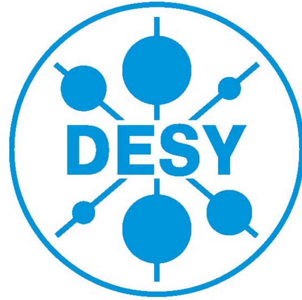


# CMS Pixel Upgrade

## DESY Summer Student Programme 2014 - Project Report

Arthur Bolz, Heidelberg University, Germany

September 11, 2014


### Abstract

My work during the DESY Summer Student Programme 2014 in the CMS group is presented. DESY participates in the development of a new CMS pixel detector to be completed in 2016/2017. In particular, detector chips and modules are currently tested and calibrated at DESY and a fraction of the modules for the final detector will be bump bonded here.
Contributions to two projects were made and are presented in this report. First, a method to measure the quality of individual bump bonds without having to destroy the bonded chip is discussed. Secondly, the efforts to program an interface for the CMS pixel detector, which allows to integrate it into DESY EUDAQ data acquisition framework are described. The latter was part of the preparations for testing the CMS pixel detector at the DESY testbeam.

# Contents

**1 Introduction** **3**

**2 The CMS Pixel Detector** **4**
2.1 ROC Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
2.2 Bump Bonding . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

**3 Non-Destructive, Qualitative Bump-Bond-Test** **6**
3.1 Fitting Procedure . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
3.2 Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

**4 DESY Testbeam preparations** **12**
4.1 Writing a CMSPixel-Producer for EUDAQ . . . . . . . . . . . . . . . . . 14

# 1 Introduction

The Compact Muon Solenoid (CMS) [1] experiment is one of the four large experiments at the Large Hadron Collider (LHC) at the European Organization for Nuclear Research (CERN) in Geneva, Switzerland. It is a general purpose detector which allows to study a wide range of particle physical questions.
At the very center of the detector in close proximity of the interaction points the CMS pixel detector forms the core of the CMS tracking system. It consists of three barrel layers with radii from 4cm to 11cm covering a pseudorapidity range $-2.5 < \eta < 2.5$ and two endcap layers on either side. It counts a total number of roughly 65 million silicon pixels with a size of $100\mu$m times $150\mu$m. All the details of the CMS tracking system, as well as extensive descriptions of all other detector components is given for example in [2].
While the CMS tracking system has performed very nicely so far [3], the very good performance of the LHC and particularly the steady increase in luminosity requires a detector upgrade. Already in 2012 the number of proton interactions per bunch-crossing exceeded the nominal value [4] so in order to maintain tracking performance at a higher detector occupancy the current CMS pixel detector needs to be replaced by an improved version. Modifications focus on improving the readout scheme in order to achieve faster readout and on reducing the material budget in order to improve the detector resolution. Also, an additional fourth barrel layer will be included.
Among other institutes, DESY contributes to the detector development. Modules are tested and calibrated at DESY and a fraction of them will be bump bonded here. The detector development is currently in the final stage. The design of the silicon pixel sensors

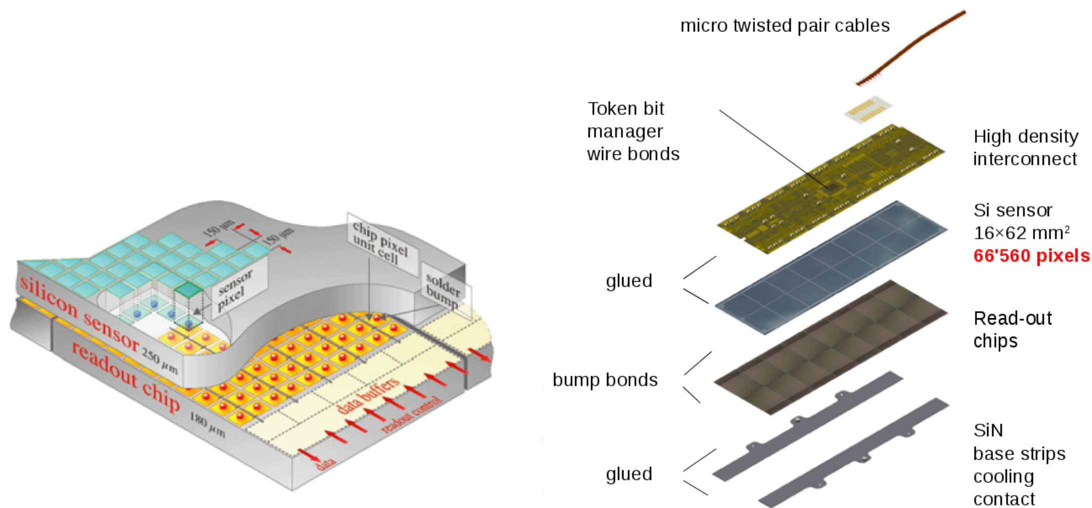


Figure 1: Left: Schematic illustration of a hybrid pixel detector where sensor and readout chip come from separate wavers and are bonded together by bump solder bonds. Right: Composition of a CMS pixel detector module. Source: CMS Pixel Group, DESY

