Challenges and Solutions to Next-Generation Single-Photon Imagers

THÈSE N^{O} 7136 (2016)

PRÉSENTÉE LE 19 AOÛT 2016

À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR
LABORATOIRE D'ARCHITECTURE QUANTIQUE
PROGRAMME DOCTORAL EN GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Samuel BURRI

acceptée sur proposition du jury:

Prof. J.-Ph. Thiran, président du jury
Prof. E. Charbon, directeur de thèse
Prof. J. Langowski, rapporteur
Dr X. Michalet, rapporteur
Dr R. Ischebeck, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2016

Für Ulrike

Zusammenfassung

Eine immer grösser werdende Menge an Anwendungen profitiert vom Detektieren und Zählen einzelner Photonen. Die Mehrheit dieser Anwendungen benötigt dabei möglichst hohe räumliche Auflösungen, zumeist mindestens ein Megapixel. Diese Dissertation beleuchtet die Schwierigkeiten, die im Zusammenhang mit hochauflösenden Kameras zur Zählung einzelner Photonen auftreten, in umfassender Weise.

Dabei behandelt diese Dissertation eine grosse Zahl leistungsmindernder Probleme, die auftauchen wenn die Zahl der Pixel über ¼ Million steigt. Zur Diskussion dieser Probleme benutzt man den Begriff der Nicht-Uniformität (non-uniformity). Dazu werden Techniken zur Charakterisierung der Nicht-Uniformität verschiedener Parameter und mögliche Lösungen im Zusammenhang mit daraus resultierenden Problemen aufgezeigt. Diese Techniken sollen dabei helfen Nicht-Uniformität zu erkennen und damit verbundene Leistungseinbussen zu beheben.

Zur Bewertung der in dieser Arbeit vorgestellten Techniken wurden zwei Kameras hergestellt, die für die Detektion einzelner Photonen geeignet sind.

Die erste Kamera, SwissSPAD, besteht aus einem Sensor mit 512 x 128 Pixeln mit Dioden, die einzelne Photonen detektieren. Jeder Pixel hat darüber hinaus ein 1-Bit Speicherelement und einen Verschlussmechanismus der Verschlusszeiten bis hinunter zu 5 Nanosekunden erlaubt. Dabei hat der Verschluss eine hohe Uniformität für alle Pixel und kann mit einer Präzision im Bereich weniger Pikosekunden relativ zu einem Referenzsignal ausgelöst werden. Zusammen mit einer schnellen Datenübertragung mit einer Rate von über 10 Gigabit pro Sekunde ermöglicht dieser Verschluss eine Beschleunigung der Messzeiten in Anwendungen der Fluoreszenz-Lebenszeit-Mikroskopie sowie der Fluoreszenz-Korrelations-Spektroskopie um mehrere Grössenordnungen. Andere mögliche Anwendungen betreffen die räumlich hochauflösende Aufzeichnung von Flugzeiten einzelner Photonen und die Erzeugung von Zufallszahlen in hoher Frequenz. Kürzlich hat auch eine Anwendung super-auflösender Mikroskopie vom SwissSPAD-Sensor Gebrauch gemacht.

Die zweite Kamera, LinoSPAD, wendet die mit SwissSPAD gewonnenen Erkenntnisse an, wobei der Sensor auf ein Minimum an Funktionalität reduziert wird. Im Gegenzug enthält die Kamera eine leistungsfähigere Verknüpfung des Sensors mit einem FPGA, einem flexibel konfigurierbarem Schaltkreis. Dadurch wird es ermöglicht, den Sensor auf seine wichtigste Eigenschaft, die Photoneneffizienz, zu optimieren. So besteht der Sensor aus nur einer Zeile von Photonen detektierenden Dioden. Dafür ist aber jede Diode direkt mit dem FPGA verbunden, in welchem die komplexe Verarbeitung der Information über Photonen stattfinden kann. Zur Veranschaulichung der Möglichkeiten aktueller, preisgünstiger FPGAs haben wir eine Reihe hochauflösender Module (time-to-digital converters) integriert, um die Ankunftszeiten einzelner Photonen auf dem Sensor zu messen. Dieses System erlaubt die Erfassung einzelner Photonen und das Aufzeichnen ihrer Ankunftszeit in Histogrammen mit einer Rate von bis zu 8.5 Milliarden Ereignissen pro Sekunde. Unsere Kamera, in Kombination mit einer Laserdiode und einem Schaltkreis zur Erzeugung ultrakurzer

Lichtpulse im Pikosekundenbereich, ergibt eine vielfältig einsetzbare 3D Kamera basierend auf dem Prinzip der Laufzeitmessung von Photonen.

Diese Dissertation soll ein erster Schritt sein auf dem Weg zu einem Sensor mit einer Million Pixel, die auf einzelne Photonen detektierenden Dioden basieren. Wir glauben, dass ein solcher Sensor dank den in dieser Dissertation vorgestellten Möglichkeiten von Architektur und Verschaltung einer grossen Anzahl von Photonen detektierenden Dioden in naher Zukunft realisiert werden kann. Ausserdem glauben wir, dass die Anwendungen, die in dieser Dissertation präsentiert werden, Hinweise geben auf eine grosse Zahl zukünftiger Anwendungen, in welchen die hergestellten Kameras bisherige Detektionssysteme gewinnbringend ersetzen können.

Schlüsselwörter

Einzelphotonendiode, SPAD, CMOS, Kamera, TDC, Phänomene im Bereich der Pikosekunden, Photonen

Abstract

Detecting and counting single photons is useful in an increasingly large number of applications. Most applications require large formats, approaching and even far exceeding 1 megapixel. In this thesis, we look at the challenges of massively parallel photon-counting cameras from all performance angles.

The thesis deals with a number of performance issues that emerge when the number of pixels exceeds about ¼ of megapixels, proposing characterization techniques and solutions to mitigate performance degradation and non-uniformity.

Two cameras were created to validate the proposed techniques.

The first camera, SwissSPAD, comprises an array of 512 x 128 SPAD pixels, each with a one-bit memory and a gating mechanism to achieve 5ns high precision time windows with high uniformity across the array. With a massively parallel readout of over 10 Gigabit/s and positioning of the integration time window accurate to the picosecond range, fluorescence lifetime imaging and fluorescence correlation spectroscopy imaging achieve a speedup of several orders of magnitude while ensuring high precision in the measurements. Other possible applications include wide-field time-of-flight imaging and the generation of quantum random numbers at highest bit-rates. Lately super resolution microscopy techniques have also used SwissSPAD.

The second camera, LinoSPAD, takes the concepts of SwissSPAD one step further by moving even more ‘intelligence’ to the FPGA and reducing the sensor complexity to the bare minimum. This allows focusing the optimization of the sensor on the most important metrics of photon efficiency and fill factor. As such, the sensor consists of one line of SPADs that have a direct connection each to the FPGA where complex photon processing algorithms can be implemented. As a demonstration of the capabilities of current low-cost FPGAs we implemented an array of time-to-digital converters that can handle up to 8.5 billion photons per second, measuring each one of them and accounting them in high precision histograms. Using simple laser diodes and a circuit to generate light pulses in the picosecond range, we demonstrate a ubiquitous 3D time-of-flight sensor.

The thesis intends to be a first step towards achieving the world’s first megapixel SPAD camera, which, we believe, is in grasp thanks to the architectural and circuital techniques proposed in this thesis. In addition, we believe that the applications proposed in this thesis offer a wide variety of uses of the sensors presented in this thesis and in future ones to come.

