SALT128 Wafer Probe Tests

Internal Note

Issue: 1
Revision: 0

Reference: LHCb-XX-2017
Created: February 8, 2017
Last modified: February 9, 2017

Prepared by: Christopher Betancourt[a],
[a]University of Zurich, Switzerland

Abstract

This is a description of wafer probe tests for the SALT128 mass production quality assurance. The mechanical and electrical layout of the chip as relevant for the tests, types of tests to be carried out, information on the wafer probe card and a timeline is provided.

Contents

List of Figures

List of Tables

1 Introduction

SALT (**S**ilicon **A**SIC for **L**HCb **T**racking) is a 128 channel ASIC to be used as a readout chip for silicon sensors in the UT of LHCb experiment. Quality assurance tests need to be carried out after mass production of the chip, insuring high yield and proper functionality before installation into the experiment. The large volume of chips ($\sim$ 5k) requires tests to be carried out in an automatic fashion, with each chip classified into three catagories; GOOD (full functionality of every channel), OK (at most one bad channel) and BAD (two or more bad channels). The chips will be tested on wafer before dicing.

Tests will be carried out with a semi-automatic probe station housed in a clean room located at CERN. The probe station is equipped with special imaging allowing for proper alignment and planarity. A wafer probe card will be used to make electrical contact with the chip. The probe card will be connected to a DAQ system controlled by a PC which also controls power supplies and movement of the probe station chuck, as illustrated in figure 1.

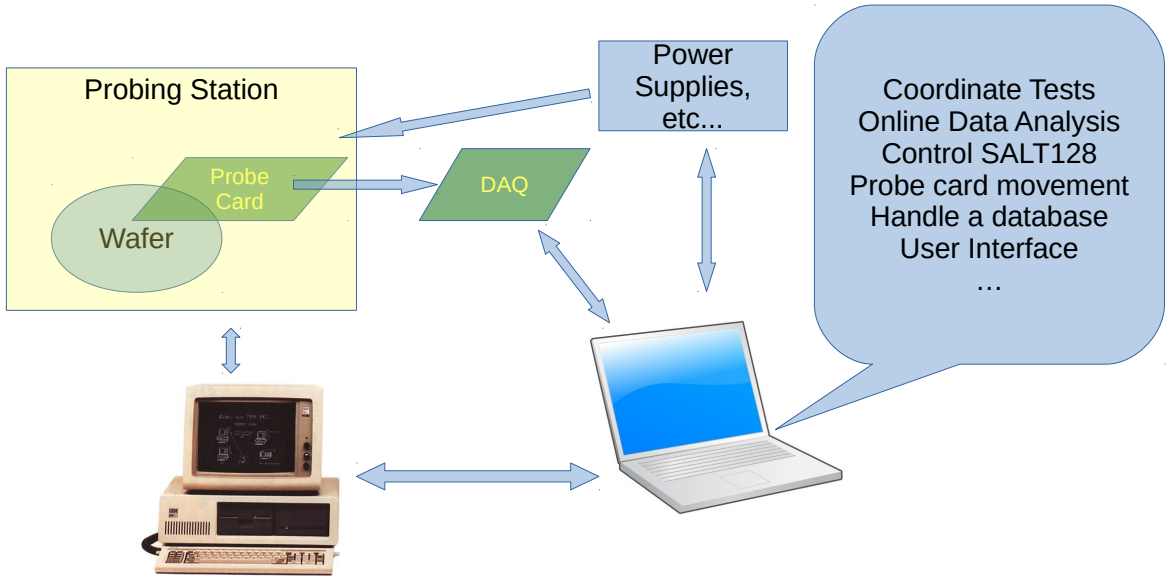

Figure 1 Illustration of the measurement set up.

