1 Performance Studies Based on Phase 1 Pixel Detector

The pixel detector will be upgraded with a geometry of four barrel layers, three forward disks, and three backward disks. Our simulations consider the upgraded pixel detector and scenarios with high number of collisions per bunch crossing (“pile-up”). The study shows the occupancy of the pixel detector as function of the pile-up. It allows us to estimate data rates and lead us to think of strategies for rejecting pile-up from interested events.

1.1 Production of Monte Carlo Samples

The simulation of Monte Carlo samples require two steps:

- A sample of electron-gun is produced in a first step. Here, one electron per event is produced in generator level together with the *GenParticles* collection. The deposities of energy in the detector components are simulated by Geant4.

- The second step uses the electron-gun samples as input. The deposities of energy are translated into detector signals (`DIGIS`). The digital signals are converted into a `RAW` format, which is the same format provided by the online system.

A offical electron-gun sample, with E_T between 2 and 50 GeV, is used as input to the second step. Here the pile up is added to the electron-gun sample. In order to simulate a scenario of 140 pile up, a total of 140 minimum bias are summed to. Table 1 lists the official electron and minimum bias samples.

Tabela 1: Official MC samples (`GEN-SIM` for electron-gun and Minimum Bias) used to create `DIGI-RAW` and `Ntuple` files.

Description	Number of Events	Official Samples
GEN-SIM	50K	/SingleElectronFlatPt0p2To50/UpgFall13-POSTLS261_V2-v1/GEN-SIM
Minimum Bias	30M	/MinBias_TuneZ2star_14TeV-pythia6/UpgFall13-POSTLS261_V2-v1/GEN-SIM

Output files of the second step have a `DIGI-RAW` format. Such samples are produced using configuration files created with a `CMSSW` tool called *cmsDriver*. Table 2 provides a list of *datasetpath* for `CRAB` corresponding to the `DIGI-RAW` files in four different pile up scenarios: 0, 35, 70 and 140 pile up. Since outputs are big files — a total size range from 12 to 400 GB — they are produced via `CRAB`. These files are located at T2_SPRACE site and each sample of pile up contains 50 K events.

Production of the TTree structure goes in a third step with the full event reconstruction, where the `DIGI-RAW` samples are taken as input into configuration files also

Tabela 2: DIGI-RAW MC samples created from official electron-gun and Minimum Bias MC samples. CRAB *datasetpaths* are provided for 0, 35, 70 and 140 pile up from top to bottom lines. Each sample has 50K events.

Data Set Paths
/SingleElectronFlatPt0p2To50/adesouza-SingleElectron_noPileUP_50K_DIGI_RAW_v6-a49c986a608faae0f4ff11329e2bb83f/USER
/SingleElectronFlatPt0p2To50/adesouza-SingleElectron_PU35_50K_DIGI_RAW_v6-61196276663e836a1b830c8b84a619d4/USER
/SingleElectronFlatPt0p2To50/adesouza-SingleElectron_PU70_50K_DIGI_RAW_v6-3752475e524a60330e4a83a2952b3625/USER
/SingleElectronFlatPt0p2To50/adesouza-SingleElectron_PU140_50K_DIGI_RAW_v6-f74d2f3945f4866a2ea25695b38d79f0/USER

created via *cmsDriver*. Four different samples, corresponding to each pile up scenario, are produced via CRAB. They are located in T2_SPRACE with sizes varying from 24 MB to 22 GB. This is the file path to them:

/pnfs/sprace.org.br/data/cms/store/user/adesouza/SingleElectronFlatPt0p2To50/MergedNtuples/.

Figura 1: RZ view of reconstructed hits in the pixel detector.

Figura 2: Pile up distributions in four different scenarios: zero (top left), 35 (top right), 70 (bottom left) and 140 (bottom right) pile up.

Tabela 3: Beam spot position and associated uncertainty.

1.2 The L1PixelTrigger Analyzer

An EDAnalyzer is a module in the CMSSW framework that allows to extract information from datasets containing simulated or collision data. The extracted information is usually saved in files browsable by ROOT, and organized in a TTree structure. The TTree may contain as many branches as needed for the specific analysis.

To build the *L1PixelTrigger* we took elements from other analyzers available in the CMSSW, between them are the *PixelTree* provided by the Tracker Detector Performance Group (DPG), and the *TrackTriggerStudy* provided by the Track Trigger Integration (TTI) group.

1.2.1 Code and Documentation

The *L1PixelTrigger* analyzer was developed in CMSSW_6_1_2_SLHC6_patch1, with the routine *mkedanlzr L1PixelTrigger*. The source code, named *L1PixelTrigger.cc*, is available at *https://github.com/jruizvar/pixel-analysis/*.

The collections retrieved by the analyzer are:

- ***PileupSummaryInfo***: Provides the number of interaction per bunch crossing. This collection stores sixteen bunch crossings (12 early, one in-time, three late). To retrieve the pileup distribution, we have to loop into the collection and use the method `getPU_NumInteractions()` for each bunch. The associated branch in the ROOT file is called *pileup*.