Keywords

CMOS, SPAD, Pixel, Camera, FPGA, TDC, picosecond phenomena, Photons

Contents

Chapter 1 Introduction

After a brief statement of the objectives for this work, this chapter aims to present the history and theoretical foundations behind single-photon image sensors, focusing especially on single-photon avalanche diodes (SPADs) and their applications. On the application side, the focus is on time-correlated single-photon counting (TCSPC) techniques, which make use of the excellent timing capabilities of SPADs.

1.1 Objectives of the thesis

The cameras ubiquitous today all around us in robotic vision, surveillance and smartphones are designed to reproduce the macroscopic observation of the environment as seen by the human eye. To achieve an authentic rendering of the environment a large amount of information related to the microscopic and quantum domain of the environment can be discarded. The most challenging aspects for this type of cameras become solely the issues of sensitivity and noise summarized in the signal-to-noise ratio SNR.

After the foundation of quantum mechanics, it became clear that the mechanisms governing our environment cannot be explained on the basis of macroscopic observation alone. The most important restriction in classical imaging approaches is the limited time discrimination, which precludes the observation of very fast phenomena that take place in the nano- and picosecond domains and below. Examples of such fast phenomena caused by the interaction of light particles, photons, and matter are various types of fluorescence where atoms or molecules are excited through the absorptions of photons and subsequent relaxation occurs usually on the time scale of hundreds of picoseconds to a few nanoseconds, a time known as lifetime.

An ideal camera would observe the full objective quantum state of the photons reaching its sensor. We know that this is not possible from Heisenberg's uncertainty principle affecting quantum observations and placing a limit on the accuracy with which the quantum state of a photon can be known. To add a small step on the path to a practical quantum camera is the main objective of this thesis.

Theoretical foundations have been researched and discussed to the point where practical, easy-to-use, and easy-to-understand cameras need to be made available to researchers and the public to give them tools to gain a better understanding of their topic of research and the world around them.

With this goal in mind, the research in this thesis started with the analysis of current single-photon sensitive imaging systems and their limitations, which led to the identification of the primary performance parameters in need of improvement. On the outset of this thesis, these were the camera resolution in both, time and space; while later data transfer speed and sensor flexibility became increasingly important.

The parameters where a SPAD based camera really differentiates itself from other camera technologies are: 1) the achievable speed or frame rate, essentially limited by the rate at which photons can be detected; 2) the time resolution, again, for single frames or single photons depending on architecture; 3) the absence of readout noise thanks to the early digitization and fully digital architectures, and 4) the compatibility with standard CMOS fabrication processes that allow high levels of integration.

Single-photon detection is used in a wide range of areas where generally the available photon flux is low or where photon arrival timing plays an important role, or both. Examples are detectors for fiber optic communication, astrophysics, quantum computing, medical imaging, range finding, and materials science.

This chapter presents the state-of-the-art in the field of single-photon imaging and the following two chapters present the cameras developed and used for this work. Chapter 2 describes SwissSPAD, a 512 × 128 pixel wide-field SPAD image sensor with global gating and fast readout, Chapter 3 presents LinoSPAD, a 256 × 1 pixel SPAD line sensor with reconfigurable pixel architecture thanks to the close integration with a FPGA. 0 discusses challenges and possible solutions for non-uniformity issues encountered in higher resolution SPAD imagers, and Chapter 5 presents practical results obtained with our sensors.

1.2 Photon-counting cameras

Chronologically speaking, the first photon-counting cameras were based on photo-multiplier tubes (PMTs) and micro-channel plates (MCPs). In recent years, with the creation of high quality charge-coupled devices (CCDs) and complementary metal-oxide semiconductor (CMOS) sensors, photon-counting has become possible in these technologies, while speed still is a limitation due to the inherent readout architectures. SPAD imagers, on the contrary, offer single-photon sensitivity with unmatched timing resolution and readout speed.

1.2.1 Photo-multiplier tubes

In the late 19th and early 20th century the photoelectric effect [1,2,3] and quantization of light in a discrete number of photons was discovered and explained. It did not take very long before the photo-multiplier tube (PMT) was invented as the first device that could be used to detect single photons. In the West, its invention is generally attributed to V. K. Zworykin et al. [4]; though, recent discussions seem to recognize that the invention was made in Russia by L. A. Kubetsky whom Zworykin visited a short time before his famous publication [5].

Figure 1.1 shows the schematic of a PMT with an optional scintillator to increase the detection efficiency for high-energy photons or particles. A PMT is a vacuum device multiplying the number of single electrons freed from a photo-cathode by impact ionization on a series of accelerating dynodes where multiple secondary impact ionizations take place. The total amplification factor typically between 10^3 and 10^6, makes it possible to sense single photon events.

The quantum efficiency of the photo-cathode plays an important role as only the fraction of photons that is converted into electrons can be detected. A scintillator placed above the photo-cathode can be used to generate multiple lower energy photons from single high-energy photons or particles greatly enhancing the chance of detection in these applications.

PMTs have excellent noise and jitter performance making them useful in many applications where extremely low photon counts need to be detected, like for example in astrophysics. The largest market for PMTs with scintillators is nuclear medicine with PET and SPECT cameras. The downside of PMTs is the requirements for high voltages to drive the amplification, the bulky size, and their sensitivity to magnetic fields.

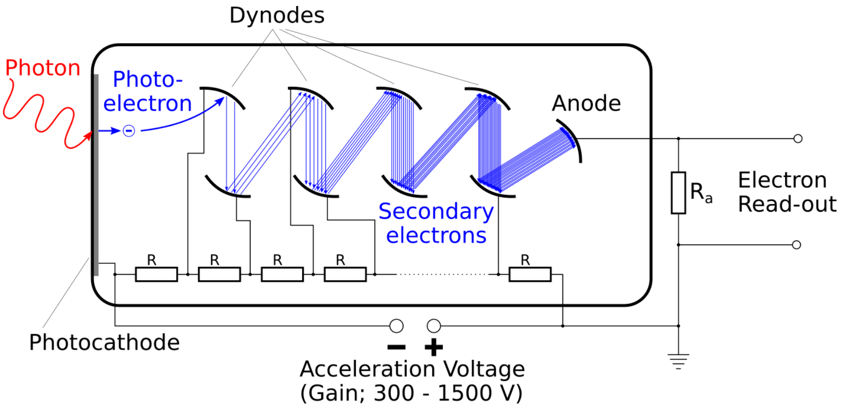


Figure 1.1: Schematic of photo-multiplier tube. A photon is converted into a primary electron on the photocathode, which then generates a larger number of secondary electrons when it impacts the dynodes after acceleration through a high electric field. At the anode, the electron current is multiplied by 10^3-10^6 depending on the number of amplification stages and electric field strength. A scintillator in front of the photocathode can be used to detect high-energy photons or other particles by converting them into multiple visible photons. (Original image by J. Krieger; public domain.)

As PMTs have been available for many decades, they have been extensively studied. An exhaustive description of these studies is beyond the scope of this thesis. We refer instead to a recent edition of the freely available book “Photomultiplier tubes: Basics and Applications” by Hamamatsu [6].

1.2.2 Micro-channel plates

Micro-channel plates (MCPs), also known as micro-channel plates, can be seen as an extension to PMTs that allows retaining the position information of the photon impinging on the photo-cathode such that an image can be reconstructed based on the detected photons. The MCP is an extension for the first image intensifiers, the development of which began around the time when the first PMTs were developed. The image intensifier uses a phosphor screen behind a photo-cathode and a large electric field between the two accelerates electrons such that multiple photons are produced from the high energy with which they impact on the phosphor crystals. The first generation of such devices, working with active infrared illumination, where already used in World War II [7,8].

