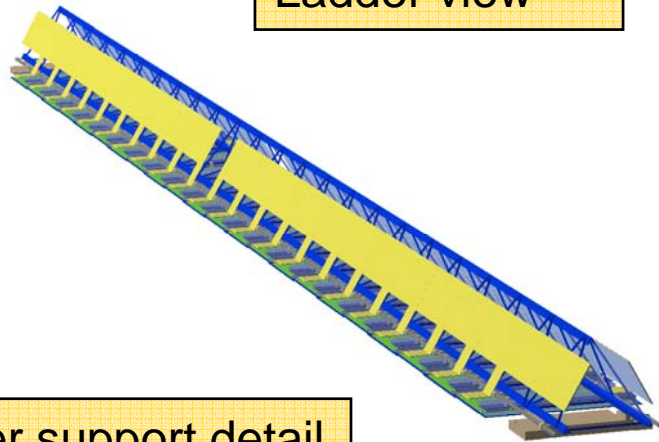
Offline week, 8-10-2007

# Project task report

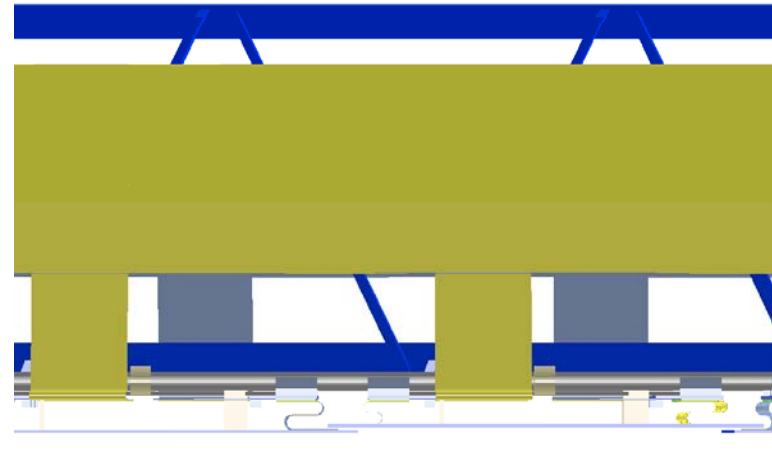
<b>Geometry &gt;&gt; Geometry as installed</b>	<b>done</b>
<b>Raw Data Format &gt;&gt; Provide DDL to equipment ID mapping</b>	<b>done</b>
<b>Raw Data Format &gt;&gt; Provide Geometrical Mapping</b>	<b>done</b>
<b>Raw Data Format &gt;&gt; Reconstruction from raw data</b>	<b>new (EF)</b>
<b>Alignment &gt;&gt; Format of Survey data and conversion into align. objects</b>	<b>Tbd (GN)</b>
<b>Event display &gt;&gt; Raw data display</b>	<b>Tbd (?)</b>
<b>Calibration &gt;&gt; SHUTTLE &gt;&gt; preprocessor algorithm implemented</b>	<b>done</b>
<b>Calibration &gt;&gt; User Requirements</b>	<b>Almost ready</b>
<b>Calbration &gt;&gt; decalibration strategy</b>	<b>Tbd (EF)</b>
<b>Calibration &gt;&gt; Offline &gt;&gt; Provide data quality control macro</b>	<b>Tbd (EF)</b>
<b>Quality Assurance</b>	<b>Tbd (EF)</b>