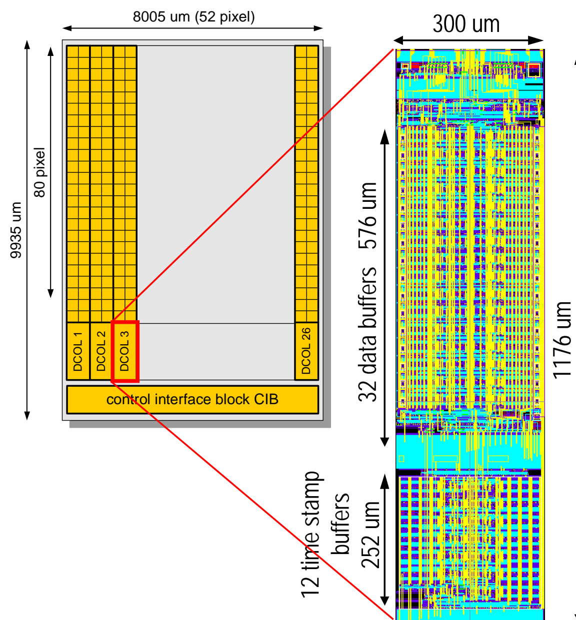


Figure 2: Design of the readout chip. Source: Beat Meier, PSI

and readout chips is complete but they still need to be produced, tested, calibrated and assembled. If things go as planned, the new pixel detector will be mounted into the CMS experiment during the shutdown in 2016/2017.
This report is structured into three parts: The first part gives a basic introduction to the CMS pixel detector. The second part describes a non-destructive method to test the quality of the bump bonding between detector sensors and readout chips. And the third part deals with preparations for testing the CMS pixel detector at the DESY testbeam, more precisely with the integration of the CMS pixel data acquisition into the DESY EUDAQ data acquisition framework.

# 2 The CMS Pixel Detector

The CMS pixel detector is a hybrid design, where sensor and readout chip (ROC) come from different wavers and need to be soldered together as is shown schematically on Figure 1. It consists of several sub-modules which again consist of one silicon sensor plate and 16 separate readout chips. The sensor ROCs are further segmented into 52x80 pixels per ROC with an approximate size of 100x150$\mu$m$^2$. The silicon sensor is of an "n-on-n" doping design so that electrons are collected. This choice, as well as many details of the ROC design, follow from the necessity of the detector being radiation hard due to its proximity to the beam line. The ROCs are connected to the sensor via bump solder bonds and collect the electrons from the sensor to convert them into well-defined digital signals that can be read out. Figure 1 shows a module and all of its components. Besides the key elements mentioned above, also mechanical support structures and a

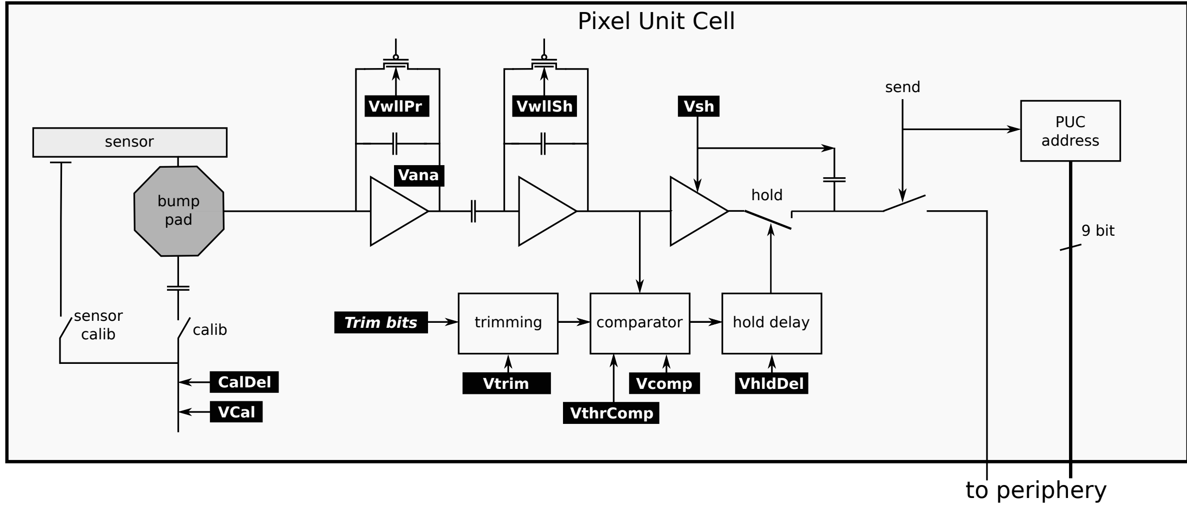


Figure 3: Schematic of the readout circuit of one pixel unit cell. The sensor is connected via a bump solder bond as indicated. The signals are then processed through a preamplifier, a shaper and a discriminator and if they exceed a given threshold they are stored to be read out by the periphery. Also included is a calibration signal to test the ROC settings and bump bond connection to the sensor. Source: CMS Pixel Group, DESY

token bit manager are part of a module. All details are given in [5].

## 2.1 ROC Design

The ROC layout is shown in Figure 2. One chip consists of three regions with different functionality. First to mention are of course the 52 columns and 80 rows of active pixels to read out the sensor. Two neighboring columns form a unit where one is the mirror image of the other. The pixels of such two columns are read out double-column-wise and the data stored in the second region of the ROC where all such double columns have separate data and time stamp buffers. The third region is the interface with the rest of the detector used to receive timing signals and merge the pixel data into the CMS data stream.

