Development of an FPGA-based DAQ system and power integrity analysis of the pixel sensor for the upcoming Belle II vertex upgrade

Tobias Blesgen

Master-Thesis in Physics made in
Physikalisches Institut

submitted to the
Mathematisch-Naturwissenschaftliche Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

September 2025

I hereby declare that the work presented here was formulated by myself and that no sources or tools other than those cited were used.

Bonn, 07.09.2025 .....
Date

.....
Signature

1. Supervisor: Prof. Dr. Jochen Dingfelder
2. Supervisor: Prof. Dr. Florian Bernlochner

Contents

1 Introduction 1

2 Project overview 3
- 2.1 Particle physics . . . 3
- 2.2 SuperKEKB . . . 3
- 2.3 Belle II detector . . . 4
- 2.4 VTX Upgrade . . . 6
- 2.5 OBELIX sensor . . . 7
  - 2.5.1 Silicon detectors . . . 7
  - 2.5.2 Pixel matrix . . . 9
  - 2.5.3 Digital periphery . . . 10

3 DAQ system 11
- 3.1 Development . . . 12
  - 3.1.1 Cocotb . . . 13
  - 3.1.2 Debugging using waveform viewers . . . 13
- 3.2 Hardware . . . 13
- 3.3 Firmware . . . 14
  - 3.3.1 Basil . . . 14
  - 3.3.2 Command Unit (CMD) . . . 15
  - 3.3.3 Receiving Unit (RX) . . . 16
- 3.4 Software . . . 18
- 3.5 Cocotb tests . . . 19

4 Power and rail analysis 23
- 4.1 Digital IC implementation methodology . . . 23
- 4.2 Methods and Motivations . . . 24
  - 4.2.1 Power analysis . . . 24
  - 4.2.2 *IR* drop . . . 28
  - 4.2.3 Electromigration . . . 29
- 4.3 OBELIX-1 design . . . 30
- 4.4 Results . . . 33
  - 4.4.1 Power analysis . . . 33
  - 4.4.2 Rail analysis . . . 40

5 Conclusion 47

List of Figures 51

List of Tables 53

CHAPTER 1

Introduction

Sensor development for particle physics experiments is a complex process that spans multiple years. The associated work does not end once the sensor is designed, but involves detailed computer aided analysis of the design and the preparation of test systems. The development of the Optimized Belle II pixel sensor (OBELIX) is a collaborative effort of multiple institutes to create a next-generation sensor for the particle physics experiment Belle II in Japan.

This thesis focuses on simulation-based work for the OBELIX sensor during the critical transition phase from the virtual design work to the physical hardware. OBELIX is designed to be integrated into the Belle II experiment as part of a planned detector upgrade – the VTX upgrade. Chapter 2 introduces the current Belle II detector, the upgrade plans and the OBELIX sensor itself.

OBELIX-1 is the first revision of the OBELIX sensor. To test its capabilities and identify issues to be improved upon in the final sensor iteration, OBELIX-1 will undergo extensive lab testing and test beam campaigns. The control and readout of the sensor will be realized by an FPGA-based data acquisition system (DAQ). In Chapter 3 the development process of the software and FPGA firmware is discussed. Together with the codebase verification testbenches are introduced and their output displayed.

Fast, dense transistor logic, as implemented in the digital periphery of the OBELIX-1 sensor, generates heat due to its power consumption. Related large currents flowing through on-chip supply nets can – when unchecked – lead to large voltage drops, while high current densities in wires are prone to producing failures over time. Investigations of all of these issues are crucial steps before chip production to ensure long living, functional sensor designs. Dedicated power and rail analyses of the OBELIX-1 digital periphery are performed in Chapter 4 covering the different types of analysis.

The final chapter Chapter 5 highlights the results presented in this thesis and gives an outlook on future developments.

CHAPTER 2

Project overview

This chapter aims to introducing the reader to the underlying motivation of capturing collider data at the Belle II experiment and to provide an understanding of the environment for which the OBELIX sensor is being developed. Section 2.1 is an short introduction of the current state of particle physics. The SuperKEKB accelerator and the Belle II detector are described in Section 2.2 and Section 2.3. A overview of the planned VTX Upgrade of Belle II is given in Section 2.4. Following this overview, the OBELIX sensor is introduced in Section 2.5.

2.1 Particle physics

In the field of particle physics, scientists investigate the fundamental forces and particles that make up the universe. At its core lies the Standard Model of particle physics, describing the interactions and properties of the known elementary particles. It has withstood many tests and successfully predicted the Higgs boson [1], but there are indications of physics beyond the Standard Model. There are countless proposed theories describing new physics (a few are listed in [2]). In order to acquire new insights and further test the predictions of the standard model, the interactions of particles at high energies are studied at particle accelerator facilities all over the world. One accelerator in particular is of great interest for this thesis: SuperKEKB.

2.2 SuperKEKB

The SuperKEKB particle accelerator, located at the KEK research facility in Japan, is an asymmetric electron-positron double-ring collider (Fig. 2.1). Electrons and positrons are accelerated in opposite directions around a 3.016 km long ring to energies up to 7 GeV and 4 GeV respectively. It is designed to reach luminosities of up to $8 \times 10^{35}\,\mathrm{cm}^{-2}\mathrm{s}^{3}$ placing it at the luminosity frontier of research [3]. This means it is set up to probe parameters of the Standard Model, such as the CKM matrix parameters [4], and search for new physics in very rare events by making measurements with large statistics. The two particle beams intersect at a point along the so-called Tsukuba Straight where the Belle II experiment (described in detail in the next section) is located. SuperKEKB initial upgrade from its predecessor KEKB was commissioned in 2016, followed by work on the Belle II detector region. In April 2020 a record-breaking luminosity of $4.71 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ was achieved at SuperKEKB, which was followed

up by even higher luminosities in the years since. [5]

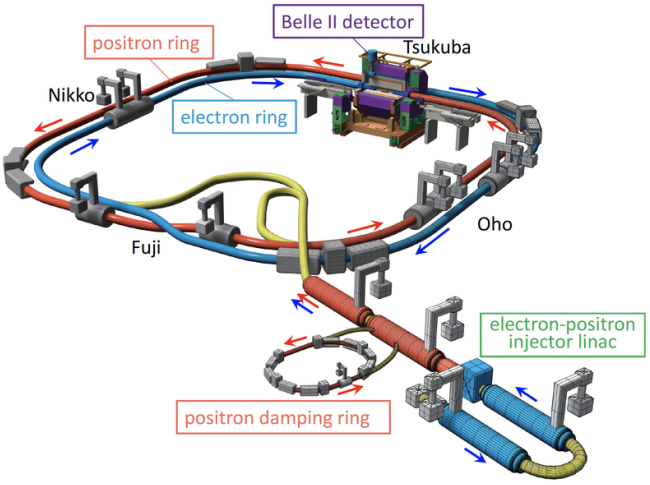


Figure 2.1: Schematic view of SuperKEKB showing the acceleration elements as well as the Belle II detector. The names of the four straights of the accelerator ring are marked in black. Taken from [3].

2.3 Belle II detector

SuperKEKB is referred to as a B-factory as the beam energies are tuned to a center-of-mass energy E_{cms} matching the $\Upsilon(4S)$ resonance. As this resonance corresponds to approximately twice the B-quark mass, large numbers of B flavoured mesons are expected to be created in the interactions. In order to observe and measure the properties of those mesons the general purpose particle detector, Belle II was built with a full solid angle coverage around the interaction point. Several sub-detectors optimized for different detection tasks are positioned in concentric onion-like layers to enable the detection of most of the created final state particles. Figure 2.2 displays the cross-section of the Belle II detector around the interaction point. The detector components shown are (information based on [6, 7]):

- The innermost detection system, the Vertex Detector (**VXD**), is designed to measure the position of charged particles close to the interaction point in six cylindrical layers. By matching captured hit locations and reconstructing the particles' trajectory belonging to the same interaction, the decay vertices of the primary particles can be determined. In the current configuration the VXD consists of two pixel sensor layers and four silicon strip layers.

  The pixel sensors of the Pixel Detector (PXD) are designed to achieve high spatial resolution while keeping the material budget to a minimum. The DEPFET sensors meet this requirement by being thinned down to 75 µm, while featuring 400 pixels per square millimeter.

  At larger radii, the expected particle flux has decreased sufficiently to allow the usage of silicon strip sensors, without risking the loss of large amounts of data due to too high activity in the readout channels. The Silicon Vertex Detector (SVD) consists of double-sided microstrip modules with a timing resolution of less than 3 ns.

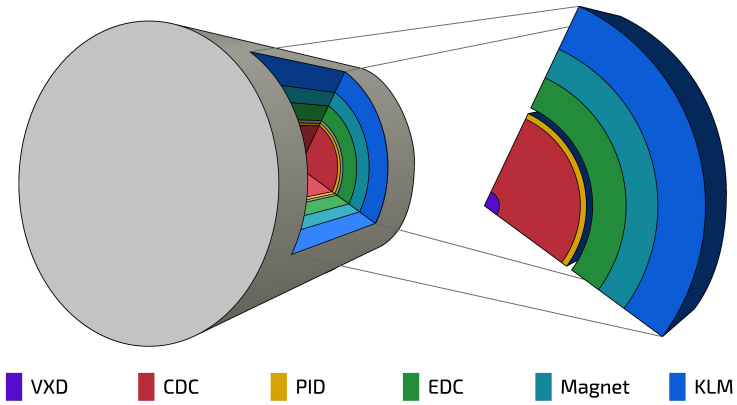


Figure 2.2: Color-coded cross-section of the central Belle II detector at the collision point.

While silicon detectors have good spatial resolution, gas-filled detectors are better suited to fill large detection volumes. The Central Drift Chamber (**CDC**) consists of $O(10^4)$ sense and field wires that capture the trajectories of charged particles in the gas-filled volume. Particles are deflected in the 1.5 T magnetic field generated by the **solenoid magnet**. By reconstructing the curved track, the particle momenta can be estimated based on the track curvatures.

The particle identification system (**PID**) was added to the detector to improve the separation between kaons and pions. It consists of two elements: a Time of Propagation Detector (TOP) in the barrel part (shown in Fig. 2.2) and an Aerogel Ring Imaging Cherenkov Detector (ARICH) in the forward end-cap. Both detectors identify particles based on the Cherenkov radiation emitted by the particles [8, chapter 11].

In contrast to the inner detectors which measure particles in a non-destructive way, the Electromagnetic Calorimeter (**ECL**) is designed to absorb the entire energy of a particle. Particles, such as electrons but also the electrically neutral photons, entering the dense CsI(Tl) crystals, interact with the material creating an electromagnetic shower. The remaining shower particles create scintillation light which is captured and digitized. The amount of detected light is proportional to the energy deposited by the initial particles in the detector.

While most particle types are absorbed in the ECL, there are two in particular which reach the outermost detector system. Those are the K_L and μ. To capture signals of these particles, denser material is used in the **KLM** (K_L and μ) detector. Active sensor material is stacked between thick iron plates to capture secondary particles created in interactions in the dense passive material. Scintillators are used as the active material to convert the particle energies to light measured by silicon photomultipliers, similar to the ECL.

Storing all data generated by the sensors in not feasible. To gather as much of data the scientists are primarily interested in as possible, while keeping the total recorded data rate as low as possible, a complex trigger system was created. Based on information collected in real time from multiple sub-detectors, the trigger system decides whether to store the event information (*trigger* the event).

The parameters dictating those decisions are fined-tuned on the signatures expected for event classes the collaboration is interested in. [9]

2.4 VTX Upgrade

To reach new luminosity goals, the focusing magnet system close to the interaction point will need to undergo changes during the upcoming Long Shutdown 2 (LS2). This upgrade was initially planned for 2027, but has now been delayed until 2032 [10]. To adapt the innermost elements of the Belle II detector to the new beam line geometry, a new fully pixelated detector system called VTX is planned to replace the current Vertex Detector. The new system targets to improve performance even in increased background environments due to luminosity increases and reduce occupancy by featuring smaller pixels. Requirements include a spatial resolution of $< 15\,\mu$m and a timestamp resolution of at least 25 ns to 100 ns. The new detector is required to withstand a total ionizing dose of 10 kGy/year and non-ionizing energy losses of around 5×10^{12} n_{eq}/cm^2/year. [11]

VTX is proposed to consists of five layers of depleted monolithic active pixel sensor (DMAPS) modules based on CMOS technology, arranged in straight cylindrical layers around the beam pipe (see Fig. 2.3), replacing both the PXD and SVD detectors. The sensor ladders are mounted with a small overlap so that all angles are covered with active sensor material.

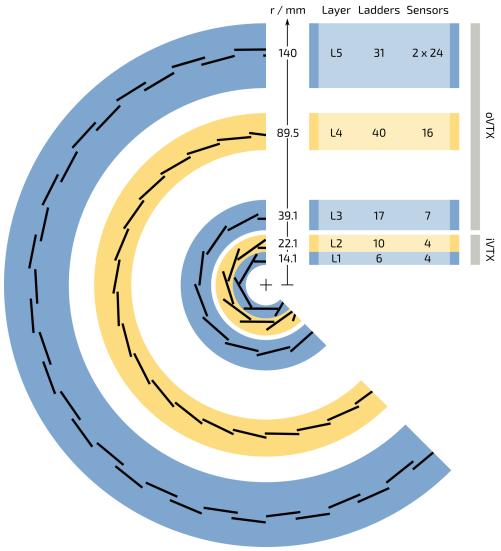


Figure 2.3: Sketch of the proposed five layers of the VTX, together with information about grouping in iVTX & oVTX, spacing and layer structure. Based on [11].

A special feature of the inner VTX layers (called iVTX) is the proposed mounting strategy. To further reduce the material budget, the iVTX avoids using a conventional support structure. Instead, four sensors, produced back-to-back that make up the iVTX modules, exclusively held only at the module edges. In the outer VTX layers (oVTX), sensors are diced individually and placed on a more traditional support structure, allowing longer ladders to be formed.

2.5 OBELIX sensor

The Optimized Belle II Pixel (OBELIX) sensor is a monolithic semiconductor pixel sensor developed for the VTX upgrade. It is based on its well-tested predecessor TJ-Monopix2 [12]. OBELIX inherits the 33.04 µm square pixel design from TJ-Monopix2, while increasing the matrix size to an area of $29.60 \times 15.33\,\mathrm{mm}^2$ and adapting its periphery to fit the needs of the VTX upgrade. The first revision of OBELIX, called OBELIX-1, is the main focus of this thesis. A second revision with bug fixes and small changes is planned to be developed based on results of extensive testing of OBELIX-1 and is expected to be the final version used in the Belle II upgrade.

The area of the chip can be divided into two regions: the pixel matrix and the periphery located along one of the long matrix edges (see Fig. 2.4). Section 2.5.1 gives an overview of silicon sensors, introducing the main working principles necessary to understand the pixel matrix displayed in Section 2.5.2. The digital logic located in the periphery (indicated in yellow in Fig. 2.4) is explained in Section 2.5.3.

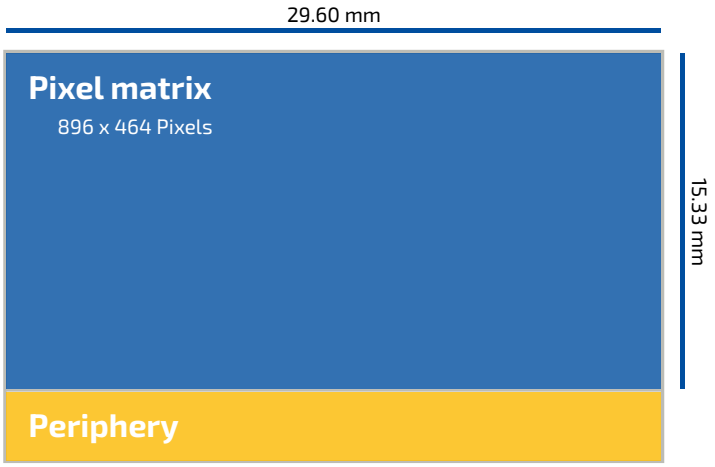


Figure 2.4: Simplified partitioning of the OBELIX floorplan into pixel matrix and periphery. Values are based on [13].

2.5.1 Silicon detectors

The detectors constructed to measure subatomic particles have evolved drastically, from bubble chambers and photographic plates, through the first wire chambers, to today's highly complex optimized detector systems able to measure single particles with high precision. While gas-filled detectors are still a crucial part of many detectors like Belle II, silicon-based sensors have gained great popularity.

In contrast to conductive materials, the electrons of semiconductors require a certain amount of energy to move freely within the material. This gap energy between the valence band and conduction band (visualized in Fig. 2.5) depends on the material. For semiconductors like silicon, the band gap is on the order of a few electronvolts while the energy difference between the bands of insulators is usually much larger.

