Department of Physics and Astronomy
Heidelberg University

Bachelor Thesis in Physics
submitted by

Jens Cravan Cisneros

born in Valencia (Spain)

2025

Hit Rate Study of an Amplifier Model in a Combined Allpix^2 and LTspice Simulation

This Bachelor Thesis has been carried out by
Jens Cravan Cisneros
at the Physikalisches Institut (PI) in Heidelberg
under the supervision of
Prof. Dr. Ulrich Uwer

Abstract

LHCb Upgrade II is planned to be completed during LHC Long Shutdown 4 (LS4), bringing with it improvements to the tracking subsystems aimed at ensuring excellent performance under the high-luminosity conditions expected during the operation of the HL-LHC. The proposed Mighty Tracker (MT) and Upstream Pixel (UP) subdetectors are foreseen to incorporate silicon pixel sensors, and the use of the same HV-MAPS based sensor for both systems is under discussion.

This thesis presents a workflow combining a charge deposition simulation in Allpix^2 with an analogue amplifier circuit simulation in LTspice. A temporal hit distribution is calculated based on the 40 MHz LHC bunch crossing frequency and a range of per-pixel hitrates. The maximum per-pixel hitrate for a (84×84)µm pixel located in a high particle rate section of the UP is calculated to be 26 460 Hz. It is shown that, for the maximum hitrate estimate, the hit-detection efficiency of the simulated circuit is $(96.858 \pm 0.012)\%$ or lower for discrimination thresholds up to 150 mV. Moreover, the hitrate and threshold dependence of the hit-detection efficiency is studied for the simulated amplifier, concluding that hit-detection efficiencies higher than 99% are only achievable for per-pixel hitrates below 10 kHz due to pulse pile-up.

Zusammenfassung

Die zur vierten Wartungspause des LHC geplante Aufrüstung des LHCb Detektors, zu Englisch LHCb Upgrade II genannt, soll Verbesserungen mit sich bringen, die eine ausgezeichnete Leistung selbst im Rahmen der erhöhten Luminosität während des Betriebs des HL-LHC sicherstellen. Die vorgeschlagenen verbesserten Spurrekonstruktionsdetektoren namens Mighty Tracker (MT) und Upstream Pixel (UP) sollen im Rahmen dieser Aufrüstung mit Silizium Pixelsensoren ausgestattet werden, wobei die Verwendung des gleichen HV-MAPS basierten Sensortyps für beide Systeme in Erwägung gezogen wird.

Diese Arbeit stellt eine Reihe an Simulationsverfahren vor, welche eine Ladungsträgerdeponierungssimulation in Allpix^2 mit einer analogen Verstärkersimulation in LTspice vereint. Eine zeitliche Verteilung der detektierten Treffer wird auf Grundlage der LHC Bündelkollisionsrate von 40 MHz und einer Reihe an Trefferraten berechnet. Die maximale Einzelpixel-Trefferrate für einen (84×84)µm großen Pixel, der sich in einem durch Teilchen hochfrequentierten Bereich des UP befindet, wird zu 26 460 Hz geschätzt. Es wird gezeigt, dass die Teilchendetektionseffizienz für diese maximale Trefferrate bei einer Diskriminatorschwelle von 150 mV weniger als $(96.858 \pm 0.012)\%$ beträgt. Des weiteren wird die Teilchendetektionseffizienz des simulierten Verstärkers in Abhängigkeit der Trefferrate und Diskriminatorschwelle untersucht, wobei als Ergebnis herausgefunden wird, dass eine Effizienz von über 99% nur für Trefferraten unter 10 kHz erreichbar ist.

Contents

Chapter 1

Introduction

The LHCb experiment, located at LHC interaction point 8, provides cutting-edge measurements of CP violation and heavy quark physics, which are crucial to our developing understanding of physics beyond the Standard Model. In pursuit of ever more precise and insightful discoveries, the Large Hadron Collider will receive an upgrade to its luminosity, resulting in an increased particle rate at each of its interaction points. With the goal of optimizing detector performance for the increased luminosity during the operation of the High Luminosity LHC (HL-LHC), the LHCb detector is planned to receive an upgrade (LHCb Upgrade II) during LHC Long Shutdown 4 (LS4), which will prepare the detector to operate efficiently in the more demanding environment created by the increased particle rate.

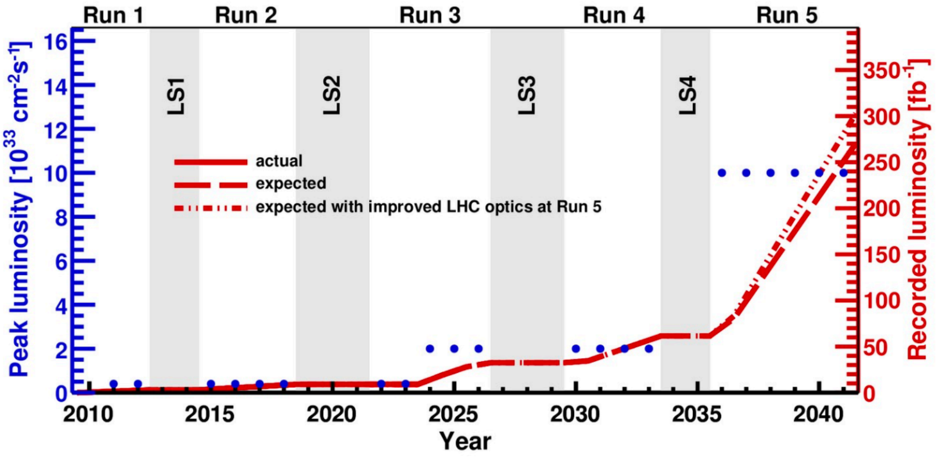


Figure 1.1: Timeline showing planned LHC Runs, Long Shutdowns, and luminosity values. [1]

As part of LHCb Upgrade II, the Scintillating Fiber (SciFi) and Upstream Tracker (UT) tracking subdetectors will receive improvements in the form of silicon pixel sensors, resulting in new detectors called Mighty Tracker (MT) and Upstream Pixel (UP). For the Mighty Tracker, monolithic sensors based on the HV-MAPS technology, called MightyPix, are under development. The same sensor type is also being discussed as a candidate for the Upstream Pixel, where a significantly higher particle rate is expected, particularly in the regions of the subdetector closest to the beam pipe.

The study presented in this thesis aims to quantify the extent to which a MightyPix2-type sensor is able to provide the hit-detection efficiency required to operate in the highest occupancy regions of the UP by investigating the performance of the analogue pixel front-end at high hitrates.

A particle interaction simulation realized in Allpix^2 is combined with an analogue circuit simulation in LTspice to provide a detailed representation of the charge amplification behavior for a realistic range of charge deposition values. The amplifier output is simulated as a continuous signal in order to investigate the impact of pulse pile-up on the observed inefficiency. A method for calculating temporal hit distributions based on a desired hitrate value by assuming discrete timegaps between subsequent hits, emulating the conditions arising from the LHC bunch crossing frequency of 40 MHz, is presented. An estimate for the highest per-pixel hitrate for a sensor installed in the UP is given, and the simulated hit-detection efficiency for this value is discussed. The effect of different discriminator threshold settings on the efficiency of the simulated amplifier circuit is shown, and a function describing the data is parametrized such that the predicted hit-detection efficiency can be calculated for a combination of hitrate and threshold values.

Chapter 2

Key Concepts

2.1 The LHCb Detector

The LHCb detector is a spectrometer designed to probe particles along a narrow cone in the forward beam direction, exploiting the directionality of some high-energy particle production processes, such as the production of b-flavored hadrons [2]. The detector incorporates multiple subsystems that work together to accomplish several tasks, including particle identification (PID) and particle tracking. The LHCb detector is scheduled to receive a systems upgrade, referred to as LHCb Upgrade II, during LHC Long Shutdown 4 in preparation for the increased particle rate expected during the operation of the High-Luminosity LHC. A schematic representation of the planned detector structure following Upgrade II is shown in Figure 2.1.

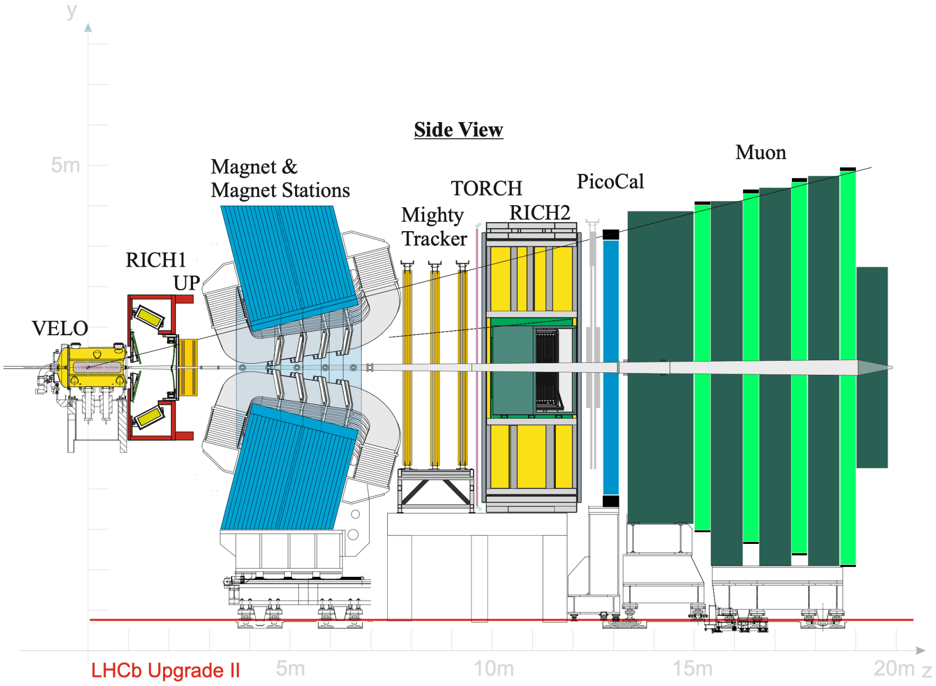


Figure 2.1: Schematic of the LHCb detector showing its structure and the position of main systems planned for LHCb Upgrade II. Image taken from [3].

With a peak instantaneous luminosity target of $1.0 \times 10^{34}\text{cm}^{-2}\text{s}^{-1}$ [1] planned for HL-LHC, the particle tracking subdetectors installed during Upgrade II must be able to withstand and efficiently detect particle rates significantly higher than those experienced during previous LHC runs. To account for the high particle rates and the associated increase in radiation damage, especially in areas close to the beam pipe, the existing SciFi detector is planned to receive an upgrade, including to its central portion, where silicon pixel sensors will be introduced. The resulting detector, named Mighty Tracker, will use scintillating fibers read out by silicon photomultipliers in the outer tracking regions where lower particle rates are expected, and silicon pixel sensors more suited to withstanding high-radiation environments and capable of providing the increased spatial granularity required for tracking particles at high rates, in the regions close to the beam pipe.

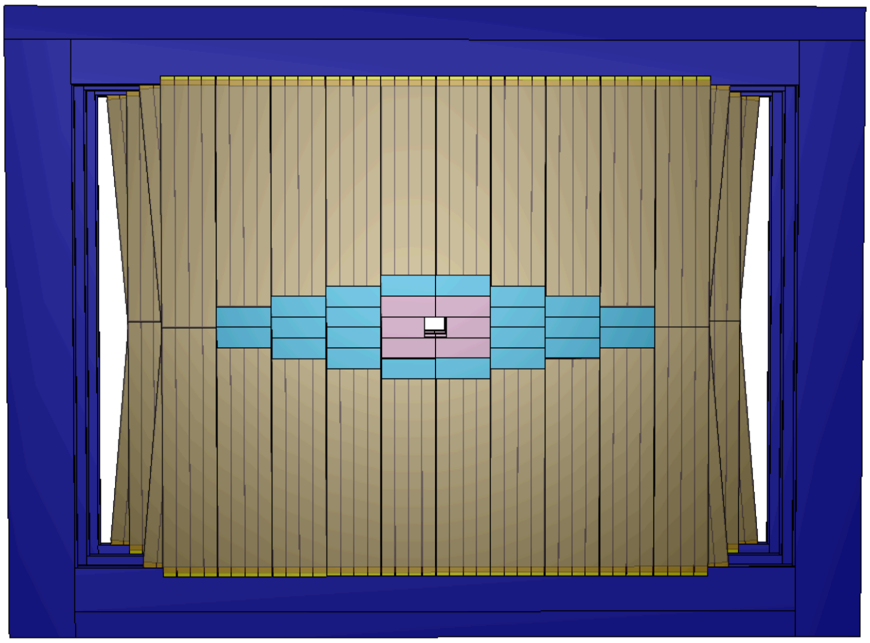


Figure 2.2: Schematic of the planned layout for the Mighty Tracker showing areas covered in silicon pixel sensors (light blue and pink) as well as scintillating fibers (brown). Image taken from [4].

Upstream of the bending magnet and behind the vertex-locator (VELO) and Cherenkov-ring detector 1 (RICH1) currently lies the Upstream Tracker (UT). Used for matching VELO tracks to tracks downstream of the magnet, it plays an important role for reconstructing full particle paths through the entire detector. In its current form, it consists of four layers of silicon strip sensors well suited for track reconstruction at a luminosity of $2 \times 10^{33}\mathrm{cm}^{-2}\mathrm{s}^{-1}$, but unable to deliver the required readout rate for the fivefold increase in luminosity expected following Upgrade II [4]. A new Upstream Tracker design based around silicon pixel sensors, named Upstream Pixel (UP), has been proposed. The use of pixels instead of strips aims to reduce the number of particles able to hit a single sensor matrix element at the same time, and the move to a different sensor technology is foreseen to improve the detector's radiation resistance.

Both the UP and the Mighty Tracker are candidates for the implementation of HV-MAPS sensor technology [4]. Using the same technology in both detectors comes with the advantages inherent to standardization, but the feasibility of HV-MAPS designs such as the MightyPix sensor for use in the high particle rate environment present in the innermost parts of the UP has to be thoroughly investigated.

2.2 Particle Detection

The detection of particles hinges on the interaction of the particle with the medium it traverses. This principle is the basis of all particle detection systems, including the silicon pixel detectors that are the subject of this thesis. Along its travel path, a particle exchanges some of its energy with the material surrounding it, leaving behind a traceable signature that is converted into a signal by the detector. The dominating mode of interaction depends on particle type, particle energy and sensing material, making different detector designs more suited for some applications than others, and vice-versa.

2.2.1 Photons

Photon interaction with matter is governed by three predominant energy loss mechanisms: the photoelectric effect, the Compton effect, and pair production. These processes are not mutually exclusive (assuming the photon is not absorbed immediately during the first interaction), but heavily energy and material dependent. The dominant process for a given photon energy and nuclear charge of the interaction material is shown in Figure 2.3.

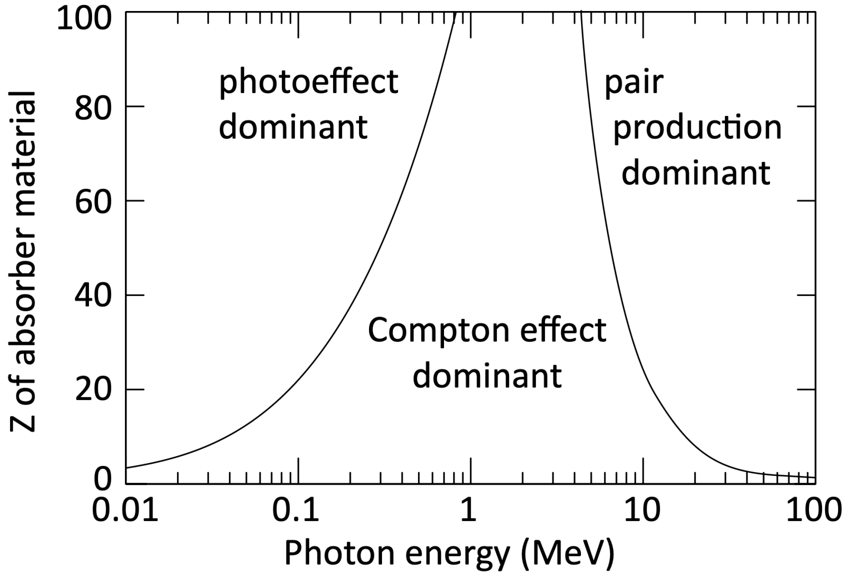


Figure 2.3: Dominant photon energy loss mechanism depending on photon energy and nuclear charge of the interaction material. Image taken from [5].

The photoelectric effect describes the absorption of photons via the complete transfer of their energy to an atom. The excess energy is released from the atom through the emission of an electron, and the vacant spot in the electronic configuration of the atom is filled by an electron from a higher shell under emission of a photon, if energetically possible. In order for a photoelectric absorption to occur, the kinetic energy E_γ of the incoming photon has to be larger than the binding energy E_B of an electron to the atom. The kinetic energy T of the outgoing electron is equal to the energetic gap between the absorbed photon and the bound electron:

$$E_\gamma - E_B = T \tag{2.1}$$

Photoelectric energy loss is the dominant process in the low-energy regime. For higher energies, the elastic scattering of photons off of shell electrons, known as the Compton effect, takes over as the dominant mechanism. The prerequisite for Compton scattering is the treatment of the shell electron as quasi-free, which is the case when the energy of the incoming photon far exceeds the binding energy of the shell electron. During the scattering, the photon transfers some of its energy to the electron and changes trajectory as a consequence. The energy E'_γ of the scattered photon is a function of its pre-scattering energy E_γ, the electron resting energy $m_e c^2$, and the scattering angle θ_γ of the photon measured with respect to the incoming trajectory:

$$E'_\gamma = \frac{E_\gamma}{1 + \frac{E_\gamma}{m_e c^2}\left(1 - \cos(\theta_\gamma)\right)} \tag{2.2}$$

For highly energetic photons, pair production becomes the dominant form of energy loss. Pair production occurs when a photon with an energy higher than twice the electron mass decays into an electron-positron pair through the interaction with an atomic nucleus. The photon energy required for this process is slightly higher than two electron masses, as the kinematics require accounting for the recoil energy transferred to the nucleus during the interaction. The kinetic energy of the outgoing electrons is sourced from the energy surplus following the decay of the photon, and sufficiently energetic electrons can continue causing traceable interactions in the detector. Very high-energy photons can set in motion an interaction cascade, where the number of particles continues increasing due to pair production and subsequent ionization until all energy is transferred to the detector.

2.2.2 Heavy Charged Particles

Electrically charged particles other than electrons and positrons are referred to as "heavy" in allusion to the large mass gap existing between even the lighter of the "heavy" particles and the electron. As a consequence of the mass discrepancy, electrons and positrons are usually treated slightly differently when it comes to describing their interactions with matter.

For heavy charged particles, energy loss through ionization is the dominant process. Ionization describes the interaction of an incoming charged particle with electrons in the atomic hull of the traversed material, ultimately resulting in the ionization of the interaction partner. A parametrization of the mean ionization energy loss per unit length is given by the Bethe-Bloch equation [6]:

$$\left\langle -\frac{dE}{dx} \right\rangle = K z^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \frac{2 m_e c^2 \beta^2 \gamma^2 W_{max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right] \tag{2.3}$$

Several shorthands are used in the equation above. The coefficient K:

$$K = 4\pi N_A r_e^2 m_e c^2 \tag{2.4}$$

Where N_A and r_e are Avogadro's number and the classical electron radius, respectively.
The maximum single-collision energy transfer W_{max}:

$$W_{max} = \frac{2 m_e c^2 \beta^2 \gamma^2}{1 + 2\gamma m_e/M + (m_e/M)^2} \tag{2.5}$$

Where M is the incoming particle's resting mass.
And the mean excitation energy I, whose dependance on the atomic number Z of the absorber can be approximated by a power law [5]:

$$I \approx 17.7\, Z^{0.85}\, \text{eV} \tag{2.6}$$

Other than these, Equation 2.3 is parametrized by the relativistic kinematic variables β (relative velocity) and γ (Lorentz factor), the charge number z of the incident particle, the atomic mass A of the absorber, and the density effect correction factor $\delta(\beta\gamma)$ [6].

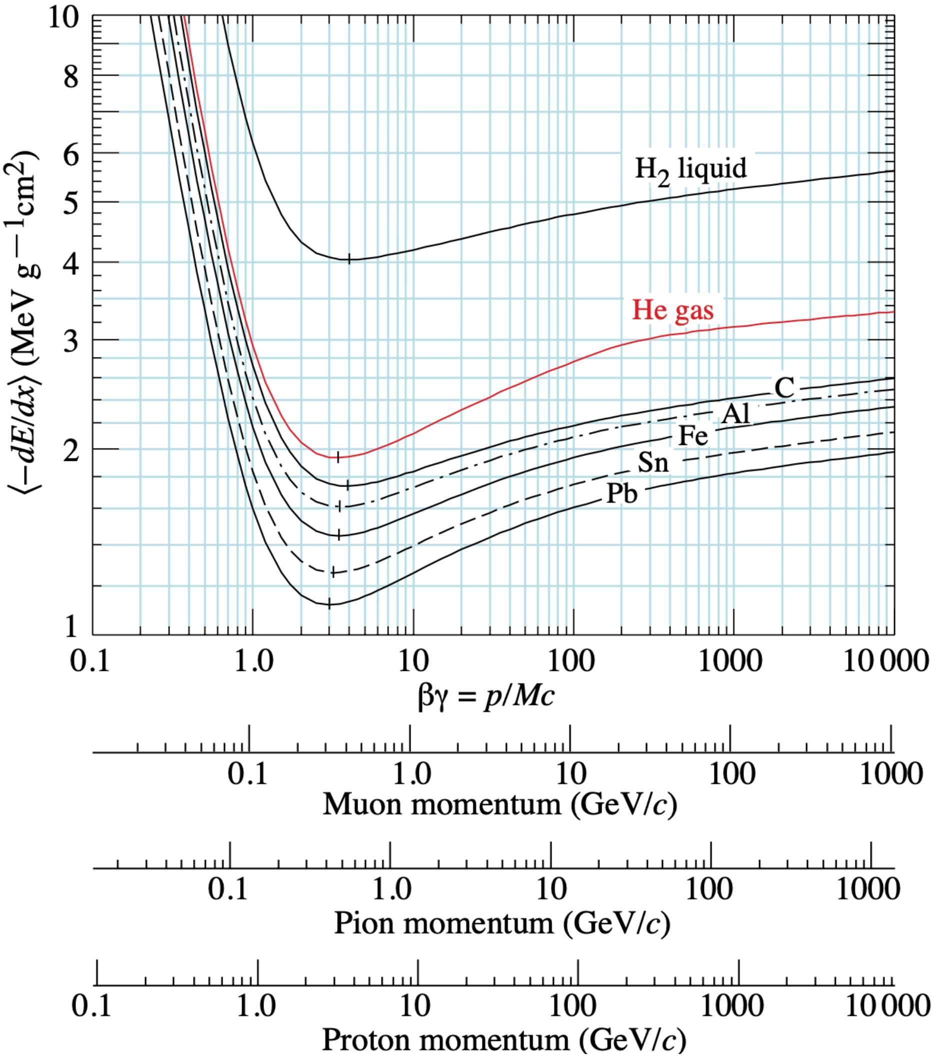


Figure 2.4: Shape of the Bethe-Bloch average ionization energy loss curves for different absorber materials. Image taken from [6].

The energy loss experienced by any individual particle over a fixed interaction path length is subject to a statistical distribution that follows a Landau probability density. This energy loss distribution has a mean matching the value calculated by the Bethe-Bloch function, and a most probable value different from the distribution's mean. In other words, the Landau distribution is asymmetric with a long tail towards high energy deposition values.

2.2.3 Electrons and Positrons

For the much less massive electron, additional contributions to the mean energy loss are made by the emission of electromagnetic radiation by the incoming particle following deflection or deceleration at high velocities (Bremsstrahlung), and the indistinguishability of incoming and outgoing particles during the interaction. In the case of the positron, the additional contributions stem from Bremsstrahlung and the possibility of electron-positron annihilation. The Berger-Seltzer equation [7] [8] incorporates these effects to describe the mean energy loss per unit length for "light" charged particles:

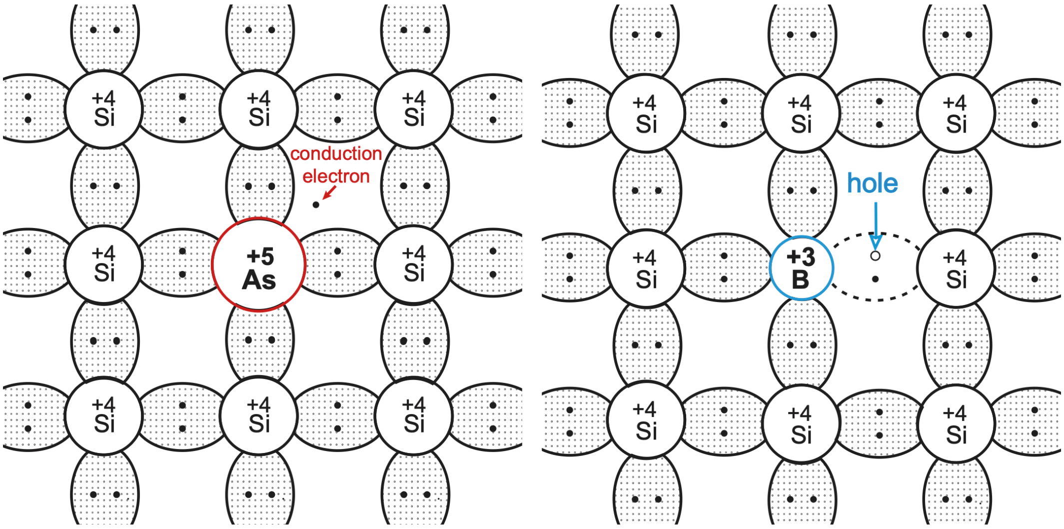
$$\left\langle -\frac{dE}{dx} \right\rangle = \rho \frac{0.153536}{\beta^2} \frac{Z}{A} \left[B_0(T) - 2\ln\left(\frac{I}{m_e c^2}\right) - \delta(\beta\gamma) \right] \tag{2.7}$$

In addition to the quantities carried over from Equation 2.3, the Berger-Seltzer formula is parametrized by the absorber density ρ, and the stopping power B_0, which itself is dependent on particle type (e^+ or e^-), momentum T, and absorber material.

2.3 Semiconductor Physics

Silicon, the material of choice for the construction of many pixel sensor types, including the MightyPix series [9] developed for LHCb, is a semiconductor. Semiconductors are materials whose electric conductivity lies between that of insulators (glass, quartz, etc.) and conductors (silver, copper, etc.). The dense arrangement of semiconductor atoms in a lattice leads to the formation of distinct energy ranges, referred to as energy bands, which are separated by so-called band gaps. The valence and conduction bands, which occupy the highest energy ranges, contribute to the electrical properties of the semiconductor. Semiconductors feature a small energy gap between the valence and conduction band compared to insulators, enabling energy state transitions induced by external influences, such as the energy transfer of a particle traversing a sensor. When an electron transitions from the valence to the conduction band, its nuclear bond is broken, leaving behind a free "hole" with positive charge in addition to the free electron, both able to move through the semiconductor lattice. In silicon, the band gap between the valence and the conduction band is 1.12 eV [5].

The occupation of the conduction and valence bands can be modified by introducing foreign atoms into the semiconductor lattice in a process known as "doping". The introduction of so-called donor atoms leads to an increase in conduction electrons; the semiconductor is thereafter referred to as "n-doped". If acceptor atoms are used instead, an excess of holes in the valence band is produced and the semiconductor is referred to as "p-doped".