The MCP operation is perhaps best explained by M. Lampton [9]:

“It is a glass wafer, perforated by millions of electron-multiplying tubes, resembling a compound eye. It can transform a dim pattern of electromagnetic radiation into a brightened, pointillist image […]”.

Figure 1.2 shows a schematic of a MCP image intensifier. The incident photons hit the photo-cathode where they produce an electron that is accelerated and multiplied in a channel behind the cathode, before being converted back into photons on a phosphor screen. Many variations and combinations of the basic principles exist, such as multiple stages or direct position sensing of the electrons at the output using special anodes. A notable example is the H33D detector using of a stack of 3 MCPs followed by position-sensing anodes [10]. Nowadays instead of using the screen directly to display the intensified image, it is coupled to an image sensor either via optics or by using a fiber optic bundle.

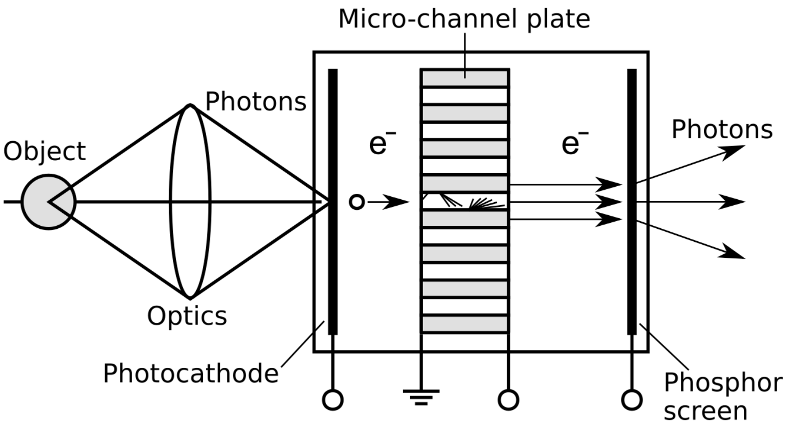


Figure 1.2: Schematic of MCP image intensifier. A photon is converted into a primary electron on the photocathode that is accelerated in an electron multiplying tube. The electron shower at the output of the MCP retains the position information of the original photon. When the electrons hit the phosphor screen, they are converted back to a higher number of photons. (Image in public domain.)

Image intensifiers can be time-gated to high temporal resolution. The main advantage of MCP photodetectors is their low noise comparable to that of PMTs and their higher spatial resolution. The main drawback is again the need for high voltages for operation, which makes it difficult to integrate them with low voltage electronics.

Additional information can be found in [11]. MCPs have somewhat merged into PMTs giving rise to different combinations and often creating confusion in the terms.

1.2.3 Charge-coupled devices

Willard S. Boyle and George E. Smith invented the charge-coupled device (CCD) in 1969 as a nonvolatile memory [12]. The inventors were awarded the Nobel Prize in Physics in 2009 for the invention, while Michael F. Tompsett is credited for extending the use of CCDs to imaging applications. CCDs have been dominating the imaging market until recently, when they have been gradually supplanted by CMOS image sensors. So much so, that in 2015 market leader Sony announced that they would halt (conventional) CCD production by 2017 [13].

Three steps are needed to operate a CCD image sensor: 1) collect photons and convert them to electrical charge 2) transport the charge (or information) from the point of collection to the sensing elements 3) measure the transported charge, converting it into an image for display or storage.

The CCD solves these basic steps in a simple yet ingenious manner. The basic idea is to collect charge carriers generated from photon absorption in a potential well in each pixel formed by an appropriately doped and biased semiconductor area creating a MOS capacitor. After collection, the charge packets are transported from one well to the neighboring by applying a voltage sequence to shift potential wells. Once the charge reaches the end of the transport chain, it is converted in a so-called floating diffusion, and, later, into a digital signal for display or storage.

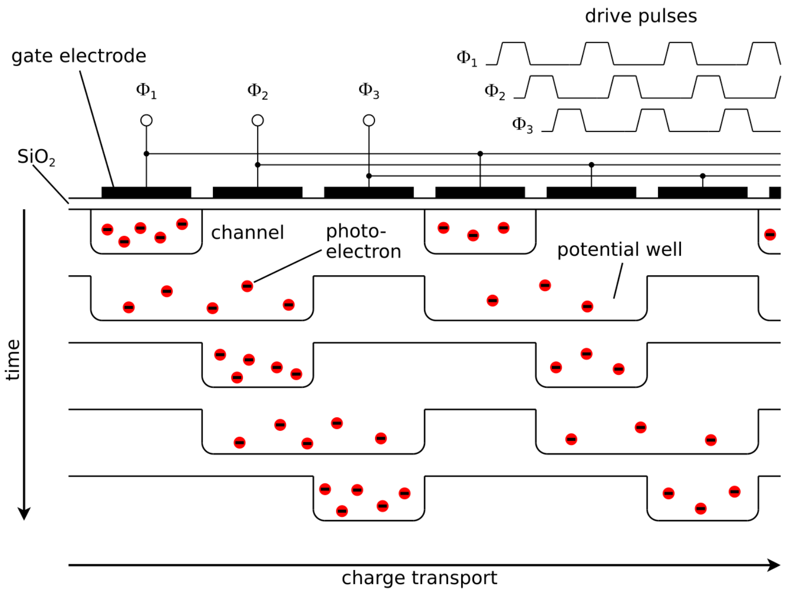


Figure 1.3: Charge transport illustration for a 3-phase CCD. The charge in the well is transported by sequencing the control voltages of adjacent gates. (Image inspired by [14].)

Figure 1.3 illustrates the charge transport mechanism in a CCD. Charge packets are moved from one place to the next by applying a sequence of gate control signals, essentially creating moving potential wells. The simplified architecture of a linear device with surface channels shown here was soon improved upon, with buried channels for increased charge transfer efficiency and better noise characteristics. Backside illumination was introduced to reach a fill factor of 100% and cooling was used from the beginning in scientific CCDs for additional noise reduction.

In 1976, image sensor development driven by the requirements of the large space telescope (LST) later named Hubble Space Telescope (HST) resulted in a 800 × 800, buried channel, backside-illuminated device with polysilicon gate [15] much like the CCDs widely used in image sensors until recently when they were replaced with additionally enhanced CCD or CMOS.

A quite natural development with the CCD was to couple it with an image intensifier employing an MCP and thereby giving birth to the intensified CCD (ICCD, [16]). The combination of a vacuum device operating at high voltages and a solid-state device was not optimal though as the image artifacts of MCPs were now imported to the CCD. The inherent possibility of gating the intensifier was on the other hand very welcome since more time can be used to read a CCD without smearing when a shutter blocks light during this time.

Smearing appears in CCDs and other image sensors when the light accumulation time on individual pixels is not the same for the whole sensor. It can be prevented easily by using a shutter to block light from reaching the sensor while it is read out. This can become impractical for larger sensors and reduces the photon efficiency by the fraction of time used for the readout process. In CCDs, the problem was addressed with interline- and frame-transfer sensors, which provide a second charge storage space for each pixel, where the image is quickly transferred to before being read out, thus preventing image smear. The price to pay for this solution is increased chip area and, in the case of the interline CCD, reduced fill factor.

Sometime later, the vacuum device in the ICCD was replaced with a solid-state solution again, like the sensor itself, when the electron-multiplying CCD (EMCCD, [17]) was invented. In an EMCCD, image amplification takes place during special multiplication stages in the charge transfer process. By using higher electrical fields, impact ionization is induced multiplying the charge collected by the CCD. EMCCD were almost immediately a viable alternative to ICCDs and replaced them quickly where no gating was needed.