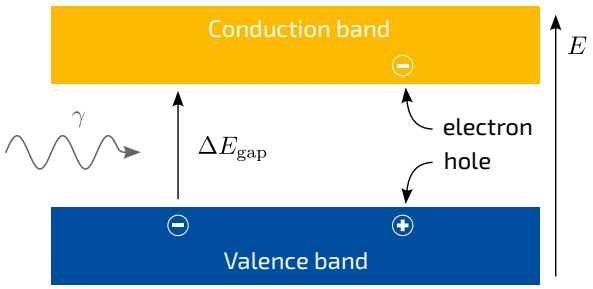


Figure 2.5: Semiconductor energy band structure, showing the excitation of an electron into the conduction band, leaving a positively charged hole in the valence band

Properties of semiconductor materials can be modified by introducing elements with one more (n-doped) or one less (p-doped) valence electron. At the edges between doped areas the charge carriers (electrons from n-doped material and holes from p-doped material) diffuse into the opposite region and recombine (shown in Fig. 2.6 (a)). The remaining ions in this area act as fixed space charges, creating an electric field counteracting the diffusion (Fig. 2.6 (b)). The region free of mobile charge carriers around the doping boundary is called the depletion region. By applying an external reversed bias voltage to the junction the depletion width can be widened.

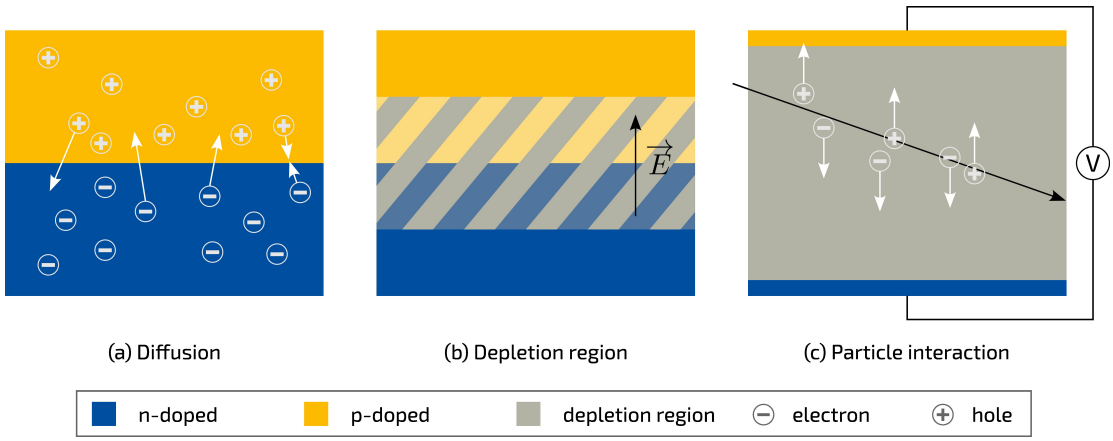


Figure 2.6: Working principle of a pn-junction, showing the initial diffusion of charge carriers (a), the depletion region and resulting electric field (b). In (c), the depletion width is increased by applying an external voltage. The motion of electrons and holes generated by a charged particle is shown as well.

Charged particles traveling through the depletion region create electron-hole pairs along their tracks (displayed in Fig. 2.6 (c)). The electrons and holes drift along the electric field towards the respective ends of the pn-junction. A charge signal induced by the movement of these charges can be measured by a readout electrode at one end of the junction. Spatial information can be obtained by segmenting the electrodes.

External readout electronics can be used to process the signals (hybrid approach), but this comes at the price of more connections and a second silicon chip, increasing the material budget in the particle's path. An alternative is the so-called monolithic approach, where the readout is implemented on the same silicon wafer as the sensor. Fig. 2.7 displays the cross-section of a depleted monolithic active

pixel sensor (DMAPS). The CMOS transistors (mixture of n-based and p-based metal oxide silicon transistors) are located at the surface of the chip. Implant regions, called p- and n-wells, are used as the substrate of the transistors and the charge collection electrode. Since a direct contact between the sensor's p-substrate and PMOS n-well would act as a second collection well, deep p-wells are introduced to shield the n-wells. More detailed descriptions of different configurations of DMAPS and material details can be found in [8, 12, 14].

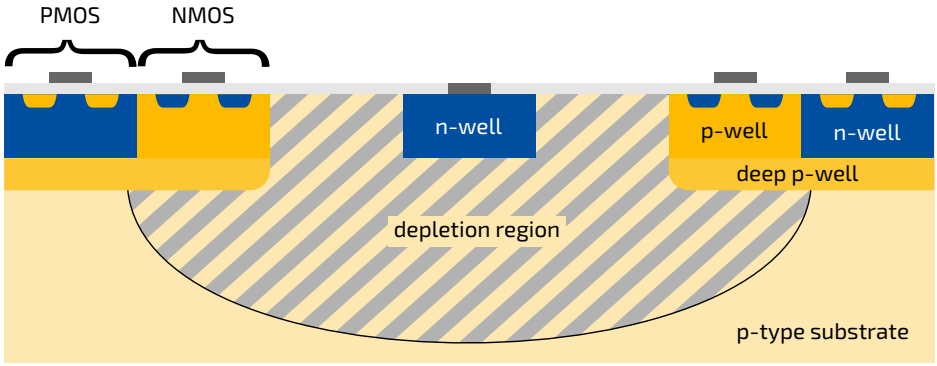


Figure 2.7: Sensor cross-section demonstrating a DMAPS design, showing a central depletion region around the collection n-well. CMOS technology is implemented at the surface, using deep p-wells to shield the n-well. This is not the OBELIX cross-section. Based on [12].

2.5.2 Pixel matrix

The pixel matrix implemented in OBELIX consists of 896×464 pixels. The CMOS logic implemented on pixel-level covers the first two stages of the readout chain: amplification and digitization, shown in Fig. 2.8. A voltage amplifier is used to amplify and shape the signal. In the discriminator stage, the amplifier output is compared to a tunable threshold voltage. The threshold consists of two parts: a global threshold value set for the entire matrix and an individual trim correction set for each pixel individually, implemented to compensate local deviations. The discriminator's output is set to one as long as the input voltage is larger than the threshold. The length of the active signal, called Time-over-Threshold (ToT), depends on the input signal amplitude and therefore is related to the energy deposited by the traversing particle. The leading and trailing edges of the ToT signal are sampled with a 21.2 MHz clock and propagated to the digital periphery via a synchronous column-drain readout architecture [12].

The matrix is laid-out in 896 columns, consisting of 464 pixels each. It is structured in double columns, sharing large parts of their CMOS logic as well as a common data output. Hit information sent to the digital periphery includes the address of hits and their ToT edge timings. [12, 13]

The matrix features a charge injection circuit, which can be used to inject charges in selected pixels. An 8-bit DAC generates a voltage pulse that charges a capacitor in the pixel, connected to the readout circuit. This functionality is used for testing purposes as described in Section 3.1.1.

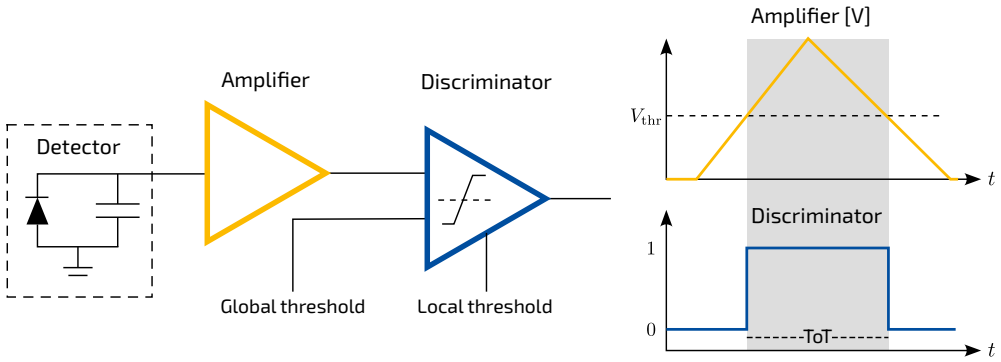


Figure 2.8: Readout chain implemented in the pixel consisting of an amplifier and discriminator. On the right, the generation of the ToT signal is illustrated. Based on [12].

2.5.3 Digital periphery

OBELIX's digital periphery consists of modules covering control, storage and communication tasks, among others.

The **Trigger Unit** (TRU) was introduced to OBELIX to add the ability to triggering events – which was not supported in TJ-Monopix2. The module is structured in multiple trigger groups, each assigned to receive the data of four double columns. The hit data of the double columns is merged and stored in an intermediate trigger memory. The data is only read out from memory if a trigger with a matching trigger ID is issued. OBELIX additionally features a triggerless mode, in which all captured hit data is transmitted. Data from all trigger groups is merged and sent to the Transmission Unit.

The outgoing communication is managed by the **Transmission Unit** (TXU). This includes the hit information coming from the TRU and register information requested by the user. Data blocks are decoded and sent to an LVDS (Low Voltage Differential Signaling) driver to be transmitted to the DAQ system. The driver transmits one bit at each clock edge of the 169.7 MHz clock (double data rate), thereby increasing the total data rate. Details of the data structure and further data handling can be found in Section 3.3.3.

The **Control Unit** (CRU) reads commands sent by the DAQ system and controls other parts of the chip by changing register values and sending control and configuration signals. In Section 3.3.2, the command protocol is introduced in the context of the Command Unit. The CRU hosts the Global Register Table, which contains hundreds of individually addressable configuration registers.

This subsection is based on the doctoral thesis of Maximilian Babeluk, which will be published soon.

CHAPTER 3

DAQ system

In order to connect the OBELIX chip to the outside world, it is integrated into a data acquisition system (DAQ system). All communication to and from the OBELIX sensor is controlled by the intermediate DAQ hardware. While the final system implemented in the Belle II detector will differ, a more flexible setup has to be designed for lab and test beam measurements. This test DAQ system is shown in the top part of Fig. 3.1. It consists of aq custom printed circuit board with passive components carrying the OBELIX sensor, the readout board hosting an FPGA module and the software on the user's computer.

Instructions sent by the user are transmitted from the computer to the readout board, where they are translated into commands and distributed to the OBELIX sensors via DisplayPort connections. In return, encoded hit information and register values are sent back to the FPGA, decoded and stored until requested by the user. The DAQ system can also be connected to an external Trigger Logic Unit (TLU) in order to forward trigger commands to the sensor.

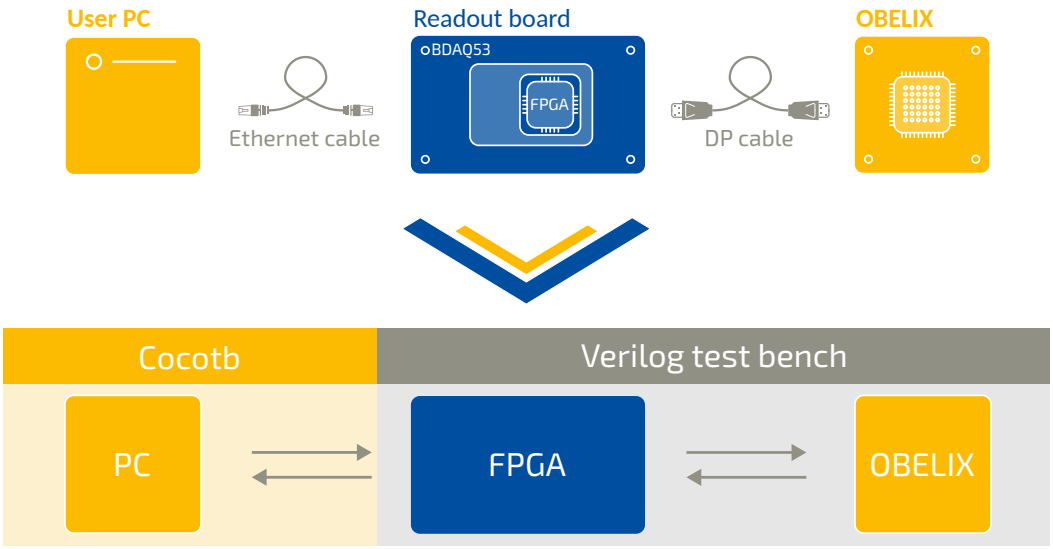


Figure 3.1: Comparison between the hardware setup (top) and the equivalent simulation setup (bottom).

This chapter focuses on the development of the FPGA-based test DAQ system. Section 3.1 gives

an overview of the complete development flow, introducing methods and software components used. The hardware deployed in the DAQ system is detailed in Section 3.2 focusing on the BDAQ53 board. The firmware implemented on the FPGA and the software controlling these modules are described in Section 3.3 and Section 3.4. Tests verifying the functionalities of the DAQ system are displayed in Section 3.5.

3.1 Development

The development of a sensor-specific DAQ system must start with the elements closest to the device under test (DUT), since otherwise testing of the functionalities would not be possible. In the case of the setup shown in Fig. 3.1, the communication between the DUT (OBELIX) and the FPGA (field programmable gate array) on the readout board must be implemented first.

To set up an FPGA a binary file dictating the configuration of lookup tables and interconnects is written onto the hardware. This binary file is generated from firmware code written in a hardware description language (HDL) such as Verilog. The task-specific firmware modules of the DAQ system are described in detail in Section 3.3 and cover communication with the sensor and the user's computer.

In addition to the firmware, Python-based software enabling the interaction between user and DAQ system is developed. This software consists of an interface for controlling the FPGA's modules as well as higher-level functions, reducing the quantity of code required in scan routines to interact with the system.

Due to the strong similarities between OBELIX and its predecessor chip, TJ-Monopix2, the latter's DAQ system serves as a starting point for the new system. This includes much of the TJ-Monopix2 DAQ firmware and software, enabling a greater focus on OBELIX-specific unique features rather than recreating standard components like communication protocols. Hardware-wise the same readout board – the BDAQ53 board designed for the RD53 sensors – will be used to connect to the FPGA.

An important element of developing a DAQ system for a given sensor, is testing the entire setup with that specific device. Usually, new features would be tested by flashing them onto the FPGA and running tests in hardware. however, since the OBELIX sensor does not yet exist, these tests must be performed in a simulation that closely resembles the intended setup. The translation from hardware into the simulation is displayed in Fig. 3.1: The firmware modules of OBELIX and the FPGA are instantiated in a Verilog testbench including all connections between the two devices. As the simulation of the analog domain of OBELIX (mainly the pixel matrix) would increase the run time significantly, simplified models are used to represent these elements. Even though the testbench can be simulated by most commercially available simulators, it is necessary to integrate the external communication from the user to the DAQ system using additional software. In this setup, the open-source framework cocotb was selected to perform this task, as described in Section 3.1.1.

Comparing the hardware and simulation setups in Fig. 3.1 illustrates the one-to-one correspondence between the hardware elements and their virtual counterparts. As the data flow remains the same from the perspective of the DAQ system, verification of the different functionalities enables a transition from the simulated to the hardware setup without additional development steps.

3.1.1 Cocotb

Errors in HDL code can be easily overlooked due to the complexity within and between modules. To ensure that the HDL code matches its intended behavior, a process called functional verification is performed. During this stage, key features are tested using dedicated test cases. While a majority of these tests are nowadays written in SystemVerilog, alternative approaches can make it easier for new developers to get into and add other language-depending benefits [15]. One alternative approach is based on the open source framework cocotb (Coroutine-based cosimulation testbench) [16]. Cocotb runs asynchronous to the simulation of the Verilog testbench, steering signals and accessing data. The user specifies the desired behavior by writing a simple python-based testbench, calling asynchronous functions that e.g. control clocks or provide stimulation of the DUTs inputs, while also being able to access output signals. Towards the end of the simulation, assertions are used to compare the observed outputs to expected values and thus determine whether the DUT fulfills the set goal.

The simple structure of these tests and the popularity of Python allows users to easily write tests to cover all functionalities important for the final chip. Once the testbenches are written, they can be used in continuous integration (CI) to be run after each submitted change in either software or firmware to verify that the applied changes did not break any of the main functionalities.

3.1.2 Debugging using waveform viewers

While cocotb testbenches can be used to verify a working design, they are not well suited for debugging. When encountering unexpected behavior, developers can visualize the simulated signals using a waveform viewer. The tool allows the designer to study the state of each wire and register at any given point in time. This amount of insight is big advantage with respect to the final testing in hardware, where only selected signals connected to the outside world are available.

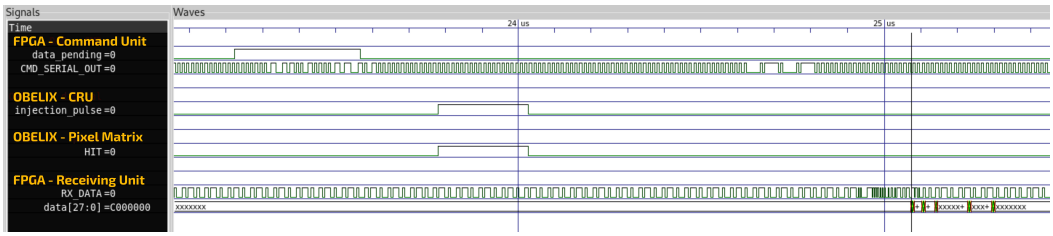


Figure 3.2: Signals from different parts of the setup plotted with respect to the simulation time. The waveforms show an injection commands issued on the FPGA and interpreted by the OBELIX's digital periphery, a hit injection in the matrix and the corresponding data received on the FPGA.

The applications reads in an activity file, produced by the simulator. Users select the signals they are interested in and display them with respect to the time (shown in Fig. 3.2). The signal names are same as declared in the HDL code. By following the signals in the firmware in parallel to the waveforms, users are able to locate bugs and understand complex interactions.