(a) n doped (b) p doped

Figure 2.5: Representation of a silicon lattice with donator (As) and acceptor (B) atoms showing the effect on the electronic configuration from different doping. Image taken from [5].

Bringing p-doped and n-doped silicon layers in contact with each other creates a so-called pn junction, where holes from the p-doped material diffuse into the n-doped material and electrons from the n-doped material diffuse into the p-doped material as a consequence of the charge carrier concentration gradient at the contact surface. The result is a volume along the pn junction where neither charge carrier type is found in excess due to recombination: the depletion region. The depth of the depleted volume can be controlled by applying an external voltage to the p- and n-doped material, a process which is known as "biasing". Applying a positive bias voltage $V_b > 0$ to the p-doped silicon substrate, known as forward biasing, leads to a reduction in depletion volume. The opposite, applying a negative voltage $V_b < 0$ to the p-doped substrate, is known as reverse biasing and leads to an increase in depleted volume.

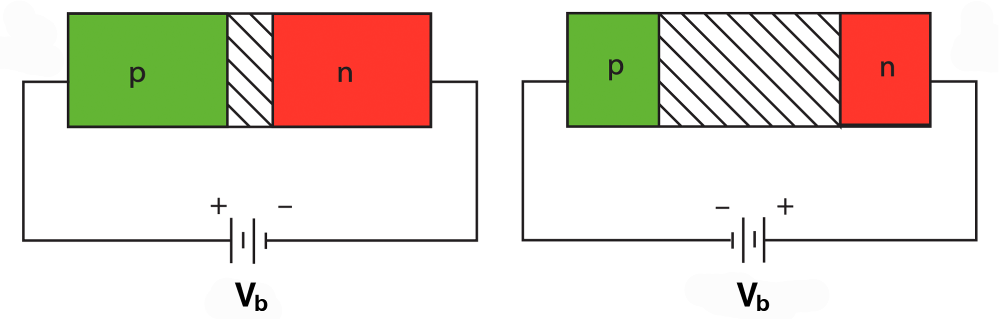


Figure 2.6: Representation of forward biasing (left) and reverse biasing (right) showing the impact on the size of the depletion zone (striped). Image taken from [5], modified.

Operating in reverse bias, the depletion depth d is proportional to the square root of the magnitude of the biasing voltage $|V_b|$, and the substrate resistivity ρ. An approximate formula for the depletion depth is given here [5]:

$$d \approx 0.32 \cdot \sqrt{|V_b| \cdot \rho} \;; \quad [d] = \mu\mathrm{m}; \; [V_b] = \mathrm{Volt}; \; [\rho] = \Omega\,\mathrm{cm} \tag{2.8}$$

An electric field is present only in the depletion region, causing charge carriers set free within its volume to drift in opposite directions, depending on their charge and the polarity of the applied voltage. This property enables the construction of particle detectors based on doped silicon operated in reverse bias, as the enlarged depletion region acts as sensing volume for traversing particles.

2.4 Signal Generation in Semiconductor Detectors

The p- and n-doped substrates of a semiconductor detector act as electrodes for the free charge carriers moving in the electric field of the depletion zone. Due to their opposite electric charges, electrons and holes are attracted in opposing directions, causing them to drift from their point of origin toward one of the doped substrate regions. Drifting charges induce a current in the electrodes, the magnitude of which depends on the total amount of charge moving and the drift velocity of individual charge carriers. This relation is commonly formalized through the Shockley-Ramo theorem [10], which relates the induced current i with the drift velocity $\vec{v}_d$ of a charge q in the following way:

$$i = q\vec{v}_d\vec{E}_w \tag{2.9}$$

The additional quantity $\vec{E}_w$ dictating the induced current is known as a weighting field. It corresponds to the electric field of the electrode seeing the current, calculated with its potential set to 1V and all other electrodes set to 0V. Equation 2.9 implies an important characteristic of signal generation: charge does not have to be collected at the electrode to contribute to the signal. An electron that recombines with a hole during its movement in the electric field will still contribute to the total induced signal, even if only until the moment of recombination.

Silicon pixel sensors divide their sensing volume into a matrix of electrodes whose signals can be read out individually, yielding 2D hit position information. One of the phenomena inherent to this design is the sharing of the charge produced along the ionization path of a single particle between two or more adjacent pixels, which can occur primarily in two ways. The first cause is the intersection of a particle with the sensor at a shallow angle of incidence, leading to charge deposition in multiple pixels. The second cause for charge sharing relates to the movement of free charge carriers deposited outside of the depletion region, where they are not subjected to an electric field. Their movement in regions with no electric field is governed by diffusion, which is essentially an anisotropic random-walk process. Charge carriers can therefore diffuse into the depletion region of any near pixel, where the electric field causes them to drift and contribute to the current or induce a new signal of their own. Diffusive contributions to charge sharing are more common for hits that occur along the edges or in the corners of pixels, where the average path length to the depletion region of a neighboring pixel is short.

2.5 HV-MAPS

Silicon pixel sensors can broadly be divided into two groups: monolithic and hybrid. Whereas hybrid pixel sensors physically separate the sensing region from the electronics and readout, connecting both through metal bump bonds, monolithic sensors incorporate the amplification and often also discrimination electronics into the pixels themselves.

HV-MAPS is short for "high-voltage monolithic active pixel sensor". As the name implies, they fall into the category of monolithic sensors, featuring an amplifier and frequently also a comparator embedded in each pixel, which deliver discriminated signals. A high biasing voltage is applied to the pixels, aiding fast charge collection through charge carrier drift.

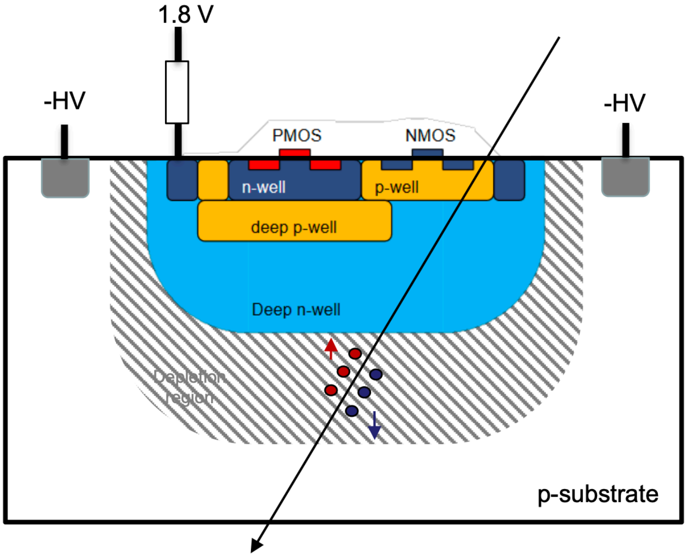


Figure 2.7: Schematic showing the lateral cross-section of an HV-MAPS pixel with CMOS electronics. Depletion region shown in stripes. [11]

As shown in Figure 2.7, depletion occurs at the boundary between a deep well of n-doped substrate and a bulk of p-doped silicon, with a high reverse biasing voltage applied to the latter substrate. The electronics are embedded into the n-well, shielding them from the electric field in the depletion zone.

The in-pixel electronic circuits are realized as PMOS and NMOS transistors. A top-level schematic showing the electric circuit components contained within one HV-MAPS pixel is provided in Figure 2.8.

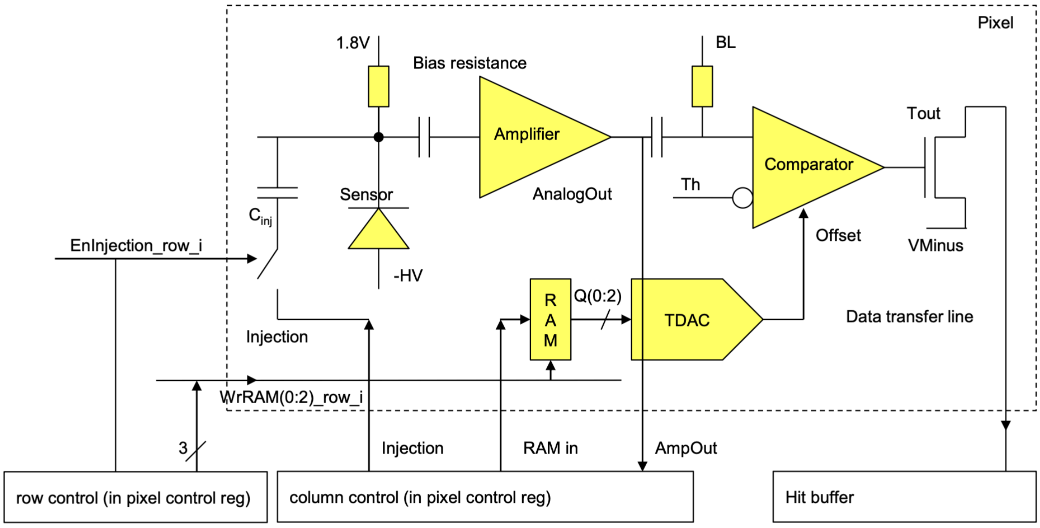


Figure 2.8: Top-level schematic diagram showing the layout of the electronics within a TelePix1 HV-MAPS. Image taken from [11].

The exact layout and design of the electronics circuit can differ between sensors, both as an adaptation to different operational requirements and as continued improvement over multiple design iterations. A more detailed description of the individual electronic circuits of an HV-MAPS can be found here [12].

As of the writing of this thesis, the MightyPix line of HV-MAPS is being developed specifically for use in the Mighty Tracker at LHCb, with the second iteration sensor called MightyPix2 [13] currently being in the submission process.

Chapter 3

Simulation Tools

In the following sections, a concise introduction to the software tools used for data gathering and analysis is given. The aim is not to provide a comprehensive overview of all features these programs offer, but to summarize their purpose and working principles as they relate to this thesis. At the end, an overview of the workflow used to produce the results shown in chapter 5 is provided.

3.1 Allpix^2 - Pixel Sensor Simulation

Allpix^2 [14] is a software package designed for the simulation of silicon pixel sensors. Its modular design enables each step of the simulation process to be addressed and configured individually, allowing for a high degree of specificity regarding the conditions and physical processes simulated. With a focus on fast simulation speeds while maintaining high accuracy and featuring built-in data analysis methods, Allpix^2 serves as a Monte-Carlo based tool for sensor performance evaluation using large datasets.

Each Allpix^2 simulation, henceforth referred to as "Allpix" simulation for brevity, can be understood as a chain of subsequent simulation steps. Every step, represented by a so-called simulation module, carries out its function before communicating the result to the next module, which uses it as input for carrying out further simulation tasks. Broadly speaking, the simulation can be divided into the following steps:

1. Definition of Global Simulation Parameters
2. Particle Interaction/Energy Deposition
3. Free Charge Carrier Propagation
4. Signal Formation and Digitization

In the context of this study, an event can be understood as the simulation of a single particle. The successful execution of steps 1 through 4 constitutes a simulated event, after which the chain can loop back to step 2 for the next event to start. Step 1 needs only be carried out once, at the start of a full run of N events.

Step number 1 defines the fixed boundary conditions underlying the simulation. These include information about the type of sensor, its position in simulated space, and the total number of events to be simulated. Allpix provides the ability to create individualized sensor configurations from scratch, choose from a set of existing ones, or even selectively modify specific aspects of an existing sensor configuration. Among the configurable parameters, pixel geometry, pixel size, and sensor thickness are worth highlighting, since they directly influence charge deposition and the calculated rate of hits on a given pixel. A summary of the most relevant simulation parameters used for generating the charge deposition data is provided in chapter 4, and the complete set is appended in Appendix A.

Having initialized the virtual lab environment with the configured sensor, Allpix moves on to step number 2. This step encompasses the simulation of particle propagation toward the sensor, as well as of particle interaction with the silicon substrate.
In Allpix, particles can be propagated in multiple ways. Most notably for this thesis, by being treated as individual constituents of a particle beam, the dimensions and shape of which are configurable parameters. The distribution of particle positions within a beam's cross section is calculated assuming a gaussian beam profile, in addition to an optional angular beam divergence setting which, when enabled, causes a broadening of the beam by a set angle.

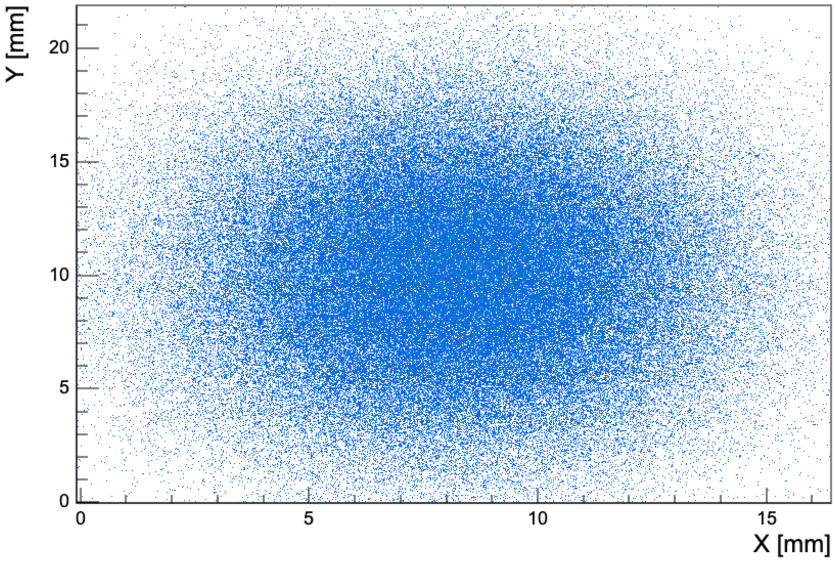


Figure 3.1: Incident particle positions for 10000 events on a MightyPix1 - type sensor. Generated in Allpix^2 by the DepositionGeant4 module.

Figure 3.1 exemplifies in-beam particle distribution by mapping the positions where each particle first intersects the sensor in an Allpix simulation. Each blue dot represents the true simulated incidence position calculated by the beam propagation module, before energy deposition and independent from digitization or clustering. The depicted distribution is intentionally very broad to showcase the decrease in particle density toward the edges of the simulated beam's cross section, and its underlying simulation is not representative of the configuration used for the rest of this study.

Modeling the physical processes that govern a particle's interaction with matter is taken care of by the Geant4 framework [15], a Monte Carlo simulation tool supporting high-energy (HE) physics calculations. It takes into account particle type, energy, and material properties of the sensor along the interaction path to simulate energy loss and the resulting charge deposition in accordance to the phenomena discussed in section 2.2. The accuracy and treatment of interactions for different energy regimes and particle types in Geant4 varies depending on the so-called "physics list" selected: a repository of equations and variables best suited to a specific use case. For use in HE collider physics, the *FTFP_BERT_EMZ* physics list provides the highest accuracy [16], and is therefore used throughout this thesis.

At the end of simulation step Nr.2, both the path of the simulated particle through the sensor, as well as the number of free charge carriers created by ionization and their positions within the sensor's volume, are known. Next, in step 3, the electric field found within a pixel's depletion volume is used in combination with a diffusion model to propagate the free charge carriers through the pixels, beginning at the starting positions calculated during the deposition step.

In principle, Allpix supports the use of realistic electric field maps sourced from Technology Computer Assisted Design (TCAD) tools, that describe the shape and magnitude of the electric field within a pixel based on accurate knowledge of pixel dimensions, substrate characteristics, as well as position and magnitude of applied bias voltages. These electric field maps are used in combination with weighting field maps in order to produce accurate charge carrier drift and signal formation simulations, as the direction and propagation speed of the charge is based on a physically accurate electric field. In practice, however, no TCAD-sourced files matching the pixel dimensions and electric fields simulated for this study were are available as of writing, and producing them lies beyond the scope of this thesis. The alternative choice is a simplified approximation of the electric field within a pixel, which assumes the field strength to follow a linear gradient of decreasing strength with increasing distance to the pixel front-side. In other words, the simplified model assumes a uniform electric field in x and y orientation, and a linear decrease in strength in z direction. The implications of this choice of electric field parametrization are discussed in chapter 4.

When simulating an event, the movement of free charge carriers is calculated either until all of them reach the defined electrodes, or up until a simulated propagation time limit is reached. The final step in the simulation, step 4, then calculates the signal seen by the pixel's front-end electronics and carries out the digitization of the signal, which is further processed for clustering and hit-detection.

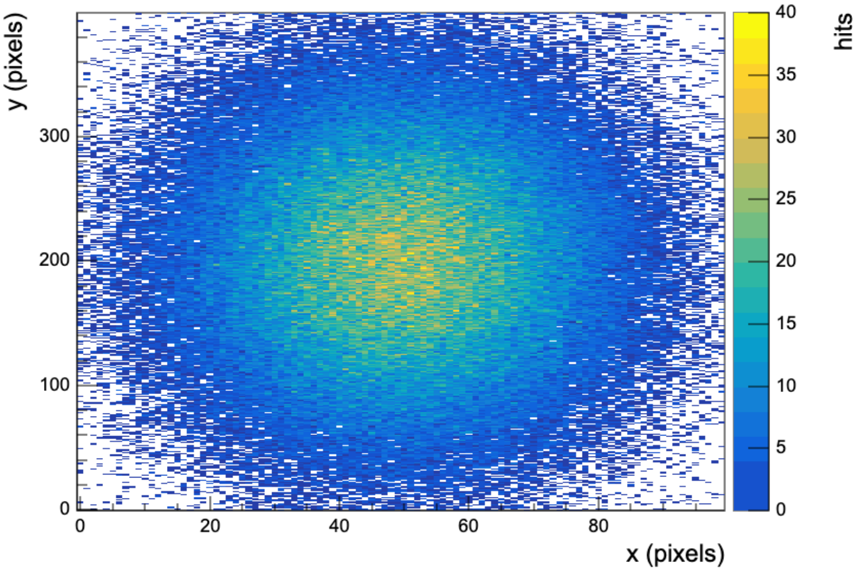


Figure 3.2: Position and number of detected hits for 10000 events on a MightyPix1 - type sensor. Generated in Allpix^2 with built-in digitization.

Figure 3.2 shows the hitmap produced after the signal generation and digitization step in Allpix for the same simulation as Figure 3.1. Being a product of the simulated electronic and digital circuits, the hitmap represents the number of hits per pixel, and only includes those events that manage to cross the detection threshold configured in the digitization module.

When used in conjunction with a weighting field map, Allpix can be configured to calculate the signal seen by the pixel's front end by treating it as a current induced by the motion of free charge carriers. For the linear electric field implemented in this thesis, that method is not applicable and signal formation is therefore limited to counting the total charge that reaches the electrode within the time limit set in the charge propagation step.

The default implementation of the charge-sensitive digitization module included in Allpix relies on a parametrized pulse response function that mimics a generic charge-sensitive preamplifier with Krummenacher feedback [17] [18]. In short, the amplification and digitization work by calculating the convolution of the collected charge pulse and a pulse response function with user-definable parameters, resulting in a voltage pulse shaped by rise and fall response times [19]. This shaped pulse is subjected to a threshold check, which determines time of arrival (ToA) and time over threshold information and stores it as digital values with a user-definable resolution.

While powerful, especially through its tight integration with the rest of the Allpix simulation chain, the customizability of the CSA Amplifier module is limited by its reliance on a parametric pulse response. For simulation tasks aimed at optimizing amplifier designs or testing the behavior of existing amplifiers for a physically realistic spread of input charges, using the built-in Allpix digitization modules demands a constant back and forth between two separate simulation programs: an electronics simulation to measure, tune and parametrize an amplifier response, and the Allpix simulation to implement and test the performance of the parametrized amplifier.

Only recently, with the introduction of the *NetlistWriter* module [20] following the 6th Allpix Squared User Workshop in May 2025, has the interface between Allpix simulations and external electronics simulation programs become streamlined. This new module replaces step 4 in the simulation chain by writing the charge pulses collected in each pixel during an event into a Netlist file, readable by a dedicated electronics simulation based on either SPICE or SPECTRE. This effectively outsources the amplification and digitization at the cost of automatic hit processing and clustering. A more detailed description of the *NetlistWriter* can be found in the following section.

3.2 LTspice - Electric Circuit Simulation

LTspice is an electric circuit simulation and plotting software built around the open-source *Simulation Program with Integrated Circuit Emphasis* (SPICE) simulator [21]. Available for a wide range of operating systems as freeware and featuring a graphical schematic capture front-end as well as waveform plotting, LTspice renders the versatility and speed of SPICE circuit simulation tools approachable.

SPICE simulations are used to analyze and predict the behavior of analogue circuits in response to AC, DC or transient signals. The topology and component dimensioning of a circuit is communicated to SPICE through a netlist file which, in addition to a node-based circuit description, includes simulation commands and analysis directives. After a simulation is completed, the current and voltage at each node over the simulated timeframe is saved, along with any commanded measurements. Simulation times tend to be in the order of seconds or minutes, allowing for frequent circuit modifications during testing or troubleshooting.

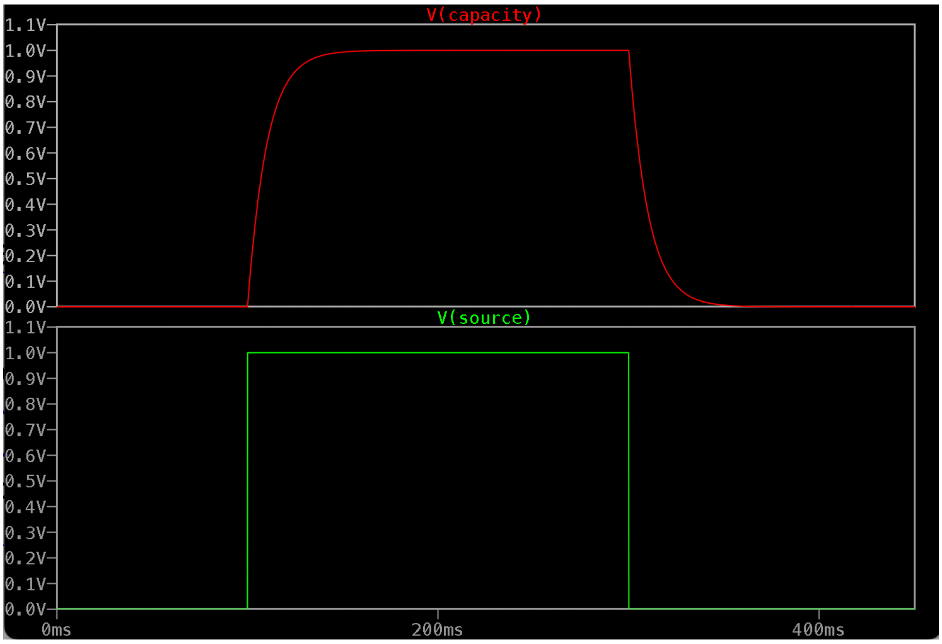


Figure 3.3: Waveform for a transient LTspice simulation showing the charge-discharge cycle of an RC circuit. Plotted in LTspice. $R = 10k\Omega$; $C = 1\mu F$

LTspice includes a library of premade circuit elements that can be incorporated into a netlist in the same way that fundamental components such as resistors or current sources would be. These range anywhere from complex integrated circuits to simpler operational amplifiers, and often include simplified or ideal versions alongside the more realistic models. One such model, the operational amplifier named *"UniversalOpamp2"*, is used in the CSA circuit designed for this thesis. The final circuit with its individual components is described in chapter 4.

Current and Voltage sources are declared as regular components in the netlist, with an additional parameter defining their temporal behavior. For transient pulses, LTspice offers the option to declare a piecewise linear (PWL) pulse, optionally sourced from a text file stored locally.

Measurement commands can be included in the netlist before a simulation is run in order to sample and save specific characteristics of a voltage or current signal. Measurable characteristics include the maximum or minimum values observed at a node, the timestamp for the first occurrence of a given value, and the value of a signal at a given time, among others. Crucially, there is no built-in method for automatic repeated measurement of the same variable type during a transient simulation. Measuring the maximum voltage of a pulse with more than one peak requires specifying the ordinal index of the peak to be measured and declaring this measurement to a unique variable, all before the simulation is run. The implications on the method used for evaluating amplifier behavior for a high number of pulses calculated in a single transient simulation is discussed in chapter 4.

A correctly formatted netlist template file can be read by the *NetlistWriter* module during the Allpix simulation. For each charge deposition event, the netlist is used as a template and, for each pixel that sees a charge during the event, a new current source named after the pixel address is written. The result of an Allpix simulation of N events is N individual written netlist files, each of them containing the current pulse and pixel address information for their corresponding charge deposition event. With the Allpix configuration chosen for this study, the shaping of the current pulse is taken care of by a fixed parametrization in the time domain, the details of which are explained in chapter 4. Output netlists can be read by LTspice and subjected to its analysis tools following a transient simulation.

3.3 Simulation Workflow Overview

The entire simulation process, starting at zero and ending at a calculated pseudoefficiency value for the simulated combination of beam, sensor, and amplifier circuit, can be summarized in four sequential steps:

1. Event Simulation in Allpix^2
2. Generation of Temporal Hit-Distribution in Python
3. Amplifier Simulation in LTspice
4. Waveform Analysis in Python

First, the configuration of the Allpix simulation is chosen according to the sensor design, particle type and energy range desired. This choice defines the frame of reference for the following steps, since all aspects of charge collection and any meaningful analysis based on the results observed after simulating the amplifier circuit, hinge on the physical processes simulated in Allpix. When step 1 is completed, the simulation has generated a netlist file for every event during which charge collection is observed in at least one pixel.

In step 2, all netlists generated during step 1 are processed by extracting the current pulse information for the largest pulse contained in each file. Each extracted current pulse is assigned a unique timestamp based on a temporal distribution calculated around a desired hit frequency. Finally, a text file containing every extracted current pulse and its timestamp, formatted as an LTspice-readable PWL pulse command, is saved. The resulting file therefore contains all pulses generated during the Allpix simulation as a continuous signal, allowing the following electric simulation to treat the currents as a single input rather than discrete and non-interacting events. This method allows current pulses that are temporally close to a one another to affect the amplifier's response, causing observable pile-up effects.

Step 3 consists of reading the PWL text file into an instance of LTspice preconfigured with the desired amplifier circuit. A single transient simulation is then run, producing a continuous voltage signal corresponding to the output voltage signal over the simulated timeframe. This voltage waveform, which contains the amplifier response to the continuous hit distribution, is exported as another text file, enabling processing outside of LTspice.

During the fourth and final step, the waveform is processed using a range of python scripts that emulate the role of a comparator and carry out the necessary calculations to arrive at a pseudoefficiency value for the simulated scenario.

Detailed explanations of each of these steps, including the relevant configuration parameters for the Allpix simulation, calculation method of the temporal hit distribution, a description of the amplifier circuit, and a breakdown of the final waveform analysis, are given in chapter 4.

Chapter 4

Data Gathering and Analysis Methods

In this chapter, the data gathering and analysis methods used for producing the results showcased and discussed in chapter 5 are described, focusing especially on the practical implementation of Allpix^2 and LTspice for data generation. Fundamental principles of the Python-based data processing and statistical analysis are presented.

4.1 Charge Deposition Simulation in Allpix^2

As described in section 3.3, running an Allpix simulation is the first step in the process of gathering data for analysis. The following describes a few key simulation parameters and their effects on the formation of charge collection pulses. The list is not exhaustive, and a copy of the full configuration files is provided in Appendix A.