Figure 3 shows the readout circuit of one pixel unit cell. The charge signals are collected from the sensor via the bump solder bond. They are first pre-amplified and then shaped. If a signal exceeds an adjustable threshold it is stored and eventually read out into the periphery buffers by the delayed readout chain. The readout unit cells of all the pixels can be configured individually by a set of adjustable registers to tune for example the discriminator threshold or adjust preamplifier or shaper.

The most relevant feature for this report is a test circuit included on every pixel for checking the readout chain settings, investigating how signals are propagated and testing the connection to the sensor. This test circuit allows to apply an adjustable calibration voltage pulse $V_{cal}$. With the help of a calibration capacitance this pulse can either be

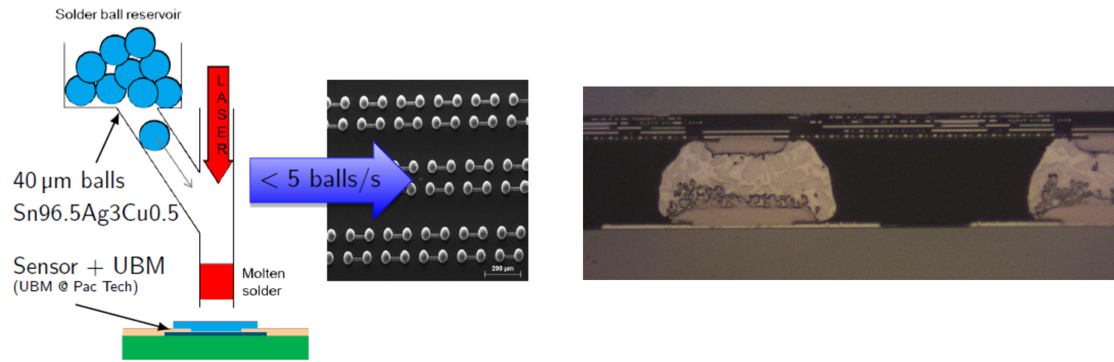


Figure 4: Left: Illustration of the operation method of a solder ball laser jetter and microscope image of solder balls attached on top of a sensor. Right: Microscope image of the cross section of a solder ball connection between a sensor and ROC pixel.

transformed into a charge signal directly in front of the preamplifier to test the readout chip. Or it can be transformed into a charge signal on the sensor via a metal pad, which forms a capacitance together with the sensor. This signal can be used to test the electrical connection between ROC and sensor via the bump solder bond as it can only be read out if such a connection is established.

## 2.2 Bump Bonding

The wafers for sensors and readout chips are produced by IBM. They are then thinned, bump bonded and glued to the support structures. The steps are performed by various institutes and companies. At DESY the bump bonding of modules for the fourth barrel layer is performed. It is done in two steps: First, the SnAg solder balls are placed on every pixel of the sensor with the help of a solder ball laser jetter, as illustrated in Figure 4. The solder balls are melted by a laser so that the can drop through a thin capillary onto the sensor surface. There they merge with the metal on top of the sensor, freeze out and thanks to surface tension form again little bullets. Secondly, the 16 ROCs belonging to one module are individually bonded to the sensor with a Femto flip chip bonder. To establish a stiff connection between ROC and sensor the solder balls are once more heated beyond the melding point and cooled down again. A cross section of one of the resulting bump bonds is shown in Figure 4, as well.

# 3 Non-Destructive, Qualitative Bump-Bond-Test

After having bonded the readout chip (ROC) to the silicon sensor, the bump bonds need to be tested. The first and most important test of course is, whether an electrical connection between sensor and ROC has been established for all pixels. Of further interest is the question of the quality of each individual bond.

The design of the ROC allows to easily test whether an electrical connection between chip and sensor has been established. On each ROC unit cell there is a little metal pad and since ROC and sensor don't touch but are separated by an air gap these pads form air capacitances with the metal layers on top of each sensor pixel. A calibration voltage pulse can be sent to this pad, where it induces a mirror charge on the sensor via the capacitance. If there is an electrical connection to the ROC this charge can flow back and is processed in the normal signal circuit. If not the bump bonding for the according pixel has failed.
This is a somewhat binary test of the bump bonds that doesn't give any information on their quality. For quality control and to understand how well the bonding method works it is desirable to monitor this quality. One way to do this is to simply cut the bonded chips apart and look at cross sections of bump bonds under the microscope. An example is shown in Figure 4. However, the chips have to be destroyed in order to do this.
It is desirable to test the quality of the bump bonds without having to destroy the chips. In fact, the quantitative test described above can be extended so that the test result also contains information on the quality of each bond. The idea behind it is rather simple. The charge introduced on the sensor via the air capacitance is proportional to this capacitance as well as the calibration pulse:

$$Q_{cals} = C_{air} \cdot V_{cal}. \tag{1}$$

On the other hand, the capacitance is proportional to one over the width of the gap between ROC and sensor, which is an indicator for the quality of the bond. Thus, by measuring the capacitance one can determine this width and take it as a measure for the bonding quality:

$$C_{air} = \epsilon_0 \frac{A}{d_{gap}} \sim \frac{1}{d_{gap}}. \tag{2}$$

The deposited charge is amplified by the ROC and to zeroth order the pulse height measured in the end is given by:

$$PH_{cals} = gC_{air} \cdot V_{cal} + const, \tag{3}$$

where $g$ is the overall gain factor and a current offset is included, while higher order effects like parasitic elements of the circuit are ignored. Since neither gain, calibration pulse nor the offset are well known, instead of only measuring the pulse height of the sensor signal one can compare it to the direct test pulse. For the direct test pulse the relation

$$PH_{cal} = gC_{cal} \cdot V_{cal} + const \tag{4}$$

holds, where $C_{cal}$ is the calibration capacitance which is the same for all pixels with high precision. Thus

$$\frac{\partial PH_{cals}/\partial V_{cal}}{\partial PH_{cal}/\partial V_{cal}} = \frac{C_{cal}}{C_{air}} \sim \frac{1}{d_{gap}}. \tag{5}$$

So measuring the dependence of the pulse height on the calibration voltage for both the direct pulse and the test pulse via the sensor allows to measure the ratio of capacitances and thus, up to a proportionality factor, the gap between ROC and sensor.

## 3.1 Fitting Procedure

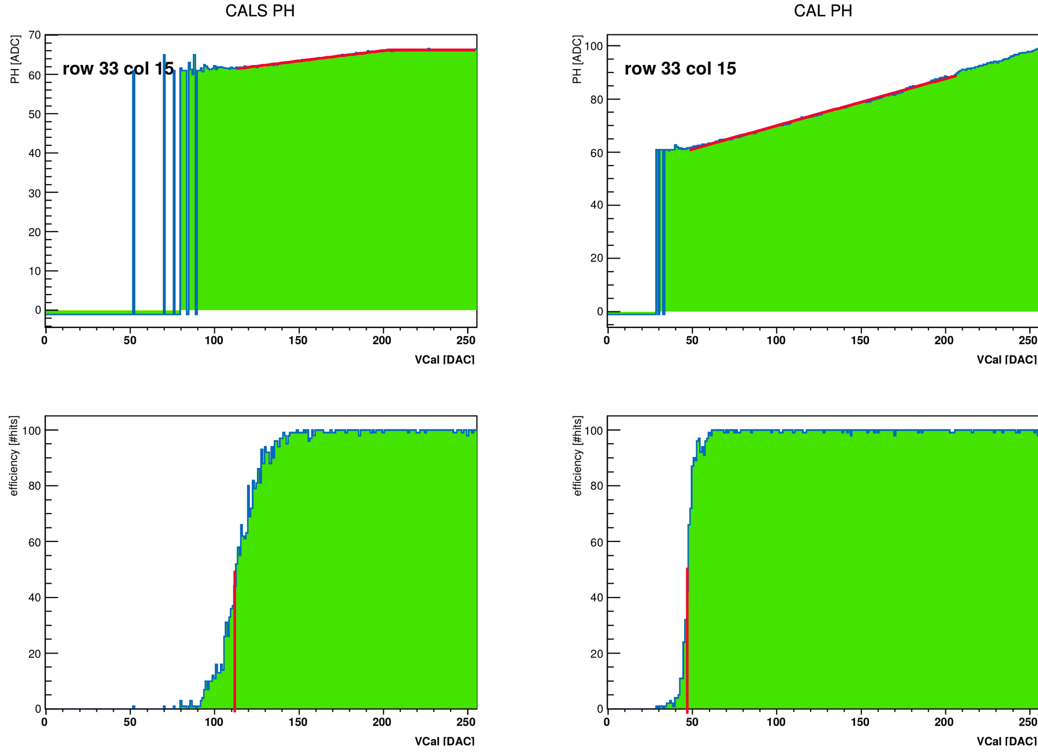


Figure 5: Top: Measured pulse height as a function of the applied calibration voltage. The left plot shows the pulse height for the sensor calibration signal, while the right one shows it for the direct signal. In red fits to the distributions are shown. Bottom: efficiency S-curves for the sensor and the direct pulse. The red lines mark the 50% threshold used as a lower limit for the fit range.

Figure 5 shows the measured pulse height for one pixel of the calibrated ROC number 3 on module D4003c depending on the applied calibration voltage. For the direct test pulse the low calibration pulse in the range ???-??? V was used, for the pulse via the sensor the strong pulse in the range ???-???V. Above threshold the pulse height shows the expected linear dependence on $V_{cal}$ until saturation is reached. In order to extract the ratio of capacitances both pulse height measurements are fitted. To account for the saturation effect the following fit model is chosen:

$$PH(V_{cal}) = \begin{cases} a_0 \cdot V_{cal} + a_1, & \text{for } V_{cal} < V_0 \\ a_0 \cdot V_0 + a_1, & \text{for } V_{cal} > V_0 \end{cases} \tag{6}$$

Only data with an efficiency larger than 50% are considered and the model parameters $a_0$, $a_1$ and $V_0$ are varied to minimize the $\chi^2$-value. In principle the current offset in Equation 3 and 4 should be the same. This was not used as a constraint in the completely independent fits, however it was verified that $a_{1,cals} \simeq a_{1,cal}$ within uncertainties.