3.2 Hardware

As shown in Fig. 3.1 the DAQ setup consists of the OBELIX sensor (introduced in Section 2.5), the readout board hosting the FPGA and a computer connected via cables. The readout board acting as

the centerpiece of the DAQ system is a custom printed circuit board (PCB) shown in Fig. 3.3. It was developed in Bonn as part of the BDAQ53 project [17]. Since its initial design for the RD53 sensors, it has been used as a versatile readout board for different DUTs such as the TJ-Monopix2.

A commercially available FPGA daughter board (the Mercury + KX2 module [18]) is situated in the center of the readout board. It features a Xilinx Kintex 7 FPGA and DRAM while also providing a clock signal used on the FPGA. The FPGA firmware is flashed via a JTAG connection. Up to five DisplayPort connections are used to interact with the devices under test, while an Ethernet cable connects the readout board to the user's computer. Power and input/output signals are distributed between the different on-board elements and outgoing connections across the four layers of the PCB. A programmable clock generator provides a 169.7 MHz clock signal that is used both for the communication between the readout board and sensor as well as reference for the DUT's clocks.

The readout board receives continuous updates, both on the firmware-wise side (e.g. support of larger data bandwidth) and on the hardware side (added features and even complete redesigns).

Figure 3.3: BDAQ53 readout board hosting the FPGA daughter board. Picture by Rasmus Partzsch.

3.3 Firmware

All interactions between the user and the OBELIX chip are handled by the DAQ system. The primary tasks are reading and writing registers as well as transmitting measured hit information. To implement these functionalities in a structured way, the DAQ system is subdivided into individual modules, designed to handle a specific task. In this section, the modules are presented and their functionalities are explained.

3.3.1 Basil

The DAQ system is designed around the open-source Basil framework [19]. All FPGA modules are connected to one central bus system. A unique 32-bit address is associated with each module, allowing the software on the user's computer to address each module individually. Each module's register is assigned a sequential index. Adding a register's index to the module's address selects the register

directly. A 32-bit-wide data bus enables the reading from and writing to the selected register. This setup makes the isolated development of individual modules possible.

During the design process of the FPGA modules, corresponding Python classes are developed in parallel to the firmware code. They include all necessary functions to interact with modules via the Basil bus system. The Python software is covered in Section 3.4.

3.3.2 Command Unit (CMD)

Control commands are transmitted from the FPGA to the sensor via an LVDS connection. The Command Unit (CMD) drives the output signal based on user instructions and trigger inputs.

The command protocol is equivalent to the one used by RD53B [20], where commands consist of multiple 8-bit symbols combined to 16-bit frames. All commands are chosen such that frames consist of an equal number of zeros and ones. This overall DC-balanced signal is an important property to ensure proper functionality of the differential signal drivers and receivers at large frequencies [21]. Frames are serialized and sent to the OBELIX chip synchronized to a 160 MHz command clock to be interpreted by the chip. The command protocol is designed with multichip operation in mind. Commands affecting all connected sensors, such as trigger and synchronization commands, are broadcasted to all DUTs. These commands are one frame long, which is important for the triggering to allow the triggering of consecutive hits. Commands meant to address single sensors include a chip ID, ensuring that the DUT's command decoder only reacts to those with a matching ID. Those directed commands consists of up to four frames and are used for configuration, local resets and injection tasks.

The Python framework provides abstract functions for each command, reducing the amount of code needed to interact with the DAQ system. The commands are written to registers of the Command Unit via the Basil bus. A state machine selects commands to be serialized and transmitted based on their priority.

Trigger Commands

In high-energy particle physics experiments, the number of measured signals in detector elements is usually too large to store every bit of information obtained by the sensors. Fast trigger circuits decide, based on a subset of event information, whether an event is read-out. A trigger command is issued for the detectors to transmit data corresponding to the triggered point in time.

Figure 3.4: Trigger delay matching illustrated in multiple steps. First the hit is delayed on OBELIX, than a small delay on the DAQ side is added to match the timing

External trigger signals will by nature arrive on the sensor with a certain delay relative to the hit. Hits on the OBELIX sensor are delayed in the Trigger Unit to match the the timing of the trigger signal. The DAQ system can be used to fine tune this delay matching by adding an addition small delay as shown in Fig. 3.4. As the delays depend on hardware setup, the delay length on the chips will need to be adjusted once a new setup is constructed.

The DAQ System allows for two types of trigger sources: external triggers and internal injection triggers. In a test setup, a trigger logic unit (TLU) issues trigger signals to multiple components.

This external trigger is received via a DisplayPort connection on the BDAQ board and is the default trigger mode. In order to test an OBELIX sensor using only the DAQ system, the CMD unit can create additional triggers following injection commands. Those triggers (called internal triggers) are only issued if the user activated the internal trigger mode on the DAQ.

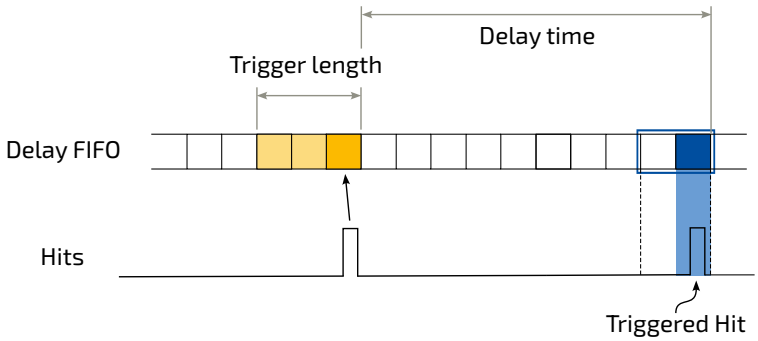


Figure 3.5: Trigger sequence showing trigger entries in the delay FIFO corresponding to hits in OBELIX. A trigger corresponding to the newest hit is stored in the FIFO (marked in yellow) with a certain trigger length set by the user. A hit arrives at the end of the OBELIX logic matching to the (blue) delayed trigger entry to be triggered successfully.

The trigger signal is checked each clock cycle, matching the period of one command symbol. Once a trigger of a hit arrives, it is delayed using as simple one-bit wide FIFO, such that the arrival of the corresponding trigger command on OBELIX matches the clock period when the hit signal arrives to the trigger unit in OBELIX. Figure 3.5 displays the entries of FIFO being delayed to send a trigger command matching their corresponding hit (marked in blue). A widened trigger window can be used to read out hits of neighboring clock periods. This can be used to deal with trigger signals arriving close to the clock edge. By increasing the `TRIGGER_LENGTH` parameter the user can adjust the number of consecutive triggers added to the delay FIFO, thus widening the triggered time window.

Trigger commands take two BCID clock periods. Thus, they have to specify whether they are meant to trigger the first, second or both BCIDs corresponding to the command. Trigger commands provided by the RD53 command protocol allow for 15 different so-called trigger patterns. In the case of OBELIX only three are needed to cover all possible trigger scenarios. Once triggers reach the last two FIFO elements, they are translated into a matching trigger pattern. At the same time a trigger flag is set to notify the state machine. Trigger commands always get priority over user induced commands, ensuring correct trigger timing.

3.3.3 Receiving Unit (RX)

All incoming data are handled by the Receiving Unit (RX). Data transmitted by the OBELIX chip to the DAQ system includes information about hits (referred to as *hit words*) as well as the content of requested registers (*register words*).

Hit words consist of 40 bits. They contain the timing information of the leading and trailing edges of the discriminator signal (introduced in Section 2.5.2) as well as the pixel address separated in column and row number.

hit word – 40 bit, 5 blocks				
leading edge	trailing edge	column	row	filler
39 ... 31	30 ... 24	23 ... 14	13 ... 5	4 ... 0

Timing information of the edges is given in units of 47.1 ns [13]. Due to the maximum ToT length expected in OBELIX the first two bits of the trailing edge can be omitted, since they can be inferred from the leading edge timing. This results in different lengths of the timing information.

Hit addresses consist of a 9-bit row and a 10-bit column number indicating the position of the hit. The column address is constructed in multiple stages during the read out, which reflects the matrix structure (shown in Fig. 3.6). Concatenating the trigger group number, the position of the double column within the group and the indicator for left/right column, results in 10-bit column values that match a straight forward numbering from left to right. Thus, there is no need for any sort of mapping in software. Since the data is sent in block of 8 bit, five empty filler bits are added at the end of the hit word.

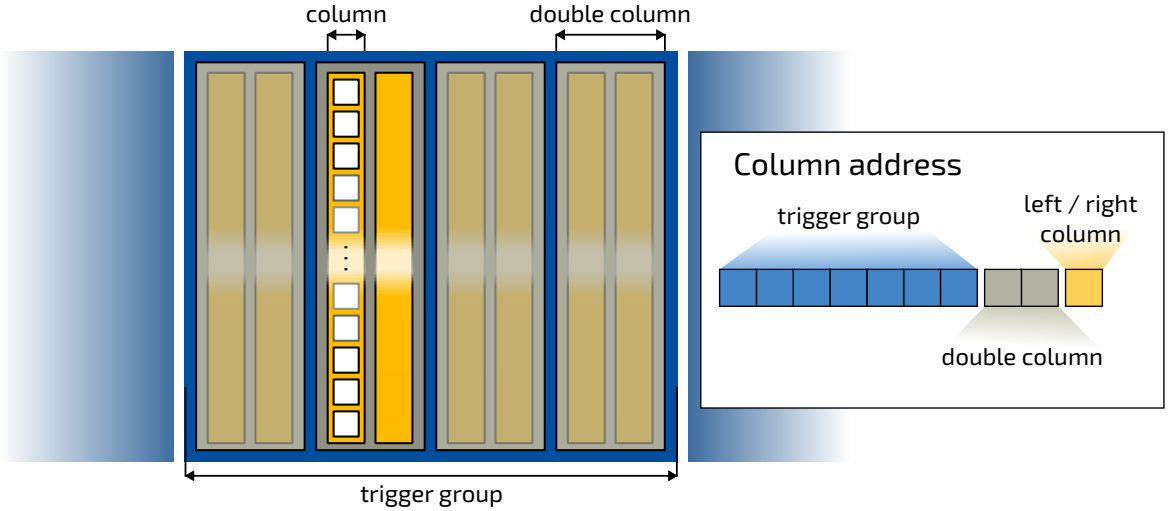


Figure 3.6: Explanation of the address structure showing the different components next to the matrix layout.

In contrast to hit words the data containing register values are only 32 bits long. They consist of the 16-bit register address and the 16-bit value of this register.

register word – 32 bit, 4 blocks	
register address	register value
31 ... 16	15 ... 0

Hit and register words are sent separately in so-called frames. Each frame can contain multiple words of the same type. The start of frame and end of frame are indicated by an 8-bit control pattern (k-words). Start of frames patterns differ for frames containing hit or register words. At large hit rates or when encountering clustered hit patterns it is likely that frames include a larger number of hit words. For transmission, these frames are divided into 8-bit blocks which are 8b/10b encoded. This ensures overall DC-balanced transmission similar to the command protocol.

Figure 3.7 shows a simplified view of the RX unit with data flowing from the left side (connected to OBELIX) towards the right. The incoming data comes from an LVDS receiver block. A deserializer

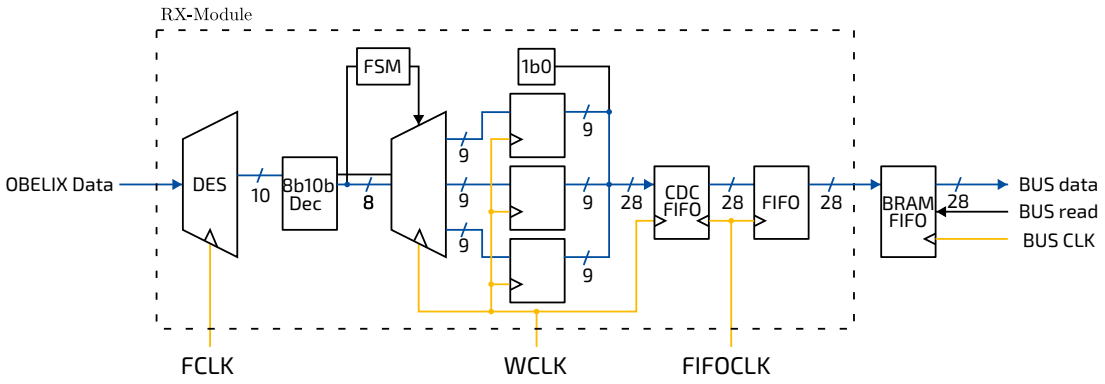


Figure 3.7: Structure of the RX module showing the deserializing, decoding and storing of the incoming OBELIX data. Clock signals are drawn in yellow and the main signal path in blue.

reconstructs the 10-bit data blocks from the serial bit stream by collecting the bits and reading them out at a slower clock frequency (1/10th of the fast input clock). After decoding the blocks, they consist of an 8-bit data word and a single k-bit used to identify k-words. To fully utilize the bus width, three consecutive data blocks (including their k-bits) are concatenated. This larger data group is forwarded into the bus clock domain for storage in the main BRAM FIFO. To associate data of different modules a 4-bit identifier is added. The data accumulates in the large BRAM FIFO until the user issues a matching readout request via the Basil bus.

3.4 Software

Users interact with the DAQ system via a Python interface, developed in parallel to the firmware. It not only contains functions to control individual elements of the FPGA modules, but also provides abstracted high-level functionalities. This level of abstraction allows to write short and easy-to-understand simulation and measurement routines. There is one Python file corresponding to each firmware module covering all local registers and fine-grained interactions. These module-specific functions are combined and abstracted into the higher level functions in one central `obelix.py` file. For example, changes to individual OBELIX registers can be made using a single high-level function call, requiring only the register name and target value. Low-level interactions with the command structures are only required during debugging or testing.

The software also provides the means to interpret received data frames. After retrieving the data from the BRAM FIFO the start-of-frame is identified, before extracting the hit or register values. There is additional logic in place to check for broken frame structures and alert the user about failed communication. At the end, the user receives one list containing all register values with corresponding address and one list with all received hit information ready for further analysis.

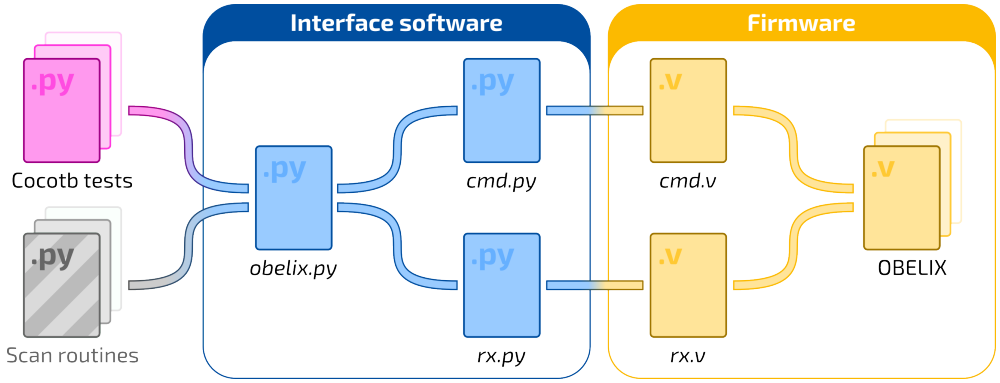


Figure 3.8: Symbolic interactions between the different files. cocotb tests and future scan routines interact with the chips via the software interface centered around the `OBELIX.py`. The module-specific Python files provide the lower-level interactions with the corresponding Verilog files of the DAQ. Additional files like `.yaml` configuration files, utility Python files and other elements of the firmware were omitted in this picture for clarity.

3.5 Cocotb tests

Before writing testbenches for any DUT, their main tasks need to be identified. For the DAQ system, these are reading and writing register data, receiving and interpreting hit frames and triggering hits. For each of these three areas, one cocotb testbench was written. The general structure is similar for all of them (as shown in Fig. 3.9).

Imports
```
import os, sys
from obelix1.tests import utils
```

```
def test_read():
```

Setup
```
    sim_thread, conf = utils.setup_cocotb()
    daq, chip = utils.init_devices(conf)
```

Test scenario
```
    test_reg = "CMD_CHSYNC_THR_HIGH_CONF"
    test_value = 3

    chip.write_command(chip.write_sync(write=False), repetitions=8)

    chip.registers[test_reg].write(test_value)
    value = chip.registers[test_reg].read()
```

Assertions
```
    assert value == test_value
```

Closing
```
    daq.close()
    utils.sim_end(sim_thread)
```

Figure 3.9: Stripped-down version of the read and write test, showing the general structure of a cocotb test. The substructures are color-coded.

First, the devices – DAQ system and OBELIX – are instantiated and configured. Before adding any test-specific code, the OBELIX sensor needs to be reset to ensure a well-known state of the chip. The actual functions implementing the interactions with the firmware modules are imported from the software introduced in Section 3.4 and combined in the primary test block to simulate the test scenario.

To validate the chip's behavior, data frames corresponding to the task at hand are recorded and then interpreted in software. Assertion statements compare output values with predefined expected values. The test is considered to be successful only if all assertions are met.

In order to interact with the chip at all, writing and reading of chip registers is crucial. The simple exchange implemented in this first test bench is an important baseline test for any code change, as any failure in this test would practically render the chip unusable. It played a central role in the development of both the CMD and RX units as well as the main Verilog test bench. A comparison of the signal waveforms and the initial Verilog code facilitated the addition of crucial features and the understanding of existing code of the TJ-Monopix2's DAQ system.