4.1.1 Sensor Setup

The chosen configuration emulates a potential MigthyPix2 [13] sensor in terms of pixel size and shape, resulting in square pixels with a pitch of (84×84)µm. This choice of pixel size and geometry ensures that the likelihood and extent of charge sharing occurring as a result of a perpendicular hit on a pixel edge is similar to the latest iteration of sensors that are potential candidates for use in the LHCb Mighty Tracker [22] and Upstream Pixel (UP) [23]. This effect is relevant for the distribution of charge collected during the simulation because, for each event, the *NetlistWriter* module assigns a current pulse to each simulated pixel individually, and the following temporal distribution calculation only assigns timestamps to the largest pulses of each event. All other factors being equal, sensors with a smaller pixel pitch will observe charge sharing more often, resulting in a higher number of low- and medium-amplitude current pulses as a result of charge diffusing to and being collected in the neighboring pixel.

The sensor is simulated with a substate thickness of 150 µm, at full depletion. As calculated from Equation 2.8 for a substrate resistivity of $\rho = 370\,\Omega\,\mathrm{cm}$ [24], this would require a bias voltage of approximately $V_b \approx 594\,\mathrm{V}$, far higher than the bias typically applied to MightyPix-type sensors during their operation [9]. Furthermore, the bias voltage has a direct impact on the charge carrier drift speed, as it controls the magnitude of the electric field in the depletion zone. The faster the charge carriers propagate, the shorter their average drift times and the lower the impact of diffusion becomes. A high bias voltage would therefore not be appropriate either in terms of adherence to a real-world implementation of a MightyPix2 sensor, or in terms of modeling a realistic charge propagation behavior. In order to keep the electric field strength in a plausible range, the bias voltage chosen for the Allpix simulation is $V_b = 263.94\,\mathrm{V}$; a value corresponding to a depletion depth of

100 µm for a 370 Ω cm substrate. In order to account for the low bias voltage and large sensor thickness, a further parameter, called *depletion voltage*, defines the bias voltage needed to observe full depletion in the simulated pixel volume, disregarding the configured resistivity of the substrate. Depleting the full 150 µm thick sensor while maintaining sensible electric field strengths is achieved by setting both the bias voltage and depletion voltage to the same value of $V_b = V_d = 263.94\,\mathrm{V}$.

A linear electric field model is applied to each pixel based on the depletion and bias voltages selected. If the bias and depletion voltages are set to the same value, as is the case here, the electric field will be present in the full pixel volume, reaching zero only at the lower boundary of the pixel. The resulting field is homogenous across the width (x) and length (y) of a pixel. The electric field strength at a depth of z for a sensor of thickness d is calculated from the bias voltage V_b and the depletion voltage V_d using the following formula, taken from [19]:

$$E(z) = \frac{|V_d| - |V_b|}{d} + \frac{2|V_d|}{d}\left(1 - \frac{z}{d}\right) \tag{4.1}$$

The depth z is measured from the plane where the bias voltage is applied; in this case, the implant side of the pixel. The direction of the electric field is given by the sign of the bias voltage. Reverse biasing results in the voltage decreasing with distance from the implant side.

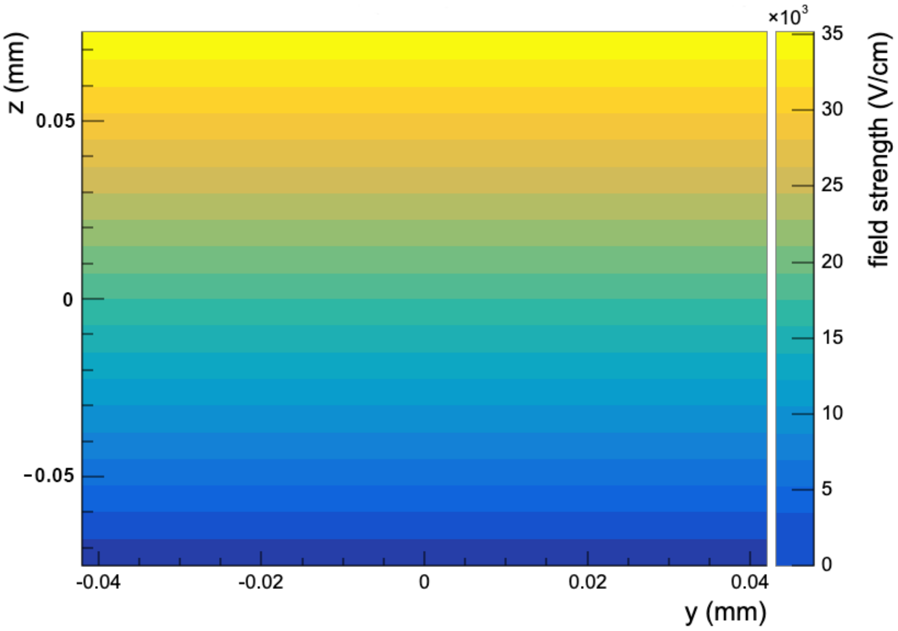


Figure 4.1: Magnitude of the electric field in a simulated pixel at position $x = 0$. Configuration matching the final analysis setup.

The ordinate axis in Figure 4.1 shows z in global coordinates, not in the relative coordinates used for describing the depth within a pixel, as used in Equation 4.1. The coordinate origin of the simulated environment lies at the center of the sensor.

As alluded to in section 3.1, this choice of electric field parametrization is a simplification and does not accurately represent the shape of the depletion zone and electric field found in a real sensor. The most accurate alternative, a TCAD-generated field map, is not available for the configured pixel geometry and depletion depth as of the writing of this thesis. An alternative model, such as a parabolic electric field, has not been chosen because the linear electric field ensures that all charge deposited within the pixel's volume can be collected.

In summary, the sensor configuration chosen for simulating particle interactions and generating charge collection pulses aims to emulate the characteristics a MightyPix2 sensor might exhibit after being subjected to large radiation doses. In this state, the sensor could experience a deepening of the depletion zone even with the original bias voltage applied [5], leading to high average numbers of deposited charge carriers and therefore large signals entering the amplifier.

4.1.2 Beam Setup

In total, the simulation run used as a source for charge collection data encompasses 100 000 events. Larger numbers are possible, but very significantly slow down the processing time needed for the analysis steps outside of Allpix. More importantly, 100 000 events carry enough statistics to represent a spread of charge deposition events ranging from the low energy to the high energy loss regime of particle interactions.

The simulated beam carries muons (μ^-) with an energy of 500 MeV. At this energy, the muons can be considered a minimally ionizing particle, which results in a comparatively low mean energy loss per event. Particles with significantly higher or lower energies would, on average, yield larger amounts of deposited and collected charge and result in fewer small signals reaching the amplifier. For studying the efficiency of an amplifier circuit, a MIP represents a "worst-case" scenario, as low charge collection events are overrepresented, yet the amplifier must be sensitive enough to detect them.

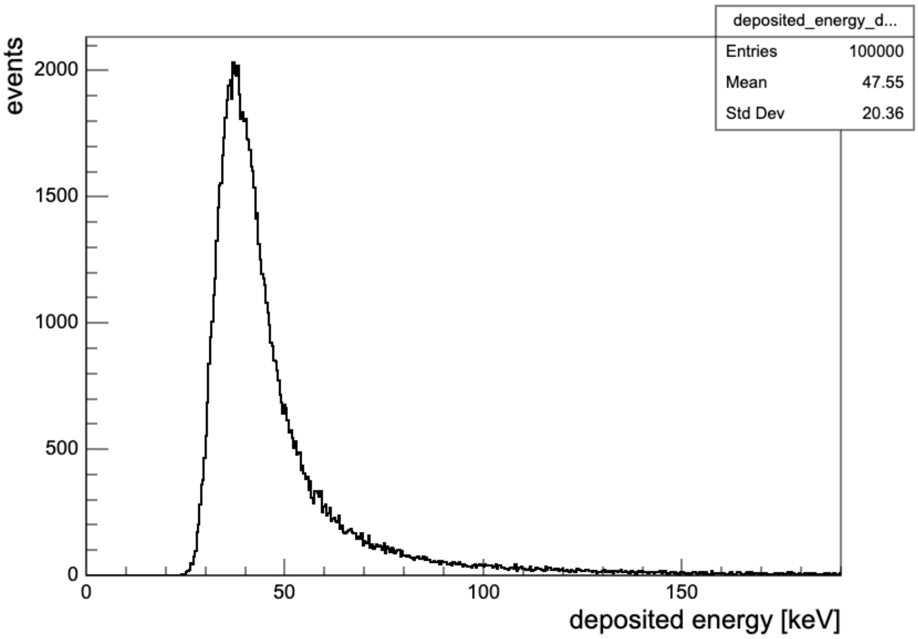


Figure 4.2: Energy deposition distribution for the 100 000 events simulated using the configuration described in section 4.1.

Particles are propagated perpendicular to the sensor and projected onto an area the size of a few pixels. To achieve this, the width of the gaussian beam profile is chosen to be double the pixel pitch in each direction, or (168 × 168)µm. The beam is aimed at a pixel by adjusting its simulated point of origin to align with the center of the pixel. This configuration ensures that none of the simulated particles miss the sensor's pixel matrix and all but guarantees the even distribution of hits within the boundaries of a pixel's surface. If the beam was perfectly localized to a single point in the center of a pixel, the effect that charge sharing due to hits on pixel edges has on the amplitude of charge collection pulses would not be accounted for.

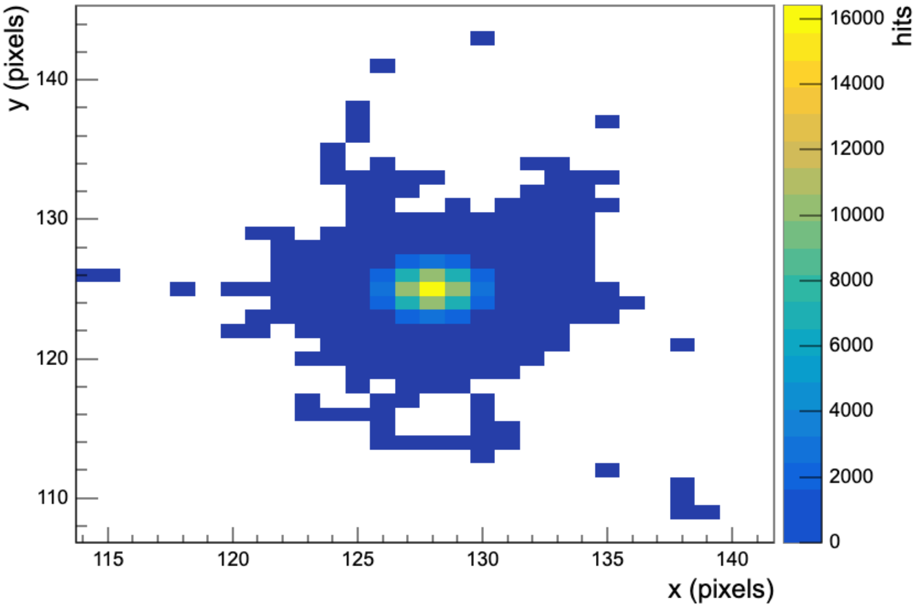


Figure 4.3: Distribution of hits on a central portion of the pixel grid for the 100 000 events simulated using the configuration described in section 4.1.

4.1.3 Charge Propagation

The propagation of charge carriers set free along the path of the ionizing particle is simulated with the *GenericPropagation* module in Allpix. Charge carriers are propagated in discrete timesteps [19] [25] by calculating their drift path based on a chosen mobility model and the electric field along their path of motion. At the end of each timestep, an additional offset is added to the calculated path to simulate the impact of diffusion. The offset is drawn from a gaussian distribution of width σ, where σ relates to the Einstein diffusion relation [26] [5]:

$$\sigma = \sqrt{2Dt} = \sqrt{\frac{2\mu k_B T}{q_e} t} \tag{4.2}$$

In Equation 4.2, μ marks the charge carrier mobility, k_B the Boltzmann-constant, T the temperature of the substrate, q_e the electronic charge, and t the length of the timestep simulated. The mobility μ is calculated using the Jacoboni-Canali model [27] for charge carrier mobility in silicon, which relates the value of μ to the magnitude E of the electric field:

$$\mu(E) = \frac{v_m}{E_c} \frac{1}{(1 + (E/E_c)^\beta)^{1/\beta}} \tag{4.3}$$

The values of the temperature dependent parameters v_m, E_c and β used in Allpix are taken from table 5 of the Jacoboni-Canali paper [27] and represent phenomenological data fits for electron and hole mobility.

Allpix allows multiple spatially close charge carriers to be propagated together to save resources and improve the speed of the simulation at the cost of some accuracy. The number of charge carriers propagated together is controlled with the *charge-per-step* (cps) parameter. Considering the homogeneity of the electric field in x and y direction, the drift trajectory of neighboring charge carriers is similar enough that propagating a moderate number of them together does not significantly impact charge collection. This is especially the case for the perpendicular trajectories of particles simulated with the configuration described in subsection 4.1.2.

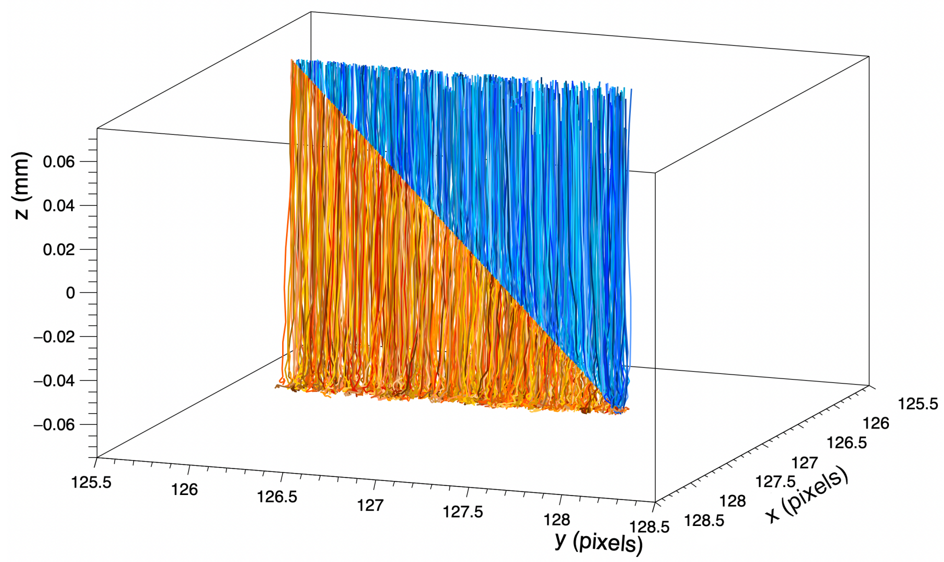


Figure 4.4: Allpix^2 simulated linegraph depicting paths of free charge carriers for a 500MeV μ^- traversing the sensor at a 45° angle. Electrons shown in blue, holes in orange. Electric field configuration identical to Figure 4.1

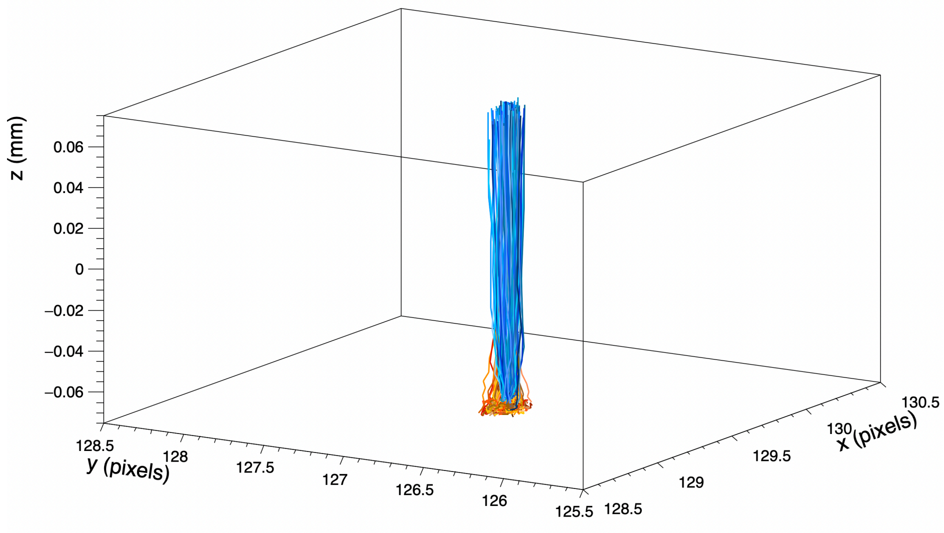


Figure 4.5: Allpix^2 simulated linegraph depicting paths of free charge carriers for a 500MeV μ^- traversing the sensor at a 90° angle. Electrons shown in blue, holes in orange. Electric field configuration identical to Figure 4.1

Figure 4.4 shows the predictably linear propagation path of electrons and holes in the configured linear electric field. Areas of lower field strength at the bottom of the figure allow for a larger contribution of diffusion to the particle paths, resulting in a slightly broader distribution that narrows as the particles move towards areas of high field strength. Figure 4.5 depicts a usual charge propagation pattern for the configuration described in this section: the particle traverses the detector with a 90° angle, generating a narrow strip of free charge carriers that propagate in a roughly cylindrical volume centered around the particle intersection path. Both figures are generated using the same *charge-per-step* setting of cps $= 20$, which is also the configured choice for the final simulation.

4.1.4 Netlist Writer

The Allpix simulation is configured in such a way that propagating electrons are considered "collected" once they reach the implant side of the detector. The total amount of charge collected per event and per pixel is saved as a variable named *pixel-charge* for use by modules ahead of the *GenericPropagation* module in the simulation chain. When using the default digitization modules built into Allpix, the front-end simulation applies an amplifier response function to the *pixel-charge* value and digitizes the signal by comparing the calculated output to a threshold value in order to generate digital hits from the collected charge.

The role of the *NetlistWriter* module is to replace the built-in amplification and digitization by outputting charge pulse information to an external electronics simulation program, as explained in section 3.2. Configured to be compatible with the charge collection and electric field model described in the previous sections, the *NetlistWriter* does so by converting the *pixel-charge* value Q_p into a parametrized current pulse. The shape of the current pulse is determined by setting a fixed rise time t_r, fall time t_f and pulse width t_w, resulting in a trapezoidal pulse. The amplitude I_0 of the parametrized current pulse is calculated from the charge by integrating the current over the total pulse length and solving for I_0.

$$Q_p = \int_0^{t_r} I_0 \frac{t}{t_r}\, dt + \int_{t_r}^{t_r+t_w} I_0\, dt + \int_{t_r+t_w}^{t_r+t_w+t_f} I_0 \left(1 - \frac{t}{t_r}\right) dt \tag{4.4}$$

The chosen parametrization for the current pulse features equal rise and fall times of $t_r = t_f = 0.1$ns and a pulse width of $t_w = 6$ns. It follows from Equation 4.4 that the total charge in the current pulse can be written for this parametrization simply as:

$$Q_p = I_0 \cdot 6.1\,\mathrm{ns} \tag{4.5}$$

From which the amplitude of the parametrized current pulse follows directly:

$$I_0 = \frac{Q_p}{6.1\,\mathrm{ns}} \tag{4.6}$$

The choice of current pulse parametrization is informed by the drift times observed for multiple simulations, including one based around a more accurate TCAD-sourced electric field map, albeit for a different pixel geometry and at only $80V$ reverse bias. The main criterion for deciding on a pulse width is that the majority of charge carriers should reach the electrode in a timespan equal or less than the width of the current pulse as, according to the Shockley-Ramo theorem [10], the induced current ends once charge carriers stop moving.

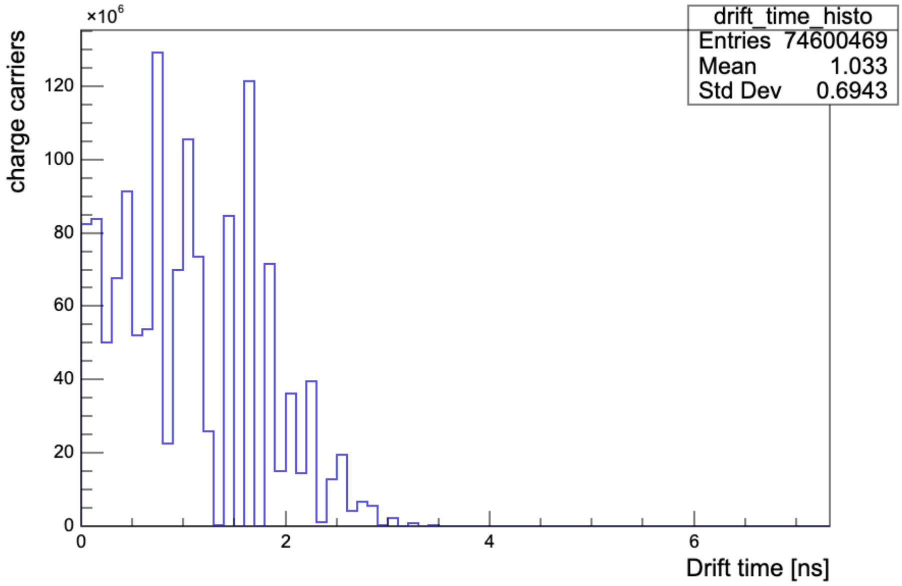


Figure 4.6: Drift time histogram for charge carriers propagated during the final Allpix^2 simulation of 100000 events.

Figure 4.6 shows that all charge carriers propagated in the linear electric field rach the implant side of the sensor within a 4 ns window, and only a few propagate longer than 3 ns. The lack of larger drift time values is a result of the full depletion of the substrate, as there are no areas within the volume of any simulated pixel where an electric field is not acting upon a charge carrier and directing it towards the electrode. Comparing the drift time distribution in Figure 4.6 with a similar plot in Figure 4.7 generated from a simulation using a more accurate TCAD-sourced electric field map shows that, in a scenario where the depletion volume is smaller than the total pixel volume, the observed drift time distribution has a long tail toward high drift times.

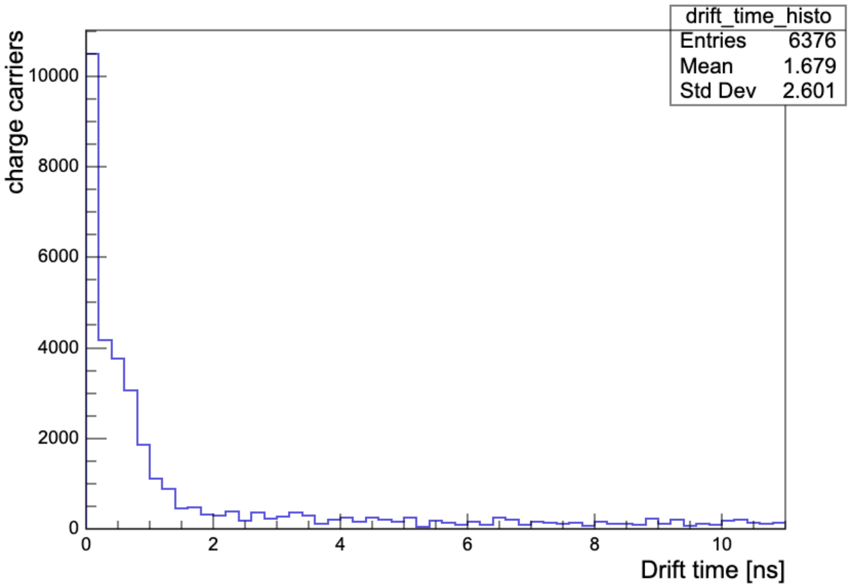


Figure 4.7: Drift time histogram for charge carriers propagated during the Allpix^2 simulation of 25 events using a TCAD-simulated electric field map. Simulated pixel size of (165 × 25)µm, sensor thickness of 85 µm, 80 V reverse bias voltage.

Additional information on the simulation corresponding to Figure 4.7 is appended in Appendix B. It is worth noting that the low number of events simulated and the difference in pixel geometry between the simulation configuration described in this section and the configuration underlying Figure 4.7 forbid direct comparisons of both distributions. Nonetheless, a qualitative assessment can still be made regarding the length of time during which a current signal is induced due to charge carrier movement, on the basis that some field-free regions would likely be present at the points furthest from where the bias voltage is applied to the pixel, even in a radiation-exposed MightyPix2. The 6ns pulse width figure has therefore been calculated by applying an additional factor of 1.5 to the 4ns drift time length observed in Figure 4.6.

The netlist written for each event includes the parametrized current pulse for each charge collecting pixel in the form of an LTspice-readable current source. The pulse is described using a format which contains the Amplitude of the current in Ampere, as well as the time parameters t_r, t_f and t_w.

Module Name	**Parameter**	**Value**
[*Allpix*]	Number of Events	100 000
[*DepositionGeant4*]	Particle Type	Muon (μ^-)
	Particle Energy	500 MeV
	Source Energy Spread	0 keV
	Beam Divergence	$(0, 0)$ mrad
	Physics List	FTFP-BERT-EMZ
[*ElectricFieldReader*]	Model	Linear
	Depletion Voltage	−263.94 V
	Bias Voltage	−263.94 V
[*GenericPropagation*]	Mobility Model	Jacoboni

Table 4.1: Summary of Key Allpix^2 Simulation Parameters and their Values.

4.2 Temporal Distribution of Hits

In order to study how the amplifier built into a pixel front-end would behave in a high-hitrate scenario like the one expected in the highest fluence regions of the Upstream Pixel, simulating individual amplifier pulses is insufficient. A major potential source of inefficiency is the occurrence of hits in close succession, leading to a pileup of amplifier pulses, and the subsequent loss of hit information. Observing this effect requires a simulation that calculates a continuous amplifier signal containing every event, with some underlying temporal distribution that allows for overlapping amplifier pulses. Additionally, in order to draw comparisons to the hit detection efficiency expected from the UP, the temporal distribution chosen for the simulated events must be based on the conditions experienced by the UP within the LHCb experiment.

4.2.1 Hitrate Calculation

The maximum rate of particle hits that can theoretically be registered by a sensor assuming an ideally efficient front-end, is limited by the bandwidth of the sensor's readout. Based on the expected particle rate in the LHCb Upstream Pixel after LHC Upgrade II, any sensor used in this section of the detector must be designed with a particle incidence rate of up to $R_p = 74\,\mathrm{MHz/cm^2}$ [28] in mind. The effective hit rate R_h seen by the sensor readout is higher than the particle rate R_p due to charge sharing between pixels. Quantifying this effect on the total hitrate specification for the sensor is achieved by multiplying the particle rate with the mean cluster size n_c expected for a given sensor design and particle incidence angle:

$$R_h = R_p \cdot n_c \tag{4.7}$$

According to [28], a mean cluster size of up to $n_c = 1.2$ is calculated, onto which an additional factor of up to 1.2 relating to the data transmission process is multiplied. Including a safety margin to account for unforeseen effects results in a hitrate specification larger than the particle rate by a factor of 2:

$$R_h = 2R_p \tag{4.8}$$

In [28], this relation is rounded up to give an Upstream Pixel hitrate specification of $R_h = 150\,\mathrm{MHz/cm^2}$, which is the final value used in this thesis going forward. Reading out such a high hitrate is challenging in and of itself, which makes it important to investigate wether the analogue circuitry is even capable of delivering the required efficiency before the readout is considered.