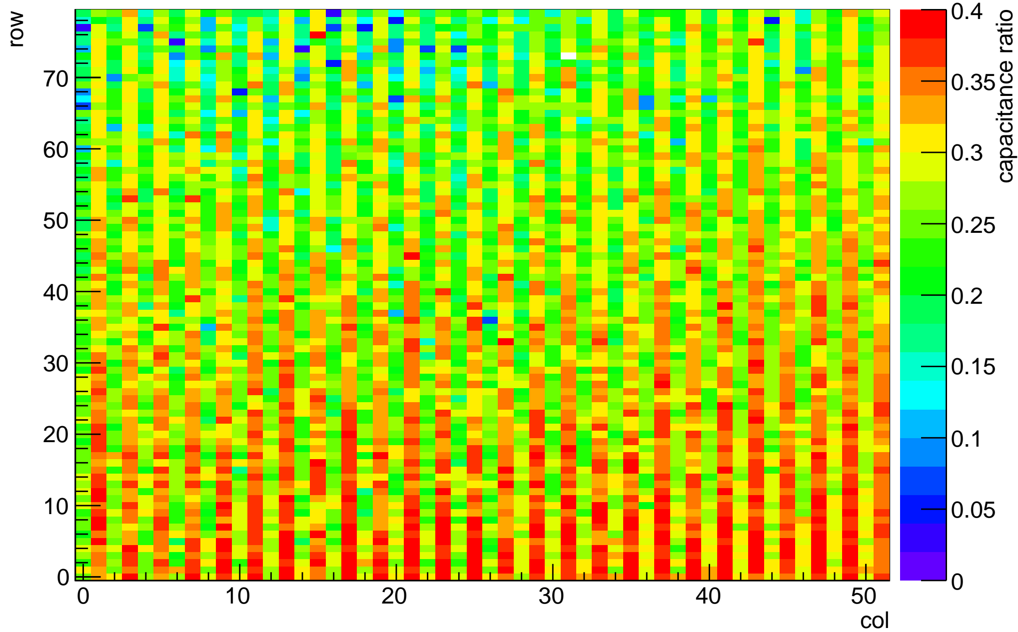


Figure 6: Pixel map for one ROC. The color code shows the ratio of the capacitances $C_{air}/C_{cal}$. The map indicates an asymmetry between even and odd columns which is not yet understood. Furthermore it shows a trend in the capacitance ratio toward lower values along the rows. In the column direction no such trend is visible. The white pixel in column 31 and row 73 indicates a missing bond.

The ratio of capacitances is given by the fit parameters $a_0$:

$$\frac{C_{air}}{C_{cal}} \simeq \frac{a_{0,cals}}{a_{0,cal}} \tag{7}$$

## 3.2 Results

The capacitance ratio can be extracted from fits to all pixels of the ROC. A map of the ROC showing the obtained values for all pixels is given in Figure 6

Two things can be observed in the map: First there seems to be a difference between even and odd columns. This effect hast not been understood yet, but is probably due to asymmetries between even and odd columns that somehow lead to different parasitic effects, which were not considered in the model. Secondly, there is a clear trend from lower capacitance ratios at the top of the chip, i.e., larger gaps, towards higher capacitance ratios at its bottom.

To visualize this trend better, one can take a look at the one dimensional projections of the map. Figure 7 shows these projections along the columns and rows of the chip. As

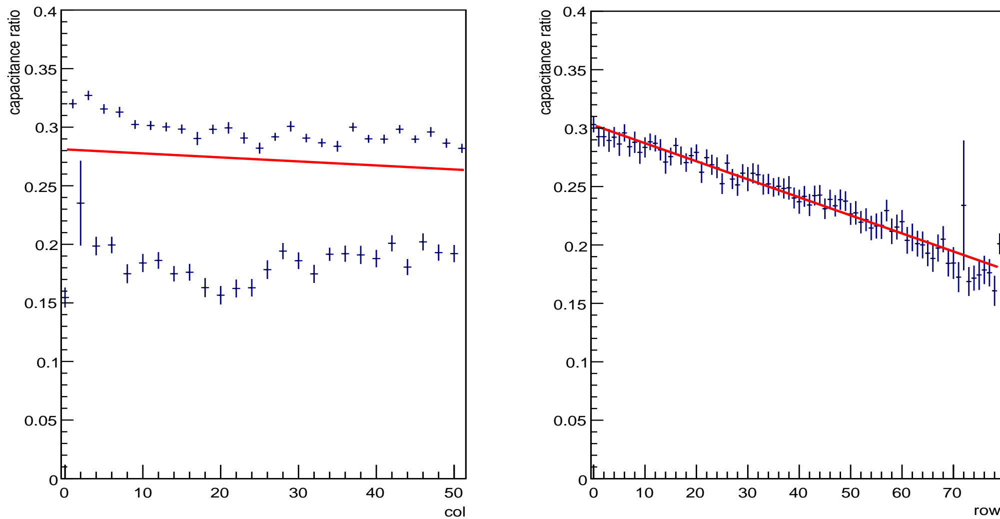


Figure 7: One dimensional projections of the two dimensional pixel map presented in Figure 6. The projections confirm the previously discussed even/odd column effect as well as the clear trend in the ratio values along the row direction.