While communication is an important part from the system's perspective, the main purpose of the sensor is taking data. The injection test uses OBELIX's digital injection functionality to mimic hits in the matrix without actually having to do a mixed-signal simulation. After an initial configuration of the functional dummy modules that replace the analog pixel matrix, the user decides on a hit pattern to be injected. The injections are performed by selecting active rows and columns. Hit signals are generated in all pixels located on an active row and column at each injection pulse.

This method of encoding active pixels by enabling certain rows and columns can result in signals injected in additional unwanted locations. These so-called ghost hits are illustrated in Fig. 3.10, showing the intended hit pattern on the left. Pixels located in the same row and column as other intended hits will also inject a signal, as both their row and column are activated. In order to avoid this effect, complex patterns have to be injected in multiple steps. One possible strategy would be to iterate through the columns one-by-one, thus injecting from left to right without causing ghost hits.

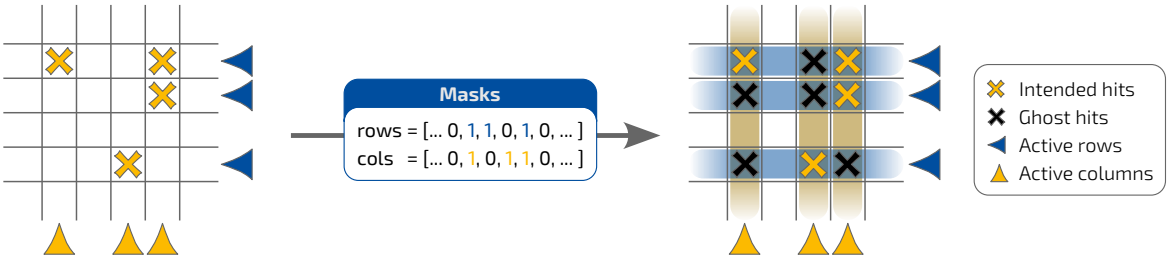
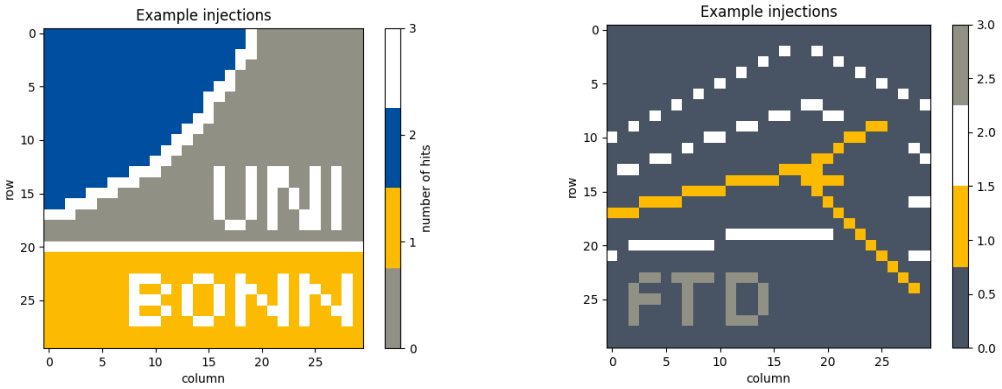


Figure 3.10: Depiction of the creation of ghost hits. On the left the intended hit pattern is shown. The active rows and columns (marked with triangles) are translated into masks and applied on the right. In addition to the intended hit locations there are other pixels located at intersections of active rows and columns.

By injecting in an iterative process and associating the number of hits per pixel to a particular color, two-dimensional color-coded test pattern can be injected. Figure 3.11 shows injection patterns based on the logos of the University of Bonn and *Forschungs- und Technologiezentrum Detektorphysik* as showcase depictions. To limit run time to acceptable periods, these test patterns were chosen to only cover a subset of the pixel matrix. For this reason only a small 3×3 pattern is injected in the basic test used in the CI pipeline.

Expanding on the basic structure of the injection test, the verification of the completely new trigger logic was developed. Additional configurations are necessary to set up the sending of triggers. These include tuning the delay period and selecting the internal trigger. In the first test run, the internal trigger command was sent in quick succession after the injection command and therefore missed the timing window of the OBELIX hit. By measuring the time difference between the hit and the trigger pulse the correct delay value is determined. In the final test a small injection pattern is set and the hits



(a) University of Bonn pattern

(b) FTD pattern

Figure 3.11: Hit maps of an injection tests showing the number of received hits per pixel. The number of injections per pixel were chosen such that the hit pattern represents a low resolution logos of the University of Bonn (a) and *Forschungs- und Technologiezentrum Detektorphysik* (FTD) (b). The hits were only injected in a smaller area of the matrix (30×30 pixels) for manageable simulation time.

are successfully triggered based on the internal trigger signal.

Test benches involving external trigger and multi-chip readout are possible future additions. Both of these tests first require additional firmware code and are not necessary for testing the basic functionalities of the DAQ system. Thus, the basic codebase is complete and covered by the three main test benches.

CHAPTER 4

Power and rail analysis

The first revision of the OBELIX sensor (OBELIX-1) has been developed over the course of several years and final analyses of the design have recently started. This chapter focuses on two analyses carried out towards the end of the development cycle: power and rail analysis. Section 4.1 introduces the general development flow of an integrated circuit sensor from the digital perspective. In Section 4.2 the different types of analysis are explained and the basic theory is discussed. The design of OBELIX-1's digital periphery is discussed in Section 4.3 before the results of the power and rail analyses are presented in Section 4.4.1 and Section 4.4.2.

4.1 Digital IC implementation methodology

The development of any integrated circuit (IC) is a time-consuming endeavor, often requiring the input specialists across multiple related fields. As introduced in Section 2.5, monolithic sensors can be divided into a digital and an analog domain. The development of both regions requires very different design flows, tools and domain-specific expertise. Therefore, they are usually carried out by separate design teams. Figure 4.1 shows an idealized, abstract development flow of the digital

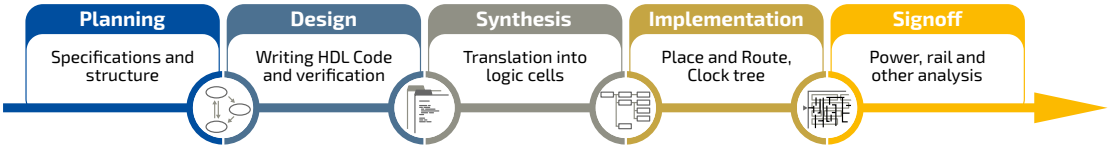


Figure 4.1: Development flow of a digital IC design, showing some of the most important steps of each development stage.

domain. After initial planning, the designers describe the digital chip logic in Register Transfer Level (RTL) code. Verification engineers add test benches similar to those introduced in Section 3.5 to verify the design. During synthesis, a tool parses the RTL code into abstract cells which are mapped to technology-specific cells. Most of these cells will be standard cells supplied by the foundry in one of their complete design kits (CDK). CDKs additionally include other technology-specific information such as design rules, technology characteristics and timing tables. Digital designers use custom cells only for very specific tasks. These could include efficient memory blocks or optimized driver cells. The cells will be placed within a floorplan and connected to other elements – a process called routing

– during the physical implementation. The clock tree, consisting of a network of buffers and wires, is also generated in this stage, connecting the clocks to the sequential logic implemented on-chip.

Once the implementation stage runs to completion, the developers run analyses, testing the design's performance and electrical properties. These signoff checks cover not only the power and rail analysis introduced in Section 4.2, but also timing analysis and detailed design rule checks, ensuring that the design will be manufacturable by the foundry. Problems found in this stage might be fixed by running additional optimization tools. If the root of the problem is more fundamental, designers might need to go back and change parts of the design, thus starting an additional iteration of the development chain. Simpler analyses are usually already performed at earlier design stages, but because of the time-consuming nature of these detailed tests, they only run during the final development stages.

4.2 Methods and Motivations

The analysis of large-scale designs consisting of millions of transistors, can realistically only be performed using sophisticated software. Even these complex tools rely on simplified models and approximations to estimate design properties such as the chip's total power consumption. This section serves as an overview of the underlying calculations performed in the power and rail analyses and highlights the importance and applications of the results. All formulas introduced in this section are based on [22].

4.2.1 Power analysis

The electronics implemented on chips consume power. Reliable estimations of power consumption – both average as well as peak consumption values – are crucial design parameters to develop working systems. The most obvious constraints come from power supplies which must be able to supply peak currents fast enough to ensure working operation of the chip. Especially for sensors developed for particle physics experiments, heat dissipation and the related need for cooling elements are a second major consideration based on the power consumption. As increased power dissipation will heat up the silicon, more effective cooling is needed to ensure the expected performance and low leakage currents in the temperature-dependent transistors. As increased cooling usually requires larger cooling systems, more material must be placed within the detector. Particles traversing this additional material lose energy and may be deflected, resulting in less accurate measurements in subsequent subdetectors. The power analysis results will also be used during rail analyses to estimate the currents drawn by the cells. Rail analyses will be covered in the following two subsections.

The power consumption of a circuit can be broken down into multiple contributions from each cell. During switching from one output state to another, external loads are charged or discharged (shown in Fig. 4.2). These capacitances are not only related to other connected cells but also the parasitic capacitances introduced by the connecting wires. At each 0-to-1 transition the total load C_{load}, driven by the cell's output, is charged from 0 V to the supply voltage V_{DD}. While part of the energy will be stored in the load capacitance, there is also energy dissipated by the PMOS transistor. The stored energy will be dissipated over the NMOS transistor during the next 1-to-0 transition. The total energy lost in one complete low-to-high and back-to-low-cycle is:

$$E_{\text{sw,cycle}} = C_{\text{load}} V_{\text{DD}}^2. \tag{4.1}$$

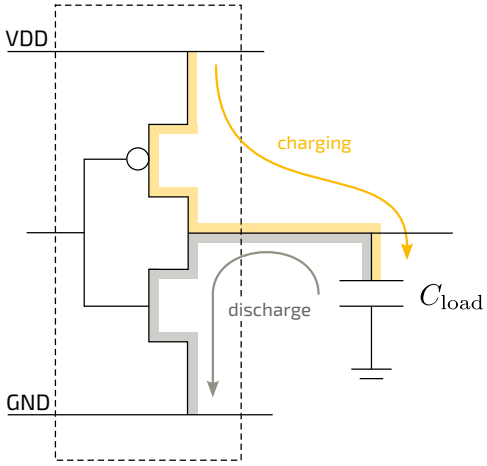


Figure 4.2: Charging (yellow) and discharging (grey) currents contributing to the switching power.

For a given switching activity $f_{0\to1}$, representing the number of charge cycles per unit time, the switching power of a cell can be described as:

$$P_{\text{sw}} = C_{\text{load}} V_{\text{DD}}^2 f_{0\to1}. \tag{4.2}$$

To further visualize the different contributions the switching power can be divided into the clock frequency f_{clk} and the transition probability $p_{0\to1}$, leading to:

$$P_{\text{sw}} = V_{DD}^2 f_{\text{clk}} p_{0\to1} C_{\text{load}}. \tag{4.3}$$

The supply voltage V_{DD} and clock frequency f_{clk} are chosen early on in the design process and are therefore only changed during later development stages if other optimizations are insufficient to reach the necessary design specifications. Transition probabilities depend on the functionalities of the modules. These probabilities are best optimized during the RTL writing stage by writing more efficient logic. Power consumption can sometimes also be reduced by switching off circuit regions only needed during specific time intervals. Clock-gating circuits implement this functionality by deactivating the clock input of sequential blocks, thus stopping the block's operations.

The final contribution, the external load C_{load}, consists of the capacitances of connected cells and the parasitic capacitances of the wires themselves. Reducing fan-outs (number of connected cells) or adding buffer stages decreases the capacitance seen by the initial cell. In many cases this merely shifts the power dissipation to the added driver cells or to the cells now driving the secondary cells previously connected to the initial output. The second capacitance contributor, coming from the wires, depends not only on the fan-out of the cell but also the distance between the cells and the local environment. Modern routing tools usually come with different levels of optimization settings. By investing more computational resources and runtime, the tool can slightly decrease the capacitance seen by the wire. Lowering the number of vertical power-grid vias and optimizing the chip's logic can also help in this regard, by reducing the density of metal connections (routing congestion) around the signal wire.

During switching, losses occur not only due to external capacitances but also internal contributions, shown in Fig. 4.3. Similar to switching power losses, capacitances within a single cell (yellow and grey in Fig. 4.3) are charged and discharged. The term describing the load dependent power

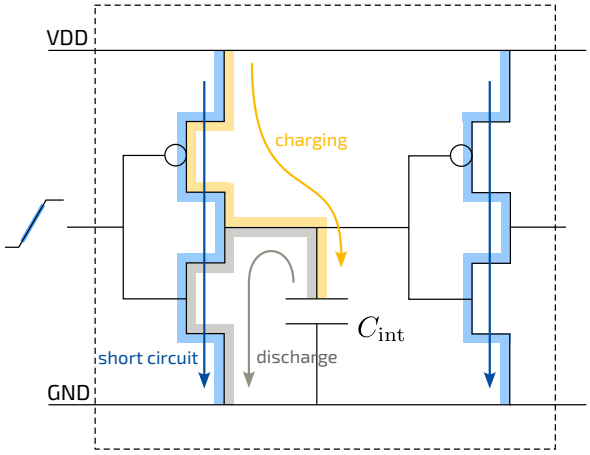


Figure 4.3: Currents contributing to the internal power losses in a symbolic CMOS cell. The blue paths show the short circuit created when both gates conduct. The yellow and grey currents charge and discharge the internal capacitor.

consumption, introduced in Eq. (4.3) also applies to these inner loads. A second source of internal power consumption during the switching process is the short-circuit current. In an ideal cell, the PMOS and NMOS transistors shown in Fig. 4.3 would never conduct simultaneously and would switch states at the same instant. In reality, input signals will always have a finite rise and fall time. During these transition intervals both transistors conduct simultaneously for a short duration t_{sc}, creating a direct current path from the supply rail to ground. The average power P_{sc} lost due to short circuits can be estimated to be:

$$P_{\text{sc}} = t_{\text{sc}} I_{\text{peak}} V_{DD} f_{0\to1}. \tag{4.4}$$

Saturation current I_{peak} and short-circuit times t_{sc} depend on the input signal's edge slope (signal slew) and transistor design. A short-circuit capacitance $C_{\text{sc}} = t_{\text{sc}} I_{\text{peak}} / V_{DD}$ can be introduced, to rewrite Eq. (4.4) to match the structure of the switching power:

$$P_{\text{sc}} = C_{\text{sc}} V_{DD}^2 f_{0\to1}. \tag{4.5}$$

Combining short-circuit power with the power consumption of the internal load C_{int}, results in an internal power contribution of:

$$P_{\text{int}} = C_{\text{int}} V_{DD}^2 f_{0\to1} + C_{\text{sc}} V_{DD}^2 f_{0\to1} \tag{4.6}$$

$$= (C_{\text{int}} + C_{\text{sc}}) V_{DD}^2 f_{0\to1}. \tag{4.7}$$

Unlike switching power related loads, the capacitances encountered in Eq. (4.6) cannot be optimized at RTL level or by the routing tool, as they depend on the cell's layout. Most cells used in digital design come from CDKs and are already well optimized. Thus, the cell's designs are usually not modified, instead designers select a cell better suited to the specifications. Alternatively, custom cells matching the desired behavior can be designed. Since this process takes a lot of effort and time, custom cells are developed only for special cases not covered by the cells included in the CDK. For power analysis, CDKs often provide internal power data for each cell as a function of input slope and supply voltage.

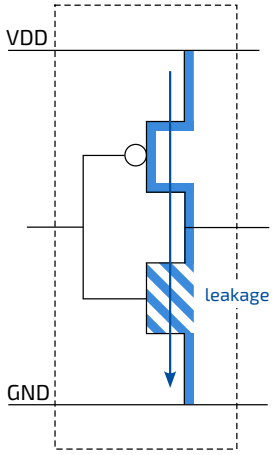


Figure 4.4: Leakage current leaking through one of the transistors.

The third power category is the static or so-called leakage power (Fig. 4.4). Static power is the only power contribution occurring outside switching periods. While ideal field effect transistors do not conduct below the threshold voltage, real transistors have a nonzero subthreshold current I_{leak}. This leakage current I_{leak} depends on cell design and external parameters such as the temperature. The dissipated leakage power P_{leak} is described by:

$$P_{\text{leak}} = I_{\text{leak}} V_{\text{DD}}. \tag{4.8}$$

Due to the leakage current's temperature dependency, effective cooling is important to reduce static power losses.

The total power P_{total} of a cell can be expressed as the sum of the three contributions, like:

$$P_{\text{total}} = P_{\text{sw}} + P_{\text{int}} + P_{\text{leak}}. \tag{4.9}$$

By plugging in the terms introduced in Eq. (4.3), Eq. (4.6) and Eq. (4.8), the power totals:

$$P_{\text{total}} = C_{\text{load}} V_{DD}^2 f_{0\to1} + (C_{\text{int}} + C_{\text{sc}}) V_{DD}^2 f_{0\to1} + I_{\text{leak}} V_{\text{DD}}. \tag{4.10}$$

In high-speed digital designs, capacitance-based contribution dominate over static power consumption.