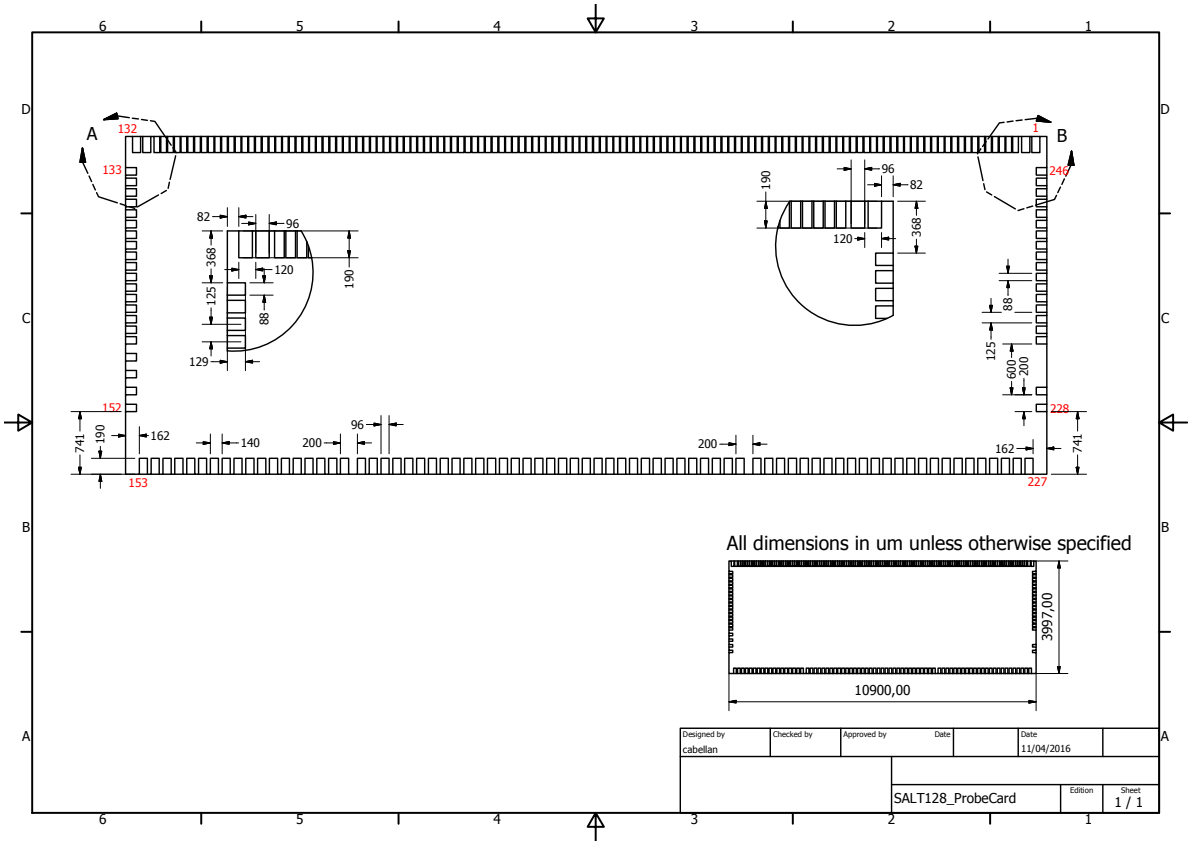

Figure 2 Layout of the pads on SALT128.

2 SALT128 layout

SALT128 will contain a total of 246 pads: 128 input pads which will not be probed during the wafer probe tests and will be bonded to hybrids after screening, 4 ground pads, two on each side of the 128 input pads, 78 pads on the opposite to the input pads which will be tested during wafer screening and also subsequently bonded to hybrid, and two sets of 18 pads which will only be used during wafer screening. A layout of the chip indicating the pad locations and their pin numbers is shown in figure 2. A list of pads to be probed during wafer screening is indicated in table 1.