# Geometry

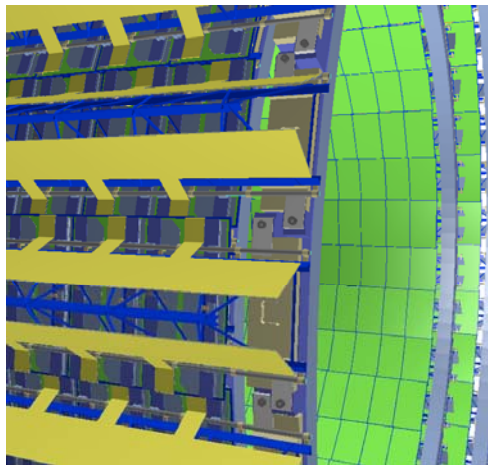
Ladder view



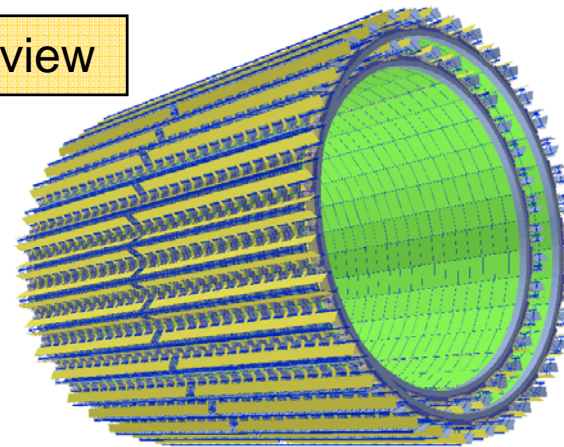
Ladder side detail



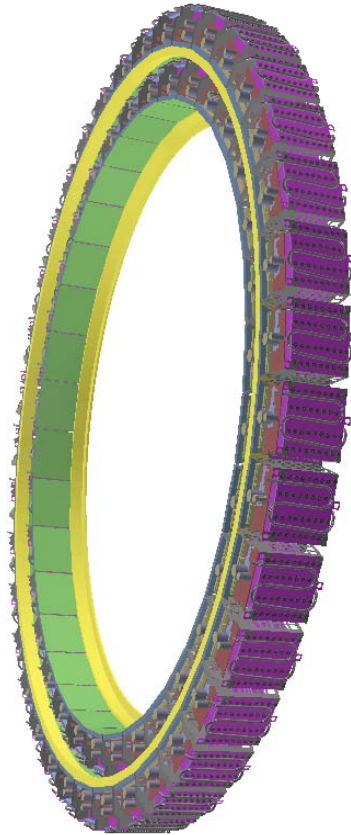
Ladder support detail



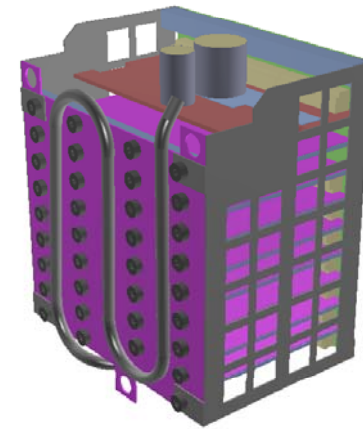
Layers view



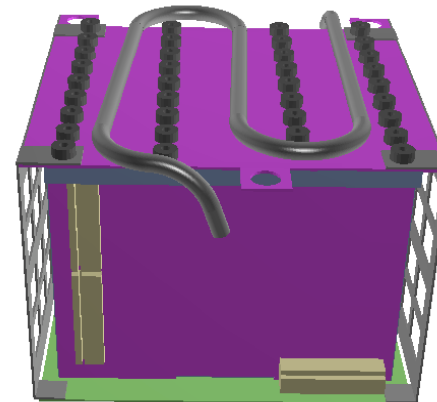
# Geometry



Endcaps and endcaps support



Endcap details



---

## Raw Data -> DDL map

- Coded in AliTStreamSSD (with the correct 32-bit decoding for rawdata)
- Revised on the basis of the latest HW map
- For each DDL associates AD/ADC to module index (500-2198)
- Both AliTDDLRawdata (in encoding) and AliTStreamSSD (in decoding) make use of the DDL map