During analysis, power consumption of each cell will not be calculated from scratch at each time step. Instead, cells are characterized beforehand, including writing power consumption, timing and other properties in lookup tables, relating these values to external loads and input slews. The analysis tool then loads this library and retrieves power values depending on each cell's load. A file containing input signals of all cells (crucial for $f_{0\to1}$ in Eq. (4.10)) can be obtained from netlist-based functional simulations – similar to the HDL simulations in Section 3.1.1.

4.2.2 *IR* drop

The resistance R of a wire is proportional to its length l and inversely proportional to its cross-section σ:

$$R = \rho \frac{l}{\sigma}, \tag{4.11}$$

with the proportionality constant ρ being the resistivity of the wire material. In small designs, wire resistance can usually be ignored. For metal wires of large designs with connection lengths in the order of millimeter or even centimeter the wire resistance can cause so-called *IR* drops. According to Ohm's law:

$$V = I \times R, \tag{4.12}$$

a current I running through the wires results in voltage drops V depending on the total ohmic resistance R between the signal current source and destination (shown in Fig. 4.5). Especially in wires carrying

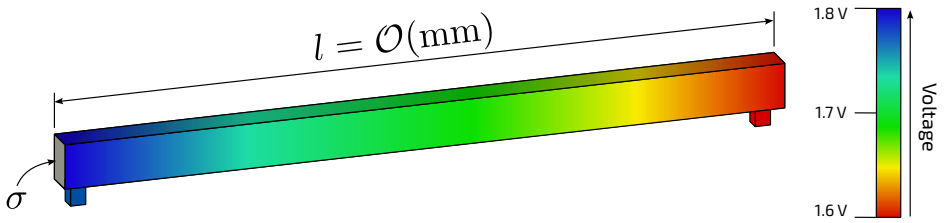


Figure 4.5: Voltage drop over a long wire caused by the wires resistance.

large currents, such as supply rails, *IR* drop can significantly affect the potential seen by connected cells. Local shifts in the supply or ground voltage will have multiple consequences. The driving strength, which dictates how fast signals can be propagated, depends on the supply voltage. If the local potential is affected by *IR* drop, signal timing at the following cells may shift enough for data to arrive too late to be captured, causing a functional failure. Reduced transistor voltages make logic more susceptible to signal fluctuations caused by noise. In the worst case, noise can cross the reduced logic threshold, potentially flipping the output state. Besides the functional consequences in extreme cases transistors can become partially conductive, increasing the power consumption of the cell.

IR drops can best be reduced by increasing the affected wire widths and adding additional connections from the power supply to power-hungry chip regions. Typically, a power grid of equally spaced metal stripes distributes both the supply voltage as well as the ground connection over the entire chip region. To locate regions that require wider or denser power grids *IR* drop analysis is performed.

IR drop analysis has to be performed with sophisticated tools due to the complexity of the analysis. This is not only due to the large number of cells but also due to the fact that there are multiple connections between the cells and the power supply as well as the time-dependence of the *IR* drop. Since voltage drop is proportional to the current I, which is determined by the cells' switching activity, the drop depends strongly on the circuit's logic behavior. As the sequential logic will only change states after a clock edge, the chip's current will usually peak in sync with the clock period. During the power analysis switching activities of the cells have already been considered. Intermediate files generated during power analysis, containing the dynamic behavior of the cells, can be read in by the tool for the rail analysis. Color-coded plots and written reports generated by the tool help the designers to locate problematic chip regions.

4.2.3 Electromigration

Large currents not only cause voltage drops but, in extreme cases, can also damage the wires themselves. At high current densities, moving electrons strike the metal ions frequently enough to transported them along the wire. If the chip operates under these conditions over an extended period of time, the damage to the wire alters the wire's structure. This can result in regions of reduced (voids) or increased (hillocks) wire widths as shown in Fig. 4.6. While hillocks might put the chip area under

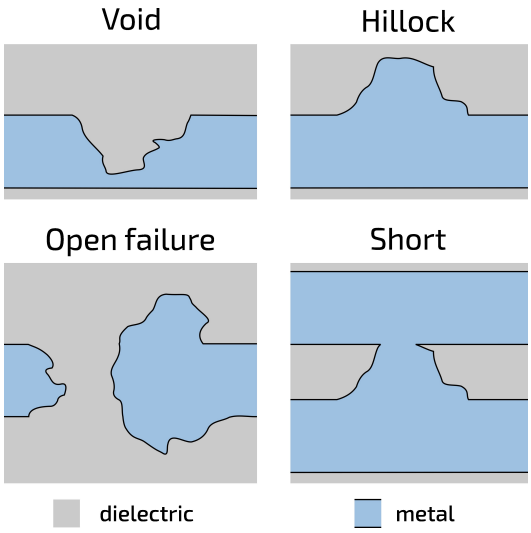


Figure 4.6: Four basic consequences of electromigration. The upper one (void and hillock) might affect performance, the lower two (open failure and shorts) will likely cause functional failures.

mechanical stress [23], voids reduce the cross-section of metal wire. Since a smaller cross-section increases the current density and heats up the area, electromigration accelerates, carrying away metal ions even faster [24]. Over time, this can create an open wire failure, shown in the lower left of Fig. 4.6. If this occurs in a critical area of the chip, an open wire failure can render the chip non-functional.

A second dangerous outcome, wire shorts, is related to the hillocks. If a hillock grows large enough over time to reach another metal (as shown in the lower right in Fig. 4.6), it shortens the two wires. Similar to open wire failure, shorts will potentially have a large impact on the overall functionality of the chip. Electromigration is most dangerous for wire carrying large DC currents like the supply network, as alternating currents tend to be less damaging due to self-healing effects [22].

During rail analysis electromigration can be studied in parallel with the *IR* drop analysis. Three metrics are investigated, capturing different effects: average, peak and RMS (root-mean-square) currents running through wires and vias. By comparing the calculated currents to the limits set by the foundry, weak points can be identified. Based on these limits and analysis results, the cross-section of the power grid wires can be adjusted and if necessary connection network redesigned.

RMS current analysis is best suited for low frequency nets, such as DC-dominated nets. Since self-healing effects are not covered, the analysis of high frequency nets is not accurately displayed by RMS results. RMS analysis is well suited for determining the sizing of supply net connections and does include effects of so-called self-heating or Joule heating.

A better-suited approach when studying high frequency AC nets is an analysis based on average currents, as self-healing – based on the alternating direction of the electron and thus transported ion

flow – is covered. The limit between the RMS and average regime can be placed around a frequency of 1 Hz as noted in [25]. Thus, analysis of typical signal nets should be performed using average currents studies. Estimates of the median time to failure $t_{1/2}$ based on Black's Law:

$$t_{1/2} = \frac{A}{j^n} \times \exp\left(\frac{E_a}{kT}\right), \tag{4.13}$$

require the average current I_{avg} to calculate density $j = I_{avg}/A$. The other parameters current-density scaling factor n, wire cross-section dependent parameter A, layer-specific electromigration activation energy E_a and temperature T are design- or environment-depend and not based on the current simulation. k is the Boltzmann constant.

The third category, peak current analysis, focuses on short but very intense currents. Extreme examples of such currents are electrostatic discharges (ESDs), which can be very dangerous if the design is not properly prepared to handle them.

All three analysis must be performed to identify the worst results for each chip element. A more detailed description of these electromigration studies can be found in [25, 26].

4.3 OBELIX-1 design

As of summer 2025, the OBELIX-1 design is nearing completion. Extensive work has been put into the design and functional verification, allowing first signoff analyses can be run. The design is based on the Tower Semiconductor 180 nm technology[1]. It consists of a six-metal-layer stack shown in Fig. 4.8. All metal layers of this process are made of aluminum.

Standard cells are implemented below the metal stack, using polysilicon (Poly) and the lowest metal layer M1. The cells are designed to fit in an evenly spaced alternating pattern of ground (`GNDD`) and power (`VDDD`) rails. This means that the cells orientation (indicated with a triangle in one corner) is inverted every second row as shown in Fig. 4.7.

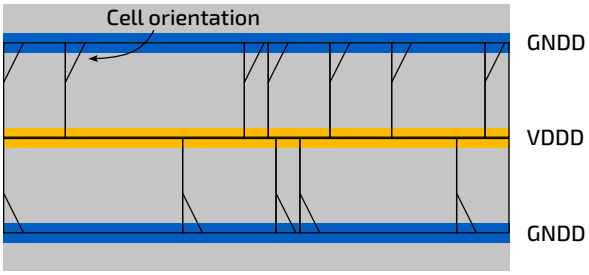


Figure 4.7: Standard cell positioning and orientation between power and ground rails.

The metal layers M2 to M5 are used during routing to connect the cells and ports. As some custom cells include metal connections on higher metal layers, the routing tool has to avoid these blocked areas as well as via, connecting lower layers with higher layers. For power distribution an alternating grid of parallel ground (`GNDD`) and supply (`VDDD`) stripes is implemented on the top metal layer M6. These stripes distribute the power, supplied at the chip bottom, evenly over the periphery using a power ring.

[1] https://towersemi.com/technology/process-technology-offerings/

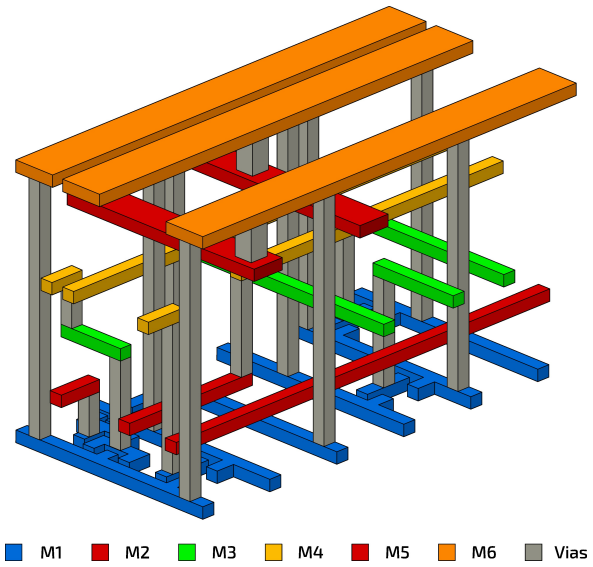


Figure 4.8: Multilayer design of a six metal layer integrated circuit, showing the different metal layers using the Cadence tools color code.

In the simulations each power stripe is connected to the power supply individually, while in reality a horizontal power stripe will distribute the power to the vertical stripes. Every eighth power-ground stripe pair is connected to lower metal layers with via stacks. The seven stripes in between lack these dense via connections as it was estimated that otherwise the routing tool would struggle connecting cells on different sides of the via stacks.

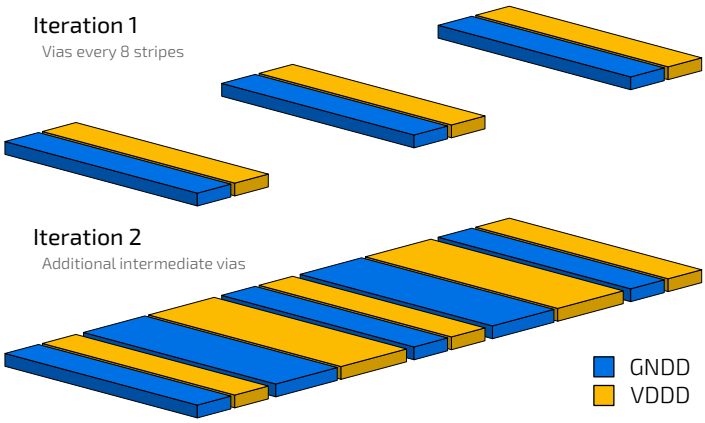


Figure 4.9: Two iteration of top level power stripes. During the second iteration additional vias were added to connect to the M1 layer.

During Section 4.4.2 focusing on the *IR* drop analysis, the need for a second iteration of the power network will be discussed. The changes in the second design are partially illustrated in Fig. 4.9. Thick top metal stripes were added between the existing structures. Additional via stacks were added in regular patterns, reducing the current density through the old via stacks and stripes. Keeping routing congestion in mind, these new via stacks are smaller and less frequent.

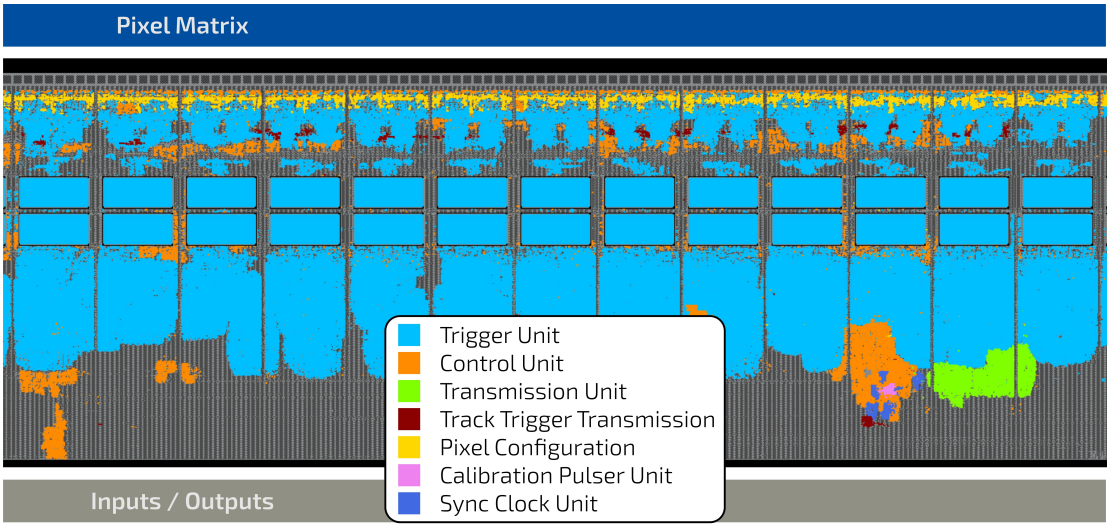


Figure 4.10: Central section of the digital periphery with color coded module cells.

The analyses discussed in this chapter focuses on the digital periphery. In Fig. 4.10 a central section of this area is shown, displaying the main modules in different colors. The pixel matrix is located close to the top edge, while inputs and outputs – both for signals and power – are located below the bottom edge. Communication-focused modules like the Trigger and Command Unit are primarily situated closer to the lower edge, leaving space for the large Trigger Unit on top. As discussed in Section 2.5.3 the Trigger Unit is subdivided into multiple trigger groups, each responsible for 4 double columns. This structure is fairly well visible in the figure, showing thin vertical empty areas between pairs of trigger groups. The top part, closest to the matrix, handles the data merging, digital injections and additional functionalities. Towards the center large rectangular SRAM elements are located. These custom blocks - called macros - are placed manually in advance on the floorplan. The memory block closer to the top edge is used by the left trigger group, while lower one is associated with the trigger group of the right four double columns. The second stage of the trigger memory requires less memory. It is realized with standard cells and is located below the macros. At the very top of the digital periphery elements of the Pixel Configuration are located. These cells drive configuration signals on the matrix, encountering large loads. As discussed in Section 4.2.1, the switching power scales with the load capacitance. Therefore these cell are expected to play a major role in the power consumption during the initial configuration phase.

During the redesign of the power network custom driver cells were added to the top of the digital periphery. These cells drive the so-called Bunch Cross Identification (BCID) signal on the matrix. This signal is used for time stamping and thus needs to be delivered quickly along the columns. As cross talk of these signals is a significant concern, it is planned to reduce the signals voltage swing. The cells used in the analyses are the driving cells used in TJ-Monopix2 and do not yet include the reduced voltage range. Thus the two design iterations are still comparable. Once the final design of the improved cells is implemented, additional analyses need to be rerun, as the changes will have positive impacts on the power consumption.

The digital periphery measures 29.6 mm in width and 2.43 mm in height, with a $3.88 \times 0.52\,\mathrm{mm}^2$ cutout at the lower-left corner.

4.4 Results

For this work the power and rail analysis tool Voltus [27], developed by Cadence, is used. Its output consists of text reports and plots viewable through the graphical user interface. A mixture of both types of results will be used in this section.

Both static and dynamic power analysis require realistic net activities to create accurate results. A Cocotb testbench is used to simulate the response of the digital periphery on a hit-rate of 120 MHz/cm^2. This hit-rate equals the requirement set for the VTX upgrade. [13] Hit signals are simulated on the Trigger Unit's inputs and propagated through the Trigger groups to the Transmission Unit. The power and current estimates of analyses based on the simulation's activity file will be conservative limits, since this requirement includes a 71.7 % safety margin above the expected hit rate in Belle II.

As of now the simulation only runs with a reduced matrix size of 16 double columns. The corresponding reduced implementation is shown in Fig. 4.11, featuring the Trigger Unit stages for the leftmost double columns and a central grouping of other digital units. This simplified smaller design makes analyses and simulations faster to run and allows for rapid implementation-testing cycles, crucial at this stage of development. On the flip side, the power results of this configuration will not be easily scalable to the final large implementation. The small design will be used to understand the contribution of the different cells implemented in the design as well as general power analysis trends. For a first order estimate of the total design power a vectorless analysis (without activity file) will be run on the full design, to get a second average power estimate. In rail analyses the small design is well suited to locate issues and calculate accurate results, as the leftmost design is similar to the final TRU structure, and the analysis depends more on local properties than the chip wide power analysis.