Pin	Name	Probed?
1	VSSAI	?
2	VSSAI	?
131	VSSAI	?
132	VSSAI	?
133	PRE_BUF1	weak no
134	SH1_BUF1	weak no
135	SH2_BUF1	weak no
136	SH_BUF1	weak no
137	AP_BUF1	weak no
138	AN_BUF1	weak no
139	I_BUF1	weak no
140	VSSPSTD_BUF1	weak no
141	VSSA_BUF1	YES
142	VDDA_BUF1	YES
143	ADC_INP	weak no
144	ADC_INN	weak no
145	V_S2D_DAC	YES
146	V_PULSE_DAC	YES
147	V_SLVS_VBIAS_DAC	YES
148	V_SLVS_INREF_DAC	YES
149	SCAN_ENABLE	YES
150	SDI	YES
151	SDO	weak no
152	TEST_MODE	YES
153	VDD	YES
154	VSS	YES
155	VDD	YES
156	VSS	YES
157	VDDPST	YES
158	ID0	YES
159	ID1	YES
160	ID2	YES
161	I2C_SCL	YES
162	I2C_SDA	YES
163	RST_N	YES
164	MAIN_CLK_N	YES
165	MAIN_CLK_P	YES
166	VSSPST	YES
167	VDD	YES
168	VSS	YES
169	VDD	YES
170	VSS	YES
171	VDDAPST	YES
172	VSSAPST	YES
173	VDDADC	YES
174	VSSADC	YES
175	VDDADC	YES
176	VSSADC	YES
177	VREFD	YES
178	VSSADC	YES
179	VREFD	YES
180	VSSADC	YES
181	VDDADC	YES

182	VSSADC	YES
183	VDDADC	YES
184	VSSVCM	YES
185	VDDVCM	YES
186	VSSVCM	YES
187	VDDVCM	YES
188	VSSAPST_IN	YES
189	VDDAPST_IN	YES
190	VSSAPST_IN	YES
191	VDDAPST_IN	YES
192	VSSAI	YES
193	VDDA	YES
194	VSSAI	YES
195	VDDA	YES
196	VSSAI	YES
197	VDDA	YES
198	VSSA	YES
199	VDDA	YES
200	VSSA	YES
201	VDDA	YES
202	VSSA	YES
203	VDDA	YES
204	VSS	YES
205	VDD	YES
206	VSS	YES
207	VDD	YES
208	VSSPST	YES
209	DDR_TFC_N	YES
210	DDR_TFC_P	YES
211	DDR_OUT0_N	YES
212	DDR_OUT0_P	YES
213	DDR_OUT1_N	YES
214	DDR_OUT1_P	YES
215	DDR_OUT2_N	YES
216	DDR_OUT2_P	YES
217	DDR_OUT3_N	YES
218	DDR_OUT3_P	YES
219	DDR_OUT4_N	YES
220	DDR_OUT4_P	YES
221	DDR_OUT5_N	YES
222	DDR_OUT5_P	YES
223	VDDPST	YES
224	VSS	YES
225	VDD	YES
226	VSS	YES
227	VDD	YES
228	DATA_CLK_OUT_N	YES
229	DATA_CLK_OUT_P	YES
230	V_SH_DAC	YES
231	V_KRUM_DAC	YES
232	V_PRE_DAC	YES
233	VMCD	YES
234	VCMC	YES
235	VCMB	YES

236	VCMA	YES
237	VDDA_BUF0	weak no
238	VSSA_BUF0	weak no
239	VSSPSTD_BUF0	weak no
240	I_BUF0	weak no
241	AN_BUF0	weak no
242	AP_BUF0	weak no
243	SH_BUF0	weak no
244	SH2_BUF0	weak no
245	SH1_BUF0	weak no
246	PRE_BUF0	weak no

Table 1: List of pads to be tested during wafer screening.

3 Probe station, wafer probe card and DAQ

All wafer probe tests will make use of a semi-automatic probe station located in a clean room at CERN. A photo of the set up is shown in figure 4.

4 Functionality Tests

In order to ensure proper functionality of a chip, a series of tests must be executed succesfully from start to finish. Each chip will be catagorized into three groups: GOOD, OK and BAD. GOOD chips are fully functional in every channel and are ready for dicing. OK chips have one bad channel; they may be used in the experiment depending on the overal yield of GOOD chips. BAD chips will have two or more bad channels and are not suitable for use in the experiment. A preliminary list of tests is as follows:

1. Power tests
   Check proper power consumption of ANA and DIG parts after power-up and reset
2. Communication (I^2C) tests
   $\rightarrow$ There are about 450 registers $\sim$ 10kb total amount of data, which should not be a with limited FPGA RAM
   (a) Serializer registers
   (b) DSP registers NZS path
   (c) DSP registers rest
   (d) SEU registers
   (e) Basing DACs and monitoring ADCs basic functionality test
   $\rightarrow$ Need to estimate time needed to test all values of DAC (input from Carlos needed)
3. DLL, PLL, Bandgap tests (using internal monitors) and configuration
   $\rightarrow$ DLL, PLL values close to 0 ADC would mean proper functionality
   $\rightarrow$ We need to learn how to configure different blocks
   (a) DLL test SALT documentation section 5.1.2.
   (b) PLL test SALT documentation section 5.2.1.
   (c) Bandgap test and readout of temperature sensor
4. Serializer and Deserializer tests and configuration
   $\rightarrow$ Need to check if we can move first byte and correct pattern, then this works correctly
   $\rightarrow$ The PLL should be fully functional before we get to this stage
   $\rightarrow$ Need to see if Serializer works before testing Deserializer

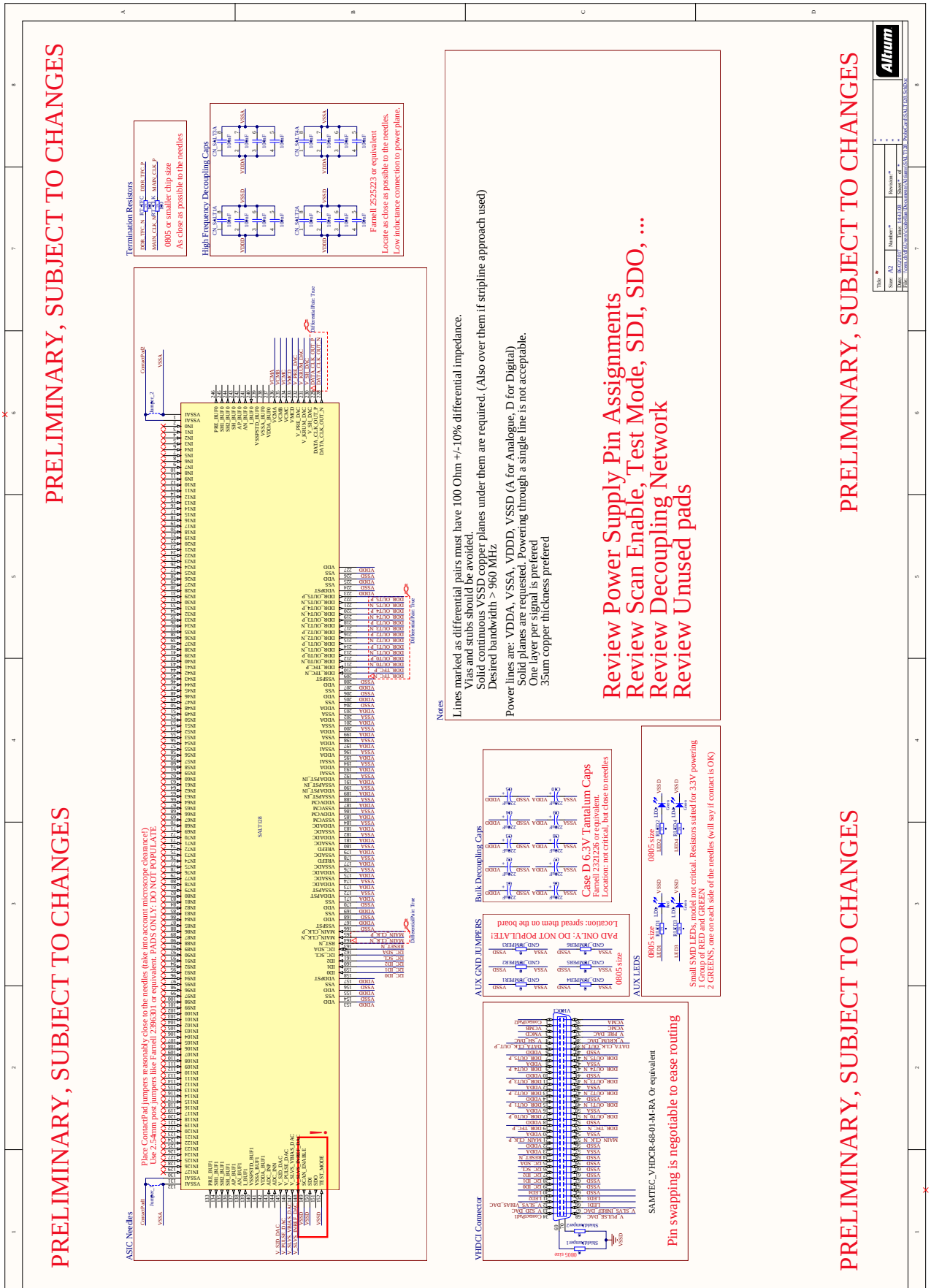

Figure 3 Schematic of the SALT128 chip with the input pads labeled and lines to the DAQ.

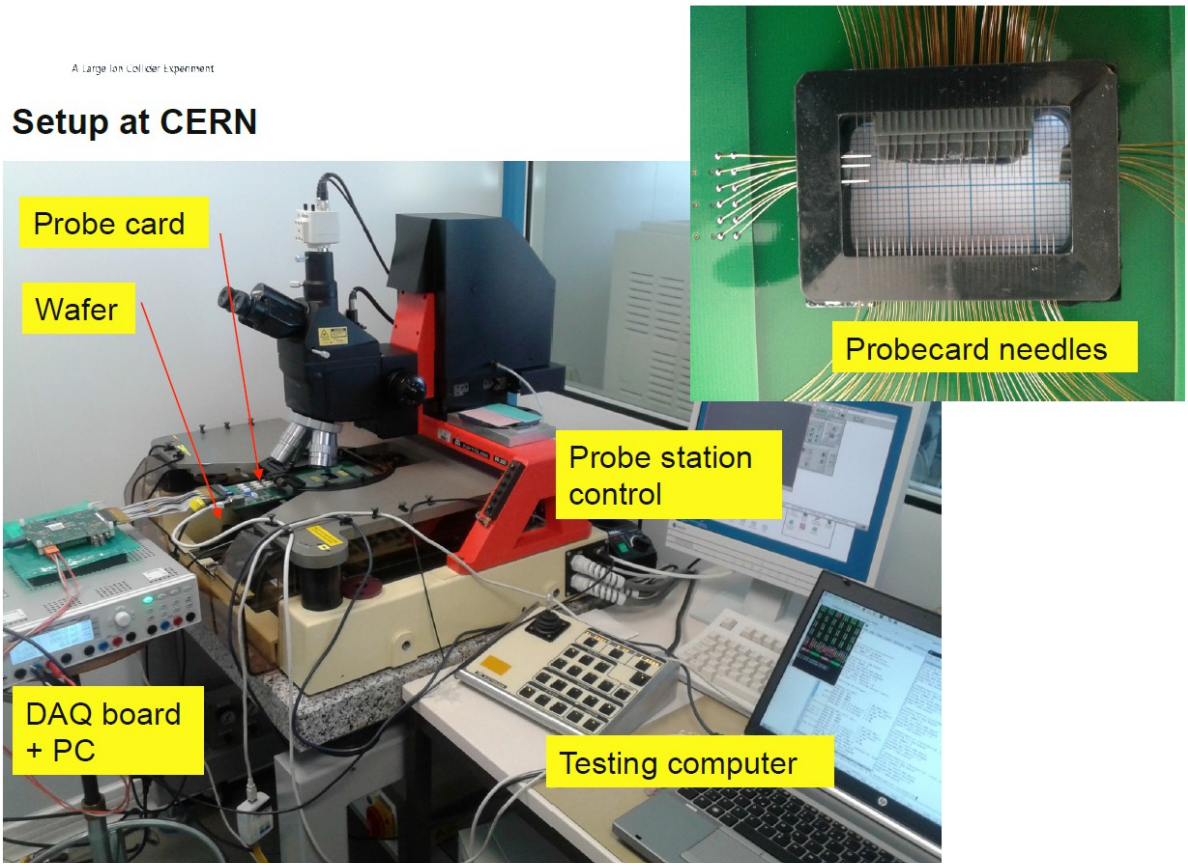

Figure 4 Picture of the semi-automatic probe station with connections to the PCs and DAQ (PRILIMARY PICTURE, TAKEN FROM *P. Riedler, 5th ALICE ITS, MFT and O2 Meeting*).