Assuming a MightyPix2-Type sensor with a pixel pitch of (84x84)µm and an even distribution of hits across the surface of the sensor, the per-pixel hitrate R_{pix} is calculated by multiplying the surface area $A_{pix} = (84\,\mathrm{\mu m})^2$ of a pixel with the final hitrate specification:

$$R_{pix} = R_h \cdot A_{pix} \tag{4.9}$$

To study wether or not an amplifier circuit could in principle provide the hit detection efficiency required of the UP, the hitrate of the "hottest" pixel, the pixel experiencing the highest particle rate, is calculated by considering an uneven distribution of incident particles on the sensor and applying an additional safety margin to the calculation. The resulting offset factor for the hottest pixel is estimated to be 2.5, which is multiplied with the result of Equation 4.9 to yield a maximum per-pixel hitrate of

$$R_{pix} = 2.5 \cdot R_h \cdot A_{pix} = 26\,460\,\mathrm{Hz} \tag{4.10}$$

This value represents the calculated upper limit of hits for any MightyPix2-sized pixel used in a sensor installed on the Upstream Pixel at LHCb, and is therefore used as an orientative hitrate ceiling for testing hit detection efficiency of the simulated amplifier circuit.

4.2.2 Hit Distribution

Hits originating from particle interactions during an LHC bunch crossing will occur at discrete time intervals with a minimum separation equal to the bunch crossing time of $t_b = 25\,\mathrm{ns}$. The first step in calculating the temporal distribution of particle hits is therefore to divide the time domain into 25 ns bins, representing the discrete bunch crossing times at which a distinct particle could traverse a pixel and generate a new current signal.

Assuming a fixed per-pixel hitrate R_h, a value can be assigned to the probability P_h of seeing a hit within a given number k of bunch crossings from a previous hit. Given that for each bunch crossing there are only two possible outcomes (seeing a hit or not seeing a hit in the pixel), and that particles stemming from distinct bunch crossings share no causal relation with each other, the temporal distribution of hits is a geometric probability distribution [29]. Geometric distributions describe the probability that the first success in a series of independent Bernoulli trials with success probability p is observed after k attempts. The probability P_h of a hit being observed within exactly k bunch crossings of the previous hit is therefore given by

$$P_h(p,k) = p \cdot (1-p)^{k-1} \tag{4.11}$$

where $p = p(R_h, t_b)$ is a hitrate-dependent quantity describing the probability of seeing a hit within a timespan equal to the length t_b of a bunch crossing time interval.

$$p(R_h, t_b) = R_h \cdot t_b \tag{4.12}$$

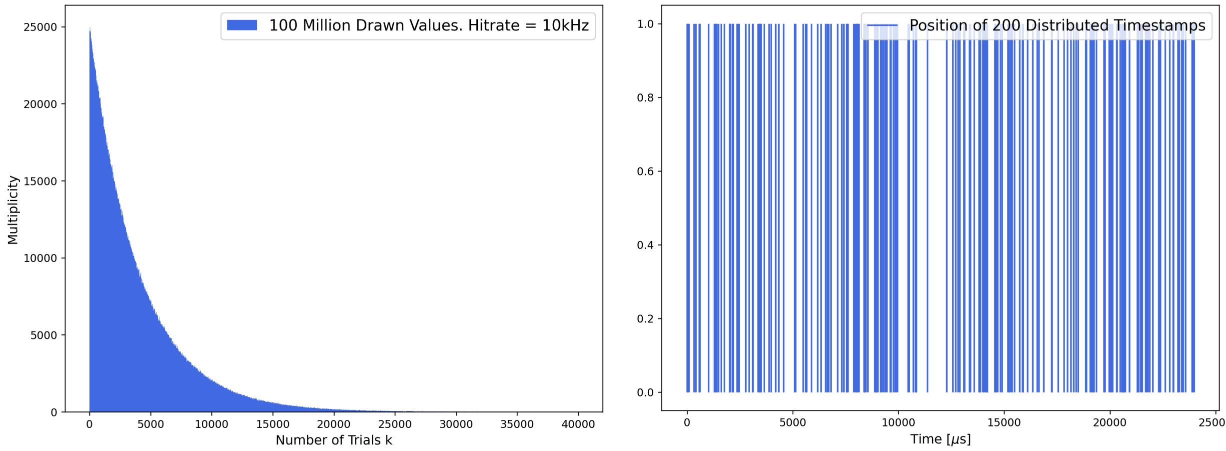
Combining Equation 4.11 and Equation 4.12 yields a hitrate- and bunch-crossing-dependent formulation of the binned hit probability distribution:

$$P_h(R_h, t_b, k) = (R_h \cdot t_b)\,[1 - (R_h \cdot t_b)]^{k-1} \tag{4.13}$$

Following the Allpix simulation and the creation of all netlist files, the amplitude of the largest current pulse in each netlist is extracted using a python script. The script then calculates the probability distribution from Equation 4.13 based on the desired hitrate and bunch crossing time, and draws from it one k-value for each extracted current pulse. The drawn value k is then multiplied with the bunch crossing time t_b to generate a time Δt_h corresponding to the total distance from the previous current pulse.

$$t_{i+1} - t_i = \Delta t_h = k \cdot t_b \tag{4.14}$$

Using the current pulse parametrization from subsection 4.1.4, the extracted pulse amplitude and the hit time difference Δt_h generated from the drawn k-value, a piecewise-linear pulse segment is written for each extracted current and added in sequence onto the previous segment. The result is a single PWL pulse command containing all of the largest current pulses from every netlist generated during the Allpix simulation, with pulse timestamps assigned to them based on the geometric distribution for the configured hitrate.



(a) Probability Distribution

(b) Temporal Distribution Example

Figure 4.8: (a): 10^8 k Values drawn from a geometric distribution calculated with Equation 4.13 for $R_h = 10\,\text{kHz}$ and $t_b = 25\,\text{ns}$. (b): Position of 200 values distributed with the method described in Equation 4.13 and Equation 4.14 for $R_h = 10\,\text{kHz}$ and $t_b = 25\,\text{ns}$. Lines are position markers of the calculated timestamps, not current pulses.

For the final analysis, PWL pulse distributions for 13 hitrates between $R_h = 2000\,\text{Hz}$ and $R_h = 26\,000\,\text{Hz}$ have been generated using the same set of 100 000 netlists created with the Allpix configuration described in section 4.1. For every hitrate value, 5 distinct temporal hit distributions have been created to incorporate some statistical variation in the produced data.

4.3 Amplifier Signal Generation in LTspice

To recreate the behavior of a charge sensitive amplifier, a simple electric circuit has been designed using component templates built into LTspice. The charge pulses generated using the method described in section 4.1 and combined into a continuous signal through the method described in subsection 4.2.1 are read into LTspice as a current source, which produces the simulated amplifier voltage output. The following subsections describe the design and dimensioning of the amplifier circuit.

4.3.1 Amplifier Circuit Design

The chosen amplifier design consists of an operational amplifier connected in an inverting configuration to a charge-sensitive *RC* circuit element [30]. This circuit design does not aim to perfectly recreate the amplification characteristics of a typical CSA used in HV-MAPS sensors such as MightyPix, which usually incorporate more complex Krummenacher or linear feedback circuits [12]. The extent and implications of this discrepancy are demonstrated in subsection 4.3.2.

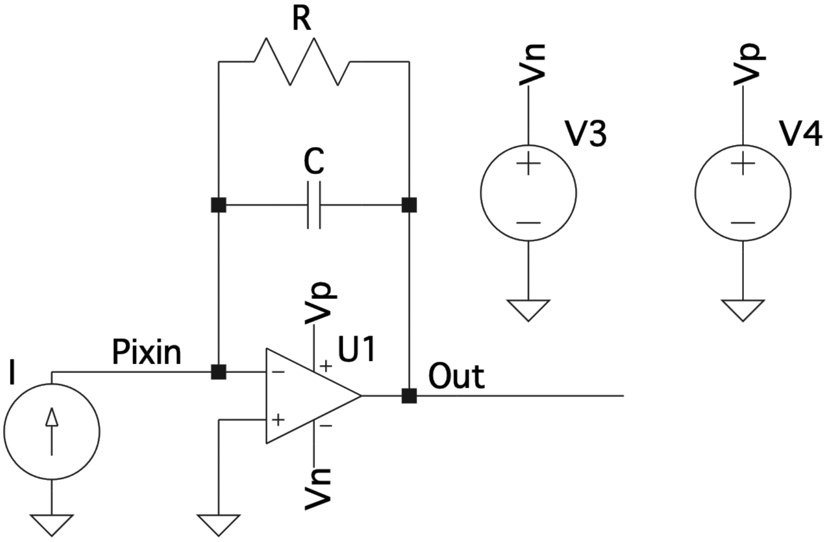


Figure 4.9: Schematic representation of the amplifier circuit used for simulating a charge-sensitive amplifier in LTspice. Pixel input "Pixin" and amplifier output node "Out" marked by name. Op-Amp voltage supplies shown as floating connections.

The operational amplifier used for the circuit is the LTspice default model named *UniversalOpamp2*. In its implemented configuration, it simulates an operational amplifier with no noise and a lossless linear voltage amplification behavior up to a saturation voltage defined by its voltage supply. Its intrinsic open-loop DC gain is $A_G = 10^6$. The inverting Op-Amp input is connected to the pixel input node, named *Pixin*, the non-inverting input is grounded, and the output feeds into the node named *Out*. The operational amplifier's voltage supply rails are connected to 2 DC voltage sources, one supplying a negative, the other a positive voltage.

The charge-sensitive feedback circuit consists of a capacitor and a resistor connected in parallel to the pixel current input node and the amplifier output node. Both the resistor and the capacitor are modeled by default LTspice components and behave like ideal resistive and capacitive circuit elements. A current source, configurable to take the PWL pulse generated from the Allpix simulation (4.1) and pulse distribution (4.2), delivers a current signal into the *Pixin* node.

4.3.2 Dimensioning the Amplifier Circuit

The effective gain of an amplifier circuit based around an ideal operational amplifier with a feedback loop is determined almost exclusively by the behavior of the feedback circuit. In the case of the amplifier described in subsection 4.3.1, the feedback circuit consists of an RC element, which reduces the calculation of the maximum voltage output V_{max} to the typical capacitance-voltage-charge relation, based on the charge Q_{in} contained in the input current pulse and the capacitance C_f of the feedback capacitor:

$$V_{max} = \frac{Q_{in}}{C_f} \tag{4.15}$$

In practice, as soon as charge begins building up in the feedback capacitor, the potential difference between the input and output nodes induces an immediate discharge current through the feedback resistor R_f, which counteracts the ongoing capacitor charge process. As a result, the true maximum output voltage achievable with the circuit for a given input current is slightly lower than calculated in Equation 4.15. The effect is minor but still observable in the simulation, as demonstrated in the simulated amplifier characteristics shown later in this section.

The width of the amplified output pulse is dominated by its falling edge, which follows an exponential decay curve typical of a resistive capacitance discharge process:

$$V_{fall}(t) = V_{max} \cdot \exp\left(-\frac{t}{\tau}\right) \tag{4.16}$$

The time constant τ is given by the product of feedback capacitance and feedback resistance:

$$\tau = R_f \cdot C_f \tag{4.17}$$

The shape of the amplifier output pulse can therefore be controlled in two ways: the amplitude V_{max} is set by adapting the feedback capacitance C_f to the expected input charge Q_{in}, and the pulse length is controlled by adjusting the value of the feedback resistance R_f.

To arrive at final R_f and C_f values, the amplifier response to a range of injected charges has been measured for a Run2021v2 sensor (also known as TelePix) using a special setup developed for HV-MAPS sensor testing within the Mu3e research group at the Physikalisches Institut in Heidelberg. Run2021v2 is an HV-MAPS based research prototype sensor consisting of $(165 \times 25)\,\mu\text{m}$ sized pixels; a more rigorous description of its characteristics can be found in [31]. Crucially, the amplifier circuit built into Run2021v2 pixels is similar in structure to the amplifier used in other HV-MAPS sensor designs such as the MightyPix series, allowing comparisons with the simulated circuit to be made. The full test setup is thoroughly explained here [32]. In short, the tested sensor is connected to a circuit board allowing charge injection directly into individual pixel front-ends using an injection voltage V_{inj} in combination with the in-pixel injection capacitance C_{inj}. A parameter controlling the amplifier's feedback and therefore the width of the amplified output pulse for individual pixels can be set in discrete steps on a computer connected to the circuit board. The amplifier output is monitored and measured on a digital oscilloscope. The injected charge Q_{inj} is calculated as follows:

$$Q_{inj} = V_{inj} \cdot C_{inj} \tag{4.18}$$

In accordance with the measurement shown in [33] and its stated design value, the injection capacitance is taken to be $C_{inj} = 1\,\text{fF}$. Safe injection voltages range from $0.3\,\text{V}$ to $1.5\,\text{V}$ for the tested sensor.

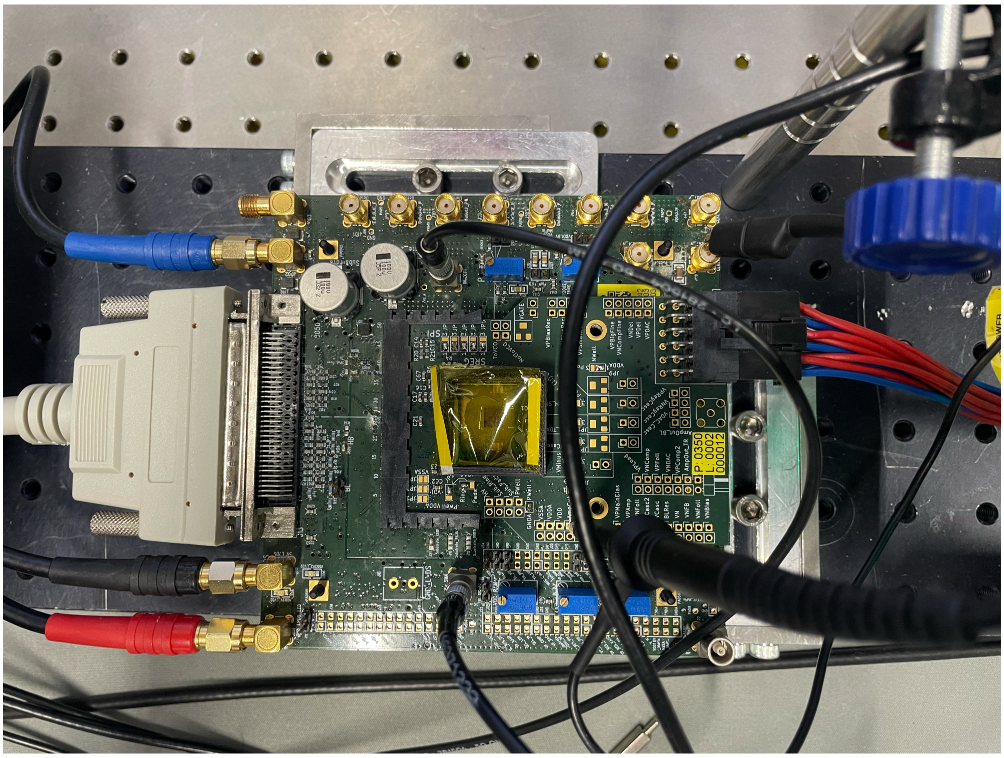
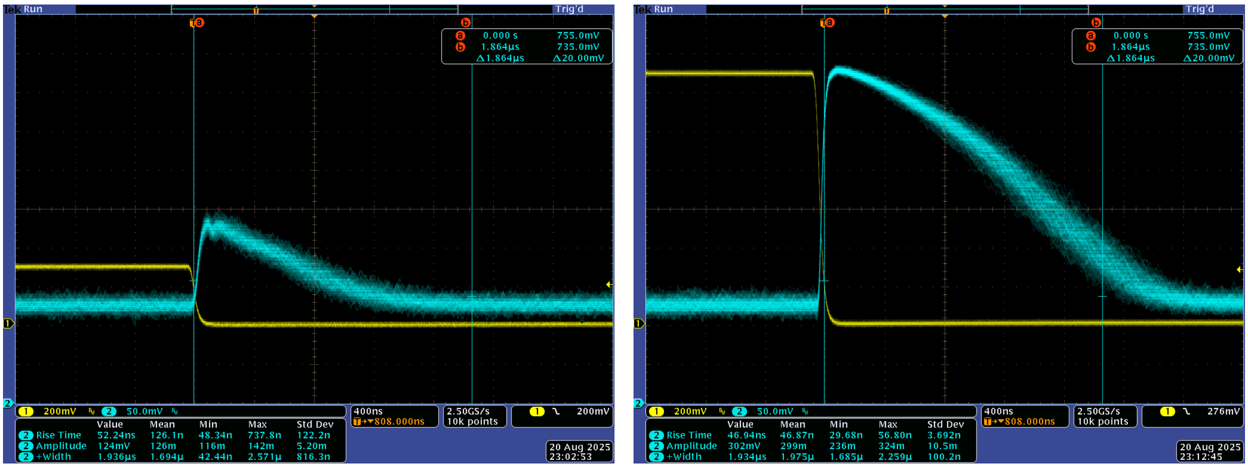


Figure 4.10: Amplifier characterization test setup with Run2021v2 sensor mounted on a PCB. Voltage supply and readout data cables shown on left side of the image. Sensor located beneath the Kapton foil.



	Value	Mean	Min	Max	Std Dev
2 Rise Time	52.24ns	126.1n	48.34n	737.8n	122.2n
2 Amplitude	124mV	126m	116m	142m	5.20m
2 +Width	1.936µs	1.694µ	42.44n	2.571µ	816.3n

20 Aug 2025 23:02:53

(a) $V_{inj} = 0.3\,\mathrm{V}$

	Value	Mean	Min	Max	Std Dev
2 Rise Time	46.94ns	46.87n	29.68n	56.80n	3.692n
2 Amplitude	302mV	299m	236m	324m	10.5m
2 +Width	1.934µs	1.975µ	1.685µ	2.259µ	100.2n

20 Aug 2025 23:12:45

(b) $V_{inj} = 1.3\,\mathrm{V}$

Figure 4.11: Oscilloscope screenshots taken during the injection measurement showing amplifier output signals for different injection voltages. Blue channel shows the amplifier output, yellow channel the injected voltage.

The width of the amplifier pulse is measured by the oscilloscope as the time between the amplifier output voltage rising above 15% of its maximum value on the rising pulse edge and falling below 15% again on the falling edge. Measurements are taken only after the mean and standard deviation of the amplitude and width measurement parameters stabilize.

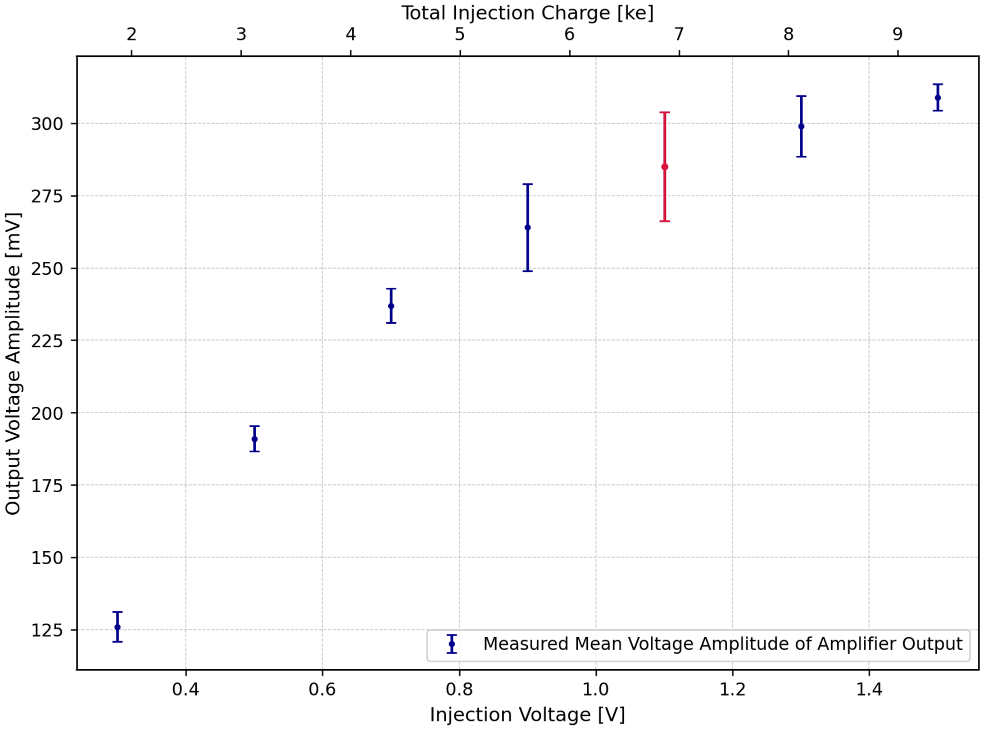


Figure 4.12: Mean amplitude value of output voltage pulses for different injection voltages and corresponding injection charges measured on a Run2021v2 sensor.
Dimensioning reference point highlighted in red.

The amplification characteristic plotted in Figure 4.12 shows a linear amplitude progression only in the low-charge regime. Beggining at medium and moving toward higher injection charges, the amplifier shows saturation effects that flatten the amplification curve. Considering the shape of the amplifier response implied by the measurement and its saturation value, the third highest injection voltage, $V_{inj} = 1.1\,\mathrm{V}$, can be taken as a reference for the dynamic range of the simulated amplifier. Substituting V_{max} for the measured output voltage at $V_{inj} = 1.1\,\mathrm{V}$ and Q_{in} for Q_{inj} in Equation 4.15 yields a feedback capacitance of $C_f = (3.86 \pm 0.25)\,\mathrm{fF}$ for the circuit in LTspice. In the final simulation, the configured capacitance value corresponds to the mean of $C_f = 3.86\,\mathrm{fF}$.

Judging from the maximum amplifier amplitude observed in Figure 4.12, the measured circuit has a peak voltage output above $300\,\mathrm{mV}$ and below $350\,\mathrm{mV}$. In the simulated amplifier, this saturation behavior is accounted for by providing the op-amp with a voltage supply of $V_{sat} = 330\,\mathrm{mV}$, limiting its maximum achievable output.

Calculating the feedback resistance R_f from the measured pulse widths requires rewriting of Equation 4.16 to solve for the RC time constant by assuming the pulse width T_w to be the time at which $V/V_{max} = 15\%$.

$$R_f = -\frac{T_w}{C_f \ln(0.15)} \tag{4.19}$$

An implicit assumption made in this method is that the pulse rise time has a small contribution to the total pulse width in comparison to the falling pulse edge. This assumption still holds true for large pulses like the one used for calculating C_f.

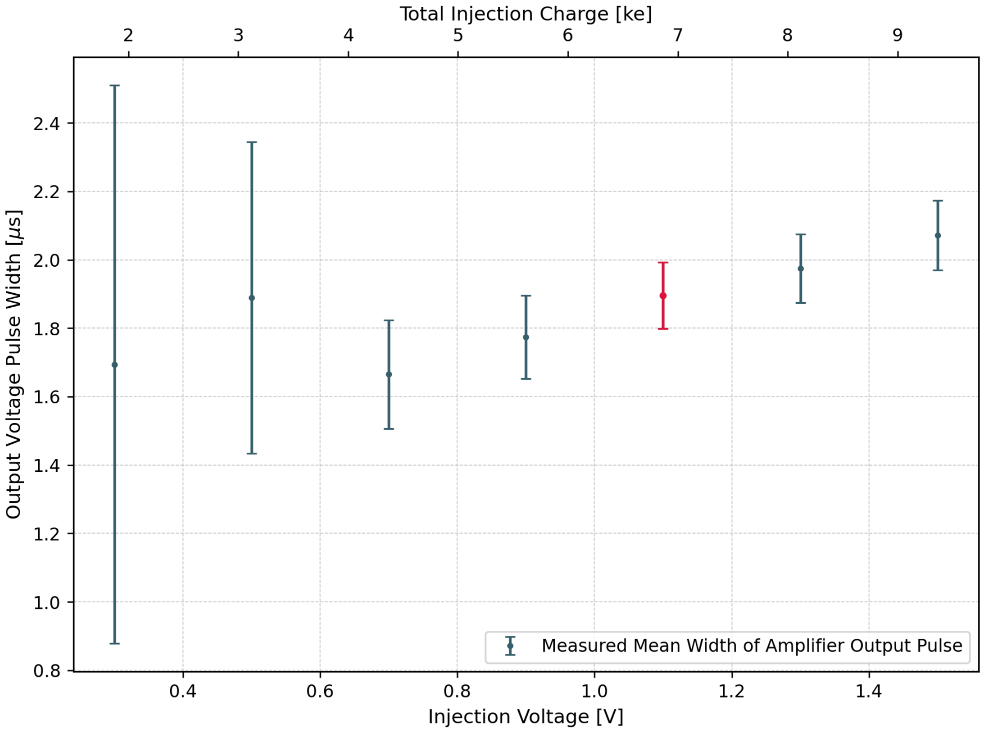


Figure 4.13: Mean width of output voltage pulses for different injection voltages and corresponding injection charges measured on a Run2021v2 sensor. Dimensioning reference point highlighted in red.

Due to per-run variations in the injection voltage, amplifier noise present in the output pulses, and the low 15% threshold used for defining start and stop times, the pulse width measurements taken in the low injection voltage regime show high uncertainties. For later measurements, 15% of V_{max} lies above the noise floor and the uncertainty decreases. Calculating the feedback resistance R_f with Equation 4.19 by plugging in the highlighted pulsewidth $T_w = (1.9 \pm 0.1)\,\mu\text{s}$ results in a value of $R_f = (259 \pm 13) \cdot 10^6\,\Omega$. The final circuit configured for the simulation uses a value of $R_f = 259\,\text{M}\Omega$.

Quantity	**Value**	**Uncertainty**
Capacitance C_f	$3.86 \cdot 10^{-15}$ F	$0.25 \cdot 10^{-15}$ F
Resistance R_f	$259 \cdot 10^6\,\Omega$	$13 \cdot 10^6\,\Omega$
Amp. Supply V_{sat}	$330\,\text{mV}$	–

Table 4.2: Final Dimensioning of Simulated Amplifier.

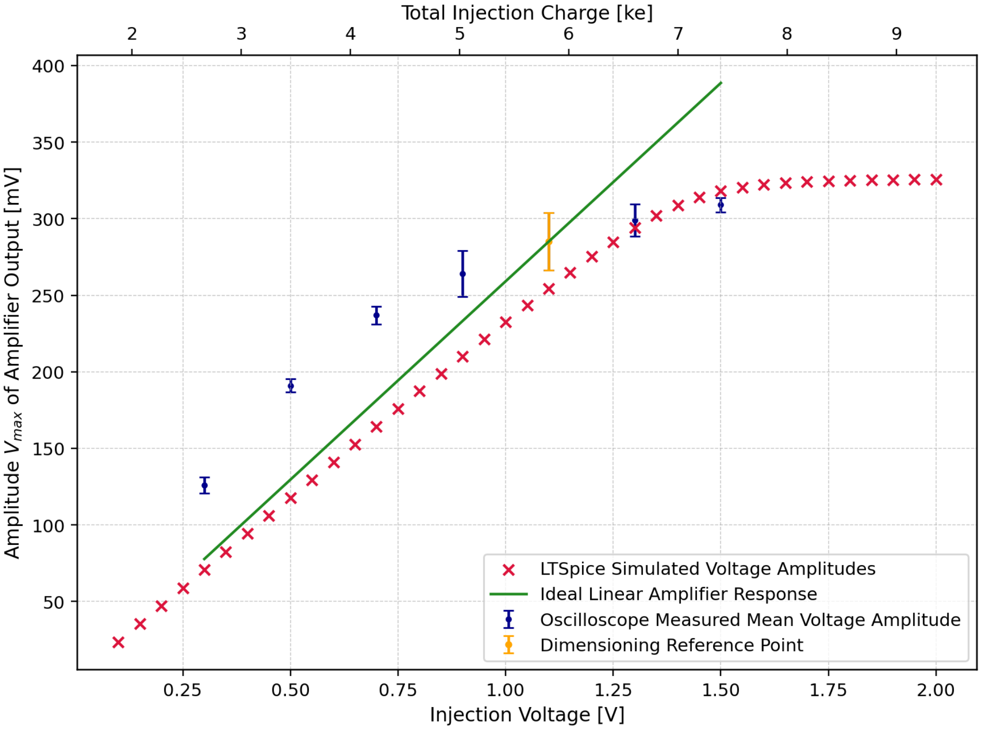


Figure 4.14: Simulated, measured, and ideal amplifier output amplitudes for a range of injection voltages. Simulated circuit parameters identical to Table 4.2.