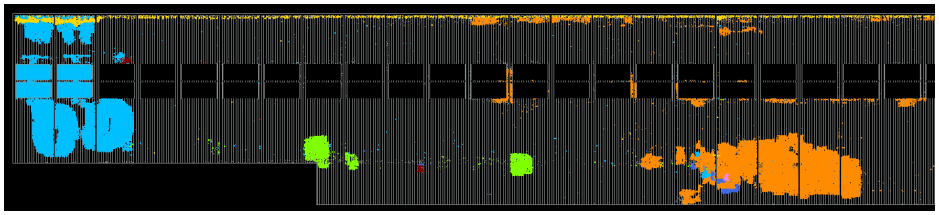


Figure 4.11: Reduced 16 double column design used in simulation. The same color code of the different modules is used as in Fig. 4.10, highlighting the change in cell density.

4.4.1 Power analysis

The activity file obtained from simulation covers the injection and readout of many hits, totaling a simulated duration of 76 µs. Since an analysis of the whole simulated timeframe is neither reasonable nor necessary, a suitable time window with reasonably high activity has to be identified. Voltus's integrated waveform analyzer is used to search for the time windows of a given length (in this case 120 ns) with the largest cell activity. Usually the top results include the very beginning of the simulation since many signals change during this period. This period is useful for extreme case studies, but does not represent an expected average of typical operation. For analyses centered on standard operation – like average power consumption – it makes sense to also identify a period that more closely resembles the standard operation. The vector profile tools report lists the values shown in Table 4.1.

During the analyses the two intervals reported with the highest activity will be analyzed – referred

Table 4.1: Top five activity results from the vector analysis run of the simulation activity file. The fourth column notes which periods were chosen for the analysis.

Start [ns]	**End** [ns]	**Activity**
0	120	39420
75840	75960	11690
30840	30960	11650
81240	81360	11410
66240	66360	11240

to as the early (startup) period and late (typical operation) period. These extreme cases are chosen to make sure that analysis results do not underestimate activity fluctuations. Especially the early period will likely overestimate the real startup sequence as in simulations all configurations happen simultaneously, while in reality they will be performed in sequence. Accurate startup simulations will need to be done in future studies to obtain more accurate results.

Static power analysis

The average power consumption of the simulated periphery totals 52.84 mW during the late period. Figure 4.12 illustrates the different power sources' contributions. Looking at the three power types (values are given in the legend), the results matches the expectations. The two capacitance based contributions – internal and switching power – make up 99.98 % of the power consumed. Leakage power contributions are negligible.

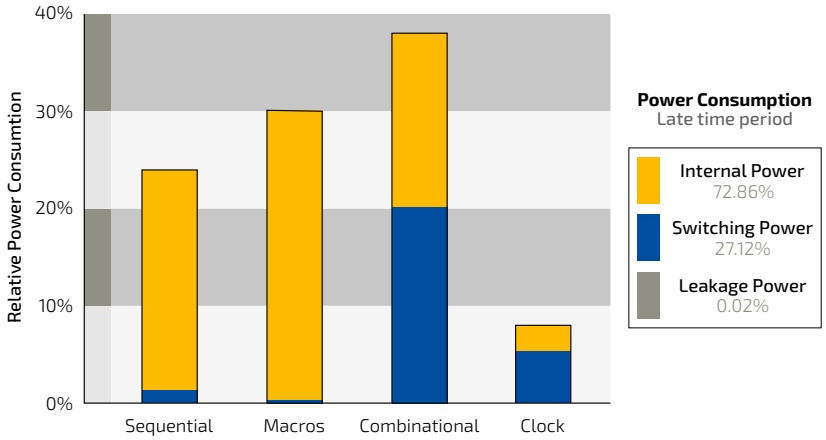


Figure 4.12: Power consumption of different logic types and power categories during the late period.

The power consumption depicted is divided in the contributing logic cell types. Standard logic is subdivided into sequential logic – logic changing its state at a clock edge – and not clocked combinational logic. Together they make up more than 60 % of the total power consumption. The second highest contribution comes from predefined macro cells, contributing around 30 %. In this design the largest macros are the memory cells, used in the TRU, referred to as trigger memory. The large contribution by these cells is also obvious when displaying each cell's total power consumption in the Voltus GUI (as shown in Fig. 4.13).

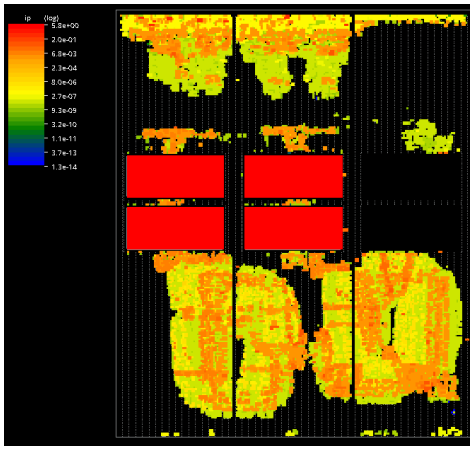


Figure 4.13: Cell-based total power consumption during the late period. Created with Voltus [27].

The remaining 7.97 % are due to the clock distribution network. It can be observed that in sequential logic the internal power clearly dominates, while in combinational logic switching and internal power contributions are similar. Sequential logic typically consists of more complex cells, which leads to larger internal power consumption due to higher internal loads. Sequential logic often does not directly drive large external loads, but instead buffers are used to distribute the output signal in case of high fan-out. This shifts the switching power from sequential to combinational cells. Thus, the observed distribution matches expectations.

Comparing the average power 52.84 mW of the late period to the early period's power consumption of 1.368 W, a clear contrast becomes obvious. In the start-up sequence the main contribution comes from the combinational logic, making up 95 % of all power accounted for (shown in Fig. 4.14).

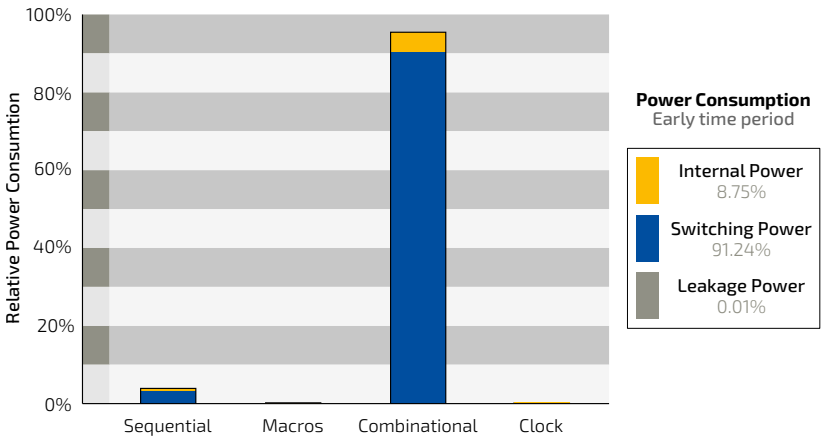
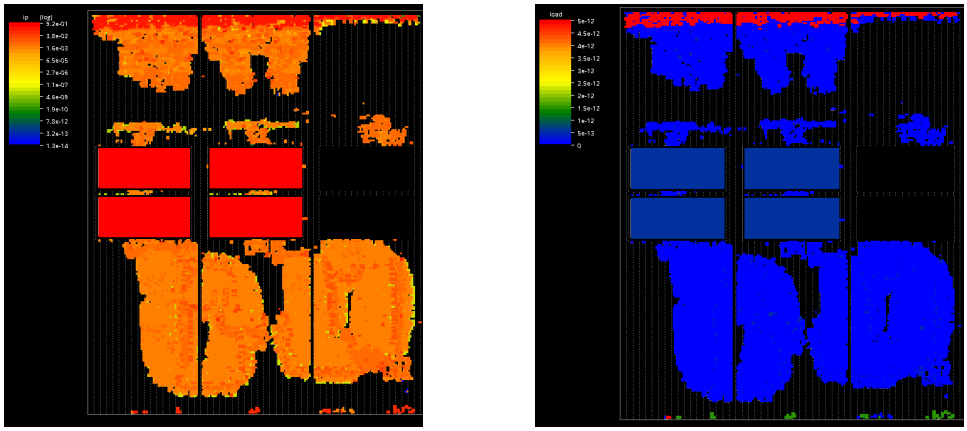


Figure 4.14: Power consumption of different logic types and power categories during the early period.

Almost all of this power is attributed to cell's switching power. The plotted power contribution of the cells, shown in Fig. 4.15(a), can be used to study this behavior. The largest power consumptions per cell are located at the very top and in the center (Trigger memories). As the plot does not weight the power consumption by the cells area, it is the sheer number of red-colored cells at the top that



(a) Power consumption (b) Load

Figure 4.15: Cell-based total power consumption (a) during the early period and loads (b) seen by the cells. Created with Voltus [27].

outweighs the contribution of the memory macros. These cells are driving configuration signals on the pixel matrix. Due to the size and structure of the matrix, the load seen by these driving cells is significantly larger than the load between cells in the periphery. This is depicted in Fig. 4.15(b), showing the load of each cell. The load seen by the driving cells exceeds 5 pF, while more than 99 % of the other cells only see loads in the femtofarad range. As discussed in Section 4.2.1 the power consumption scales linearly with the capacitance, explaining the relation between the two figures. As resets and configurations, which are performed at the very beginning, affect the matrix, these driver cells contribute greatly during the early and not the late period.

The power consumption is not only important for the power supply but also for the heat dissipation of the chip. In both cases the power consumed by the whole periphery is important. Scaling the values obtained from the 16 double column setup is not straight forward, as one has to identify cells which would need to be multiplied for the additional double columns and elements, like the command unit, which stay the same, no matter the matrix size.

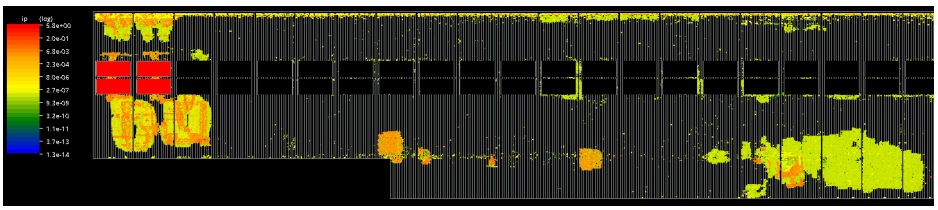


Figure 4.16: Cell-based total power consumption during the late period, showing a larger area. Created with Voltus [27].

Figure 4.16 depicts a larger area of the periphery including the elements that are not trigger group specific. The power consumption is depicted during the late period. Based on this picture it is difficult to gauge the ratio of power that can be associated purely to trigger group specific cells. By placing a lower estimate of the percentage p_{trig} of the trigger group specific power at around 60 % and the upper limit at 90 % one can scale the total power consumption $P_{\text{tot},N_{\text{DC}}}$, like:

$$P_{\text{tot},N_{\text{DC}}} = P_{\text{tot},16} \times \left(1 + p_{\text{trig}} \times \left(\frac{N_{\text{DC}}}{16} - 1\right)\right) \tag{4.14}$$

for a given number of double columns N_{DC}. In the complete design N_{DC} will be 448. Plugging in the 52.84 mW yields a power range of 0.909 W for the lower bound and 1.337 W as the upper limit. While this wide range is not very precise it can still be used to calculate a rough estimate.

Knowing the differences of the early and late power distributions gives reason not to use the same scaling for the early period as for the late period. In Fig. 4.17 a wider section of the periphery is shown. Looking more closely at the top edge, it becomes clear that large portions of the driver cells have already been implemented for all 448 potential double columns. The exact amount is once again difficult to identify. As a first order estimate $p_{\text{trig}} = (35 \pm 20)$ % is assumed based on closer study with

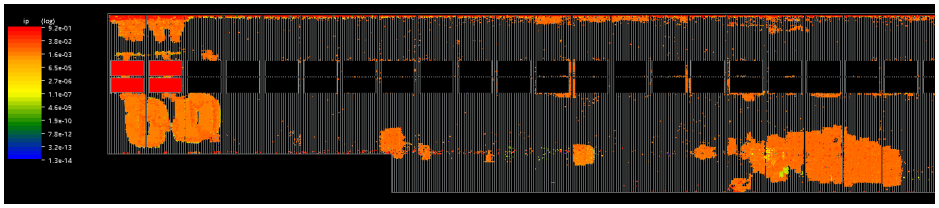


Figure 4.17: Cell-based total power consumption during the early period, showing a larger area. Created with Voltus [27].

the Voltus user interface. This results in a total start-up power estimate of $P_{\text{tot},448} = (14.3 \pm 7.4)$ W. Chip startup power consumptions similar to this are not feasible in hardware. In simulation all configurations are executed in parallel. The best option to avoid these extreme power consumptions is to configure different chip components in sequence, spreading out the power consumption over multiple clock cycles. To optimize this sequence, dedicated power up and configuration simulations will need to be performed.

Instead of relying on estimated scaling factors, an alternative approach is the simulation of the complete design without activity files. For this vectorless analysis Voltus requires only the dominant clock frequency and a global input activity. In the digital periphery most logic is based on the 20 MHz clock. The input activity, describing the probability of an input signal changing, was estimated to be around 0.2. Using these values Voltus propagates the switching probability through all cells and uses the cells power estimates provided in the CDK to calculate the total power consumption (shown in Fig. 4.18). Since all inputs have the same switching probability, all trigger groups behave similar.

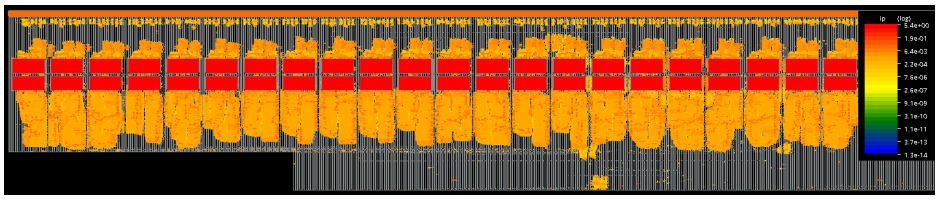


Figure 4.18: Cell-based total power consumption calculated based on the statistical approach, showing a very uniform distribution over all trigger groups. Created with Voltus [27].

This is not very realistic, as one would expect to only see large activities in trigger groups which had a

recent hit in the matrix. This analysis approach is less accurate than the usage of simulation-based cell activity, but can be used to generate a general estimate if netlist-based simulations are not available.

The total power reported by the tool is 1.128 W. This value matches well with the limits calculated based on the later period of the activity file. The contributions of the power types can be compared with the distributions of the activity file-based results, displayed in Fig. 4.19. Similar to the later

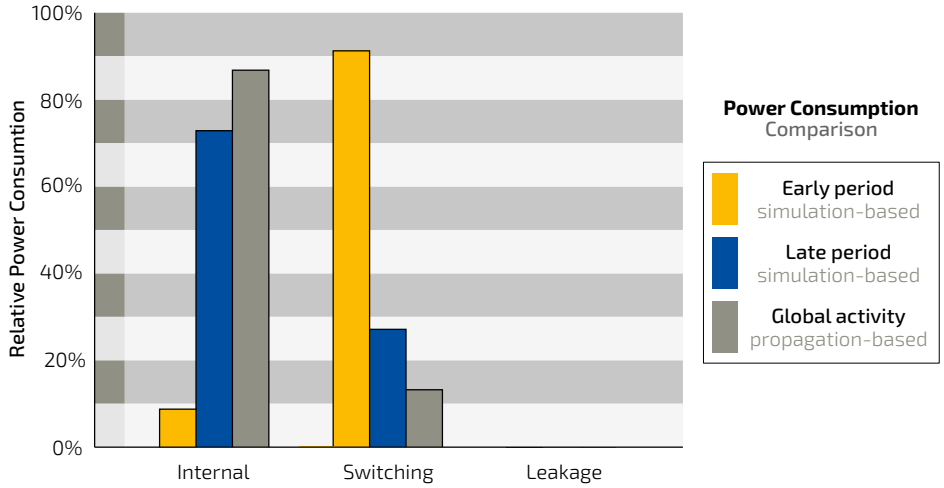


Figure 4.19: Comparison of the relative power distribution of the two activity file-based periods and the propagation-based approaches. In all three analyses the leakage current was less than 0.05 % of the total power consumption.

period's result, the propagation-based analysis shows the largest power consumption coming from internal power consumption. The two distributions do not match perfectly, as deviations of more than 10 % remain. This is expected due to the inaccurate nature of this method and highlights the importance of simulation-derived activities. Still, the similarities in both the total power as well as the overall distribution trend support the results shown so far during the later operation stages.

In chapter 7 of [12] the power consumption of TJ-Monopix 2 is discussed during idle operation. The power associated to the periphery was measured to be around 354.2 mW, with the clock distribution being identified as the main power contributor. There are multiple differences between OBELIX and its predecessor, which make a direct comparison of the total power value difficult. OBELIX features a larger amount of new modules and features implemented in the periphery like triggering. As seen in the floorplan (Fig. 4.10) the size of those iteration – such as the Trigger Unit – adds many transistors. TJ-Monopix 2's matrix is smaller, thus reducing the amount of matrix size dependent cells. The activity file used in the power analysis of OBELIX is based on a simulation of an intentional overestimated large hit rate, while TJ-Monopix2's measurements were conducted during idle operation.