1.2.4 Active pixel sensors in complementary metal-oxide semiconductors

While the photon collecting pixels in CCDs are passive elements, an active pixel sensor (APS) is based on active, amplifying pixel circuits. When an APS is realized using common complementary metal-oxide semiconductor fabrication processes it is classified as CMOS APS or CMOS image sensor (CIS). Active pixel sensors were actually realized before CCD sensors, but could not be made to work well because of too large parameter variations and were quickly overtaken by CCDs. CIS eventually reappeared as low quality image sensors that could be implemented alongside other electronic and it was only from 1993 [18] onwards that large-scale image sensor development in CMOS was undertaken, leading to the image sensors now ubiquitous in everyday devices and to the announcement of Sony to stop CCD production.

Figure 1.4 shows the CMOS active pixel in its simplest configuration using three transistors. A photodiode generates electrical charges in a capacitance formed by its own junction and the gate of M_{sf}. After an accumulation period has elapsed, M_{sel} is activated by the ROW selection signal connecting the pixel to readout line COL to sense the signal. Once the signal has been recorded, the pixel charge is emptied through M_{rst} by activating the RST signal.

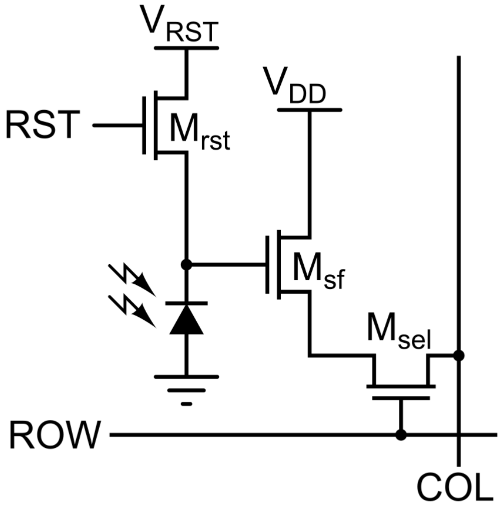


Figure 1.4: Circuit schematic of a 3T CIS pixel as in [19]. A photodiode generates electrical charge collected in the capacitance of its junction and the gate of the source follower transistor M_{sf}. M_{sel} is activated by the ROW signal and used to connect M_{sf} to the readout line COL to sense the signal induced by the collected photo-charge. RST is used to switch on M_{rst} to reset the photodiode capacitor for the next charge integration period.

The most important advantage of CIS compared to CCD is their immediate compatibility with electronic circuits on the same substrate close to the sensor. Without the need to modify the fabrication process to build the readout electronics, this causes CIS to be low-cost imagers. Additionally, CIS can directly profit from the advancements of CMOS fabrication processes with shrinking feature sizes and lower power operation.

The inclusion of active devices in each pixel of an image sensor has the drawback of reducing the active area, and thus the fill factor of the sensor. A low transistor count is therefore preferred, especially for fabrication processes with larger feature sizes. The wiring needed to connect the pixels also absorbs a portion of the incident light for front-illuminated sensors. Microlenses on the pixels can be employed to reclaim insensitive area due to electronics.

Typically, the 3T pixel is read by cycling on the rows of the sensor producing an image line by line (rolling shutter mode). This process creates distortions in moving objects because the integration period is not the same, but it is shifted from one row to the next. An additional transistor can be used to solve this issue and introduce a global gating (global shutter mode).

Important progress in CMOS APS noise performance has been made with the introduction of correlated double- and multi-sampling techniques through which CMOS performance began to rival that of CCD [19]. The main technology required to perform CDS, is the pinned photodiode, invented by Nobukazu Teranishi [20,21], that enables the transfer of all charges accumulated during exposure to a temporary analog storage unit, known as floating diffusion, which transforms charges into a voltage that can be read out independently. Pinned photodiodes eliminated charge sharing during readout, while allowing low parasitic capacitance at the photodiode, thereby ensuring high conversion gain.

A more recent development is the introduction of scientific CMOS (sCMOS), which incorporates multiple advanced techniques resulting in CIS with performance comparable to the most advanced CCD and EMCCD cameras suitable for scientific applications [22]. Hybrid approaches, bump-bonding a CCD substrate to a CMOS carrier with readout electronics were also studied [23] for some time but introduced more problems in the fabrication than were solved through advancements in purely CMOS designs.

At this time, there are still large investments being made in the research of CIS coupled in part with generic research in advanced CMOS. There are even CCD like structures now implemented purely in CMOS [24]. This brings us to the main topic of this thesis that is single-photon avalanche diodes in standard CMOS that will compete with the classical CIS.

1.3 Single-photon avalanche diode and imager

From the CMOS image sensor and the basic 3T pixel shown in Figure 1.4 a CMOS single-photon avalanche diode (SPAD) based sensor can be understood as a CMOS image sensor with a SPAD in the place of the usual photodiode. This section introduces SPAD based image sensors and the associated terminology used throughout this work.

The SPAD is a silicon diode made like most diodes from a p-n-junction. The avalanche in its name comes from the operating regime of the diode when a large negative bias voltage is applied. In that regime, the electric field in the diode is large enough that a single electron traveling through the diode gains enough energy to free additional electrons on impact and an avalanche of electrons is observed. The single-photon in SPAD then comes from the fact that one single photon is enough – with some probability – to trigger the avalanche of electrons in the device, causing a macroscopic current to be observed when the photon is absorbed by the SPAD.

1.3.1 Operation of a SPAD

A silicon single-photon avalanche diode is a p-n junction biased above its breakdown voltage and connected to an avalanche quenching and recharge circuit. Figure 1.5 a) shows the simplest such circuit composed of a SPAD, a voltage source and a ballast resistor. Part b) shows the current/voltage characteristic of the SPAD and defines the three operating regimes: breakdown, reverse and forward bias. The voltage at which the diode becomes conductive between the reverse bias and breakdown regime is the diode's breakdown voltage. Parts c) and d) of Figure 1.5 show a SPAD operating cycle and the anode voltage during one photon detection respectively. The duration of a detection cycle is on the order of tens to hundreds nanoseconds.