Figure 4.14 shows the amplifier characteristic for the oscilloscope measurement alongside the amplifier characteristic taken from the LTspice simulation dimensioned using the parameters from Table 4.2. A straight line representing the ideal linear amplification behavior for the same parameters according to Equation 4.15 is also plotted for reference. The simulation shows good agreement with the Run2021v2 amplifier in the high injection charge and saturation regimes, but fails to model the high amplification of the physical amplifier circuit in the low and medium injection charge domains. The LTspice amplitude curve consistently falls below the theoretical maximum achievable voltage, likely as a consequence of the feedback current preventing the full amount of input charge from reaching the capacitance C_f. The simulated circuit exhibits a smooth transition into the saturation regime, where the maximum output voltage asymptotically approaches $V_{sat} = 330\,\text{mV}$.

The simulated circuit also matches the measured pulse width for high injection charges, as shown in Figure 4.15. Once again, the behavior of the LTspice circuit is more linear than the Run2021v2 amplifier circuit, showing only a very slight increase in total pulse length over the expected constant value for higher injection charges. Notably, the simulated pulse length is consistently higher than the calculated T_w by around $0.1\,\mu\text{s}$, partly as a consequence of disregarding the rise time of the pulse for the calculation.

The good agreement between the simulated and measured circuits in the high-amplitude regime means that large amplifier pulses, which make up the majority of the simulated events due to the attempt made at replicating an irradiated sensor with large depletion during the Allpix^2 simulation, are a fairly close match for the real amplifier in the conditions simulated.

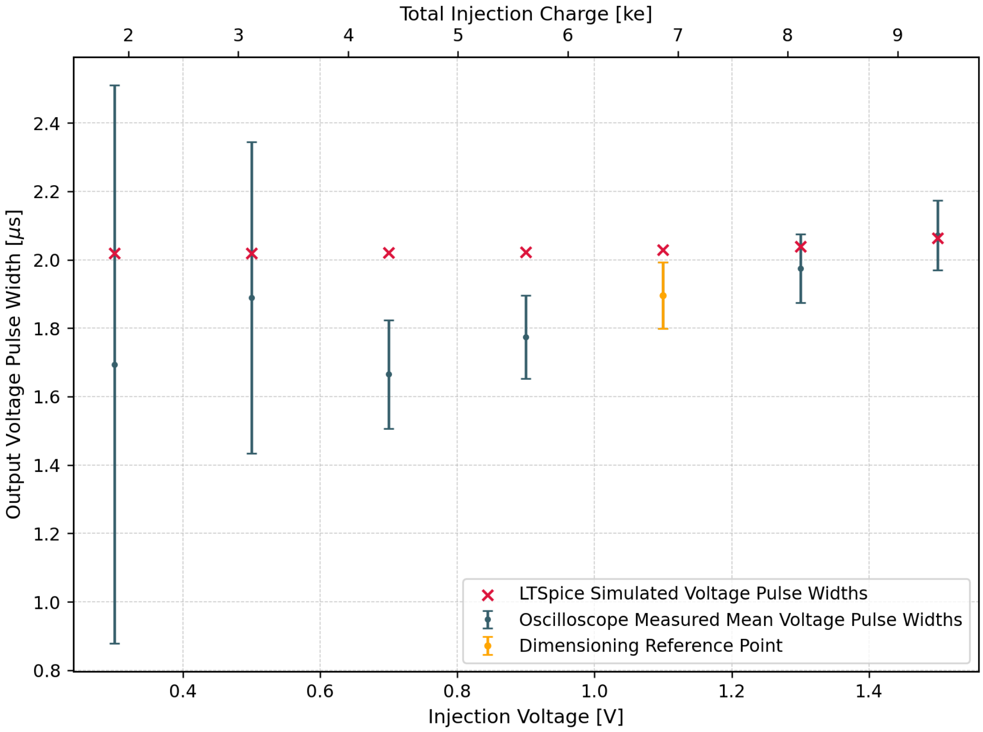


Figure 4.15: Measured and simulated pulse widths for the same injection voltages. Simulated circuit parameters identical to Table 4.2.

4.4 Waveform Analysis

Having simulated particle charge deposition in Allpix, calculated a temporal hit distribution, and designed a charge-sensitive amplifier circuit, the last step in the data generation workflow is the creation of a set of amplifier output waveforms from which conclusions about the hit detection efficiency of the amplifier circuit can be drawn. This goal is achieved by inputting each of the PWL files containing the unique distribution of current pulses into LTspice individually, running the electronics simulation, and exporting the waveform for the voltage at the amplifier output node as a data text file. Exporting the waveform data, as alluded to in section 3.2, is a necessary step due to the limitations of the measurement tools built into LTspice. It is also a source of errors stemming from the resolution of the exported file, the causes and implications of which are discussed in section 5.4.

Being based on the simulation described in section 4.1, each waveform generated in LTspice contains the amplifier response to a total of 10^5 individual current pulses. The hit detection pseudoefficiency of the simulated circuit for each PWL file is analyzed in python by emulating a comparator. Defining a fixed voltage threshold Λ ($[\Lambda] = \text{Volt}$), counting the number of "comparator activations" n_c, and dividing this number by the total number $N_p = 10^5$ of pulses in the input current yields the hit detection pseudoefficiency ϵ for the chosen combination of simulation parameters:

$$\epsilon = n_c/N_p \tag{4.20}$$

A "comparator activation" is counted each time the value of the amplifier output voltage observed in the analyzed waveform completes a full cycle of crossing above and falling below threshold. This way, amplifier pulses that are timed close enough to merge together above threshold in a pulse pile-up, are only counted as a single hit. Pulses that pile-up, but are far enough apart from each other and/or low enough in amplitude that they merge below the threshold, are still counted as two individual detected hits. A practical example demonstrating the handling of different pile-up pulses is shown in Figure 4.16.

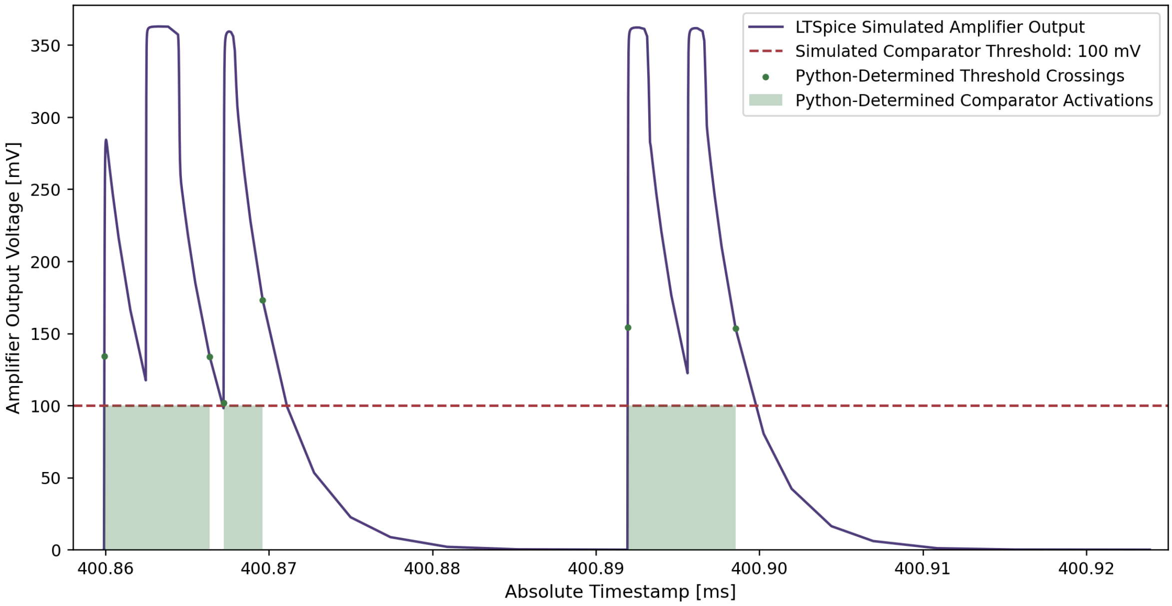


Figure 4.16: Portion of simulated amplifier output waveform demonstrating the hit-detection and comparator-emulation method used for calculating pseudoefficiencies in this thesis. The LTspice circuit configuration used for producing this waveform differs from the one described in subsection 4.3.1 in the saturation voltage configured.

Low-amplitude pulses that fail to reach the threshold are not counted as hits, and therefore contribute to the observed hit detection inefficiency. They can, however, be counted and differentiated from inefficiencies caused by pile-up.

Figure 4.16 also showcases some of the shortcomings of the waveform analysis method, all relating to the limited resolution of the waveform after the export from LTspice. Mainly, the pulses exhibit jagged rather than smooth edges, that only approximate the shapes they represent. The exponential voltage decay, for instance, consists of multiple linear segments, making it inadvisable to conduct time-over-threshold studies using the exported waveforms. For very large pulses, such as the second from the left in Figure 4.16, the transition out of saturation does not occur smoothly, creating a sharp angle followed by a straight drop before continuing the regular exponential decay. These irregularities are not present when viewing the waveforms in LTspice, leading to the assumption that they are caused by the exporting process. Precautions have been taken so that no "true" hits are missed by the code, but a limited amount of over-counting can still occur, as demonstrated in chapter 5.

In acknowledgement of the fact that the value ϵ derived through the methods presented above reflects a simplified, simulated version of reality, it has thus far been referred to as a *pseudoefficiency*. In the following chapters, in service of readability, ϵ will be referred to simply as an efficiency.

Chapter 5

Hitrate vs. Efficiency

In this chapter, the results for the hitrate study carried out using the method described in chapter 4 are presented. A single Allpix^2 simulation run is used as a source of current pulses, from which 5 temporal current pulse distributions are generated for each investigated hitrate, totaling 65 distinct current signals corresponding to 13 hitrate values. Each of the 65 signals is converted to an amplifier waveform in LTspice, and subsequently analyzed for 11 different simulated comparator thresholds, resulting in a hit-detection efficiency value unique to its combination of current signal and threshold. Finally, the five efficiencies corresponding to the same hitrate and threshold values are averaged and the standard error of the mean calculated. The averaged efficiencies and their uncertainties are plotted in Figure 5.1.

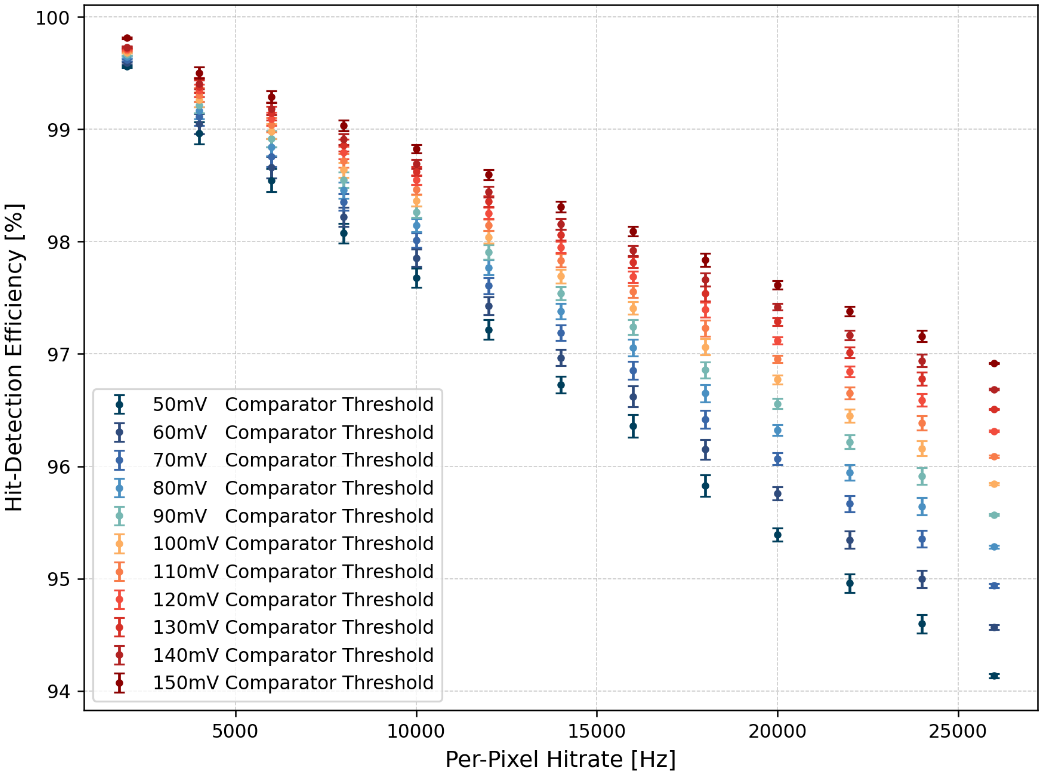


Figure 5.1: Average hit-detection efficiencies recorded for a range of simulated hitrates and comparator thresholds.

For a fixed comparator threshold, a linear decrease in the simulated hit-detection efficiency is observed as a function of per-pixel hitrate. The choice of threshold has a larger impact on the simulated efficiencies at high than at low hitrate values: the efficiency spread between thresholds at the lowest simulated hitrate 2000 Hz is merely $(0.256 \pm 0.015)\%$ compared to $(2.781 \pm 0.010)\%$ for the highest hitrate $R_h = 26\,000\,\text{Hz}$. This implies a similar or identical y-axis intersection coupled with a threshold-dependent slope for the linear efficiency curves.

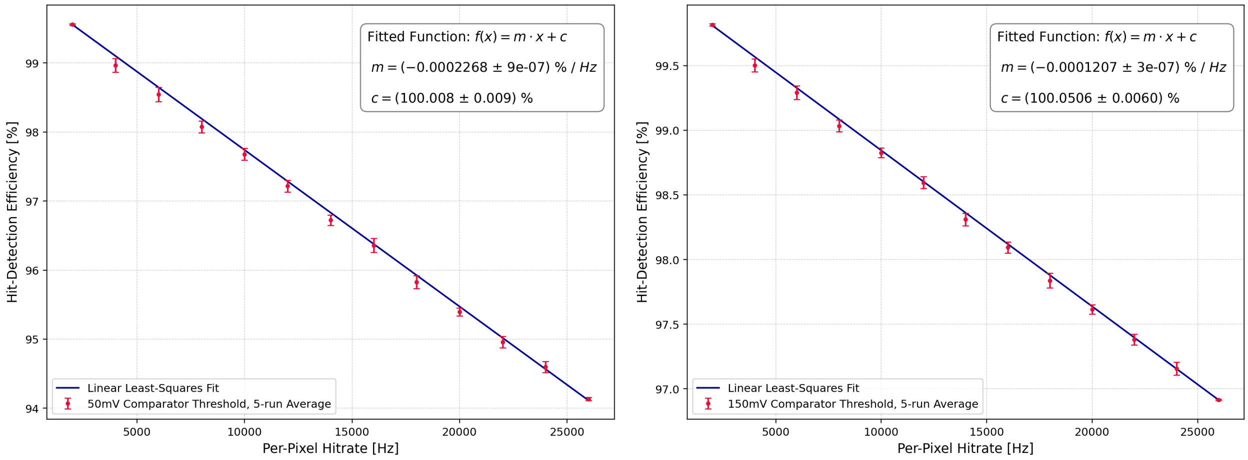
5.1 Modeling the Data

An equation modeling the hit-detection efficiency ϵ as a function of the per-pixel hitrate R_h and the comparator threshold Λ has been parametrized using the simulated results shown in Figure 5.1 as a reference:

$$\epsilon\left(R_h, \Lambda\right) = m\left(\Lambda\right) \cdot R_h + c \tag{5.1}$$

The parameters $m(\Lambda)$ and c represent the efficiency loss per hitrate, and the extrapolated maximum efficiency, respectively. For a fixed threshold Λ, Equation 5.1 is a linear function taking the hitrate R_h as its variable, matching the linear progression observed for the simulated efficiencies.

In order to arrive at values for the parameters $m(\Lambda)$ and c, a linear function is fitted to the observed efficiency progression for each value of Λ. Figure 5.2 shows example fits for the efficiency data recorded at the highest and lowest thresholds, together with the fit parameters calculated.



(a) $\Lambda = 50\,\text{mV}$

(b) $\Lambda = 150\,\text{mV}$

Figure 5.2: Example of linear fits to simulated hit-detection efficiency data for two thresholds.

The parameter c, which represents the y-axis intersection extrapolated from the fit data, also describes the theoretical maximum hit-detection efficiency that Equation 5.1 can compute. From the 11 linear fits calculated for the different threshold values, the average value $\bar{c}$ of the extrapolated maximum efficiency is $\bar{c} = (100.003 \pm 0.019)\%$. The uncertainty of $\bar{c}$ is calculated as the quadratic sum of the standard error of the mean and the average uncertainty of the fit. An efficiency larger than 100% is unphysical, so the upper bound for the parametrization of Equation 5.1 must be $c_{max} = 100\%$. The middle value calculated for $\bar{c}$ is above 100% likely as a result of the overcounting of hits, which adds an offset consisting of "fake" hits onto the observed comparator activations, leading to efficiencies larger than 100% in some edge cases, especially towards lower hitrates, as shown in section 5.2.

There are two different approaches to choosing a final value for c. The first is to choose a conservative value based on the calculated uncertainty of $\bar{c}$, for example by halving the uncertainty remaining below c_{max} and subtracting it from 100%. This approach would arrive at a final value of $c = 99.992\%$. The second approach is to accept the maximum value $c_{max} = 100\%$ as the final parameter, but constrain the domain of Equation 5.1 to hitrates above a certain minimum. Considering that the second approach more closely reflects the simulated data (compare value of $\bar{c}$) and that it does not impose external assumptions about the expected maximum efficiency onto the model, the final value chosen for the parametrization is $c = 100\%$.

The slopes of the linear fit functions are identified with the parameter $m(\Lambda)$ from Equation 5.1. When plotted against their corresponding threshold values, the slopes follow a pattern that can be approximated by a quadratic function:

$$m(\Lambda) = a\Lambda^2 + b\Lambda + \gamma \tag{5.2}$$

The shape of the slope vs. threshold plot is fitted using this quadratic approach, which yields values for the shaping parameters a, b and γ. The $m(\Lambda)$ values and the quadratic fit are shown in Figure 5.3.

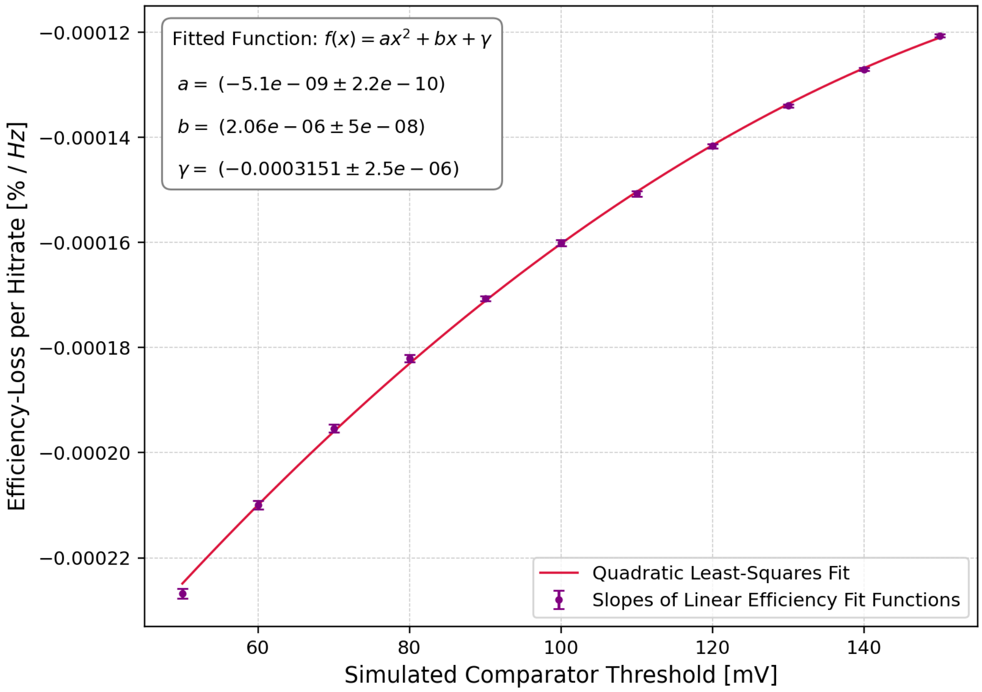


Figure 5.3: Slopes m of the linear functions fitted to the hit-detection efficiency data, shown alongside a quadratic fit and its calculated parameters.

Using the parabolic model for the efficiency loss per hitrate parameter $m(\Lambda)$, Equation 5.1 can be rewritten as follows:

$$\epsilon(R_h, \Lambda) = \left(a\Lambda^2 + b\Lambda + \gamma\right) \cdot R_h + c \tag{5.3}$$

Parameter	**Value**	**Uncertainty**
a	$-5.10 \cdot 10^{-9}$	$2.2 \cdot 10^{-10}$
b	$2.06 \cdot 10^{-6}$	$5 \cdot 10^{-8}$
γ	$-3.151 \cdot 10^{-4}$	$2.5 \cdot 10^{-6}$
c	100	$*$

Table 5.1: Summary table of fit parameters used for the parametrization of Equation 5.3.

(*): The uncertainty on c is not given because the chosen value does not match the middle value calculated for $\bar{c}$ and because a physical uncertainty would have to be unilateral, not allowing for values above 100%.

The two-dimensional field described by Equation 5.3 assigns an efficiency $[\epsilon] = \%$ to every pair of hitrate and threshold values (R_h, Λ) where $[R_h] = \mathrm{Hz}$ and $[\Lambda] = \mathrm{mV}$. Strictly speaking, the parametrization found using the simulated data only holds within the range of simulated hitrates and thresholds, meaning $(R_h, \Lambda) \in [2000\,\mathrm{Hz}, 26\,000\,\mathrm{Hz}] \times [50\,\mathrm{mV}, 150\,\mathrm{mV}]$. Furthermore, every pair of hitrate and threshold values that is not part of the original dataset generated by the simulation and shown in Figure 5.1, for example $(R_h, \Lambda) = (3600\,\mathrm{Hz}, 112\,\mathrm{mV})$, is assigned a predicted efficiency based on the interpolation derived from the fit functions used for the parametrization. With these considerations in mind, Figure 5.4 shows a contour plot for the parametrized efficiency as a function of hitrate and threshold.

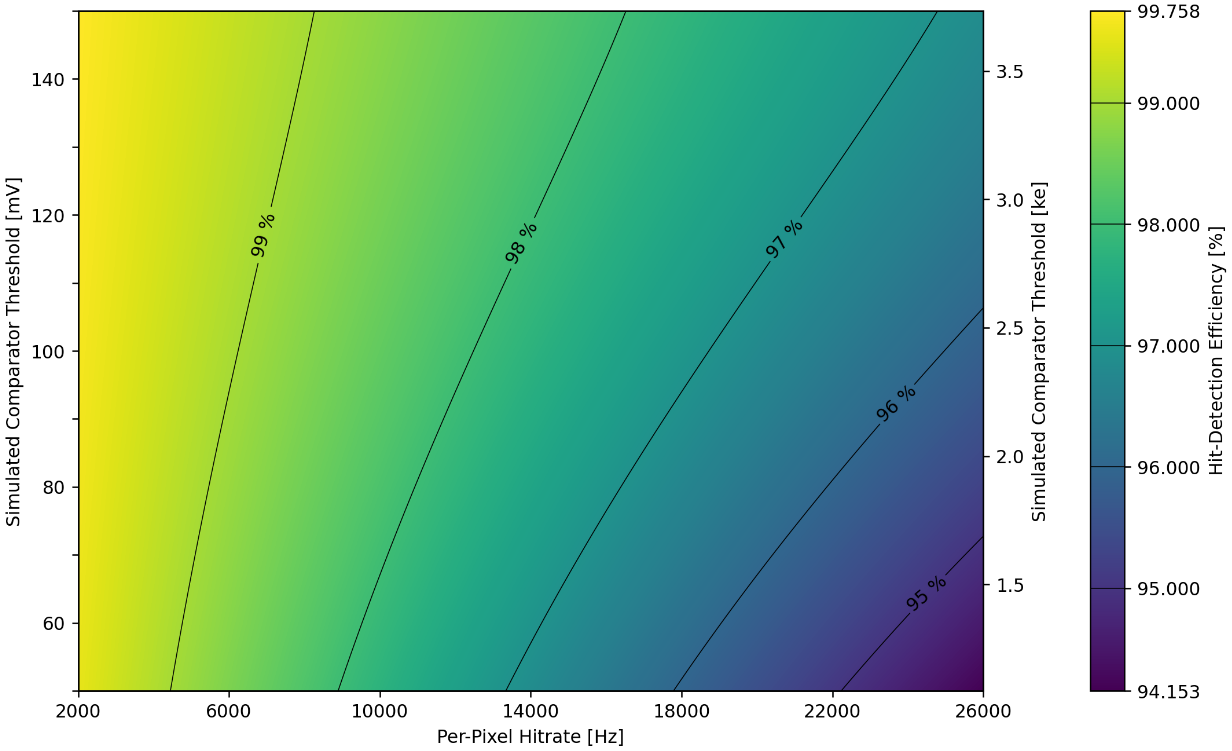


Figure 5.4: Contour plot of the parametrized Equation 5.3. The charge-equivalence of the threshold voltage is calculated from the CSA feedback capacity C_f using Equation 4.15.

It is important to emphasize that the results shown in Figure 5.4 reflect the conditions and parameters underlying the simulation chosen for this study, so they should be treated as orientative and understood as beholden to the configurations described in chapter 4.

5.2 Validating the Model

To test the validity of the parametrized efficiency equation, the hit-detection efficiencies have been simulated for threshold values not included in the original set of 11 thresholds, and subsequently compared to the "predicted" value calculated with Equation 5.3. The results are displayed in Figure 5.5.