---

# Raw Data -> Reconstruction

- Reconstruction from simulated data works successfully!
- **Reconstruction from real rawdata fails!**
- Present version of the V2 cluster finder for rawdata expects data sorted wrt modules (i.e. a pattern like mod1/strip1, mod1/strip2...mod2/strip1,mod2/strip2,...)
- The pattern of rawdata from the electronics is instead mod1/strip1, mod2/strip1, ... mod1/strip2, mod2/strip2 ...
- The largest level where rawdata are mixed up is the AD module (which reads 12 modules)
- Several solutions found:
  1. Sorting rawdata in LDCs
  2. raw2digit conversion (but would involve the whole ITS)
  3. **buffering rawdata for each AD (12x1536x32 bits) and select strips according to ADC index**
- **Work in progress (highest priority!)**

# Calibration parameters: source and use

Pedestal run	<ul style="list-style-type: none"><li>■ Noise</li><li>■ Pedestal</li><li>■ Common mode</li><li>■ Bad Channels</li></ul>	<ul style="list-style-type: none"><li>Noise in OCDB</li><li>Bad channel map in OCDB</li><li>Pedestal and common mode as ref. data</li></ul>
Special run?	<ul style="list-style-type: none"><li>■ Gain</li></ul>	<ul style="list-style-type: none"><li>Gain in OCDB</li></ul>
DCS	<ul style="list-style-type: none"><li>■ Temperature</li><li>■ Currents</li></ul>	<ul style="list-style-type: none"><li>In OCDB as ref. data</li></ul>

---

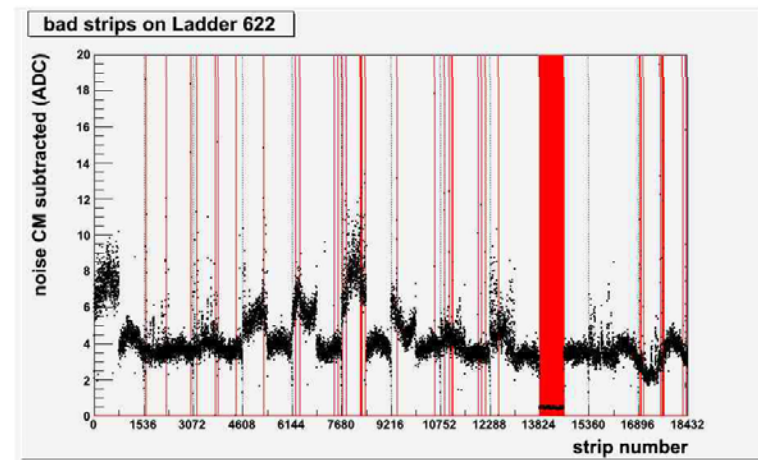
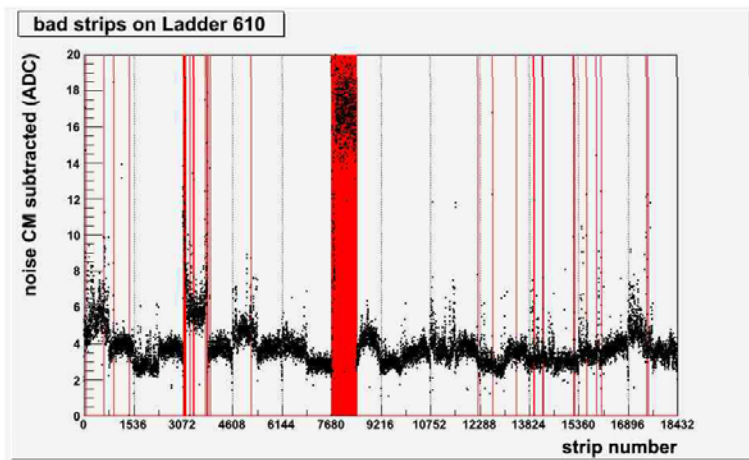
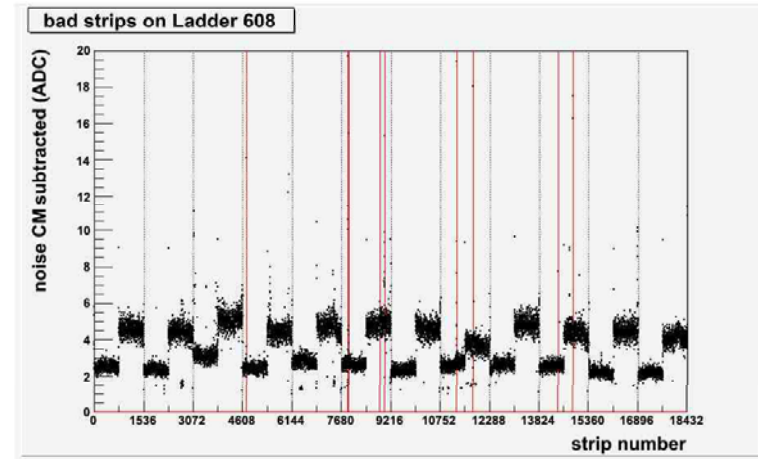
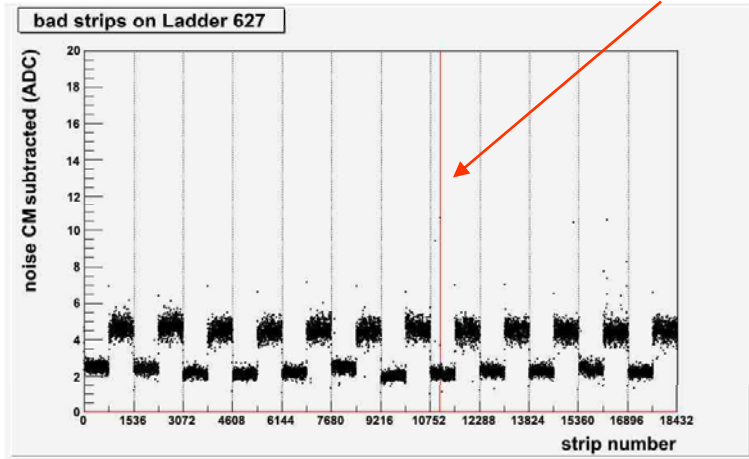
# Calibration strategy