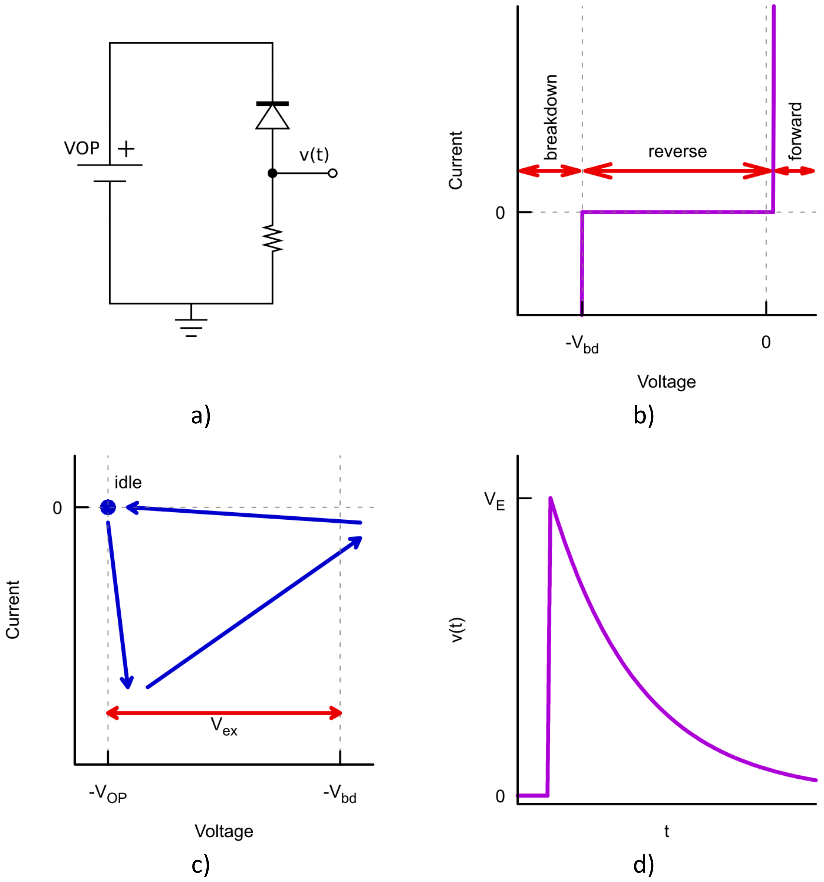


Figure 1.5: SPAD circuit and operation. a) shows the simplest SPAD circuit for single-photon detection, b) shows the current-voltage characteristic of the SPAD and its operating regimes, c) the idle operating point and one detections cycle and d) the anode voltage v(t) during one detection. (Adapted from [25].)

The SPAD will sustain the voltage above breakdown over its terminals and remain in idle state as long as there are no free charges in the region of high electric field that could start an avalanche. A photon that is absorbed in this region and converted into an electron-hole pair causes the now separated charges to be rapidly accelerated and multiplied, creating a current through the device. As the current through the device and ballast resistor rises so does the voltage over the resistor according to Ohm's law. This leads to a reduction of the voltage on the terminals of the SPAD, bringing it to the operating point where the electron avalanche is no longer sustained and multiplication stops. The free charges in the diode are then evacuated through the resistor and the voltage over the SPAD restored when it returns to the idle operating point.

Upon photon absorption in the multiplication region of the SPAD and the onset of the avalanche, the voltage across the ballast resistor rises sharply indicating the arrival of a photon. The signal at the anode of the SPAD is directly digitized using a comparator, in the simplest case a CMOS switch or inverter.

The rising edge of the voltage curve indicates precisely the moment when a photon was detected and a timing circuit connected to the digitized signal of the SPAD is used to obtain the photon arrival time in time-correlated single-photon counting applications. In applications where timing information is not needed, the signal is used to increment a photon counter or is directly routed off the sensor.

1.3.2 Fabrication of SPADs

Silicon SPADs are implemented in two main styles. The first style uses a p+ - π-p-n structure [26] and is known as reach-through APD (RAPD) or vertical APD from the extension of the depleted region spanning from anode to cathode vertically through the structure. Thanks to its deep junction the multiplication region is further away from the surface and the RAPD reaches high quantum efficiency up to the absorption limit of silicon around 1.1 µm. The drawback of the deep absorption region is higher timing uncertainty due to the charges drifting until they reach the multiplication region.

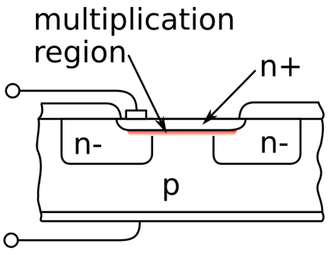


Figure 1.6: Early planar SPAD. A planar SPAD is formed in a CMOS process using a n+ implantation in a p-type substrate. A guard-ring is formed from n- implants. (Image adapted from [27].)

The second implementation style yields shallower structures fabricated in silicon by implanting p and n layers in CMOS compatible processes to form the diodes. The group around Cova have been particularly active in the research of these devices since the 1970s. They describe a variety of structures [28], which were already used to measure fluorescence decay-time constants [27]. The common denominator of these planar structures is a region of high electric field where avalanche multiplication occurs that is surrounded by a guard ring preventing excessive field strength at the extremities to ensure uniform breakdown voltage and sensitivity across the device.

Figure 1.6 shows an early planar SPAD as described by Haitz [29] and implemented by Cova [27]. The multiplication region is formed between the shallow n+ implantation and the p-type substrate and a deeper n- guard ring is used to achieve uniform electric field. In more recent SPAD structures, all contacts are made from the top using implants unlike shown in the image.

The first SPADs were not fabricated in processes also used for electronic circuits but rather were singular devices or small arrays that had to be connected to interfacing electronic for biasing, avalanche detection and quenching. It was with the integration of SPADs in standard CMOS combined with fully digital readout electronics that they became approachable for more applications and the realization of larger arrays became possible.

The first SPADs implemented in a fully standard CMOS fabrication process was shown by Rochas et al. in 2003 [30] and was made in an 800 nm twin tub technology. After this breakthrough, novel SPAD architectures in smaller technologies and integrated arrays followed in rapid succession. SPADs in 350 nm appeared in 2006 [31], in 180 nm with shallow trench isolation (STI) also in 2006 [32], in 130 nm in 2007 [33], in 90 nm in 2010 [34] and in 65 nm in 2013 [35].

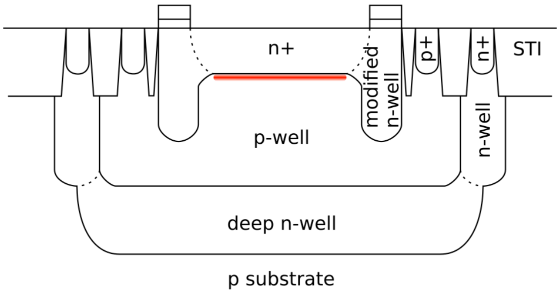


Figure 1.7: SPAD in 65 nm standard CMOS. The multiplication region is defined by the n+ - p-well junction and is surrounded by an n-tub guard ring. (Image redrawn from [35].)

Figure 1.7 shows one of the latest SPADs in standard CMOS integrated in a 65 nm technology [35]. The basic structure explained above is still present with the multiplication region surrounded by a guard ring to prevent premature edge breakdown (PEB). Today SPADs exist in a wide variety of implementations, within n-type or p-type substrates, with shared or isolated wells and with shallower or deeper junctions to optimize certain of the typical characteristics explained in section 1.3.4.

Specially modified CMOS fabrication processes are also investigated for SPADs often targeting better noise performance compared to the standard processes not optimized in this regard. A hybrid approach combining different processes for the SPAD and electronics is also possible at the expense of fabrication complexity.

1.3.3 Integrated quenching and recharge for CMOS SPADs

When SPADs are implemented in CMOS, they need to be interconnected such that the desired bias voltage can be applied over the SPAD and that avalanches can be sensed and processed. Figure 1.8 shows three examples of SPAD front-end circuits differentiated by the way, in which the SPAD bias is restored after an avalanche event. Figure 1.9 shows the associated waveforms for the SPAD anode voltage during a detection cycle.