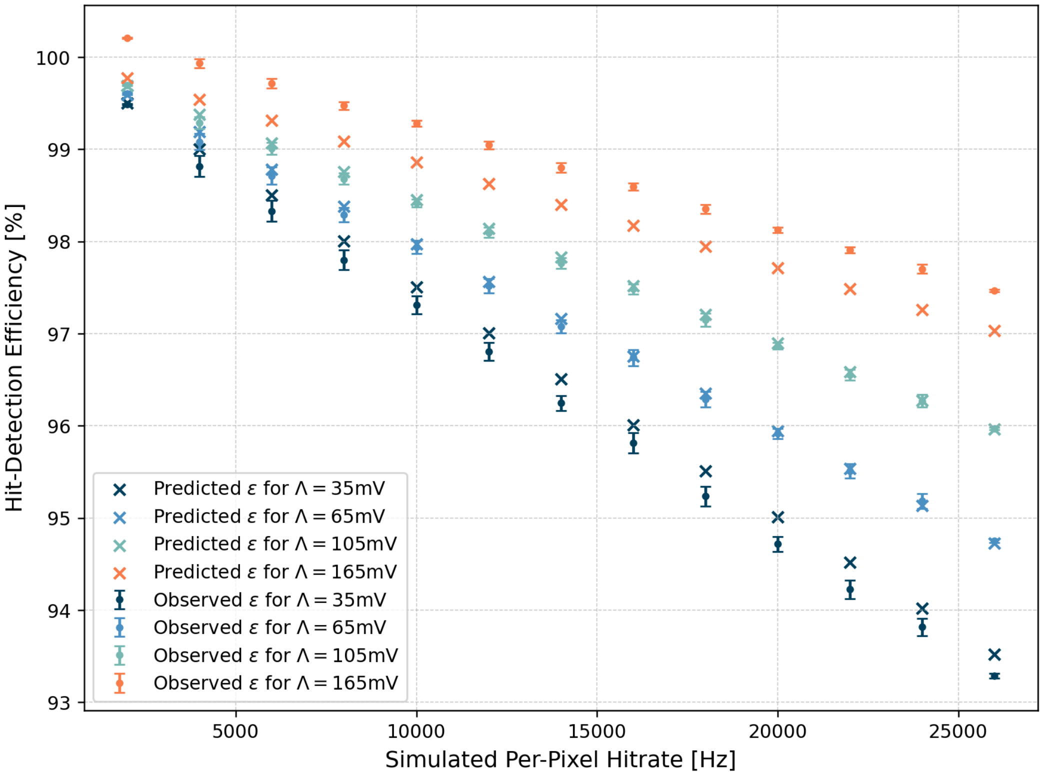


Figure 5.5: Observed (simulation) and predicted (Equation 5.3) efficiencies for different threshold and hitrate values.

The observed and predicted efficiencies only show good agreement for threshold values within the interval of $[50, 150]$ mV. Lower values, such as the threshold $\Lambda = 35$ mV shown in Figure 5.5, exhibit moderate agreement with the predicted efficiencies, but lie generally below the predicted value. The disagreement is worse for thresholds higher than the maximum original simulated value of $\Lambda = 150$ mV, where the observed hit-detection efficiencies lie very significantly above the predicted ones. Importantly, one observed value for a threshold of $\Lambda = 165$ mV even lies above 100%, indicating overcounting in the simulation's waveform analysis step. The overcounting is not unique to this combination of hitrate and threshold, and its impact and possible origin is explored in section 5.4.

In order to investigate the hit-detection efficiency for a hitrate higher than the maximum value of $R_h^{max} = 26\,460$ Hz calculated in subsection 4.2.1 and above the original set of hitrates used for parametrizing the interpolation model, an additional set of temporal current pulses and their respective amplifier waveforms has been produced for a hitrate of $R_h = 27\,000\,\text{Hz}$. A comparison between the predicted and observed hit-detection efficiencies for this high hitrate is provided in Figure 5.6. The data shows generally good agreement with the predicted values calculated from Equation 5.3, especially for thresholds between 100 mV and 140 mV. The outliers at 150 mV and below 90 mV deviate from the observed efficiencies by less than 0.11%, but still indicate that excellent agreement between predicted and observed efficiencies is retained only for thresholds in the range between 100 mV and 140 mV when simulating hitrates larger than 26 000 Hz.

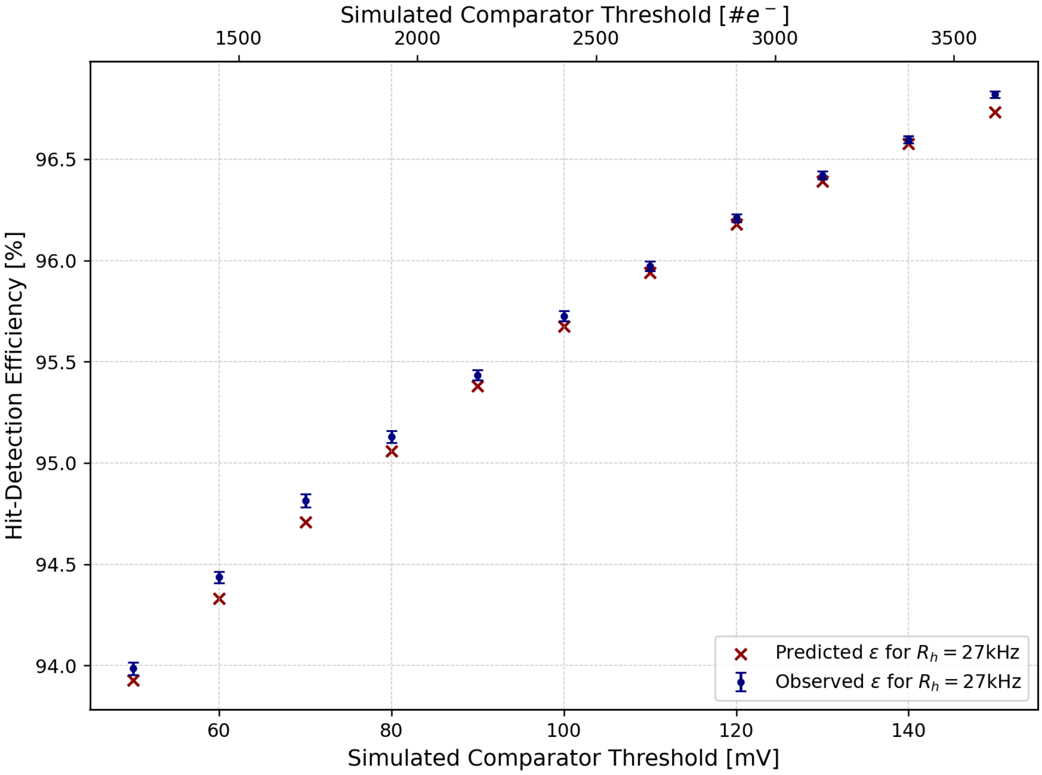


Figure 5.6: Observed (simulation) and predicted (Equation 5.3) efficiencies for a hitrate of $R_h = 27000\,\text{Hz}$ as a function of the simulated comparator threshold.

The linear decrease in hit-detection efficiency as a function of hitrate (Figure 5.1), and the good agreement shown by a quadratic fitting model for describing the slope of these linear functions (Figure 5.3) for thresholds in the interval $[50, 150]$ mV, allows Equation 5.3 to match the efficiency of the simulated data well, albeit only for $(R_h, \Lambda) \in [2000\,\text{Hz}, 26\,000\,\text{Hz}] \times [50\,\text{mV}, 150\,\text{mV}]$. For thresholds above $\Lambda = 150\,\text{mV}$ and hitrates lower than $R_h = 2000\,\text{Hz}$, the parametrization found for Equation 5.3 is a bad descriptor of the efficiencies observed following the simulation process, primarily due to the fact that overcounting of hits leads to impossibly high hit-detection efficiencies at those configured values of R_h and Λ. A clarifying example is shown in Figure 5.7, where the observed and predicted efficiencies for a hitrate of $R_h = 1000\,\text{Hz}$ are plotted as a function of the simulated comparator threshold. Observed and predicted efficiencies match very closely for thresholds up to and including $\Lambda = 70\,\text{mV}$, after which the observed efficiencies exhibit a sudden jump to values above 100%, where their distribution fluctuates without showing a clear pattern. The lower the simulated hitrate and the higher the simulated threshold, the more likely it is for overcounting to occur. It is important to note that the errorbars plotted for the observed hit-detection efficiencies underrepresent the actual uncertainty, since they do not include the systematic error made by overcounting. A more detailed discussion on overcounting can be found in section 5.4.

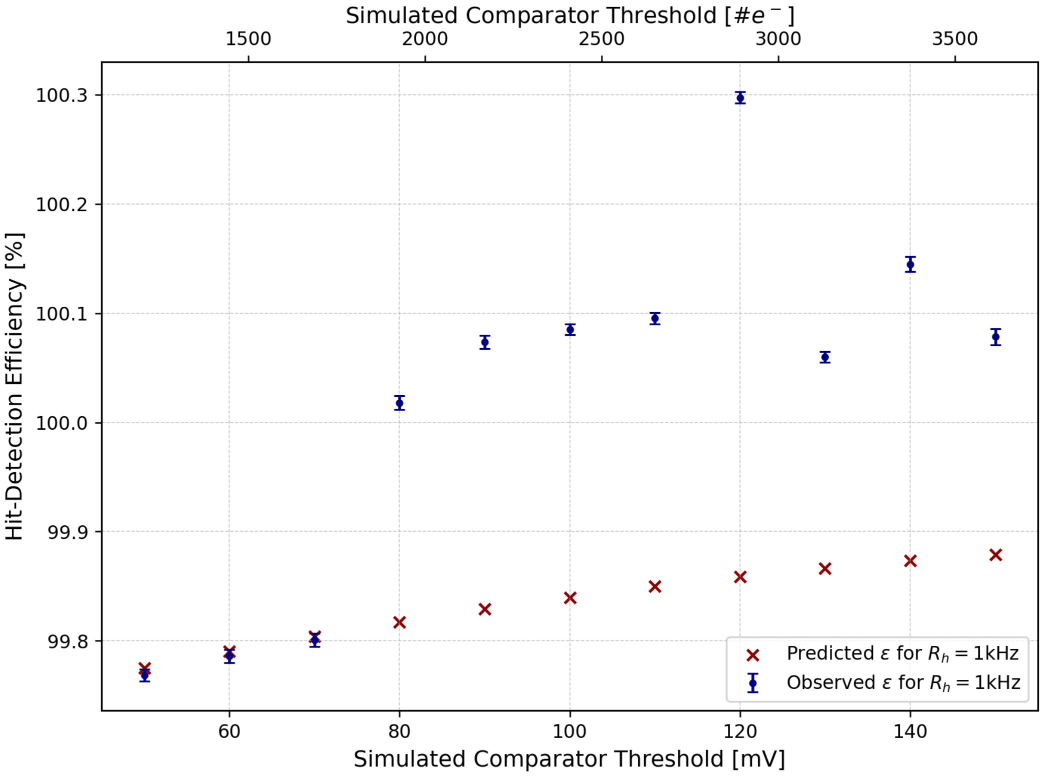


Figure 5.7: Observed (simulation) and predicted (Equation 5.3) efficiencies for a hitrate of $R_h = 1000\,\mathrm{Hz}$ as a function of the simulated comparator threshold.

Extrapolating the predictions made by Equation 5.3 beyond the hitrate and threshold bounds upon which the model has been parametrized not only leads to increasingly inaccurate results, but is also of somewhat limited utility. Overcounting, the main source of systematic uncertainty in the efficiency data (disregarding the inaccuracies of the simulations with respect to "real" sensors), is most impactful for low hitrates, where the performance of existing sensor designs is understood to be adequate. Furthermore, operating a "real" sensor at low thresholds increases the likelihood of noise hits, as outliers are more likely to cross the threshold the lower it is set. Very high thresholds are also problematic, as true events with low amounts of charge deposition are be discriminated by the comparator if the threshold is set too high. Additionally, higher thresholds incur a penalty in time resolution as they lead to a slower comparator response. This effect, known as timewalk, is discussed in section 5.6 for the simulated circuit.

5.3 Efficiency at Maximum Hitrate

As shown in subsection 4.2.1, the estimated upper bound for the hitrate per MightyPix2 pixel installed in the UP at LHCb is $R_h^{max} = 26\,460\,\mathrm{Hz}$. In contrast to the 5 temporal hit distributions generated for the rest of the investigated hitrates, 15 have been generated for R_h^{max} to provide a broader statistical sample. Figure 5.8 shows the simulated hit-detection efficiency plotted against the simulated comparator threshold for R_h^{max}. A continuous line representing the predicted efficiency calculated with Equation 5.3 is added to aid in interpolating between the measured points, as the good agreement between the observed and predicted efficiencies for thresholds between $\Lambda = 100\,\mathrm{mV}$

and $\Lambda = 140\,\mathrm{mV}$ suggests that the model is a reasonable fit for the data in this range.

Overall, the distribution of observed efficiency values for R_h^{max} carries similarities to the distribution plotted for $R_h = 27\,\mathrm{kHz}$ in Figure 5.6, as is expected from the similarity of the two hitrate values. Most notably, the thresholds for which the observed efficiencies deviate from the predicted values are the same at $\Lambda \in \{60, 70, 80, 90, 150\}\mathrm{mV}$, and when a deviation is present, the observed efficiency lies above the predicted value. The largest gap for R_h^{max}, defined as the difference between the observed and predicted hit-detection efficiency, is $(0.073 \pm 0.018)\%$ and is observed at $\Lambda = 70\,\mathrm{mV}$. The largest gap between the observed and predicted efficiencies for the thresholds that lie between $\Lambda = 100\,\mathrm{mV}$ and $\Lambda = 140\,\mathrm{mV}$ is $(0.016 \pm 0.015)\%$. The datapoint for the observed efficiency for R_h^{max} at $\Lambda = 150\,\mathrm{mV}$ lies above the predicted value.

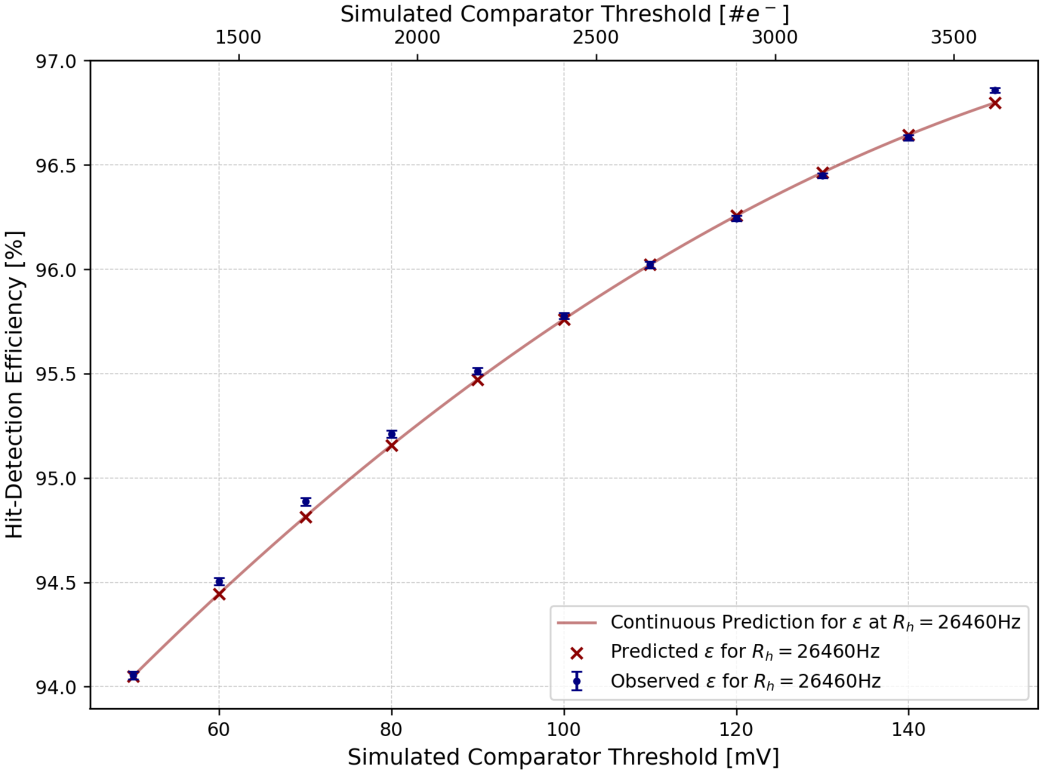


Figure 5.8: Observed (simulation) and predicted (Equation 5.3) efficiencies for a hitrate of $R_h^{max} = 26\,460\,\mathrm{Hz}$ as a function of the simulated comparator threshold.

According to the Upstream Pixel specification [28], an in-time efficiency higher than 99% is required of the sensor design to meet performance expectations. The in-time efficiency is not identical to the hit-detection efficiency, as it also requires the time resolution of the sensor to be good enough that the convolution of the hit-detection efficiency and timing efficiency together equal 99% or higher. Only when assuming an otherwise ideal sensor, meaning perfect time resolution, and no noise present in any electrical component, can the hit-detection efficiency discussed here be equated to the in-time efficiency. However, since the in-time efficiency can only be worsened by the time resolution, not improved, a hit detection efficiency higher than 99% is a necessary condition to meet the in-time efficiency requirement.

None of the observed hit-detection efficiencies reaches the required 99%, as they all lie below 97%. Of the efficiencies observed for a simulated hitrate of $R_h^{max} = 26\,460\,\text{Hz}$, the first to cross 96% is the datapoint for a threshold of $\Lambda = 110\,\text{mV}$, with a value of $\epsilon_{110} = (96.022 \pm 0.014)\%$. The observed efficiencies for other thresholds which surpass 96% are $\epsilon_{120} = (96.244 \pm 0.013)\%$, $\epsilon_{130} = (96.445 \pm 0.011)\%$, $\epsilon_{140} = (96.631 \pm 0.013)\%$, and the final simulated efficiency $\epsilon_{150} = (96.858 \pm 0.012)\%$. The threshold setting required to observe a hit-detection efficiency higher than 96% for the simulated amplifier circuit and the hitrate R_h^{max} is therefore $\Lambda_{min}^{96\%} = 110\,\text{mV}$, which equates to an injection charge of roughly 2650 electrons when calculated with Equation 4.15 assuming linear amplification with the feedback capacitance $C_f = 3.86\,\text{fF}$. However, it is important to note that overcounting may shift some observed efficiencies upward, especially at the higher threshold settings, so the true lowest threshold required for observing any of the efficiencies plotted might be even higher.

5.4 Overcounting

An example of an overcounted pulse taken from the waveform of one of the signals used to produce Figure 5.7 is shown in Figure 5.9. The overcounting here takes place due to a short and sudden irregularity in the shape of the waveform. The original PWL file only contains one current pulse at the shown timestamp, so the second spike visible in the plot is likely an error or bug caused during the process of exporting the waveform, as the resolution of the output is limited, allowing for such sudden spikes. Judging by the magnitude and shape of the pulse shown, it is likely that the overcounted part of the waveform belongs to the falling edge of the large pulse to its immediate left. Importantly, calling this effect "overcounting" is only true in the sense that the "false" pulses being counted are not linked to the simulated current pulses. They are "true" counts in the sense that a comparator activation is indeed triggered by the shape of the waveform and not by a bug in the code.

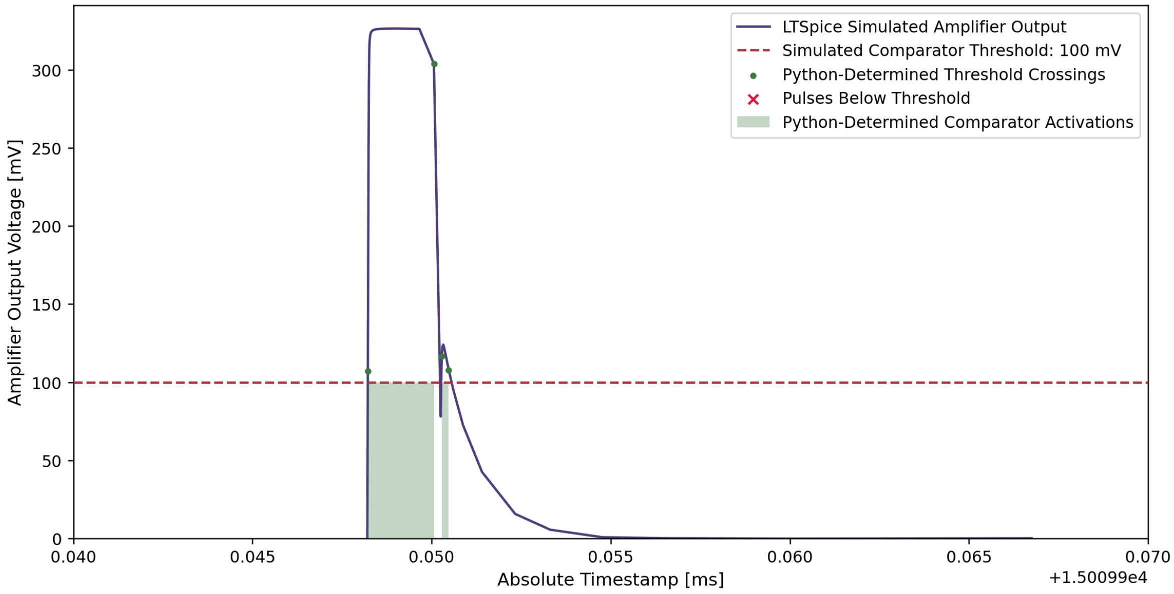


Figure 5.9: Amplifier output pulse analyzed for hit detection showing overcounting.

In order to quantify the impact of overcounting on the observed hit-detection efficiencies, two temporally equidistant hit distributions have been generated for the 10^5 current pulses generated in Allpix^2. The first corresponds to a "high hitrate" scenario based on $R_h^{max} = 26\,460\,\text{Hz}$, where the time gap Δt_h between two adjacent pulses is set to $\Delta t_h = 1/R_h^{max}$. The second corresponds to a "low hitrate" scenario, where a hitrate of $R_h = 1000\,\mathrm{Hz}$ is assumed and the hit time gap set to $\Delta t_h = 1 \times 10^{-3}\,\mathrm{s}$. The goal of these two distributions is to investigate the amount of hits counted for a waveform where amplifier voltage signals do not merge into one another causing pile-up, therefore eliminating the main variable in hit-detection inefficiencies. The number n_{over} of overcounted hits can be determined by counting the number of comparator activations n_c and subtracting from it the difference between the total number of simulated current pulses N_p and the number of amplified pulses with amplitudes below threshold n_{low}:

$$n_{over} = n_c - (N_p - n_{low}) \tag{5.4}$$

This method works by assuming that, without pile-up, all "true" pulses above threshold will be identified, and some "false" pulses will be counted by the comparator as well. Since overcounting happens only when a signal is present above threshold and that signal is not a "true" pulse, but some sort of artifact in the waveform, pulses that do not cross the threshold have to be subtracted from the total number of pulses in order to isolate the effect. As an example, if 100100 comparator activations are counted, but 200 of the 10^5 pulses lie below threshold, the total number of overcounted pulses is 300, not merely 100.

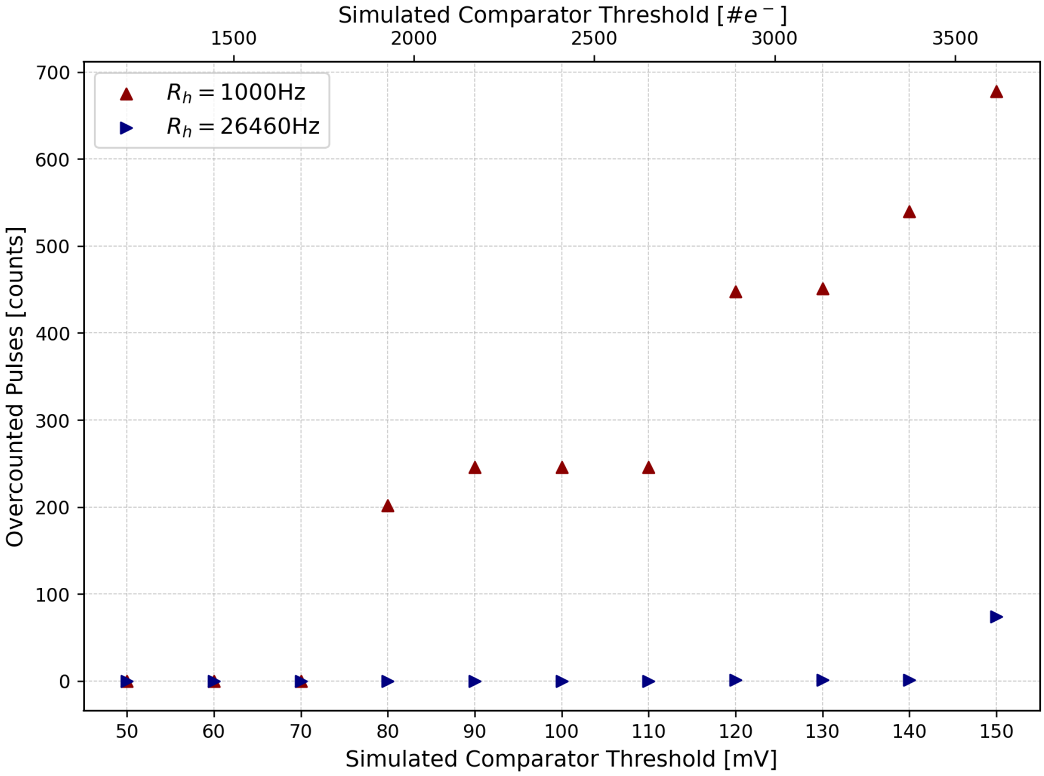


Figure 5.10: Number of overcounted pulses as a function of simulated comparator threshold for two hitrates. Pulses simulated at equidistant timestamps to prevent pile-up.

Overcounting is most prevalent at low hitrates and is worse for higher thresholds. The step-like shape of the overcounted pulses implies that the irregularities in the waveform shapes happen at discrete pulse heights (voltage values), further pointing towards overcounting being a digital artifact. Overcounting has only been encountered late in the process of writing this thesis, partly due to its effect remaining hidden in the original set of analyzed data. Searching for an affected pulse, such as the one shown in Figure 5.9, remains a manual task that is complicated by the fact that only relatively few of the 10^5 pulses exhibit overcounting at moderate hitrates. For these reasons, the exact mechanism causing overcounting is not yet understood, and neither are the causes for why overcounting is so much more common at lower hitrates. A possible explanation is the length of the simulated waveform: with a hitrate of 1000 Hz, a waveform containing the amplifier response to 10^5 pulses is around 100 seconds long, plus or minus a few fractions of a second due to the statistical fluctuation of the temporal pulse distributions. If the LTspice waveform export has a dynamic temporal resolution, longer waveforms might exhibit a lower density of points per unit time than shorter ones, possibly causing resolution-related bugs to occur. For clarity, the LTspice command disabling waveform compression has been used for every simulated waveform in this thesis.

For R_h^{max} the impact of overcounting appears to be minimal within the range of tested thresholds, as only the last value $\Lambda = 150\,\text{mV}$ shown in Figure 5.10 appears to exhibit more than 1 overcounted pulse. The efficiencies shown in the previous section can therefore likely be taken as valid, though with the caveat that the effect of overcounting for waveforms exhibiting pile-up is not currently well understood.

5.5 Source of Inefficiencies

The equidistant hit distribution used for quantifying the effect of overcounting in the previous section is also useful to get an idea of the source of hit-detection inefficiencies, as it produces a current pulse distribution where no pulses are close enough in time to cause pile-up in the amplifier. The only other source of hit-detection inefficiencies simulated with the methods described in this thesis are pulses with low associated charge, that fail to cross the configured comparator threshold. Without pile-up, these low-charge pulses should be isolated and easily countable.
Due to the configured depth of the simulated depletion zone in each pixel being 150 µm, the majority of particles simulated in Allpix^2 deposit enough charge to surpass even the highest simulated threshold of $\Lambda = 150\,\text{mV}$. Figure 5.11 shows the distribution of the charge contained within each current pulse written by the *NetlistWriter* during the Allpix^2 simulation. Three threshold values are converted to a charge using Equation 4.15 and plotted alongside the distribution for reference.