already seen on the map, there is a significant even odd effect along the column direction but otherwise no tilt between chip and sensor. Along the row direction, however, a significant tilt can be observed with a deviation of the capacitance values at the edges from the mean value of more than 20%.
In order to make sure that these variations are in fact due to variations of the gap width, one has to compare the capacitance ratios to this width. Without destroying the chip this cannot directly be measured. What can be done instead is to measure average height of the first and last row and column of a ROC relative to the flat sensor. Since the ROC width is constant, this height goes linear with the gap width.
Figure 8 shows the maximum deviation of the capacitance ratio from the mean value plotted against the according ROC height difference. The deviation is determined by a linear fit to the projections of the capacitance ratios. On the left the tilt for two ROCs of module D4003c along rows and columns is shown, on the right the tilt for 7 ROCs of module D4004c. Data for more chips wasn't available. The data points have a rather large spread, which is not within the included, barely visible uncertainties. Only "statistic" uncertainties from the fit of the projections are included while dominant systematic uncertainties were neglected. As was mentioned these arise for example from parasitic impedances that can be different from chip to chip and were not included in the model. The uncertainties for the ROC height measurements were estimated to be $1\mu$m. The data for module D4003c were measured more precisely than for module D4004c so the results there should be more reliable.
Altogether, the plots give some evidence for a relation between the measured capacitance ratio and the gap width between ROC and sensor. To further establish this assumption,

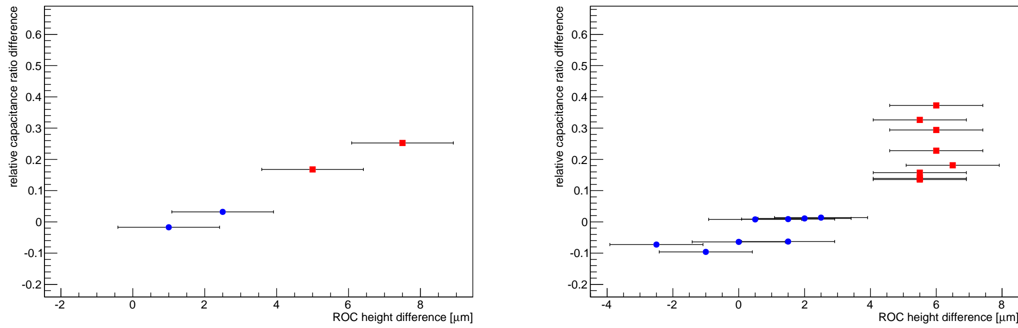


Figure 8: Correlation between the variation of the capacitance ratio and the variation of the ROC height across the ROC. All values obtained for module 4003c are shown on the left, all values obtained for module 4004c on the right. Red points indicate a tilt along the row, blue points a tilt along the column direction.

more precise measurements need to be performed.
The method used here to determine the capacitance ratio by fitting the gain curves turned out to be very inefficient. In order to be able to fit a line to the gain curves, they have to be measured many times to account for large statistic fluctuations. This is very time consuming. In the meantime, a new method to measure the air capacitance by measuring the noise via a fit of the gain curves, which is stable even at very low statistics, had been developed. With this method the correlation between ROC height and capacitance ratio could be confirmed, and since it provided faster and better results the method described above was abandoned.

# 4 DESY Testbeam preparations

The CMS pixels are to be tested at the DESY-Test beam. The DESYII accelerator for electrons(positrons) is primarily used as a pre-accelerator for the PETRAIII storage ring. In addition it can generate electron(positron) beams which can be used to test detectors under development [6].
The test beam allows the user to select fairly monoenergetic electrons in an energy range 1-6 GeV. In addition DESY provides him with the EUDET Pixel Telescope [7], a set of 6 MIMOSA26 monolithic pixel sensors which allow for very high resolution tracking.