Based on this list, a larger power consumption is expected in the OBELIX design in comparison to its predecessor. More accurate analysis based on activity files including all 448 double columns, as well as first measurements with the manufactured OBELIX-1 chip will be required before OBELIX-2 signoff. Still, the results of this section can be used as first estimates for the research in viable cooling solutions and the requirements for power supplies.

Dynamic power analysis

Almost all logic cells implemented in the digital design are either directly clocked or depend on sequential input signals. Power and currents drawn by these cells are thus not uniformly distributed over time, but are time-dependent. Thus, they are not fully described by looking at average power consumed over multiple clock cycles. In this subsection the dynamic power behavior of the digital periphery will be analyzed, based on the same simulations as used before. Analysis results of the time-dependent power consumption are best visualized by plotting the time-dependent variables with respect to the time. Figure 4.20 shows the ground and supply current during the first 120 ns. A regular pulsed structure is visible on both power rails. The larger current peaks are spaced at intervals of (50.0 ± 0.5) ns. The dominant clock frequency of the digital periphery is set to $f_{20\text{MHz}} = 20\,\text{MHz}$ in simulation – in the real design a slightly deviating clock frequency of 21.2 MHz will be used. Translation the peak spacing Δt to a frequency f_{peaks}:

$$f_{\text{peaks}} = \frac{1}{\Delta t} = (20.0 \pm 0.2)\,\text{MHz} = f_{20\,\text{MHz}} \tag{4.15}$$

confirms that the peaks are related to the 20 MHz clock.

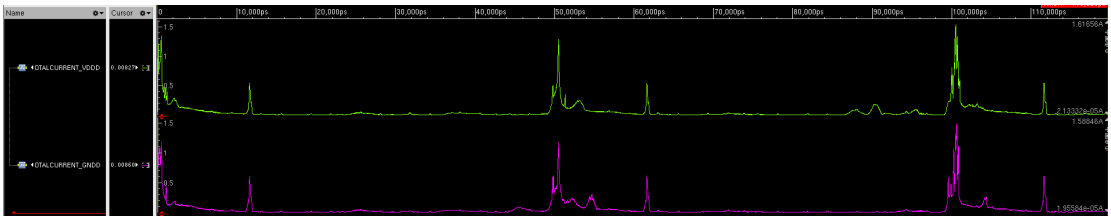


Figure 4.20: Total supply (top) and ground (bottom) current shortly after the early period. Created with Voltus [27].

Looking at the shape of each current spike, different features can be identified. The signals do not follow a simple gaussian-like shape, but are composed of multiple contributions with different widths, heights and delays. The variety of widths indicates contributions of cells with different driving strength and load, as large loads and lower driving strength will result in longer charging or discharging periods. Peak heights depend on the amount of cells with the given characteristics, adding up the current amplitudes. Large cells and long serial chains of combinational logic result in a delayed response of cells waiting for preceding signals. Comparing the large array of contributions during the early stage (Fig. 4.20) with the waveforms of the late period in Fig. 4.21, illustrates the shift of active cells in the two simulation stages. As the matrix configuration – absent in during the late period – is associated to large loads, one would expect slow, wide contributions primarily in the early period. This expectation is confirmed by the simulation results, which show almost exclusively short peaks in standard operation.

A comparison of the ground and supply currents also reveals differences in their shapes. By closer inspection of the waveforms in the graphic user interface the ground current can be observed to rise earlier than the supply current. This can be explained by the CMOS cell structure. The charge carriers of NMOS transistors, used to pull down the output voltage, are electrons, in contrast to the hole carriers of PMOS transistors. Due to the higher carrier mobility of the electrons NMOS transistors will be able to switch faster. This results in the faster ground current responds in the CMOS technology used

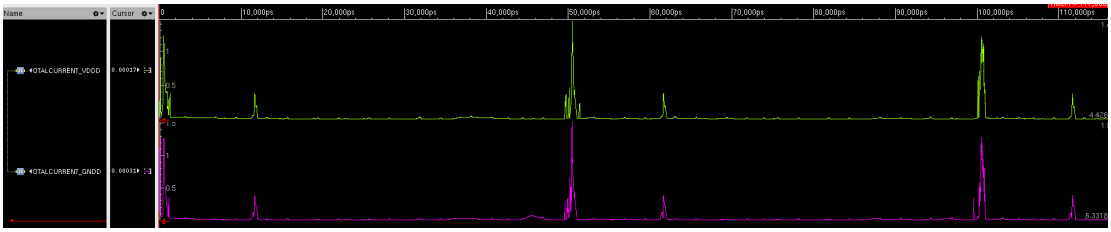


Figure 4.21: Total supply (top) and ground (bottom) current during the later period. The time shown on the horizontal axis is shifted by 75 996 ns. Created with Voltus [27].

in this chip.

The amplitude of the current peaks varies up to 1.5 A. In a real setup power supplies would struggle to produce the narrow peaks, due to the sharp rise time. Thus, translating the peak currents I_{peak} using

$$P_{\text{peak}} = V_{\text{DDD}} \times I_{\text{peak}} = 1.8\,\text{V} \times (1.50 \pm 0.05)\,\text{A} = (2.70 \pm 0.09)\,\text{W} \tag{4.16}$$

into an expected peak power, would overestimate any realistic value. Still this calculation is valuable for the development of the LDOs supplying the voltages.

A similar effect, spreading the peaks over even longer periods in reality, is based on the models used to describe the pixel matrix. In the simulation cells driving signal on the matrix only see the large total capacity of the matrix, while neglecting connection resistances. More accurate models replacing the capacity with an RC model, shown in Fig. 4.22, will slow down charging and discharging processes. Consequently, the current would distributed over a longer period of time, reducing the peak height.

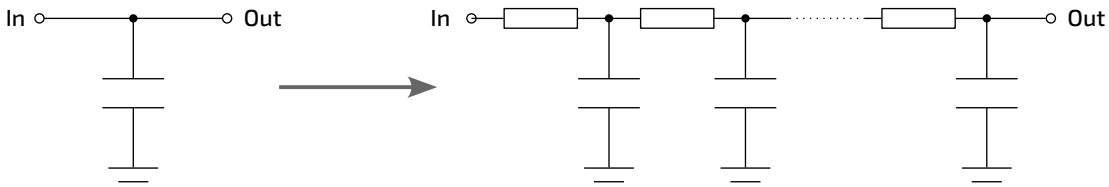


Figure 4.22: Comparison of different models of the pixel matrix, showing the used simple capacity model (left) and more detailed RC model (right).

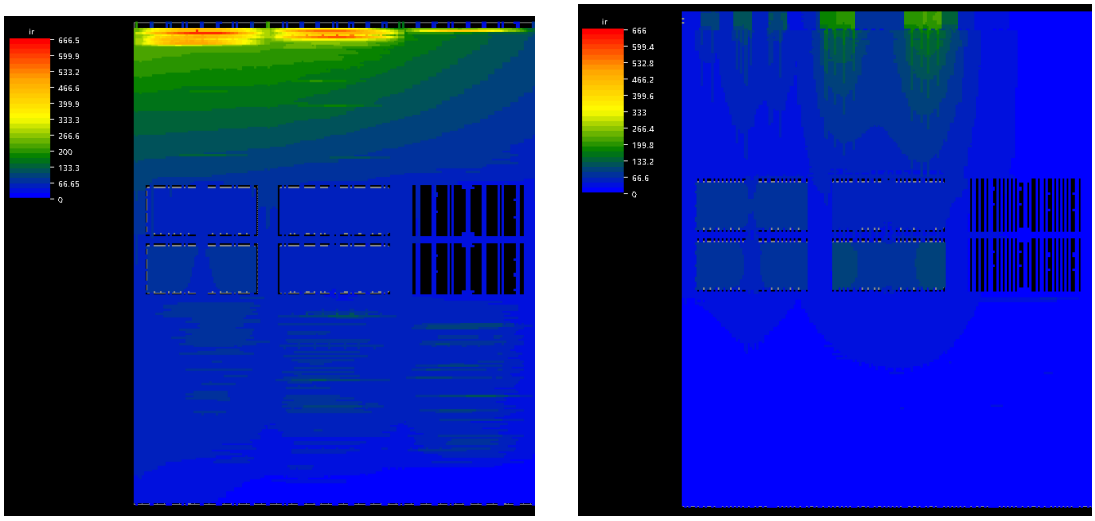
The designer can also deliberately spread out the current peaks by modifying the design. These modifications include reductions in drive strengths or the introduction of local capacitances – so-called decoupling capacitors. These decap cells could be added between the LDOs and matrix driver cells, to act as a local decoupling element, stabilizing the supply against sudden current demands.

4.4.2 Rail analysis

The large current observed during power analysis can have a large impact on the design performance and durability. To calculate the voltage drops and locate potentially overloaded connections in the design, *IR* drop analysis and electromigration analysis are performed.

IR drop analysis

The voltage drop during the early and late periods is shown in Fig. 4.23. The local voltage drops experienced from cells are color-coded relative to the maximal voltage drop within the given time frame. Cells access the power net through equally spaced M1 power rails, which are connected to the



(a) Early period (b) Late period

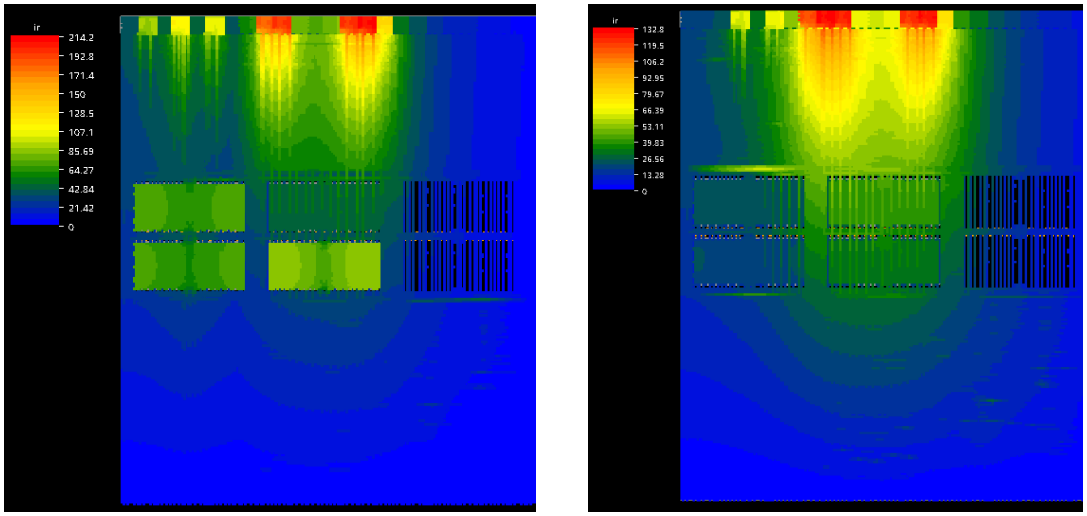
Figure 4.23: *IR* drop during the early (a) and late (b) period of the first design iteration. Created with Voltus [27].

top metal power stripes via vertical metal vias inserted every second Trigger Group. The consequences of the relatively large via spacing can be seen in the *IR* drop plots. In Fig. 4.23(b) the yellow-red areas of the highest voltage drops extend across the width of 8 double columns – from one via stack to the next. Cells located in the center of trigger group pairs are the furthest away from the vias and therefore show the worst voltage drop. In addition to the horizontal rails, the vertical stripes are also partially visible in the depiction.

The maximum *IR* drop observed during typical operation is 225 mV. This means that some cells operate at only 87.5 % of their intended voltage. As discussed in Section 4.2.2 large drops like these can have a significant impact on circuit operation and need to be avoided by better design. Even larger voltage drops can be observed during the early period, where the most extreme *IR* drops exceed 660 mV, which corresponds to more than one-third of the total supply voltage. It is unlikely that the chip would be able to function correctly under these conditions. As one would expect, based on prior power discussions, these extreme drops occur at the very top of the digital periphery, where the cells responsible for driving signals onto the matrix are located. To supply these cells with enough power, without causing excessive voltage drops, a better-connected power network is required.

The second iteration introduced in Section 4.3 adds regularly spaced vias in the space between the existing via stacks. This configuration ensures that the distance from one cell to the next via is on average considerably shorter than in the initial design. Increasing the number of vias also reduces the current running through each via individually, which is crucial to decrease the voltage drop.

The *IR* drop plots of the second design iteration are shown in Fig. 4.24. The highest *IR* drop during the startup sequence is reported as 214.2 mV. This is a significant improvement compared to the initial design value of 666.5 mV. A maximal drop of 132.8 mV seen in the late period also represents



(a) Early period (b) Late period

Figure 4.24: *IR* drop during the early (a) and late (b) period of the second design iteration. Created with Voltus [27].

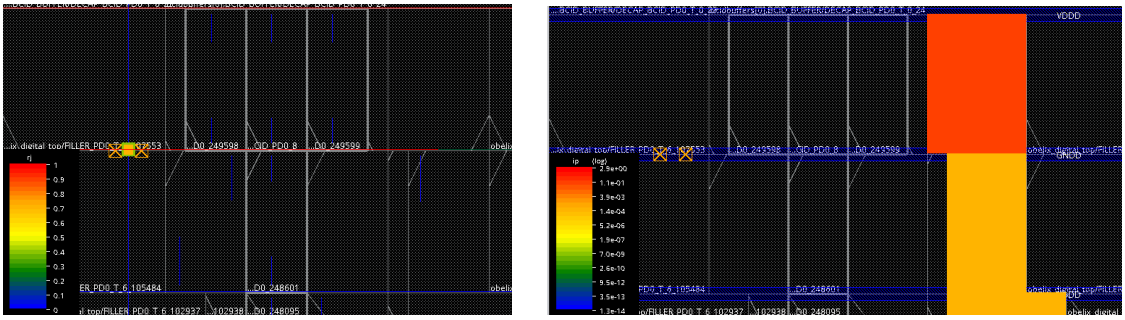
an improvement. These values show the positive impact of the added metal, but are still suboptimal. Further iterations will be required to achieve acceptable signoff values of less than 100 mV. Possible redesign strategies will be mentioned in Section 4.4.2.

As expected, the largest voltage drop occurs furthest from the supply, at the very top of the design. In this iteration the BCID driver cells are located at this edge. The voltage drop is not uniformly distributed across the individual BCID driver macros, instead there are two maxima. These maxima do not originate from the BCID marcos themselves, as all of them drive the same timing signal on equal loads. An inspection of the cells located below reveals a non-uniform cell distribution. Regions with the highest *IR* drop also contain large numbers of cells. As there are also areas further to the left with comparable numbers of cells without large *IR* drop, the voltage drop is not directly dependent on density itself. Instead, the cell types, activity and the corresponding current draw must be taken into account. Fig. 4.25 shows the metal connection with the highest average current. It is an M1 power rail segment located in one of the two regions with high voltage drop. In the right depiction the power consumption of the cells is displayed, showing two power-hungry cells connected to this ground rail.

Additional cells with comparable power consumption are located in most affected areas, providing a possible explanation for the observed non-uniform voltage drop distribution.

The *IR* drop plots of the second iteration also visualize the changes in via positioning. Instead of wide regions of extreme *IR* drop spanning eight double columns(Fig. 4.23), the voltage drops are more localized (Fig. 4.24). The example of the worst case (Fig. 4.25) shows large currents being located around the individual vias, thus confining the affected area to rail segments located between pairs of via connections.

A future iteration could further improve the voltage drop by distributing cells with large power consumption more evenly over the different rail segments. Such a strategy would limit the maximum current drawn by individual rail segments, thereby reducing the local voltage drop. On a larger scale, this would also promote greater uniformity, mitigating extended areas of increased *IR* drop. As



(a) Average current (b) Power consumption

Figure 4.25: Metal connection with the highest average current. (a) depicts the average current in units of current limits set by the foundry and (b) the total power consumption of the cells. Next to filler cells there are two cells with large power consumptions connected to the same GNDD rail. Created with Voltus [27].

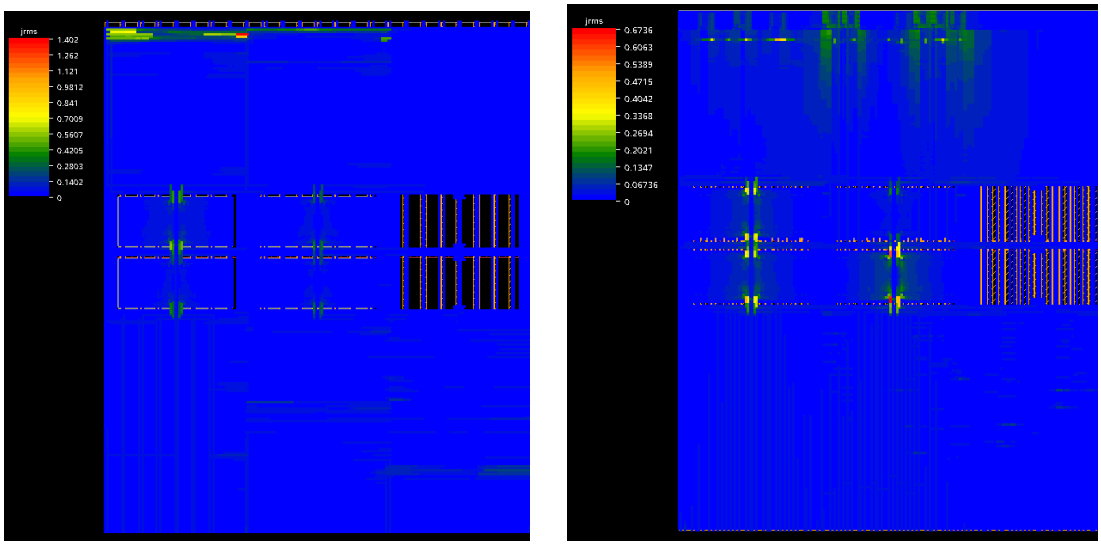
the current implementation is mainly based on optimized signal routing distances, these additional constraints could negatively impact the signals timing and need to be studied if such steps would be implemented.

