Status and Performance of the Compact Muon Solenoid Pixel Detector

Souvik Das[a], for the CMS Pixel Collaboration

[a] *Cornell University, Ithaca, NY 14853, USA*

Abstract

The Compact Muon Solenoid (CMS) experiment at the LHC incorporates a 66 million channel silicon pixel detector at its center for track seeding and precise vertexing. The hardware and data acquisition software of the pixel detector are briefly introduced in this paper. Experience with installation is succinctly recounted and some of the calibrations performed during the commissioning period are described. Preliminary performance numbers like charge collection and hit residuals from cosmic ray tracks are presented.

Key words:
Compact Muon Solenoid, CMS Pixel Detector

1. The CMS Pixel Detector Hardware

The 66 million channel silicon pixel detector at the center of the CMS apparatus consists of a central barrel called the BPix and two pairs of end-cap disks called the FPix. The BPix is made of three 53 cm long concentric cylindrical layers located at mean radii of 4.3 cm, 7.2 cm and 11.0 cm. The FPix consists of two pairs of disks placed 34.5 cm and 46.5 cm from the center of the barrel. Together they cover pseudorapidities of up to 2.5 around Interaction Point 5 of the LHC as illustrated in Figure 1. A discussion of the detector geometry can be found in [1].

The detector is segmented into 15,840 Readout Chips (ROCs). Each ROC has 4,160 pixels bump-bonded to it in 80 rows and 52 columns. All pixels have the same dimensions of 150 μm (length) $\times$ 100 μm (width) $\times$ 280 μm (depth). The length points along the z direction in the BPix and the ϕ direction in the FPix. The front-end electronics on the detector are connected to the control and readout boards in the counting room through optical links, digital for control and analog for readout. A detailed description of the readout chip can be found in [2].

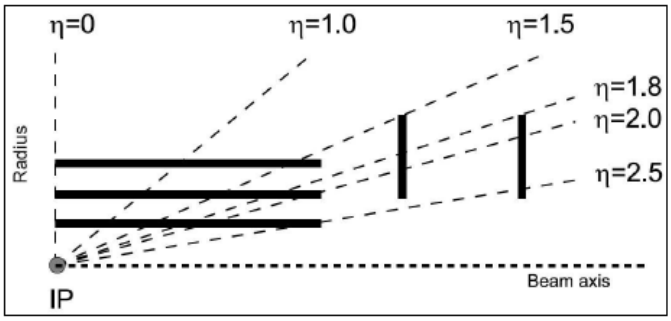

Figure 1: Pseudorapidity coverage of the CMS pixel detector.

2. The Data Acquisition Software

The CMS pixel detector's Data Acquisition Software, also known as the Pixel Online Software, is based on two libraries: XDAQ (see [3]), a Cross-platform Data Acquisition library written in C++ and RCMS, a Run Control and Monitoring System library written in Java.

Apart from being a C++ library of data acquisition tools, XDAQ incorporates a web-server executable that XDAQ-based applications plug into at runtime. This allows us to monitor and control these applications over the world wide web. XDAQ applications are typically used to communicate with VME crate controllers that control electronic boards in crates in the counting room, which in turn program or receive readout from the front-end electronics on the pixel detector. One crate of electronic boards is controlled by one PC with one XDAQ application. Front End Controller (FEC) boards used to control and program the ROCs are supervised by a XDAQ application called the *PixelFECSupervisor*. Front End Driver (FED) boards used to receive and digitize data from the ROCs are supervised by the *PixelFEDSupervisor*. Timing and Trigger Control CMS Interface (TTCci) boards that relay or generate clock and trigger signals are supervised by the *PixelTTCSupervisor*. All of the above applications run on different PCs that have access to the VME crate controllers and are coordinated for calibrations and taking data by one XDAQ application, *PixelSupervisor*, running on a central PC.

Java applications based on the RCMS library and hosted by Apache Tomcat servers form the web-interface for CMS Run Control. An RCMS application called *PixelFunctionManager* serves as the interface between *PixelSupervisor* and CMS Run Control. It also starts up the various XDAQ

Email address: sd259@cornell.edu (Souvik Das)

applications mentioned above on their respective PCs.

3. Installation and Quick Checkout of the Pixel Detector

The BPix detector, which was assembled at the Paul Scherrer Institut (PSI, Switzerland), was lowered into the detector cavern on July 23, 2008. It was slid into place on rails that had been installed between the strip tracker and the beam pipe. Its cooling tubes, power cables and optical fibers were then connected. Preliminary tests of clock and trigger distribution, front end programmability, quality of readout signals and bias connections using standalone software developed at PSI allowed us to spot and correct faulty power cables and dirty optical fibers immediately. By July 29, 2008, the BPix worked sufficiently well to begin installation of the FPix.