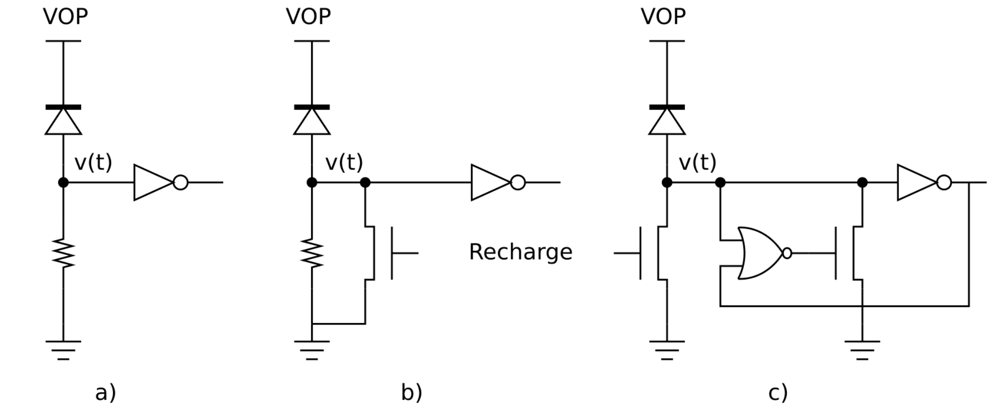


Figure 1.8: SPAD quenching and recharge examples. The SPADs are operated at a bias voltage of $V_{\mathrm{OP}} = V_{\mathrm{bd}} + V_{\mathrm{ex}}$ above the breakdown voltage. All circuits use passive quenching stopping the avalanche when the voltage over the SPAD drops below the breakdown. Circuit a) shows passive recharge, b) shows single-slope active recharge and c) shows double-slope active recharge [36].

The simplest circuit shown in a) uses a passive recharge. When an avalanche occurs, it is quenched when the voltage over the resistor reaches V_{ex} and brings the voltage over the SPAD below its breakdown voltage. Once the avalanche is stopped, the bias over the SPAD is restored to V_{OP} as the avalanche charges drain through the resistor. The circuit in b) adds a simple active recharge in the form of a transistor acting as a current source that recharges the SPAD using a constant current after an avalanche. A more advanced recharge mechanism is shown in c) and named double-slope active recharge [36]. A transistor configured as current source keeps the SPAD biased for detection and starts a slow recharge after an avalanche. A larger transistor accelerates the recharge on a voltage interval chosen through careful sizing of the transistors constituting the inverter and NOR-gate.

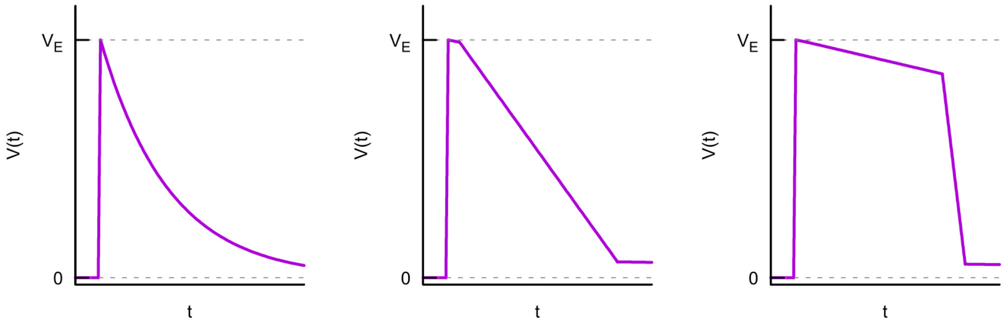


Figure 1.9: SPAD anode voltage for different recharge implementations. The waveforms show the SPAD anode voltage for a detection cycle using passive, single-slope active and double-slope active recharge. Typical recharge times are tens of nanoseconds.

The resistors shown in recharge implementation a) and b) are implemented using MOS transistors. Active recharge schemes offer the advantage that the dead time of the SPAD can be better controlled for more uniform detection and timing response in an array, especially under higher illumination. The double-slope recharge offers the added advantage of reducing afterpulsing by giving the avalanche charges more time to evacuate the SPAD before restoring the bias quickly to return to the idle state ready for the next detection.

Active quenching schemes can also be implemented by adding a switch over the SPAD and using it to reduce the SPAD bias to a set voltage below the breakdown quicker than through the avalanche alone. Active quenching and recharge schemes can be effectively combined into a shutter for the SPAD to define the photosensitive time window with great accuracy.

1.3.4 SPAD metrology

Now it is time to introduce the basic terms of SPAD metrology used throughout the remainder of this thesis in the characterization and comparison of the fabricated SPAD arrays. A much more detailed introduction to the metrology of SPADs can be found in [37].

Breakdown, excess and operating voltage

Probably the most important and defining parameter for an avalanche diode is its breakdown voltage V_{bd}. The breakdown voltage of a SPAD is the voltage above which impact ionization occurs and current multiplication takes place when free charges are present in the diode. The breakdown voltage of an integrated device is measured from the I/V characteristic or, for digital outputs, using its dark count rate and sweeping the bias voltage.

The operating voltage of a SPAD corresponds to the voltage at which the SPAD is in idle state. The operating voltage and the breakdown voltage are related through the excess bias voltage by:

$$V_{\mathrm{ex}} = V_{\mathrm{OP}} - V_{\mathrm{bd}} \tag{1.1}$$

Dark count rate (DCR)

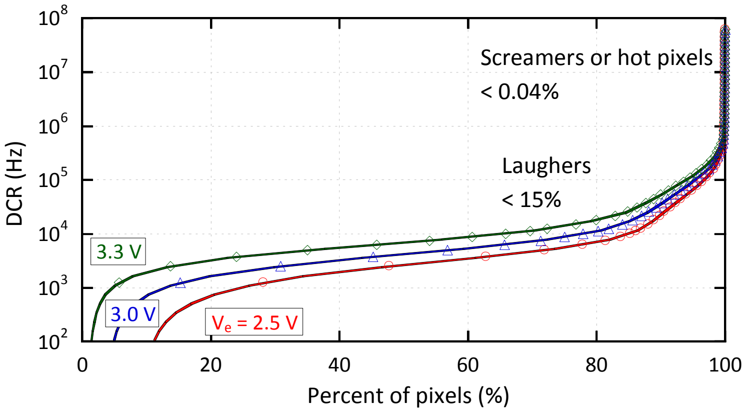


Figure 1.10: Cumulative DCR as a function SPAD population. (Courtesy: Yuki Maruyama, JPL, Pasadena, U.S.A.)

The dark count rate (DCR) of a SPAD measures its noise in the unit of avalanches per second without any incident photons. Dark counts are a result of thermal processes in the SPAD junction and of tunneling effects. Thermal DCR can be reduced by operating the SPAD at a lower temperature such that DCR becomes dominated by tunneling effects. For a given SPAD size and fabrication technology, DCR depends on

V_{ex} and temperature. For an array, it is typically reported as cumulative distribution to show the fraction of hot pixels with very high DCR, as shown in Figure 1.10. Hot pixels are due to defect sites in the silicon, where hot SPADs are recognized as devices with more than two orders of magnitude in DCR above the median and laughers more than one.

Photon detection probability (PDP)

The photon detection probability (PDP) of a SPAD is defined as the probability with which a photon hitting the multiplication region of the SPAD generates an avalanche. PDP is defined as follows:

$$PDP = \Pr(avalanche\,)\,QE, \tag{1.2}$$

where the quantum efficiency (QE) of a SPAD depends mostly on the wavelength of incoming photons and Pr(*avalanche*) depends on the excess bias voltage applied to the SPAD. PDP is commonly reported as a function of wavelength, where planar SPADs with shallow junction depth as used in this thesis typically show highest sensitivity between 400 nm and 500 nm. Figure 1.11 shows a typical PDP from a SPAD published by Chockalingam Veerappan as a function of wavelength and excess bias voltage (a). Note the PDP compression at high excess bias voltages (b). This compression is useful in the case of large arrays of SPADs, as it reduces the sensitivity of PDP on variations in the breakdown voltage, thereby enabling high uniformity across large arrays.