5. Basic TFC commands and data packet tests

   (a) SYNC (FeReset)

   (b) HeaderOnly, BxVeto, BxReset

6. NZS data packet and Pedestal Subtraction tests (in meantime also mask register could be tested)
   → Need to send a pedestal value to each channel. How do we upload pedestal value to every channel several times?
   → Need to send to each channel several mask registers
   → Need input from JC, since he is testing this now

7. ADC and front-end baseline correction DAQ tests
   → Need to discuss with JC, he is doing baseline correction tests → Want to test baseline DAQ and test ADC
   → Send some data, min code, max doce, middle code and make sure DAQ are reading correctly and see how they change
   → Through the I^2C prgram DAQ, and through ADC program a few values and check
   → Are we going to record for individual channels or not?
   → Need to get offset and spread, which is needed to correct pedestals. Need to have several points here to find correct value.
   → We can read all channels at the same time, since this is doen in parallel
   → We are testing Trim DAQ and base ADC functionality
   → How wide a range can we test? Not the full range of ADC but we need to test full range of Trim DAQ.
   → No ADC at this point, only an offset (this is a static measurement, since we only change the offset)
   Points 1-7 have validated most important blocks

8. Front-end performance tests (using internal calibration)

   (a) Gain, Offset, Crosstalk
   → 5 point Gain curve. Can do for every channel in parallel (or every fourth channel). Will there be a problem of occupancy of the chip?
   → What about timing, delays, etc.?
   → Need to set correct value of delay in DLL before measurement of gain and offset. Can get some basic numbers from current tests.
   → Can we use internal calibration even if pad is connected to something else?
   → Do we need to test the input pad? Probably not, since for these tests we are only using internal calibration
   → Need to define how many input needles we need for the probe card
   → Will a huge load capacitance (i.e. when sensor is connected) affect the performance of ths chip? Should ask people who design ASICs for ATLAS. Do we need to test inputs? Should decide this soon, but maybe it's not an issue
   → Need to define minimum number of needles needed for power (See point above about how many needles needed on probe card). Carlos, Chris, Iaroslava and Olaf can figure this out. Olaf will check what was done for the Beetle.

   (b) Noise
   → To get a correct response from the ASIC, measure each point (of the 5 points in gain tests) several times to make sure noise is correct. First we need to know how much time is needed for this test, and then we can determine how many measurements to do for each point.
   → Memory (RAM) in FPGA is limited, can't do 200 points in parallel for example. Depending on self correlation of noise, if we don't sample at max speed, but lower, we can get the same or better statistics with less points. But this will depend on type of noise → As long as we can't perform S-curves, we can only estimate from ADC. Accumulate some number of samples and draw histogram. Can learn from JC to clarify this.

9. MCM tests
   → Mean number of channels above threshold and mean common value. → If pedestals during measurements in point 6 are set properly, MCM could be tested partially during measurements in points 7 and 8
   → In any case, we can test four situations: raw data, data after masking, after pedestal, after MCM

10. ZS tests
    → Same situation as point 9. Could be tested in parallel to 7, 8 and 9. → Very easy to switch from zero supressed to non-zero supressed at any time. → Can test with and without ZS in same data stream

11. RAM and Trunc packet tests
    → We could fill up memory somehow to get a special packet truncation. Fill memory and observe several kinds of truncation packets
    → Final ASIC will be 2 or 3 trunc packets

12. TFC counters and Snapshot command tests
    → Can be done in parallel with other tests, since these are continuously running
    → Check if value in the counters are correct
    → Check if snapshot commands are working alright. → DSP is crucial, but TFC might not be critical

13. DACs tests (using external pads and ADCs) → Use relays to test from one channel to another.
    → Can op-amp have some influences on results of linearity?