- Improve algorithm ability to correctly calculate pedestal/noise (e.g. comparing from one run to the other)
- Keep pedestal, common mode, temperature and current as reference data for later checks
- ... but have a good monitoring system which notify strange behaviors during calibration and take actions (e.g. rerun calibration)



# Noise and bad channels (from tests)

Bad or Dead



---

# Calibration -> Noise

- Detector Algorithm (ITSSSDPEDda.cxx) for pedestal/noise calculation committed to cvs
- Purpose: provides noise level, uploads pedestal/noise to the electronics (to be tested)
  - Three new AliRoot classes support the DA: AliITSTChannelDaSSD, AliITSHandleDaSSD, AliITSMModuleDaSSD
  - Three new classes (AliITSTNoiseSSD, AliITSTGainSSD and AliITSTBadChannelsSSD) support storing of the calibration parameters in the OCDB.
- Simulated effect of noise and dead channels in the reconstruction (talk schedule for rec session tomorrow)

---

# Calibration: what remains to be done

- From noise to dead/noisy channel map  
(wait for filtering criteria from test data)
  - optimization of algorithms for noise calculation (e.g. is the noise gaussian?)  
Work is in progress (results from tests)

---

# Calibration -> preprocessor

- Preprocessor for noise committed at the end of August
- DA writes TObjArray of AliITSNoiseSSD objects
- Straightforward for the Preprocessor, reads AliITSNoiseSSD from the file and saves them directly to the ODCB

---

# Calibration: gain

- Gain calibration. Two solutions under study
  - Charge distribution in one-strip 1Dclusters (could be done from rawdata on LDCs);
  - Signal from digit correlated to tracks (has to be done offline after rec + access to digits)