- ***BeamSpot***: Provides the position and error of the beam spot. The associated branches are *beamSpotX0*, *beamSpotX0Error*, *beamSpotY0*, *beamSpotY0Error*, *beamSpotZ0*, and *beamSpotZ0Error*. This collection also provides details about the beam. The width and error in the transverse plane are given by *beamWidthX*, *beamWidthXError*, *beamWidthY*, and *beamWidthYError*. The spread and error along the Z direction is given by *beamSigmaZ*, and *beamSigmaZError*.

- ***GenParticleCollection***: Provides information of the event at the generator level. The multiplicity of *GenParticles* corresponds to the size of the collection, and is stored in the branch *genPartN*. Relevant information is stored in the branches *genPartEt*, *genPartPt*, *genPartEta*, *genPartPhi*, *genPartCharge*, and *genPartId*. The *GenParticleCollection* does not include the underlying particles in samples with pileup.

- ***L1EmParticleCollection***: Provides information of the event as measured by the electromagnetic calorimeter. The size of the collection is stored in the branch *egN*. Relevant information is stored in the branches *egE, egEt, egEta, egPhi*, and *egCharge*. The method `getCalorimeterPosition()` taken from the *TrackTriggerStudy* analyzer, is used to translate from cylindrical to cartesian coordinates. The output of this method is stored in the branches *egGx, egGy, egGz* and corresponds to the global position in the calorimeter.

- ***SiPixelRecHitCollection***: Provides information of the event as measured by the pixel detector. The multiplicity of reconstructed hits in the barrel is store in the branch *bHitN*. The global position is stored in the branches *bHitGx, bHitGy, bHitGz*. Relevant branches associated with the pixel barrel are *bHitLayer, bHitLadder, bHitModule*. The cluster size is stored in the branch *bClSize* .
  The multiplicity of reconstructed hits in the endcap is store in the branch *fHitN*. The global position is stored in the branches *fHitGx, fHitGy, fHitGz*. Relevant branches associated with the pixel endcap are *fHitDisk, fHitBlade, fHitModule*. The cluster size is stored in the branch *fClSize*.

1.3 Results on Phase 1 Pixel Detector Performances

- Results based on CMSSW_6 with emphasis on the effect of increasing luminosity
- and different pixel cluster algorithms.

Figura 3: E_T, η and ϕ distributions of generated particles (using single electron-gun).

Figura 4: E_T, η and ϕ distributions of L1EM particles (using single electron-gun) for 0 pile up.

Figura 5: E_T, η and ϕ distributions of L1EM particles (using single electron-gun) for 35 pile up.

Figura 6: E_T, η and ϕ distributions of L1EM particles (using single electron-gun) for 70 pile up.

Figura 7: E_T, η and ϕ distributions of L1EM particles (using single electron-gun) for 140 pile up.

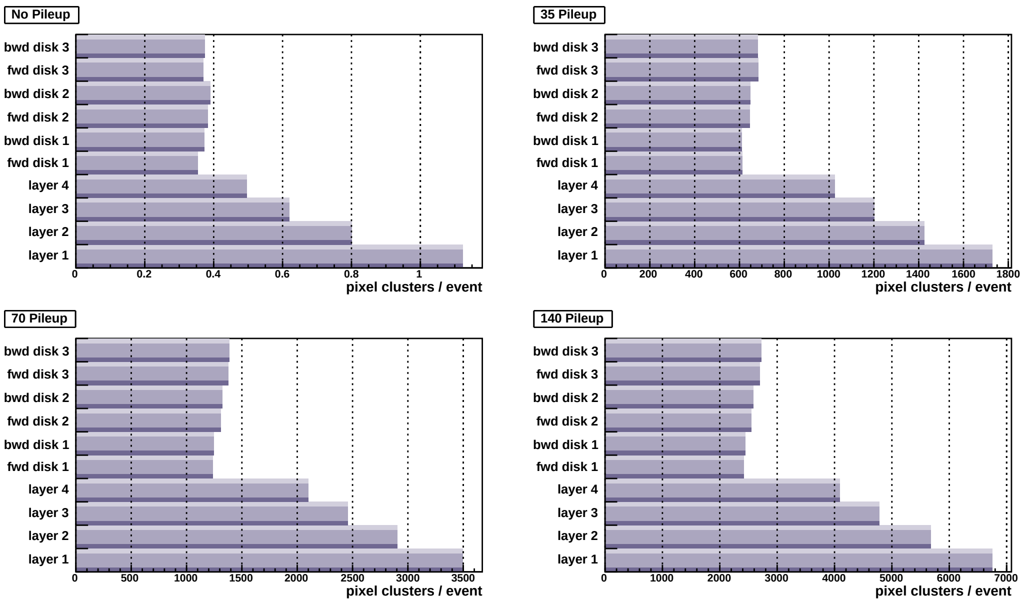

Figura 8: Number of pixel clusters per event in the different components of the pixel detector left by a single electron. Forward $(z > 0)$ and backward $(z < 0)$ disks are taken separately.