The FPix, which was assembled at Fermilab (USA), was lowered into the cavern in four parts. The FPix services were connected by July 31st, 2008. The Pixel Online Software was used to verify the mapping of the data fibers, currents drawn by the ROCs and the stability of the analog readout. On the August 7th, 2008 all access to the pixel detector were relinquished due to the very tight schedule of closing the CMS detector.

4. Calibrations during Commissioning

Several calibrations were performed on the pixel detector over the next two weeks in order to prepare it for taking cosmic ray data in the CMS global runs. A few of the interesting and important calibrations are described below.

4.1. Address Levels Calibration

The pixel ROC uses analog signals to report the position and charge deposit of every hit. In order for the FED to decipher the hit's position it must download 6 ADC levels as thresholds. These levels are discovered by injecting charge into a combination of pixels on a ROC using a calibration capacitor and noting the amplitudes of the analog signal that encode the hit's position. The combination of pixels is chosen such that all analog amplitudes are represented and possibly problematic sequences like the succession of high and low amplitudes are explored. Figure 2 shows the distribution of these 6 ADC levels sampled from a typical Address Levels Calibration.

Address level peaks may be broadened if the FED samples the analog signal at the wrong time, or if the signal is jittery due to unclean optical fibers.

4.2. S-Curve Calibration

The pixel detector depends on charge-sharing between neighboring pixels to accurately reconstruct the position of each hit. Hence the charge detection threshold and noise of each pixel are important parameters to record. These

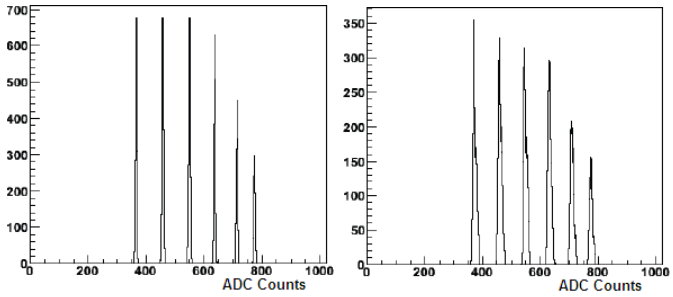

Figure 2: Distribution of address levels. (Left) Sharp address level peaks. (Right) Poor address level peaks.

are determined using the S-Curve calibration where we vary the amplitude of the charge injected into each pixel, specified by the ROC setting VCAL, and record the efficiency of response for that pixel as a function of VCAL. A turn-on curve as shown in Figure 3 is generated per pixel. The VCAL value for which the pixel responds 50% of the time is defined as its threshold. The width of the turn on region is defined as its noise.

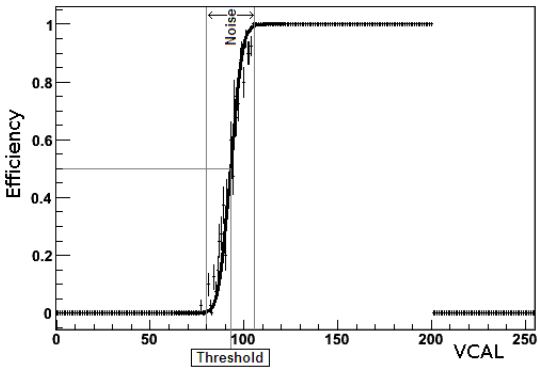

Figure 3: S-Curve with an exaggerated turn-on region.

To express the efficiency with respect to charge, one must measure how much charge a VCAL unit corresponds to in test-beam studies. Equation 1 summarizes this measurement, details of which can be found in [4].

$$charge = VCAL \times 65.5e^- - 414e^- \quad (1)$$

The distribution of thresholds for the FPix and BPix are presented in Figure 4. The distribution of noise for the FPix and BPix are presented in Figure 5. The mean and RMS of the threshold and noise in the FPix and BPix are tabulated in e^- units in Table 1. It may be noted that the thresholds and noise are well below the minimum ionizing particle of 22,000 e^-.

	Threshold		Noise	
	Mean	RMS	Mean	RMS
FPix	2870	200	85	26
BPix	2910	80	141	35

Table 1: Threshold and Noise in e^- units.