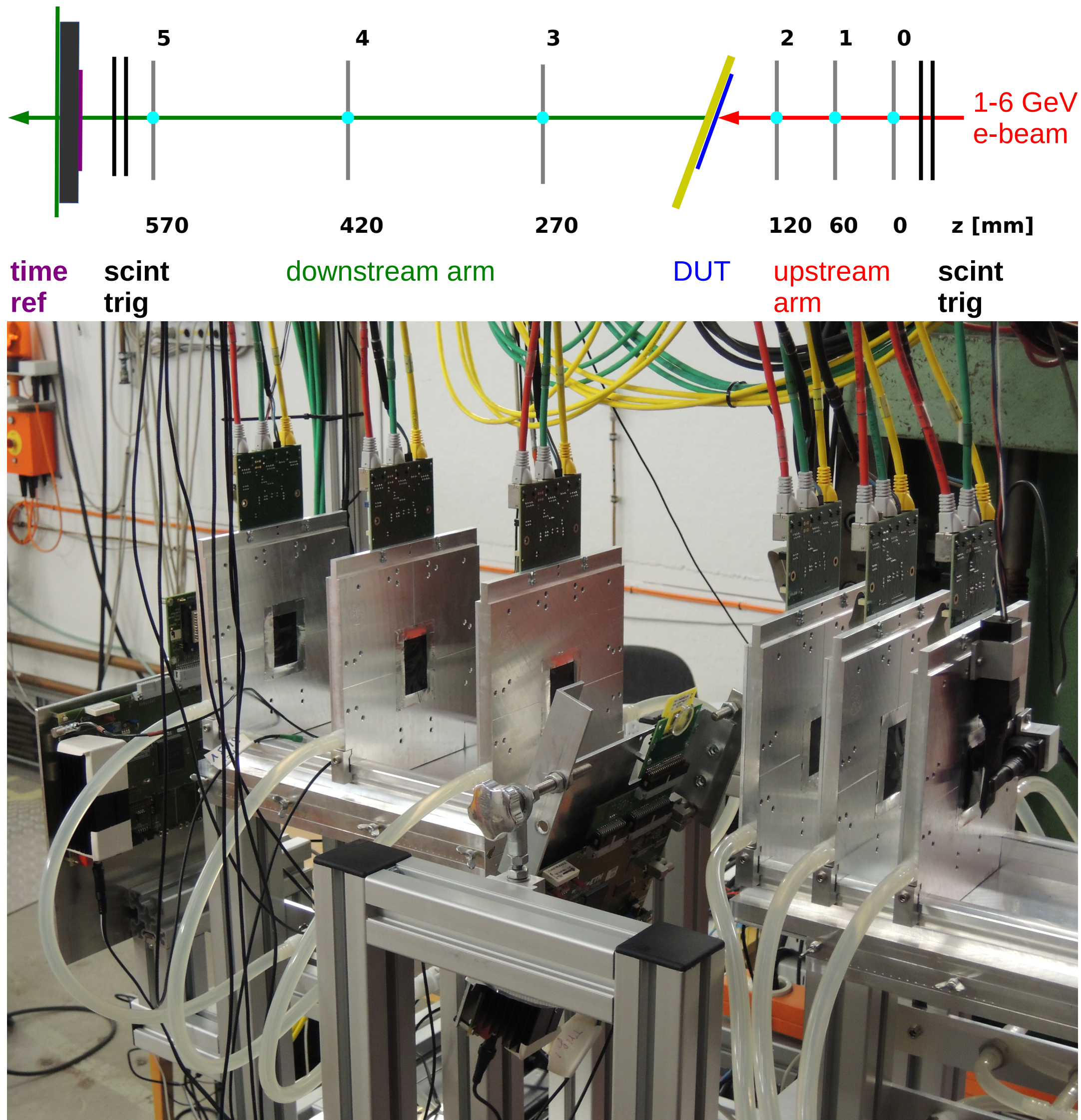


Figure 9: Top: Schematic of a typical test beam setup with the DUT placed within the EUDET telescope. Bottom: An actual picture of this setup from a previous CMS pixel test.

Altogether devices can be tested in a well defined setup. An example for such a test beam setup is shown in Figure 9.
The EUDET Pixel Telescope is controlled by the EUDAQ data acquisition framework [8]. The EUDAQ software is written in C++ and highly modular. In principle it allows to integrate the data acquisition for the device under test (DUT) which has the benefit that all test data can be collected in one data stream and stored together event by event. Consequently, no tedious offline merging of data files from various test setup components is necessary. All that needs to be done is to write an interface between the DUT daq and EUDAQ. In Figure 10 the schematic of the EUDAQ framework is shown.

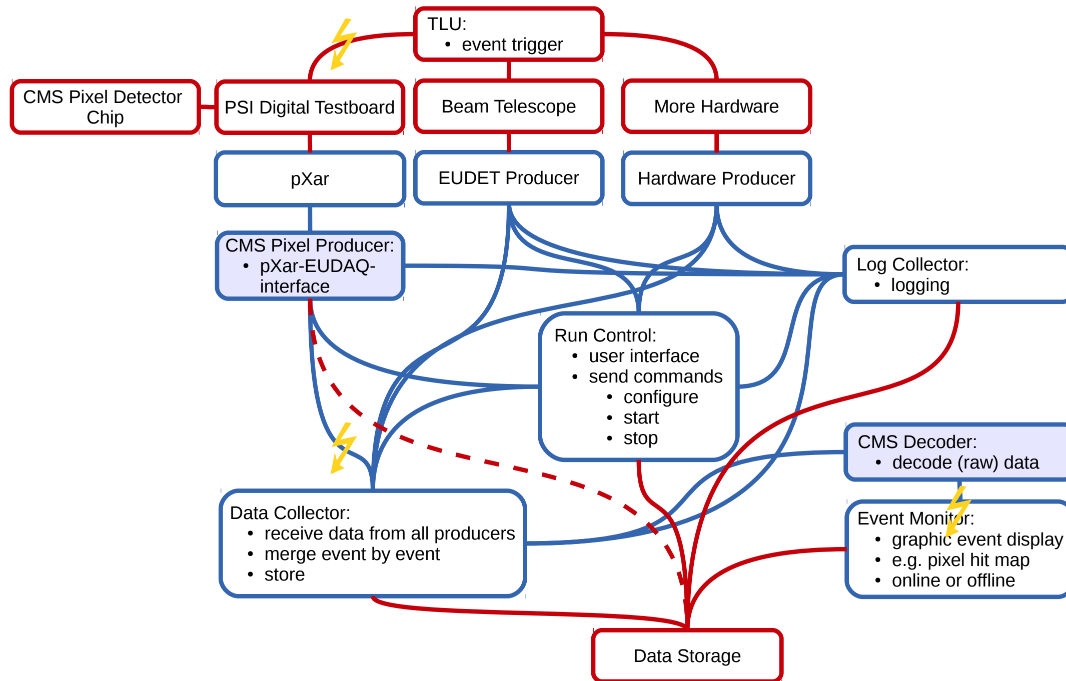


Figure 10: Flowchart illustrating the basic components of the EUDAQ data acquisition framework and how the CMS pixel detector is included. The filled boxes indicate code that was written, lines indicate connections between the various components. Lightning bolts indicate problems, that still exist but hopefully can be fixed.