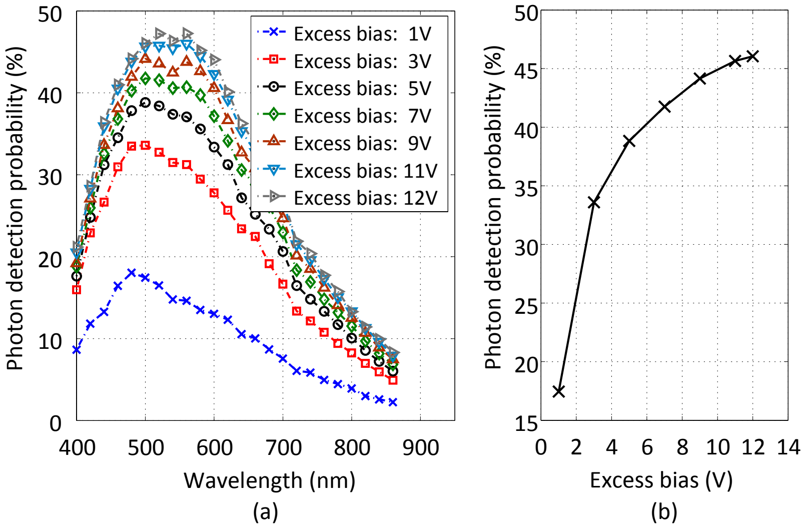


Figure 1.11: (a) PDP as a function of wavelength and excess bias voltage. (b) Peak PDP as a function of excess bias voltage. (Courtesy: Chockalingam Veerappan, [38,39])

Fill factor and photon detection efficiency (PDE)

When SPADs are implemented as arrays the sensitive area of the detector, the combined multiplication regions, cannot reach 100% due to non-sensitive guard rings and in-pixel electronics. The fill factor is defined as the ratio between the active area in an array of SPADs and the total area occupied by the array. Using the fill factor the photon detection efficiency (PDE) is then defined as:

$$\mathrm{PDE} = \mathrm{Fill\ factor} \times \mathrm{PDP}. \tag{1.3}$$

In general, fill factor is a function of area, since larger pixels can typically achieve a better fill factor. This is due to the overhead of SPADs in terms of the geometries required for preventing edge breakdown and to reduce crosstalk.

Dead time

After an avalanche occurs in a SPAD, the bias drops below the breakdown voltage and another photon cannot be detected before the bias is restored above breakdown. This time, during which the SPAD is not sensitive to photons and is effectively dead, is called dead time. In passive recharge, the SPAD insensitivity is not absolute and, even during recharge, the SPAD has a reduced but non-zero PDP. This makes it difficult to determine an exact dead time. In active recharge, on the contrary, the SPAD is held off for a precise time, thus the dead time becomes precise and programmable. The dead time, T_{dead}, limits the maximum rate at which photons can be detected, or maximum photon flux PF_{max} in a SPAD by imposing a minimum time between consecutive detections, whereas in active recharge the maximum photon flux a SPAD can detect is:

$$PF_{\text{max}} = 1/PDP \times T_{\text{dead}}. \tag{1.4}$$

In passive recharge, PF_{max} is divided by e, due to the partial PDE effect. In addition, the counts reduce upon reaching PF_{max}, because photon detections become fused in time, not enabling one to distinguish individual events. This effect, compared to the behavior of active recharge, is shown in Figure 1.12. Typical dead times are in the tens to hundreds of nanoseconds. Even though shorter dead times have been reported, it is usually preferable to keep the dead time relatively long, in order to minimize parasitic effects, such as afterpulsing.

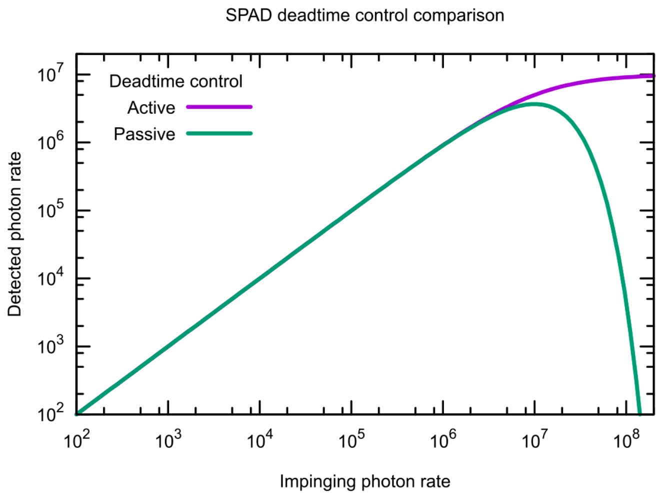


Figure 1.12: Detected vs. impinging photon rate in a SPAD with active and passive dead time control. The detected photon rate is calculated for a dead time of 100 ns and an ideal SPAD with zero noise and 100% PDP according to [40].

Timing resolution (Jitter)

The timing resolution of a SPAD is defined by its response to single and multiple photon detections. It measures the time it takes for the output voltage of the SPAD to rise over a threshold (typically 90% of V_{ex}) after a photon triggers an avalanche. Ideally, this time would always be identical regardless of the location where the photon strikes. The timing uncertainty or jitter depends essentially on the architecture, mainly the size of a SPAD and is typically in the range of a few hundred down to tens of picoseconds. It is measured in many ways, however, the standard one is based on time-correlated single-photon counting (TCSPC), in which successive exposures of a SPAD to 'Dirac' like pulses of light, such as laser pulses, are evaluated in terms of the SPAD response by means of a precise time digitizer. After multiple measurements a histogram is built, which will approach the theoretical time photon response or instrument response function (IRF) of the SPAD when a very large number of measurements are performed. The time uncertainty is usually represented in term of the standard deviation of the Gaussian fit of the IRF or the full-width-at-half-maximum of the same.

Afterpulsing and crosstalk

Afterpulsing and crosstalk are noise sources correlated to detections in SPADs. Afterpulsing refers to spurious pulses observed after a detection in the same SPAD and it is explained by avalanche charges trapped in defects and released some time later. Afterpulsing probability in a SPAD is reduced by using fabrication processes with smaller defect rate and it can be mitigated in fabricated devices by imposing longer dead times on the SPADs to give the charges time to evacuate the multiplication region.

Crosstalk occurs in SPAD arrays and is a noise source correlated to detections in neighboring SPADs. It can occur electrically when charges generated by an avalanche diffuse to the multiplication region of another SPAD. Electrical crosstalk is contained by isolating individual SPADs from one another or creating biased diffusions to collect free charges. The other way for crosstalk to occur is optically. When an avalanche occurs in a SPAD, some of the accelerated charges can generate photons instead of freeing additional electrons. When these photons reach the multiplication region of another SPAD, they can trigger an avalanche there [41].

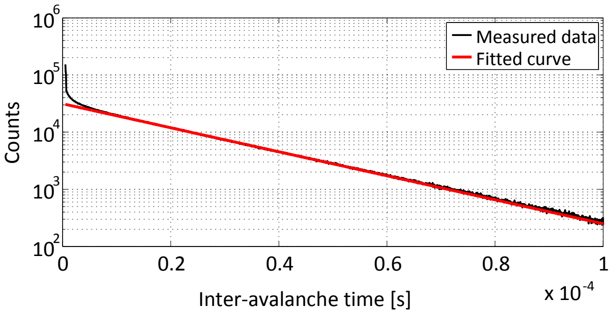


Figure 1.13: Afterpulsing. The plot shows the inter-arrival time histogram measured from a SPAD when exposed to dim light or in the dark. The afterpulsing phenomenon causes counts in excess of the normal exponential response due to Poisson statistics. (Courtesy: Chockalingam Veerappan, [38,39])

Non-uniformity

For SPADs integrated in detector arrays, the individual metrics given above are evaluated for every one of them and, ideally, they are the same for all. As this is not the case, non-uniformity measures are introduced to characterize array detectors. Typically the distributions of individual characteristics are reported in a histogram giving the mean value and the spread either as standard deviation (σ), or as full width at half maximum (FWHM). For a normal distribution, the two are related by $\mathrm{FWHM} = 2\sqrt{2\ln 2}\,\sigma \approx 2.355\sigma$.