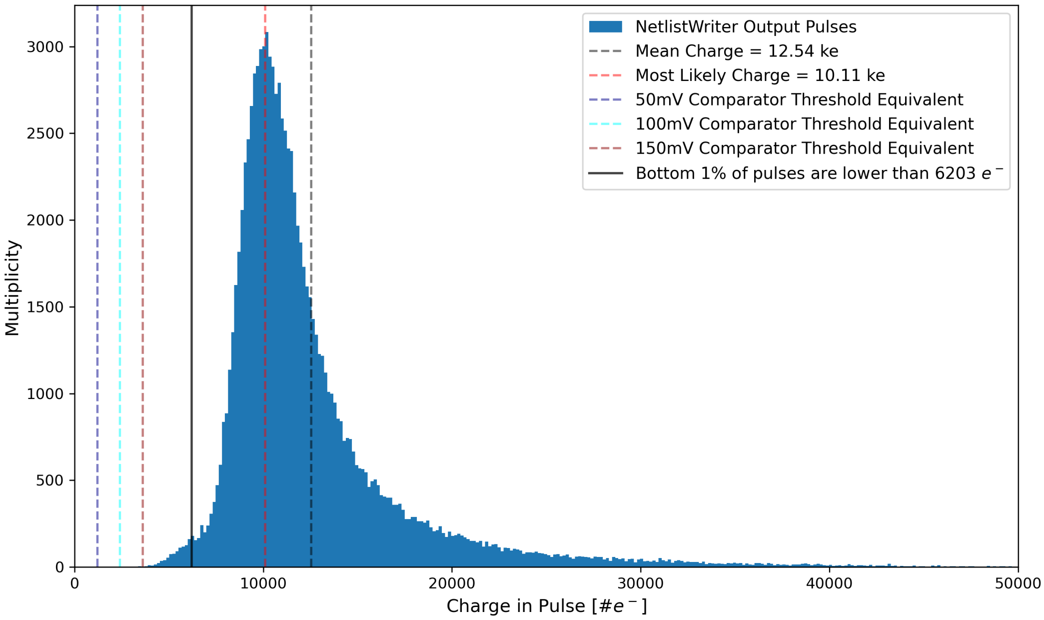


Figure 5.11: Charge contained in each of the 10^5 current pulses written by the *NetlistWriter* module following the Allpix^2 simulation.

The shape of the pulse charge distribution shown above is similar to the energy deposition distribution produced in Allpix^2 during the simulation of the 10^5 events: it resembles a Landau distribution, having a long tail towards high charge values. However, unlike the distribution shown in Figure 4.2, there is a short triangular tail to the left of the distribution, which can be attributed to charge-sharing events. Particles intersecting the simulated sensor along the edge of a pixel or in a corner, which is shared by 3 neighboring pixels, cause a measurable charge deposition in multiple pixels. The closer to the edge or corner the particle crosses the sensor, the larger the percentage of the total deposited charge that ends up being collected by the neighboring pixel becomes. Since only the largest charge values contained within each written netlist are extracted by the code and written to a current pulse for use in the LTspice simulation, those events whose charge is shared between 2, 3 or even 4 pixels appear as low-charge pulses in the left tail of this distribution.

As can be seen in Figure 5.11, even the highest tested threshold at 150 mV barely reaches the left tail of the current pulse distribution. With very few exceptions, almost all pulses entering LTspice should cross the threshold after amplification and be able to be counted as comparator activations, meaning that the overwhelming majority of the observed inefficiency should stem from the pile-up of output voltage pulses. It is worth noting that the charge-equivalence of the thresholds is calculated by plugging the threshold voltage into Equation 4.15 together with the feedback capacitance $C_f = 3.86\,\mathrm{fF}$, which assumes perfectly linear amplification. The simulated circuit does not, however, deliver perfect amplification due to the feedback current flowing through R_f, as is shown in Figure 4.14. The consequence is that the amount of charge in the input current pulse required to reach the thresholds is actually slightly larger than plotted in Figure 5.11. Using the equidistant hit distribution for R_h^{max} it is possible to compare the number of pulses identified below threshold by the waveform analysis code with the number of pulses below threshold according to the distribution of charge in the input currents (Figure 5.11).

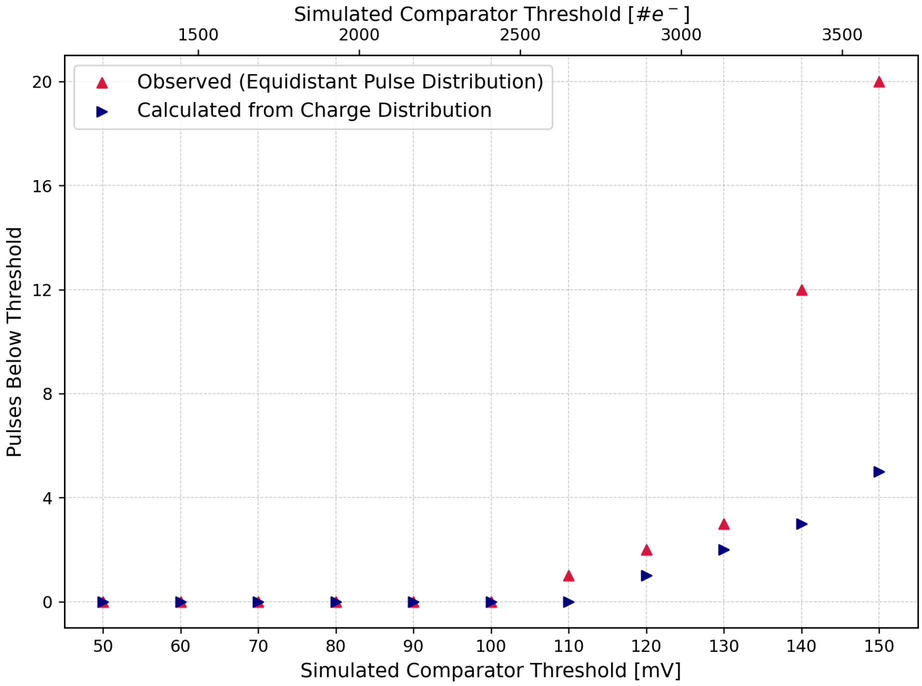


Figure 5.12: Number of amplified pulses that do not reach the comparator threshold as a function of the simulated comparator threshold.

A discrepancy is present between the number of amplifier pulses observed below threshold by the waveform analysis code (blue in Figure 5.12) and the number derived from the amount of current pulses whose contained charge is lower than the calculated charge-equivalence of the simulated thresholds (red). The discrepancy can be explained by the observed amplification behavior of the simulated circuit, as the charge needed to reach the threshold is actually higher than what is calculated assuming perfectly linear charge-to-voltage conversion.

In total, the number of pulses observed to not have reached the highest comparator threshold $\Lambda = 150\,\mathrm{mV}$ for an equidistant current pulse distribution where pile-up is not a factor, is 20. Considering that 10^5 pulses are simulated in total, this represents a maximum efficiency loss due to low-charge pulses of 0.02%. It is therefore safe to say that pulse pile-up is the dominating factor contributing to the hit-detection inefficiency observed in this study.

5.6 In-Time Efficiency

The results of the hit-detection efficiency study shown in Figure 5.1 and parametrized by Equation 5.3 indicate that higher comparator thresholds lead to higher efficiencies. Considering that observing pulse pile-up becomes less likely in the higher voltage ranges, that pile-up is the dominant contributor to inefficiency, and that the majority of particles deposit enough charge in the simulated pixel to saturate the amplifier, it is tempting to recommend a high comparator threshold based on hit-detection efficiency alone. However, doing so would miss the second major aspect in the performance of a sensor: time resolution. Particles not only have to be detected, they also have to be assigned to the specific LHC bunch crossing during which they originated in order to confidently be assigned to a track. The percentage of hits being identified within a 25 ns window is required to be higher than 99% for both the Mighty Tracker and Mighty Pixel [28].

Using LTspice, the behavior of a simple comparator can be emulated for individual amplifier pulses using the built-in waveform measurement tools. Doing this allows the measurement of a time delay t_{del}, defined as the difference between the time t_0 at which a current pulse begins introducing charge into the amplifier and the time t_{com} at which the amplified voltage pulse crosses the defined threshold and triggers the activation of the comparator:

$$t_{del} = t_{com} - t_0 \tag{5.5}$$

Since the rise time of the amplified voltage pulse is not instantaneous, it takes longer for pulses to cross higher comparator thresholds, making t_{del} larger. Low-amplitude pulses cross the threshold later than high-amplitude pulses, as the threshold's value represents a larger fraction of their total height, an effect known as *timewalk*. Timewalk has a negative impact on a sensor's time resolution and the in-time efficiency, as small pulses are invariably recognized later, causing a number of them to fall outside the 25 ns bunch crossing identification window. Timewalk can be corrected, but doing so requires offloading additional information from the pixels, which increases the bandwidth requirements for the readout of each sensor. In an ideal scenario, the time resolution of a sensor would be good enough that correcting for timewalk is not needed to achieve $> 99\%$ in-time efficiency. An example of timewalk produced with the simulated circuit in LTspice is shown in Figure 5.13. The simulated time delay for several comparator thresholds is plotted against the amplitude of the voltage pulse in Figure 5.14.

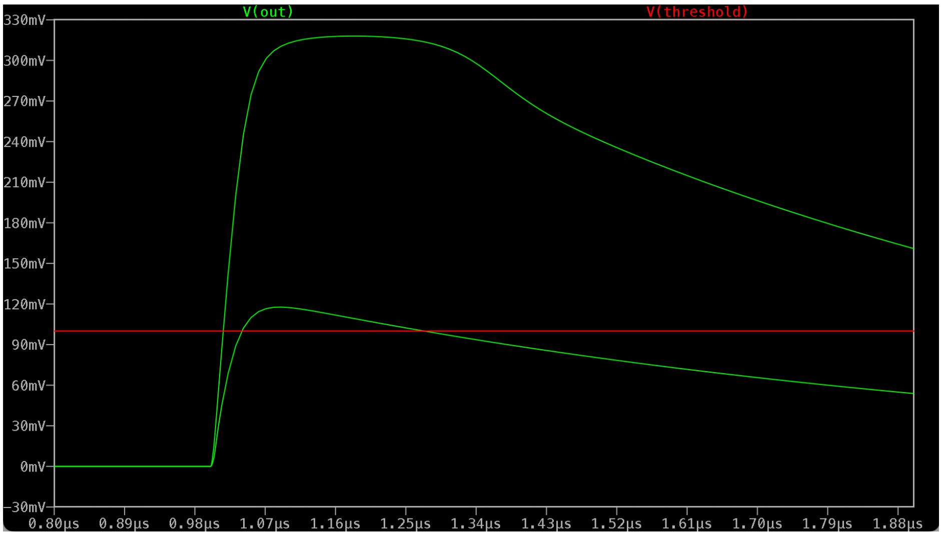


Figure 5.13: Amplifier output pulses of different heights simulated in LTspice. A threshold of 100 mV is plotted, showing the smaller pulse crossing the threshold later than the large pulse.

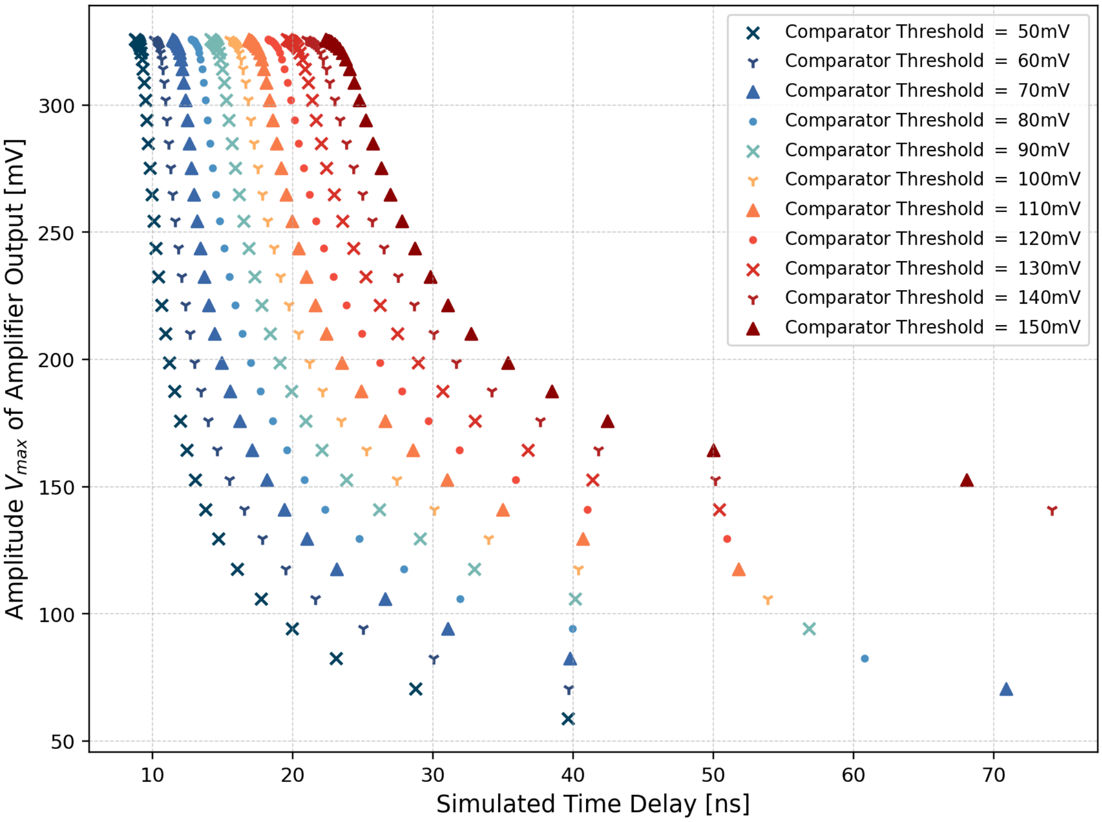


Figure 5.14: Amplifier output voltage as a function of the corresponding observed time delay. Plot generated with LTspice using discrete amounts of charge injected into the amplifier circuit.

The simulated time delay demonstrates that timewalk is most impactful for a combination of a low amplitude voltage pulse and a high comparator threshold. With increasing thresholds, Figure 5.14 shows a complete shift of the simulated time delay characteristic toward higher t_{del} values, including the minimum time delay achievable by the circuit for the largest pulses. This means that, with high comparator thresholds, the circuit not only becomes slower for low-amplitude pulses, but for any pulse. In practice, the absolute time delay plotted in Figure 5.14 is not directly the quantity impacting the time resolution of the sensor, since time resolution is commonly defined as the width of the distribution of time residuals measured between a timing sensor or track timestamp, and the device under test. Residuals measure time delay relative to another quantity, as it is not possible to know the absolute time at which the particle physically traverses any particular sensor. A constant minimum offset to the absolute time delay is therefore not influential on the time resolution, as it only impacts the absolute value of the observed residuals and not the width of their distribution. To better compare the impact of different threshold values on a relative time delay, a further plot has been created with all minimum time delays shifted to zero.

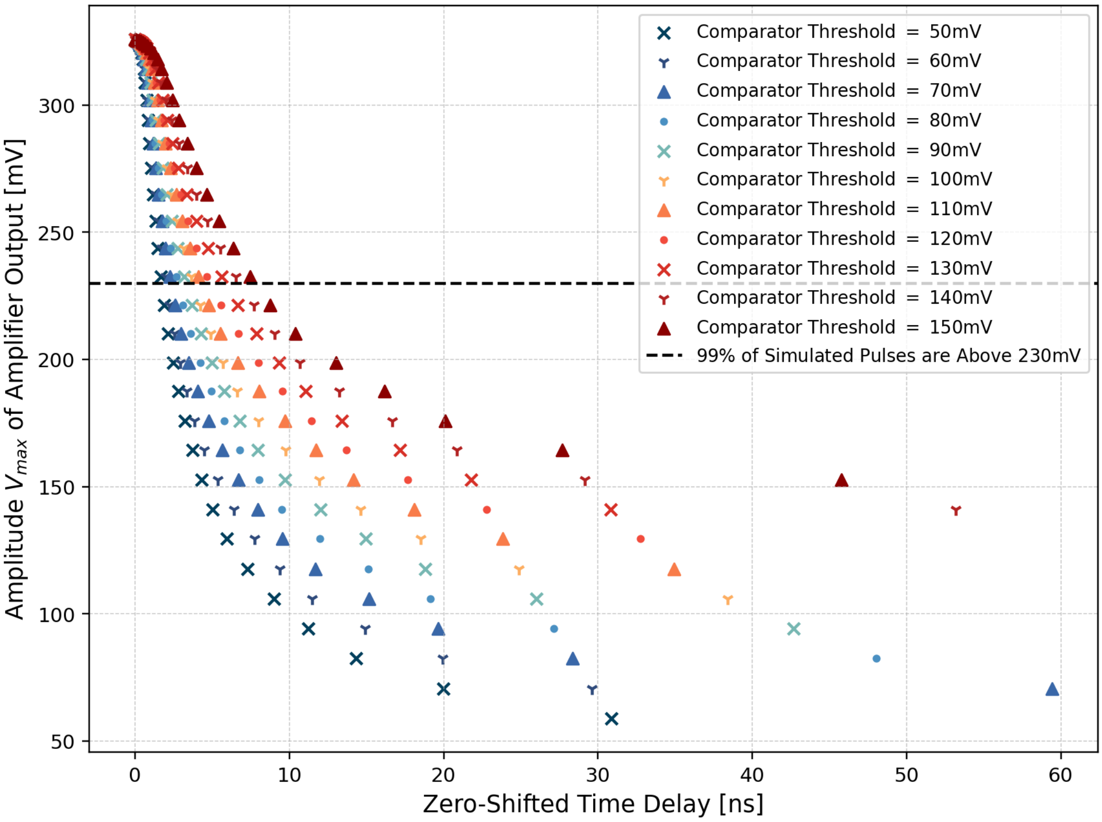


Figure 5.15: Amplifier output voltage as a function of the corresponding zero-shifted time delay. Horizontal line showing the lowest amplitude of the top 99% of simulated pulses.

The 230 mV line plotted alongside the relative time delays in Figure 5.15 represents the lowest amplitude reached by the 99% largest input current pulses after amplification. Only 1% of current pulses written during the Allpix^2 simulation carry less than $Q_{min} = 6203$ electrons of charge and fall below the dotted line as a consequence. The figure of 230 mV is arrived at by inferring the amplitude corresponding to Q_{min} from the amplifier characteristic taken of the simulated circuit in LTspice, as is shown in Figure 5.16.

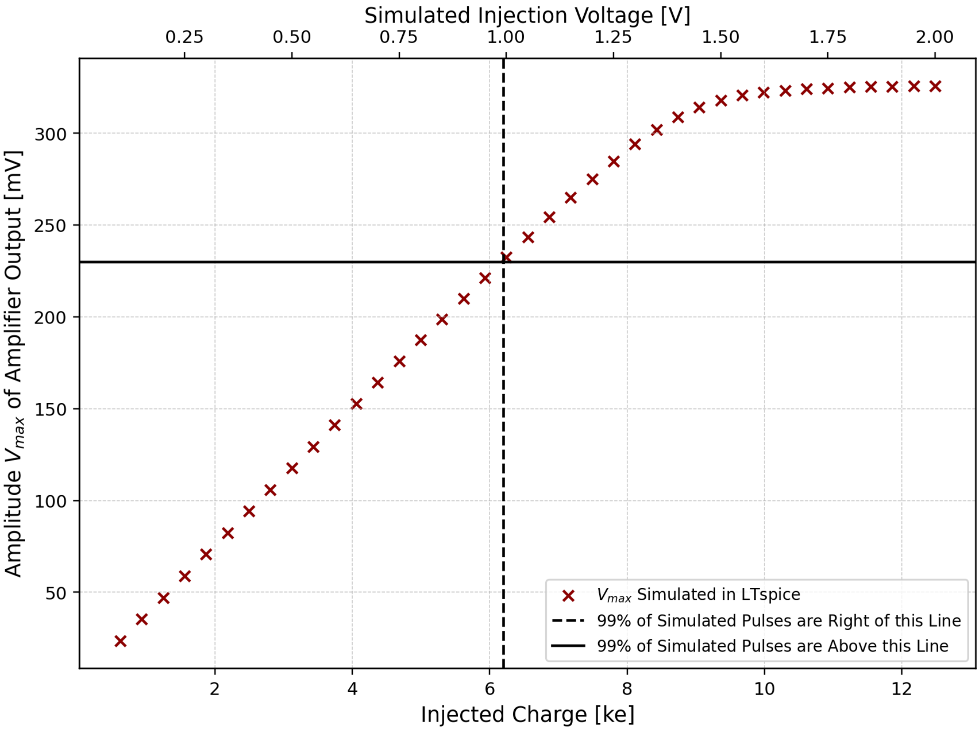


Figure 5.16: Amplifier characteristic taken in LTspice for the simulated amplifier circuit. Horizontal line showing the intersection between the linear portion of the amplifier response and the vertical marker corresponding to an input charge of $Q_{min} = 6203$ electrons.

The largest zero-shifted time delay for the 99% of simulated events with the highest charge deposition is $t_{del} = 7.47\,\mathrm{ns}$, corresponding to a comparator threshold of $\Lambda = 150\,\mathrm{mV}$. Therefore, fewer than 1% of simulated events lead to a comparator response that is slower than 7.47 ns after zero-shifting the data, even for the highest (and thereby slowest) threshold. These results suggest that the time performance of the simulated circuit is not the limiting factor for the in-time efficiency of the simulated sensor, but a more in-depth timing study incorporating the Allpix^2 to LTspice workflow has yet to be carried out.

Chapter 6

Summary and Outlook

The simulation workflow described in this thesis provides a method for approximating the behavior of an amplifier circuit under conditions modeled after those expected in the innermost part of the proposed Upstream Pixel subdetector at LHCb. The method is unique in that it implements the newly developed *NetlistWriter* module to combine a detailed particle interaction and charge deposition simulation in Allpix^2 with an equally detailed and customizable analogue circuit simulation in LTspice.

The simulated amplifier circuit designed for this thesis, consisting of an operational amplifier with an RC feedback loop, delivers an amplification behavior comparable to that of a measured Run2021v2 amplifier circuit for a feedback capacitance of $C_f = (3.86 \pm 0.25)$fF and a feedback resistivity of $R_f = (259 \pm 13)$MΩ. The agreement is best for injection charges above 7000 electrons, where the simulated amplifier begins saturating for a maximum output voltage of 330 V. The simulated and measured pulse widths match closely above that charge as well, with a value around 2 µs measured from the moment the signal crosses 15% of its amplitude during the rising edge to the moment it falls below 15% again. The simulation is not a good model for the measured amplifier at low injection charge values, where the amplitude of the simulated pulses is consistently lower than the equivalent measurement. The effect on the simulated hit-detection efficiency is minimal however, as at most 0.02% of the simulated current pulses are not registered as hits due to being discriminated by the simulated comparator threshold.

An estimate for the highest per-pixel hitrate experienced by a sensor with a pixel pitch of (84×84)µm in the high particle rate sections of the proposed LHCb Upstream Pixel subdetector is calculated to be 26 460 Hz. The maximum hit-detection efficiency achieved by the simulated circuit for this hitrate value is (96.858 ± 0.012)%, for the highest comparator threshold of 150 mV. It is suspected that this efficiency figure is overestimated as a product of overcounting, which has been observed to represent 0.074% of comparator activations for the stated combination of hitrate and threshold. This result indicates that the simulated amplifier would be unable to deliver the required in-time efficiency of 99% or higher for a pixel subjected to the estimated maximum per-pixel hitrate at the UP.

The largest simulated per-pixel hitrate value for which the observed hit-detection efficiency remains above 99% is 8000 Hz, where only the largest discrimination threshold of 150 mV is able to provide an efficiency of (99.03 ± 0.5)%. At 6000 Hz, every threshold above and including 110 mV delivers greater than 99% hit-detection efficiency, with the value for 110 mV being the lowest of the acceptance range at (99.04 ± 0.06)%. Only the lowest simulated comparator threshold of 50 mV fails to deliver over 99% efficiency at 4000 Hz, and the circuit provides greater than 99% hit-detection efficiency for any of the simulated thresholds by 2000 Hz. It is important to note that the efficiencies presented in this thesis are only single-pixel efficiencies, and do not represent the overall efficiency of an entire sensor, as secondary hits from clustering are disregarded and the gradient of particle rates across the surface of the sensor is not considered.

Over 99% of the simulated amplifier pulses have amplitudes larger than 230 mV. From the time delay characteristic of the simulated amplifier circuit it can be inferred that the upper 99% of events experience a relative (zero-shifted) time delay lower than 7.47 ns, that figure representing worst-case scenario of the slowest comparator threshold (150 mV) and a voltage pulse just barely above 230 mV. The in-time efficiency of the simulated combination of sensor and circuit is likely limited by the hit-detection efficiency, although a thorough timing study to confirm this has yet to be carried out.

Overcounting is the main source of systematic uncertainties in the data analysis, and it likely originates from the exporting of LTspice waveforms to text files for external processing. It is more common for low simulated hitrates, where it so significantly impacts the number of comparator activations that simulated efficiencies above 100% have been observed for certain combinations of low hitrates ($< 2000\,\mathrm{Hz}$) and comparator thresholds starting at 80 mV. The extent to which overcounting is a relevant effect for moderate hitrate values and in combination with pulse pile-up is currently not systematically quantified.

As it stands, the simulated combination of sensor and amplifier fails to deliver a hit-detection efficiency higher than 99% for a single pixel exposed to a hitrate higher than 10 000 Hz, predominantly due to the effects of pulse pile-up in the amplifier.

Outlook

All in all, the methods described in this thesis lay the groundwork for the creation of a highly detailed and highly customizable simulation chain in the future.

Beginning at Allpix^2, improvements can be made to the accuracy of the current signal formation by incorporating realistic electric field maps produced in TCAD, which allow the simulation of accurate charge carrier drift patterns and the resulting realistically shaped induced current pulses. Further improvements might see the introduction of a more accurate beam simulation, where the distribution of incidence angles of particles onto the sensor matches the conditions in a particular section of the Upstream Pixel subdetector, or any other simulated environment.

A clear opportunity to increase the accuracy of the results produced by the simulation chain is to incorporate the manufacturer's schematic of the pixel front-end into the LTspice electronics simulation. Doing so would allow more direct comparisons between the results obtained in the simulation and the real-life sensor being modeled. A further degree of accuracy could be achieved by simulating electronic noise, which LTspice and other similar programs support.

Further precision is possible by developing a temporal hit distribution that models the hit likelihood in a specific region of the simulated sensor based on kinematic considerations and not only stochastic ones.

References

[1] Timothy Gershon. *Status of Scoping Document review and next steps.* URL: `https://indi.to/RZYP7`.

[2] The LHCb Collaboration. "The LHCb Detector at the LHC". In: *Journal of Instrumentation* 3.08 (2008), S08005. DOI: `10.1088/1748-0221/3/08/S08005`. URL: `https://dx.doi.org/10.1088/1748-0221/3/08/S08005`.

[3] LHCb LHCb collaboration. *LHCb Upgrade II Scoping Document.* Tech. rep. Geneva: CERN, 2024. DOI: `10.17181/CERN.2RXP.HDK0`. URL: `https://cds.cern.ch/record/2903094`.

[4] *Framework TDR for the LHCb Upgrade II.* Tech. rep. Geneva: CERN, 2021. DOI: `10.17181/CERN.NTVH.Q21W`. URL: `https://cds.cern.ch/record/2776420`.