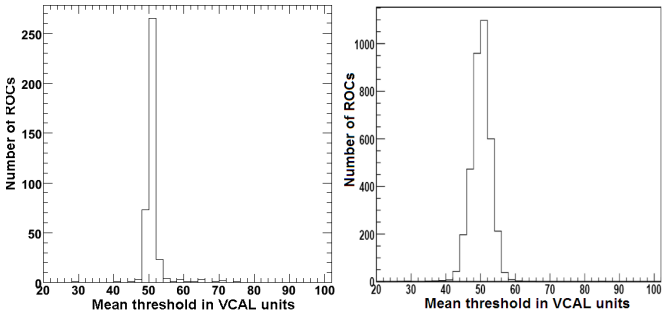

Figure 4: Threshold distribution of the BPix on the left and FPix on the right.

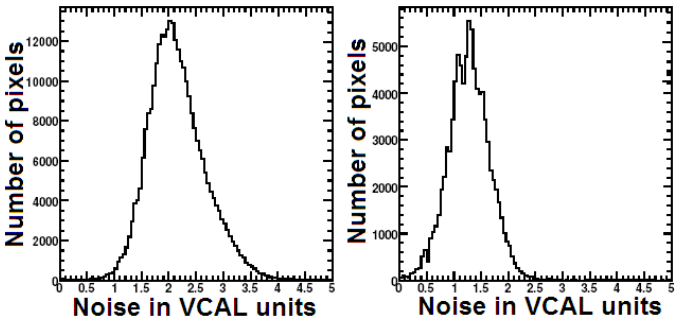

Figure 5: Noise distribution of the BPix on the left and FPix on the right.

4.3. Gain Calibration

The correlation between charge deposited (or injected) in a pixel sensor and the pulse height in the analog signal of the ROC is quantified by the Gain Calibration. Figure 6 is a Gain Calibration curve that illustrates this response curve for a generic pixel.

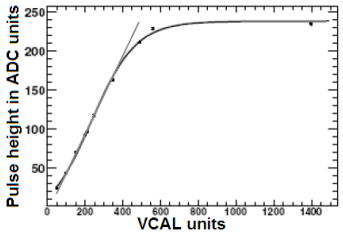

Figure 6: Gain calibration curve fitted to a straight line.

We fit the linear region of the curve with a straight line and store the slope and offset per pixel. An accurate representation of this correlation allows us to improve hit resolution and compute the charge collection in clusters.

5. Performance with Cosmic Data Taking

The results presented below are based on data taken with cosmic rays during a global run of CMS with its solenoid ramped up to 3.8 T. 370 million events were reconstructed of which 80,000 tracks were reconstructed in the pixel detector.

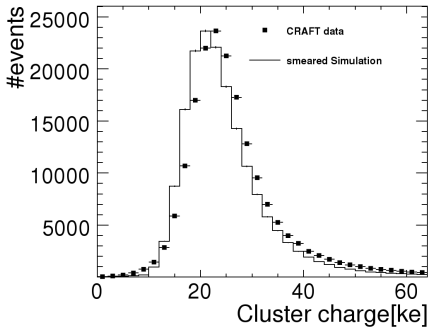

Figure 7: BPix Cluster Charge in cosmic run data and simulation of the same with vertex smearing. Two or more pixel clusters were required per hit, single clusters were discarded for this plot.

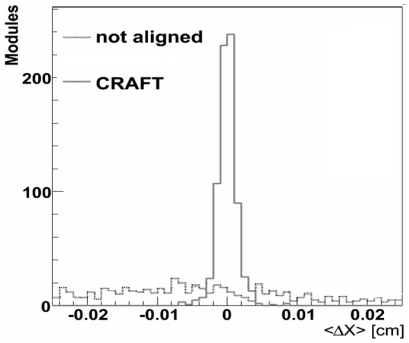

Figure 8: Distribution of the median of x-residuals in the BPix.

5.1. Charge Collection

Using the calibration constants described in section 4.3, we can measure the charge deposition in pixel clusters. Figure 7 shows the Landau-like distribution of cluster charge in the BPix normalized to the path traversed by the track within the pixel sensor.

5.2. Track Quality

For all hits that make up a track, the distance between the hit and its expected location is defined as the residual for that hit. Figure 8 shows the distribution of the median of residuals in the BPix along the local x-axis of its ROCs. Alignment with the cosmic run (CRAFT) data indicate an achievable resolution of 14 μm.

Acknowledgments

I thank Danek Kotlinski, Anders Ryd and the rest of the CMS Pixel Collaboration for pointing me to our latest results and discussing them with me. This work was supported by the NSF grant PHY-0757894.

References

[1] The CMS Collaboration, *CMS Tracker Technical Design Report*, CERN/LHCC 98-6.
[2] H.Chr. Kästli *et al.*, Nucl. Instrum. Methods A 565 (2006) 188.
[3] J. Gutleber and L. Orsini, Cluster Computing 5, (2002) 55.
[4] P.Trueb, PhD Dissertation No.17985, ETH Zurich, (2008) 48