Electromigration analysis

The large currents seen so far during the analysis can potentially be dangerous to the design over time. During the electromigration analysis the currents of both design iterations are compared to current limits set by the foundry. 0.1 % of metal connections running at these current limits for 1×10^5 hours are expected to show a change in resistance of 10 %. This corresponds to around 11.4 years of continuous operation. For most of its operational time, OBELIX is expected to work under typical conditions. Thus, the electromigration analyses are performed using the data of the later time period.

Plot comparing the RMS, average and peak currents of the two designs are displayed in Fig. 4.26, Fig. 4.27 and Fig. 4.28. In all three categories, there are chip regions of the first design iteration violating the current limits. Almost all of these violations are located at the very top of the design, matching the location of the largest *IR* drop. While the RMS and peak limits are violated by a factor of up 1.42 on these M1 rails, the average current limit I_{limit} is surpassed by a ratio of up to $I = 9.1 I_{\text{limit}}$. For a scaling factor $n > 1$ Black's Law (Eq. (4.13)) predicts a faster than $1/I$ decrease in expected median lifetime to failures with respect to the current I. Thus, a violation of more than nine times the current limit is unacceptable for a sensor which is expected to operate over multiple years. This is especially true, since individual sensors within the Belle II detector can not be replaced during operation periods. The violations are not isolated points along the wires but cover, as shown in Fig. 4.27(a), the full width of 8 double columns between the via stacks.

To reduce electromigration risks similar design modifications are applied asin mitigating *IR* drops. Larger wire cross-section and added alternative power connections between supply and affected regions are some of the typical strategies. Based on this knowledge it is not surprising to find improvements in the second design iteration, which implemented both of these strategies. Table 4.2 summarizes the worst circuit elements of both iterations in each category. In all three analysis classes, improvements can be observed. While the peak current is fairly close to the limit, the RMS current is now well within bounds. Only the average current is still clearly violating the limit, with a maximal current of



(a) First iteration

(b) Second iteration

Figure 4.26: Root mean square currents relative to their design limits of the first (a) and second (b) design iteration. Created with Voltus [27].

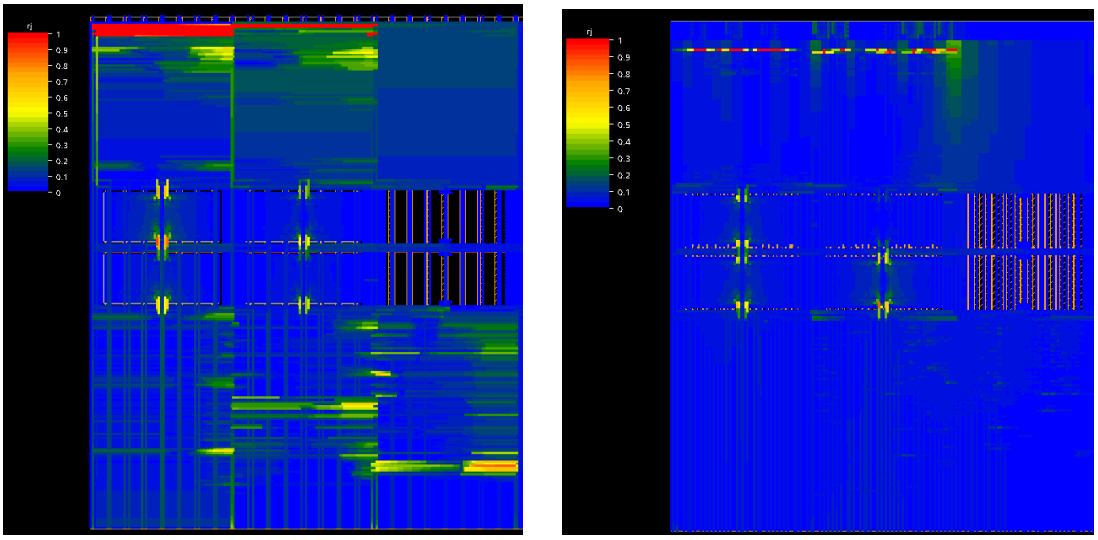
Table 4.2: Largest RMS, average and peak current violations of both iterations in units of the design limits set by the foundry.

Design	RMS current	Average current	Peak current
Iteration 1	1.424	9.061	1.402
Iteration 2	0.674	2.643	1.022

2.643 times the allowed value. This is still a significant improvement relative to the first iteration. In the different plots the locations of the largest currents remain at the top of the design as well as the power connections between and to the memory macros. Similar to the results of the *IR* drop analyses, the horizontal areas of the largest currents no longer stretch the full width of 8 double columns, but are no confined to shorter stretches of M1 rails.

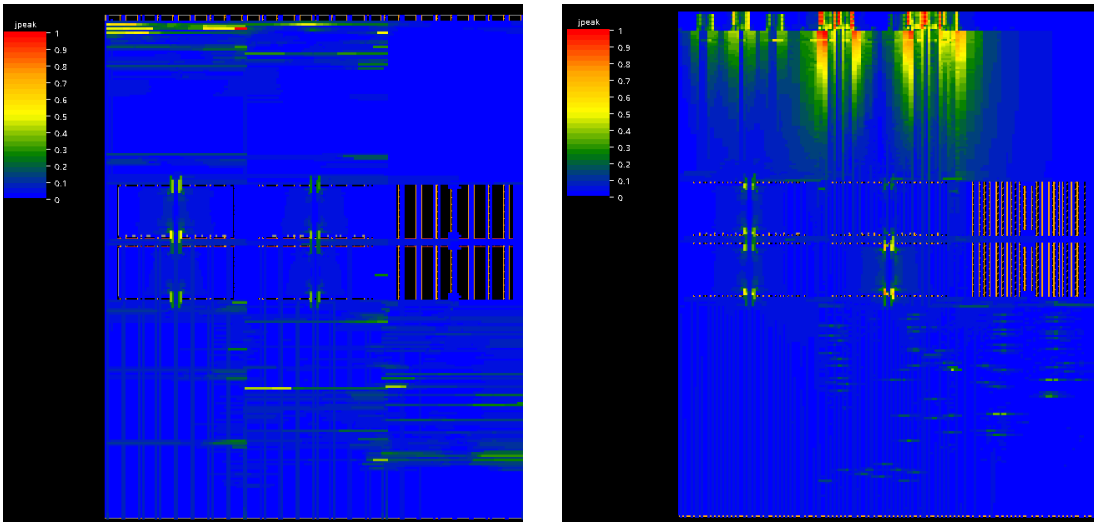
The added thick top metal stripes are clearly visible in Figure 4.28(b) and to a lesser degree in Fig. 4.26(b), as well as the rectangular BCID driver cells at the top of the periphery. As the BCID cells drive large loads, their peak current consumption is large. Depending on the local density of vias connecting the cells to the stripes, large peak currents are forced to run through only a small number of vias. This is seen fairly well in Figure 4.28(b) with large current densities concentrated around these connection points. Further design iterations could explore improved supply and ground connection strategies around the top area to decrease the peak current below the limit by either adding vias or widening the existing ones.

The comparison of the two design iterations demonstrates reduced electromigration risks. Further possible improvements, like more and wider connections as well as reordering of power-hungry driving cells are promising next steps, which will need to be implemented and tested by rerunning the simulations.



(a) First iteration (b) Second iteration

Figure 4.27: Average currents relative to their design limits of the first (a) and second (b) design iteration. Created with Voltus [27].



(a) First iteration (b) Second iteration

Figure 4.28: Peak currents relative to their design limits of the first (a) and second (b) design iteration. Created with Voltus [27].

CHAPTER 5

Conclusion

A data acquisition system for the upcoming OBELIX sensor was successfully developed using simulation tools. For the interaction with the FPGA firmware modules an easy-to-use Python interface is provided, abstracting away the low lever interaction such as command protocols and data parsing. The High-level functions are used in Cocotb testbenches, to verify the core DAQ functionalities. These tests include register interaction as well as the injections of hits. A third testbench simulates the new trigger feature, matching the delays of hit signals and trigger pulses.

A series of power integrity studies were performed for the digital periphery of the OBELIX sensor. An average power consumption of 0.909 W to 1.337 W was estimated and confirmed by a second, statistical-based approach. A first-order estimate of the startup power of (14.2 ± 7.4) W highlighted the necessity of spreading the configuration stage over multiple clock cycles.

During rail analysis, two design iterations were compared, highlighting the importance of a well connected power network. Added metal connections allowed for a reduction in *IR* drop from 666.5 mV to 214.2 mV. Inspection of the most affected metal segments revealed that clusters cells with high power consumption are the main drivers of large *IR* drops. Similar improvements between design iterations were found in electromigration studies focusing on three different current metrics. Average currents – the worst violation of the initial design – showed an improvement from 9.061 I_{limit} down to 2.643 I_{limit}. The other two metrics, RMS and peak currents, are now both close to acceptable values.

The results presented, are the first estimates of dedicated power and rail studies of the digital periphery and therefor will be used as the groundwork for further design modifications and additional studies. Such future studies will include power and rail analyses based on simulations of the full design and dedicated power-up studies. Impacts of temperature and power deviations need to be studied to cover the whole operation range. Plans for low-voltage modifications of the BCID driver cells will have large impacts on the power consumption and thus need to be tested as well. The dynamic studies could be improved using more realistic models of matrix connections, having a major impact on peak currents seen in rail studies. Finally, first studies of the final OBELIX chip, including triplication and further design improvements, can be started once OBELIX-1 is submitted.

Once the first OBELIX-1 sensors arrive, testing using the DAQ system can start. By combining functions of the Python interface compact scan routines can be created to characterize and test the sensors.

Bibliography

[1] G. Aad, T. Abajyan, B. Abbott, J. Abdallah, S. Abdel Khalek, et al. Observation of a new particle in the search for the standard model higgs boson with the atlas detector at the lhc. *Physics Letters B*, 716(1):1–29, September 2012.

[2] S. Navas et al. Review of particle physics. *Phys. Rev. D*, 110(3):030001, 2024.

[3] Kazunori Akai, Kazuro Furukawa, and Haruyo Koiso. Superkekb collider. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 907:188–199, November 2018.

[4] Ansu Johnson. Precision measurements of weak interaction parameters at belle and belle ii, 2025.

[5] Demin Zhou, Kazuhito Ohmi, Yoshihiro Funakoshi, and Yukiyoshi Ohnishi. Luminosity performance of superkekb. *Journal of Instrumentation*, 19, February 2023.

[6] J. Kahn. The Belle II Experiment. In *CERN-BINP Workshop for Young Scientists in e+e- Colliders*, pages 45–54, 2017.

[7] T. Abe, I. Adachi, K. Adamczyk, S. Ahn, H. Aihara, et al. Belle ii technical design report, 2010.

[8] Hermann Kolanoski and Norbert Wermes. *Teilchendetektoren. Grundlagen und Anwendungen*. Springer, 2016.

[9] C.-H. Kim, Y. Unno, H.E. Cho, B.G. Cheon, S.H. Kim, I.S. Lee, et al. Trigger slow control system of the belle ii experiment. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 1014:165748, October 2021.

[10] The Belle II Collaboration. The belle ii experiment at superkekb – input to the european particle physics strategy, 2025.

[11] H. Aihara, A. Aloisio, D. P. Auguste, M. Aversano, M. Babeluk, et al. The belle ii detector upgrades framework conceptual design report, 2024.

[12] Christian Bespin Valero. *Characterization of the TJ-Monopix2 Depleted Monolithic Active Pixel Sensor for High-Energy Physics Experiments*. PhD thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, November 2024.

[13] M. Babeluk, M. Barbero, J. Baudot, T. Bergauer, F. Bernlochner, et al. The dmaps upgrade of the belle ii vertex detector. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 1064:169428, 2024.

[14] Ivan Caicedo, Jochen Dingfelder, Tomasz Hemperek, Toko Hirono, Fabian Hügging, et al. Improvement in the design and performance of the monopix2 based on radiation-hard pixel detectors with high-rate capability. In *Proceedings of the 31st International Workshop on Vertex Detectors (VERTEX2022)*, volume 42, page 011021. Physical Society of Japan, 2024.

[15] Harry Foster. 2022 wilson research group functional verification study. Technical report, Siemens Digital Industries Software, October 2022.

[16] Cocotb Project. Coroutine based co-simulation library (cocotb) github repository. https://github.com/cocotb/cocotb. Accessed: 10th April 2025.

[17] M. Daas, Y. Dieter, J. Dingfelder, M. Frohne, Giakoustidis, et al. Bdaq53, a versatile pixel detector readout and test system for the atlas and cms hl-lhc upgrades. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 986:164721, January 2021.

[18] Enclustra GmbH. Mercury KX2 FPGA Module. https://www.enclustra.com/en/products/fpga-modules/mercury-kx2/, 2025. Accessed: 2025-07-25.

[19] SiLab Bonn. Basil github repository. https://github.com/SiLab-Bonn/basil. Accessed: 12th April 2025.

[20] Maurice Garcia-Sciveres, Flavio Loddo, and Jorgen Christiansen. RD53B Manual. Technical report, CERN, Geneva, 2019.

[21] M. Garcia-Sciveres and X. Wang. Data encoding efficiency in binary strip detector readout. *Journal of Instrumentation*, 9(04):P04021–P04021, April 2014.

[22] Jan M. Rabaey. *Digital integrated circuits: a design perspective*. Prentice-Hall, Inc., USA, 1996.

[23] J. Lienig and G. Jerke. Embedded tutorial: Electromigration-aware physical design of integrated circuits. In *Proceedings of the 18th International Conference on VLSI Design (VLSID '05)*. IEEE Press, 2005.

[24] X. Xuan. *Analysis and Design of Reliable Mixed-Signal CMOS Circuits*. Phd. thesis, Georgia Institute of Technology, Atlanta, GA, 2004.

[25] Jens Lienig and Matthias Thiele. *Fundamentals of Electromigration-Aware Integrated Circuit Design*. Springer International Publishing, 2018.

[26] Freddy Gabbay and Avi Mendelson. Electromigration-aware architecture for modern microprocessors, 2021.

[27] Cadence Design Systems, Inc. *Cadence Voltus™ IC Power Integrity Solution*, 2025.

List of Figures

2.1 SuperKEKB schematic with Belle II detector . . . 4
2.2 Belle II cross-section . . . 5
2.3 Proposed VTX detector cross-section . . . 6
2.4 Schematic OBELIX structure . . . 7
2.5 Semiconductor band structure . . . 8
2.6 Working principle of a pn-junction . . . 8
2.7 DMAPS cross-section . . . 9
2.8 Analog readout chain within the pixels . . . 10

3.1 Comparison of simulation and hardware setup . . . 11
3.2 Simulation waveforms . . . 13
3.3 BDAQ53 readout board . . . 14
3.4 Trigger delay matching . . . 15
3.5 Trigger related functionalities of the DAQ system . . . 16
3.6 Addressing of pixels in the matrix . . . 17
3.7 Structure of the RX module . . . 18
3.8 Code structure of the DAQ system . . . 19
3.9 Cocotb test code structure . . . 19
3.10 Injection caused ghost hits . . . 20
3.11 Showcase of injection test patterns . . . 21

4.1 Development flow of a digital IC design . . . 23
4.2 Switching power of a cell . . . 25
4.3 Internal power of a cell . . . 26
4.4 Leakage power of a cell . . . 27
4.5 *IR* drop over a wire . . . 28
4.6 Consequences of electromigration . . . 29
4.7 Standard cell positioning and orientation between power and ground rails . . . 30
4.8 Multilayer design of a six metal layer integrated circuit . . . 31
4.9 Iteration of top level power stripes . . . 31
4.10 Central section of the digital periphery . . . 32
4.11 Reduced 16 double column design used in simulation . . . 33
4.12 Power consumption of different cell types during the late period . . . 34
4.13 Close up of cell-based total power consumption during the late period . . . 35
4.14 Power consumption of different cell types during the early period . . . 35
4.15 Close up of cell-based total power consumption and cell loads during the early period 36

4.16 Cell-based total power consumption during the late period 36
4.17 Cell-based total power consumption during the early period 37
4.18 Total power consumption based on statistical approach 37
4.19 Comparison of power distributions based on different approaches 38
4.20 Current waveforms during the early period 39
4.21 Current waveforms during the late period 40
4.22 C and RC model of the matrix 40
4.23 *IR* drops of the first design iteration 41
4.24 *IR* drops of the second design iteration 42
4.25 Metal connection with the highest average current 43
4.26 RMS current comparison 44
4.27 Average current comparison 45
4.28 Peak current comparison 45

List of Tables

4.1 Top five activity results from the vector analysis run 34
4.2 Largest electromigration results of both iterations 44