[5] Hermann Kolanoski and Norbert Wermes. *Particle Detectors: Fundamentals and Applications.* Oxford University Press, June 2020. ISBN: 9780198858362. DOI: `10.1093/oso/9780198858362.001.0001`. eprint: `https://academic.oup.com/book/43645/book-pdf/50609306/9780191899232_web.pdf`. URL: `https://doi.org/10.1093/oso/9780198858362.001.0001`.

[6] M. Tanabashi et al. "Review of Particle Physics". In: *Phys. Rev. D* 98 (3 2018), p. 030001. DOI: `10.1103/PhysRevD.98.030001`. URL: `https://link.aps.org/doi/10.1103/PhysRevD.98.030001`.

[7] Stephen M. Seltzer and Martin J. Berger. "Improved procedure for calculating the collision stopping power of elements and compounds for electrons and positrons". In: *The International Journal of Applied Radiation and Isotopes* 35.7 (1984), pp. 665–676. ISSN: 0020-708X. DOI: `https://doi.org/10.1016/0020-708X(84)90113-3`. URL: `https://www.sciencedirect.com/science/article/pii/0020708X84901133`.

[8] Heiko Augustin. "Development of a novel slow control interface and suppression of signal line crosstalk enabling HV-MAPS as sensor technology for Mu3e". PhD thesis. Heidelberg University, 2021.

[9] Hannah Schmitz et al. *Mighty Tracker – Performance Studies of the MightyPix for LHCb.* 2024. arXiv: `2402.08428 [physics.ins-det]`. URL: `https://arxiv.org/abs/2402.08428`.

[10] Simon Ramo. "Currents induced by electron motion". In: *Proceedings of the IRE* 27.9 (2006), pp. 584–585.

[11] Ivan Perić. *MightyPix.* URL: `https://indico.cern.ch/event/1214423/contributions/5285760/attachments/2611842/4513971/MightyPix.pdf`.

[12] Ivan Perić et al. "High-Voltage CMOS Active Pixel Sensor". In: *IEEE Journal of Solid-State Circuits* 56.8 (2021), pp. 2488–2502. DOI: `10.1109/JSSC.2021.3061760`.

[13] *Mightypix 2 - Architecture and User Manual.* URL: `https://mightypix2.docs.cern.ch/#design-changes-vs-baseline-designs`.

[14] S. Spannagel et al. "Allpix2: A modular simulation framework for silicon detectors". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 901 (2018), pp. 164–172. ISSN: 0168-9002. DOI: `https://doi.org/10.1016/j.nima.2018.06.020`. URL: `https://www.sciencedirect.com/science/article/pii/S0168900218307411`.

[15] S. Agostinelli et al. "Geant4—a simulation toolkit". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303. ISSN: 0168-9002. DOI: `https://doi.org/10.1016/S0168-9002(03)01368-8`. URL: `https://www.sciencedirect.com/science/article/pii/S0168900203013688`.

[16] Geant4 Collaboration. *Geant4 - Guide for Physics Lists, Release 11.2.* Rev8.0 - December 8th, 2023.

[17] Francois Krummenacher. "Pixel detectors with local intelligence: an IC designer point of view". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 305.3 (1991), pp. 527–532.

[18] Robert Szczygiel. "Krummenacher feedback analysis for high-count-rate semiconductor pixel detector readout". In: *Proceedings of the 17th International Conference Mixed Design of Integrated Circuits and Systems - MIXDES 2010.* 2010, pp. 412–415.

[19] Paul Schütze et al. "Allpix Squared User Manual". In: (2023).

[20] Elio Sacchetti. *Integrating Microelectronics Simulation into the Allpix Squared Framework.* May 2025. URL: `https://indico.cern.ch/event/1489052/timetable/?view=standard#1-integrating-microelectronics`.

[21] Laurence W. Nagel and D.O. Pederson. *SPICE (Simulation Program with Integrated Circuit Emphasis).* Tech. rep. UCB/ERL M382. 1973. URL: `http://www2.eecs.berkeley.edu/Pubs/TechRpts/1973/22871.html`.

[22] S. Scherl et al. "MightyPix at the LHCb Mighty Tracker — verification of an HV-CMOS pixel chip's digital readout". In: *Journal of Instrumentation* 19.04 (2024), p. C04045. DOI: `10.1088/1748-0221/19/04/C04045`. URL: `https://dx.doi.org/10.1088/1748-0221/19/04/C04045`.

[23] Shuaiyi Liu and Mingjie Feng. "Development of a MAPS-based upstream tracker for the LHCb upgrade II". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1070 (2025), p. 170044. ISSN: 0168-9002. DOI: `https://doi.org/10.1016/j.nima.2024.170044`. URL: `https://www.sciencedirect.com/science/article/pii/S0168900224009707`.

[24] *Internal Communications with Heiko Augustin.*

[25] Erwin Fehlberg. *Classical eight-and lower-order Runge-Kutta-Nyström formulas with stepsize control for special second-order differential equations.* Tech. rep. 1972.

[26] E. J. Schioppa et al. "Study of Charge Diffusion in a Silicon Detector Using an Energy Sensitive Pixel Readout Chip". In: *IEEE Transactions on Nuclear Science* 62.5 (2015), pp. 2349–2359. DOI: `10.1109/TNS.2015.2475124`.

[27] Canali Jacoboni et al. "A review of some charge transport properties of silicon". In: *Solid-State Electronics* 20.2 (1977), pp. 77–89.

[28] F. Guilloux K. Hennessy B. Leverington D. Wiedner Sebastian Bachmann L. Dittmann. "Pixel sensor specifications for the LHCb MightyPixel tracker and the Upstream tracker".

[29] Andreas N Philippou, Costas Georghiou, and George N Philippou. "A generalized geometric distribution and some of its properties". In: *Statistics Probability Letters* 1.4 (1983), pp. 171–175. ISSN: 0167-7152. DOI: `https://doi.org/10.1016/0167-7152(83)90025-1`. URL: `https://www.sciencedirect.com/science/article/pii/0167715283900251`.

[30] Marc Weber. *Elektronik für Physiker - Vorlesungsskript*. Karlsruher Institut für Technologie, 2016.

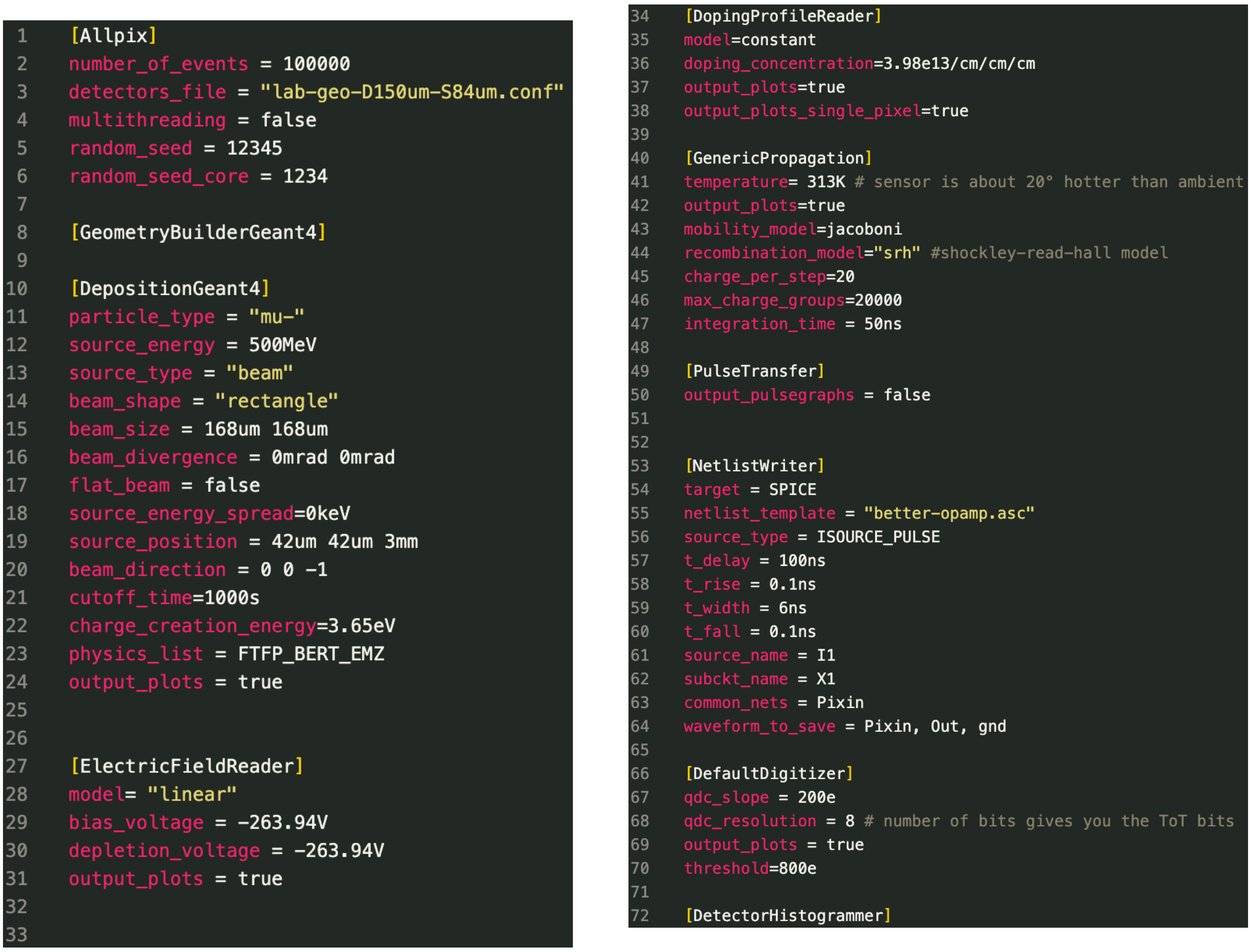
[31] Heiko Augustin et al. "TelePix – A fast region of interest trigger and timing layer for the EUDET Telescopes". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1048 (2023), p. 167947. ISSN: 0168-9002. DOI: `https://doi.org/10.1016/j.nima.2022.167947`. URL: `https://www.sciencedirect.com/science/article/pii/S0168900222012396`.

[32] Ruben Kolb. "Disentangling the Diffusive Part of the Drift-Dominated Signal Generation in a High-Ohmic Run2021v2 HV-MAPS Prototype". MA thesis. Heidelberg University, 2023.

[33] Ruben Kolb. *Charge Collection Studies for HV-MAPS*. 2024. URL: `https://indico.cern.ch/event/1323113/contributions/5823773/`.

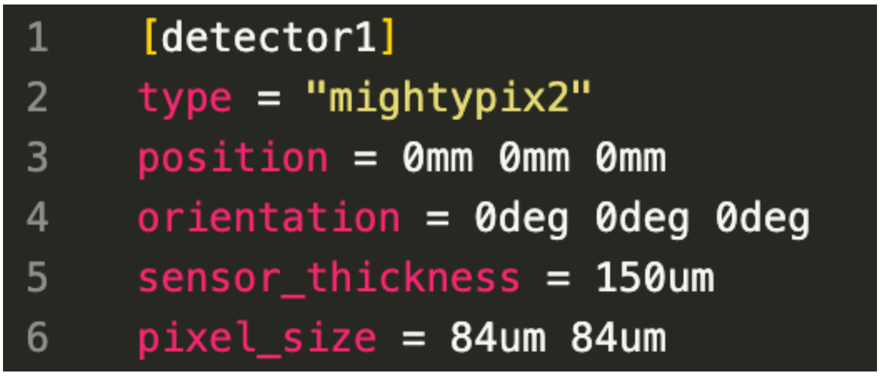
Appendix A

Record of Allpix^2 Simulation Parameters



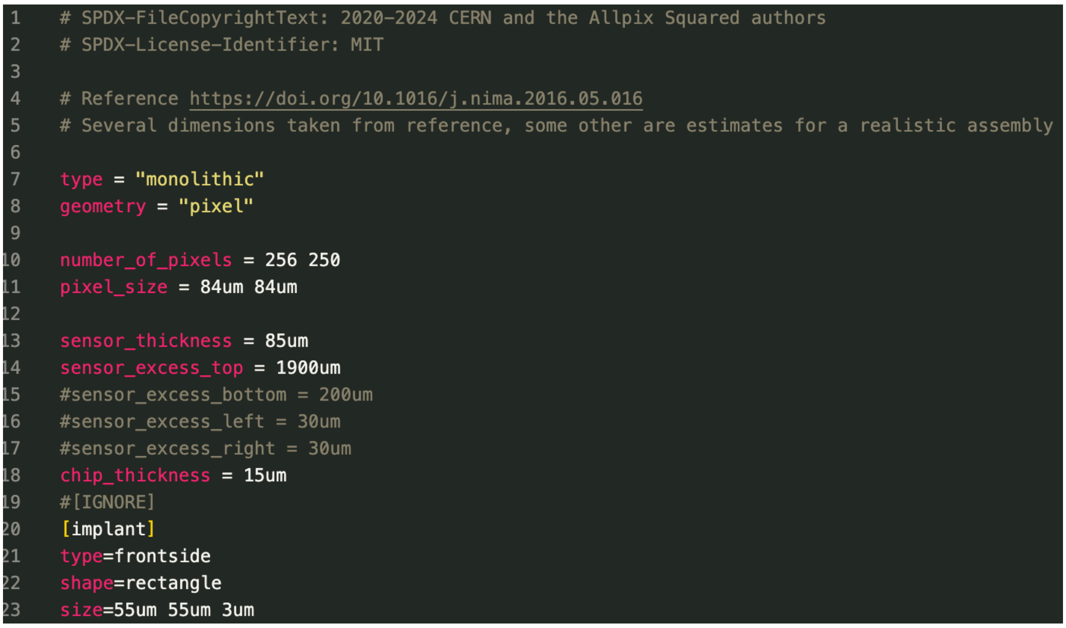
```
[Allpix]
number_of_events = 100000
detectors_file = "lab-geo-D150um-S84um.conf"
multithreading = false
random_seed = 12345
random_seed_core = 1234

[GeometryBuilderGeant4]

[DepositionGeant4]
particle_type = "mu-"
source_energy = 500MeV
source_type = "beam"
beam_shape = "rectangle"
beam_size = 168um 168um
beam_divergence = 0mrad 0mrad
flat_beam = false
source_energy_spread=0keV
source_position = 42um 42um 3mm
beam_direction = 0 0 -1
cutoff_time=1000s
charge_creation_energy=3.65eV
physics_list = FTFP_BERT_EMZ
output_plots = true


[ElectricFieldReader]
model= "linear"
bias_voltage = -263.94V
depletion_voltage = -263.94V
output_plots = true


[DopingProfileReader]
model=constant
doping_concentration=3.98e13/cm/cm/cm
output_plots=true
output_plots_single_pixel=true

[GenericPropagation]
temperature= 313K # sensor is about 20° hotter than ambient
output_plots=true
mobility_model=jacoboni
recombination_model="srh" #shockley-read-hall model
charge_per_step=20
max_charge_groups=20000
integration_time = 50ns

[PulseTransfer]
output_pulsegraphs = false


[NetlistWriter]
target = SPICE
netlist_template = "better-opamp.asc"
source_type = ISOURCE_PULSE
t_delay = 100ns
t_rise = 0.1ns
t_width = 6ns
t_fall = 0.1ns
source_name = I1
subckt_name = X1
common_nets = Pixin
waveform_to_save = Pixin, Out, gnd

[DefaultDigitizer]
qdc_slope = 200e
qdc_resolution = 8 # number of bits gives you the ToT bits
output_plots = true
threshold=800e

[DetectorHistogrammer]
```

Figure A.1: Full contents of the main Allpix^2 configuration file.



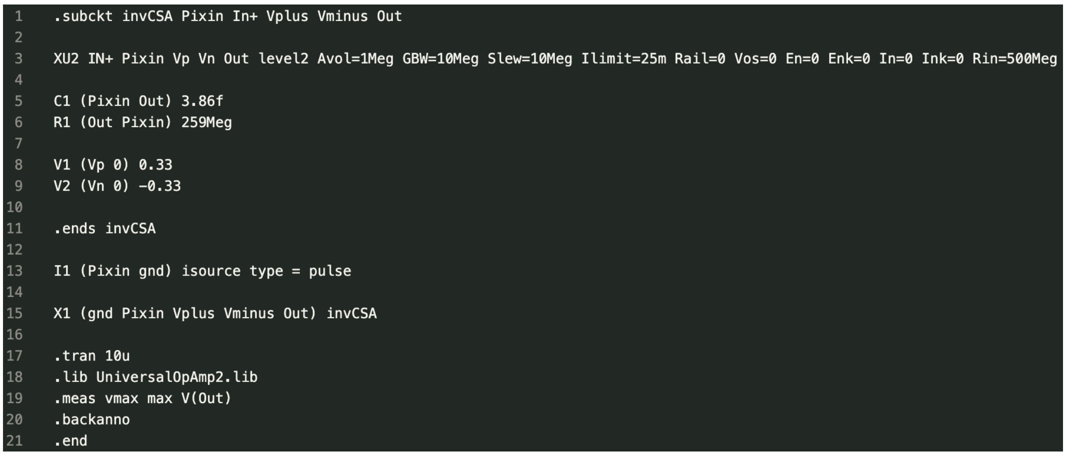
```
[detector1]
type = "mightypix2"
position = 0mm 0mm 0mm
orientation = 0deg 0deg 0deg
sensor_thickness = 150um
pixel_size = 84um 84um
```

Figure A.2: Lab geometry configuration file.



```
# SPDX-FileCopyrightText: 2020-2024 CERN and the Allpix Squared authors
# SPDX-License-Identifier: MIT

# Reference https://doi.org/10.1016/j.nima.2016.05.016
# Several dimensions taken from reference, some other are estimates for a realistic assembly

type = "monolithic"
geometry = "pixel"

number_of_pixels = 256 250
pixel_size = 84um 84um

sensor_thickness = 85um
sensor_excess_top = 1900um
#sensor_excess_bottom = 200um
#sensor_excess_left = 30um
#sensor_excess_right = 30um
chip_thickness = 15um
#[IGNORE]
[implant]
type=frontside
shape=rectangle
size=55um 55um 3um
```

Figure A.3: MightyPix2 sensor configuration file



```
.subckt invCSA Pixin In+ Vplus Vminus Out

XU2 IN+ Pixin Vp Vn Out level2 Avol=1Meg GBW=10Meg Slew=10Meg Ilimit=25m Rail=0 Vos=0 En=0 Enk=0 In=0 Ink=0 Rin=500Meg

C1 (Pixin Out) 3.86f
R1 (Out Pixin) 259Meg

V1 (Vp 0) 0.33
V2 (Vn 0) -0.33

.ends invCSA

I1 (Pixin gnd) isource type = pulse

X1 (gnd Pixin Vplus Vminus Out) invCSA

.tran 10u
.lib UniversalOpAmp2.lib
.meas vmax max V(Out)
.backanno
.end
```

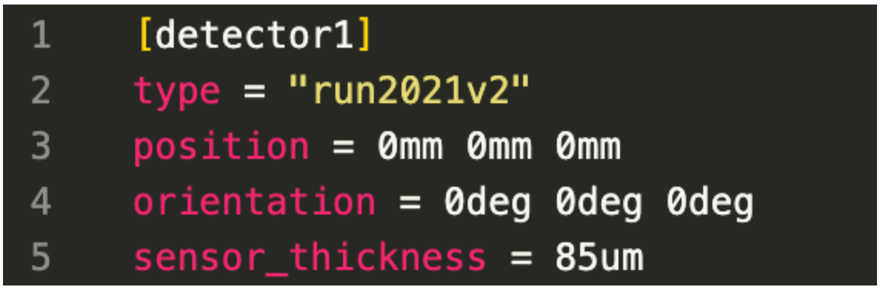
Figure A.4: Netlist template for use by the NetlistWriter.

Appendix B

Supplementary TCAD Simulation Parameters

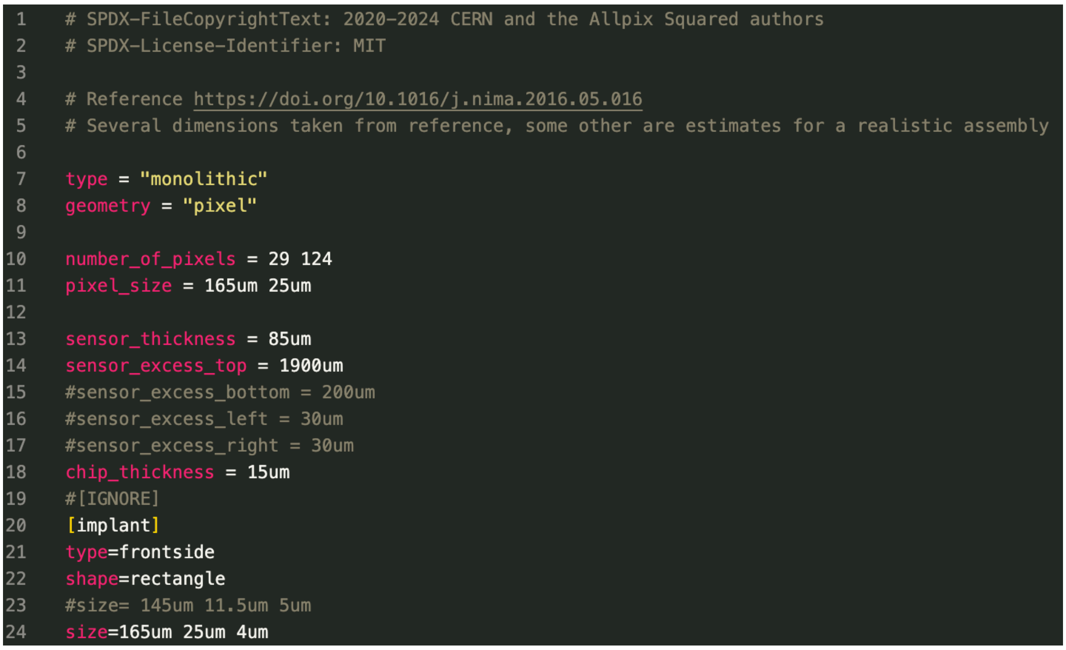
```
[Allpix]
number_of_events = 25
detectors_file = "lab-geo_telepix.conf"
multithreading = false
random_seed = 12345
random_seed_core = 1234


[GeometryBuilderGeant4]

[DepositionGeant4]
particle_type = "mu-"
source_energy = 500MeV
source_type = "beam"
beam_shape = "rectangle"
beam_size = 6mm 8mm
beam_divergence = 0mrad 0mrad
source_energy_spread=0keV
source_position = 0 0 3mm
beam_direction = 0 0 -1
cutoff_time=1000s
charge_creation_energy=3.65eV
physics_list = FTFP_BERT_EMZ
output_plots = true


[ElectricFieldReader]
model= "mesh"
file_name="TelePix_80V_ElectricField.apf"
field_mapping= PIXEL_FULL
field_scale = 1 1
output_plots = true


[DopingProfileReader]
model=constant
doping_concentration=3.98e13/cm/cm/cm
output_plots=true
output_plots_single_pixel=true

[WeightingPotentialReader]
model= "mesh"
file_name= "model_weightingpotential.apf"
field_mapping= PIXEL_FULL
output_plots= true


[TransientPropagation]
temperature= 313K
output_plots=true
output_linegraphs=true
mobility_model=jacoboni
recombination_model="srh"
charge_per_step=20
max_charge_groups=20000
integration_time = 50ns


[SimpleTransfer]
output_plots=true

[NetlistWriter]
target = SPICE
netlist_template = "better-opamp.asc"
source_type = ISOURCE_PULSE
t_delay = 100ns
t_rise = 0.1ns
t_width = 6ns
t_fall = 0.1ns
source_name = I1
subckt_name = X1
common_nets = Pixin
waveform_to_save = Pixin, Out, gnd

[DetectorHistogrammer]
```

Figure B.1: Full contents of the main configuration file for the TCAD-aided simulation mentioned in chapter 4.



```
[detector1]
type = "run2021v2"
position = 0mm 0mm 0mm
orientation = 0deg 0deg 0deg
sensor_thickness = 85um
```

Figure B.2: Lab geometry configuration file for the TCAD-aided simulation.



```
# SPDX-FileCopyrightText: 2020-2024 CERN and the Allpix Squared authors
# SPDX-License-Identifier: MIT

# Reference https://doi.org/10.1016/j.nima.2016.05.016
# Several dimensions taken from reference, some other are estimates for a realistic assembly

type = "monolithic"
geometry = "pixel"

number_of_pixels = 29 124
pixel_size = 165um 25um

sensor_thickness = 85um
sensor_excess_top = 1900um
#sensor_excess_bottom = 200um
#sensor_excess_left = 30um
#sensor_excess_right = 30um
chip_thickness = 15um
#[IGNORE]
[implant]
type=frontside
shape=rectangle
#size= 145um 11.5um 5um
size=165um 25um 4um
```

Figure B.3: Run2021v2 sensor configuration file, used for the TCAD-aided simulation.

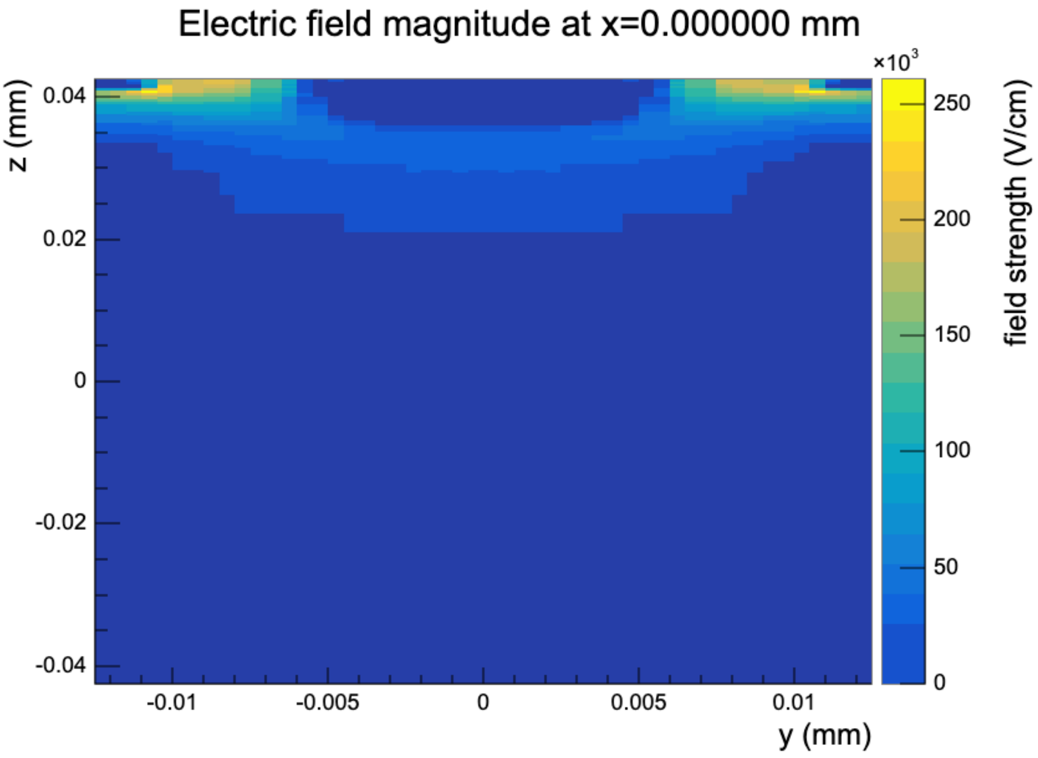


Figure B.4: Electric field map based on a TCAD model for 80 V bias voltage applied to a Run2021v2 pixel.