It consists of a various number of sub-processes with different tasks that communicate with each other via TCP sockets. The whole DAQ is controlled by the run control, from where the various devices can be configured and data runs are started and stopped. It is accompanied by a logger receiving status information from all the devices. Each detector device communicates with EUDAQ with the help a so-called producer. This producer is an software interface between the run control and the detector, which allows the run control to configure, start and stop the device. In addition, a producer needs to send out data for each event to the EUDAQ data collector. This data collector collects the data streams from all detectors and merges them into a single stream, which is then

stored into a single file. Further tools, like data monitors allowing for on- and offline data quality control are provided by EUDAQ, as well.
The goal during this summer project was to write a producer to control the CMS pixel detector and fully integrate it into the EUDAQ framework.

## 4.1 Writing a CMSPixel-Producer for EUDAQ

For testing purposes, single CMS Pixel ROCs can be controlled and read out with a digital testboard developed (DTB) at the Paul Scherrer Institute (PSI). This testboard can be connected to a PC via USB and is controlled by the pXar [9] software. It is thus not necessary to write an interface between the CMS pixel detector and EUDAQ but rather it is sufficient to write an interface between pXar and EUDAQ.
This has been achieved by implementing CMSPixelProducer class which derives from the EUDAQ standard producer class and has full access to the pXar software to control the testboard and thus the detector chip. The CMSPixelProducer allows to configure a ROC with configuration parameters specified in a EUDAQ configuration file, as well as to start and stop data acquisition runs from the EUDAQ run control.
In principle, EUDAQ requires all devices to be triggered by the same trigger logic unit (TLU) which globally defines an event. This allows to merge all file streams event by event. Unfortunately, a hardware trigger is not implemented on the PSI testboard yet. Instead the pixel ROC needs to be run in an auto trigger mode. This is acceptable during writing and testing the software, but needs to be changed before the DTB can actually be used in the testbeam.
In the auto trigger mode it is also impossible to synchronize the events recorded by the CMSPixelProducer with the events defined by the TLU and used by the EUDAQ Data Collector to merge data streams. Thus the CMSPixelProducer data stream cannot be merged with all other file streams on-line. As a temporary solution the CMSPixelProducer allows to write the data directly into a separate data file. Compiler flags in the producer class make it possible to either read out the full buffer when it's full or at the end of a run or to read out and write event by event. Data can be written in binary and ASCII format.
In preparation for the implementation of an hardware trigger, the interface between the CMSPixelProducer and the EUDAQ Data Collector has also been implemented. Alternatively this interface can be used to read out the CMS pixel data with a second EUDAQ Data Collector which only receives the CMS pixel data stream and thus doesn't have synchronization problems. Central part of this interface is a CMSPixelDecoderPlugin, which is used to decode the binary event data read out from the buffer of the CMS pixel detector and store it into the standardized EUDAQ StandardEvent format.
In principle these events can be viewed with one of the EUDAQ event monitors (offline and online). However, as for today these monitors cannot deal with events that do not contain any TLU information. An issue that hopefully can be fixed in the EUDAQ software.
Figure 10 shows the integration of the CMS pixel detector into the EUDAQ framework as it is desired. The filled blue boxes have been implemented within this project, lines

indicate existing interfaces that allow for communication. The lightning bolts indicate existing problems that need to be fixed. These are:

1. As was already discussed, triggering the PSI testboard externally is not possible, yet.

2. As a consequence, the CMS pixel detector data stream cannot be included into the global EUDAQ data stream.

3. To make matters worse, the testboard can only be read out via USB. Establishing a USB connection takes approximately 1ms, which is consequently the time needed to read out one event. This of course limits the possible data rate to 1kHz which needs to be compared to the current limit of around 8kHz caused by the $115\mu$s needed to read out the MIMOSA sensors.

4. Up to date, the EUDAQ data monitors cannot handle data files that only contain CMS pixel events.

# References


[1] CMS public home page: *http://cms.web.cern.ch/*

[2] S. Chatrchyan *et al.* [CMS Collaboration], "The CMS experiment at the CERN LHC", J. Inst. 3, S08004 (2008).

[3] M. A. Borgia [CMS Collaboration], "Operation and performance of the CMS silicon tracking detector," Nucl. Instrum. Meth. A **658**, 2 (2011).

[4] "The LHC's first long run": *http://cerncourier.com/cws/article/cern/54381*

[5] A. Dominguez *et al.* [CMS Collaboration], "CMS Technical Design Report for the Pixel Detector Upgrade", CERN-LHCC-2012-016.

[6] DESY testbeam website: *http://testbeam.desy.de/*

[7] I. Rubinskiy [on behalf of EUDET, AID Aconsortia Collaboration], "An EUDET/AIDA Pixel Beam Telescope for Detector Development," Phys. Procedia **37**, 923 (2012).

[8] EUDAQ repository: https://github.com/eudaq/eudaq

[9] pXar repository: https://github.com/psi46/pxar