The most important non-uniformity measures for SPAD arrays are those of breakdown voltage and DCR. Breakdown voltage non-uniformity leads to non-uniformity in excess bias and thus in PDP. DCR non-uniformity also leads to non-uniform light response over the array. DCR non-uniformity can be reduced by operating SPADs at lower temperatures.

1.3.5 SPAD array architectures

When SPADs are assembled into arrays to make an imager, different types of architectures can be chosen depending on the requirements. The first step in the design of a larger array of SPADs is to decide on a pixel architecture and its functionality.

A pixel can be as simple as a SPAD diode and a transistor for passive quenching as in the first SPAD arrays [42] or as complex as a SPAD diode, a Vernier delay line and thermometer coder, a frequency doubler, a 6-bit counter and a 10-bit memory, as in Megaframe [43]. A simpler pixel has the advantage of a higher fill factor, which results in a sensor with higher PDE. A complex pixel on the other hand can for example provide photon timing information concurrently on all pixels or count photon events for a long time and high dynamic range before it needs to be read out. Most SPAD sensor architectures will strike a balance between the two extremes, with a trend to simpler pixels favoring high fill factor and more shared functionality at the column level. Figure 1.14 shows several examples for pixel architectures and resulting sensors.

Functionality typically included in a pixel, besides the mandatory quenching and recharge, are a number of digital counters starting at one single bit memory. Multiple counters can be multiplexed for continuous synchronous operation of the whole sensor or switched at a fast rate to demodulate the received signal [36]. An analog counter can also be advantageous when it offers a larger counting range than a digital counterpart using the same die area. As mentioned before, a timing circuit can also be added to every pixel to generate timestamps for each detected photon.

In the second step, the pixels are assembled (at least conceptually) into rows and columns typically assigning control signals to rows and data signals to columns. Global control signals are also possible, and in the majority of designs, the pixels share common bias voltages. Pixels are accessed by addressing their row and then reading the data transferred to the column lines. Data can be for example the event count since the last reset or the recorded timestamp. Pixels without internal storage can only send live events when they are connected to the column lines [44]. This type of pixel is only interesting in high intensity applications as the overall photon efficiency is greatly reduced. The number of output lines on a sensor determines how the data is further handled until it reaches the sensor’s interface. In most cases, there is some kind of multiplexing or serialization needed to read the data of a sensor row.

Two different concepts can be employed with respect to the column organization, namely a clock driven or an event driven architecture. In the clock driven architecture, each row of pixels is read out at a fixed rate by cycling over the rows of the sensor. SwissSPAD, presented in Chapter 2, follows this approach. In the event driven approach all pixels of a column are connected in parallel to the column lines and send their row address (and additional information) when an event is detected [45]. Conflict resolution circuitry is needed to prioritize events in the case of multiple events happening at the same time [46]. Hybrid architectures are also a possibility, for example an event counter in each pixel and precise timing circuits shared for each column.

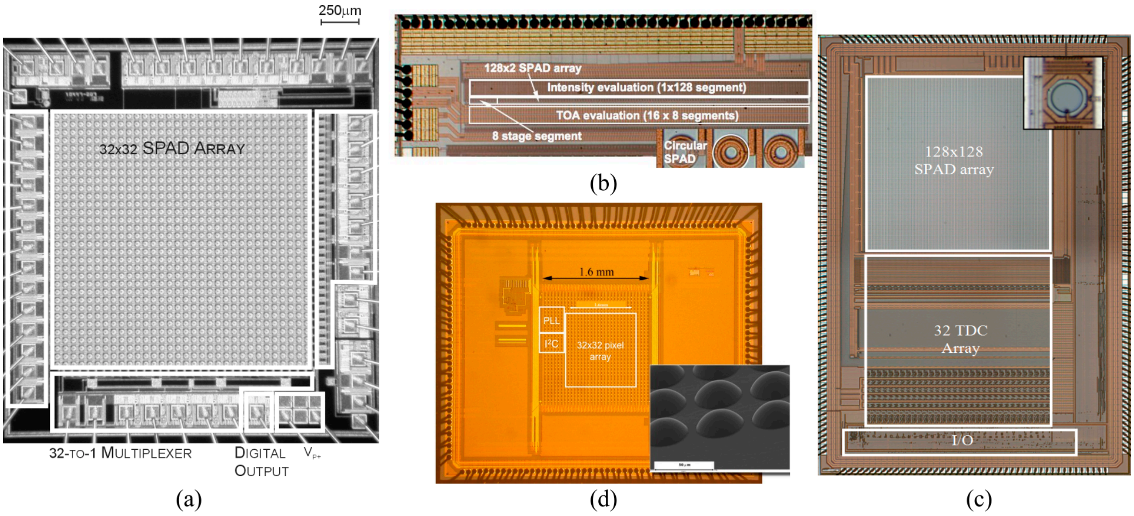


Figure 1.14: SPAD arrays and readout architectures [47]. (a) Random access 32x32 SPAD array [48]; (b) latchless access 128x2 SPAD array [49]; (c) event-driven access with column-parallel TDCs [44]; (d) pixel-parallel TDCs with microlenses in the inset [50]

Advances in CMOS process technologies have resulted in shrinking feature sizes to allow more and more functionality in a sensor and even on a single pixel while maintaining a good fill factor. Additionally, part of the fill factor lost due to in-pixel electronics and guard-rings can be reclaimed by using micro lenses. Another promising avenue for sensors with high functionality and high fill factor are various forms of CMOS 3D integration, which places the photo-sensitive front-end and processing back-end circuits on separate dies [46,51].

LinoSPAD, presented in Chapter 3, is a special sensor architecture in that the SPAD sensor itself is quite simple. It is one row of 256 pixels where each pixel has a dedicated output line. Through the close coupling with a FPGA, we create essentially a reconfigurable pixel that can be optimized in functionality for different applications. The first pixel architecture realized contains 64 column-level TDCs each shared among four pixels. This architecture evolved from event counters for each pixel to TDCs that send timestamps at a high rate, to the current state of integrated histogram engines and processing as described later.

1.4 Time-correlated photon counting

The fast time response is one of the characteristics that differentiates SPAD based image sensors from other classes of image sensors. A SPAD detecting a photon delivers a macroscopic signal with a precision in the picosecond range. Measuring the timing information of that signal allows us to assign a timestamp to individual photons and use that information to construct histograms of photon arrival times useful in many measurement applications of short-lived phenomena.

Time-correlated single-photon counting (TCSPC) is generally used in conjunction with a photon source operated synchronously to the detector. The synchronization provides a common timing reference point for the timestamps generated by the photon-counting sensor, such that histograms over multiple illumination periods can be built.

Two main techniques are used for the illumination and correspondingly for the detection. On the illumination side, one has to decide on the modulation scheme between pulsed and continuous excitation. On the detector side, the choice is between direct and indirect time-of-flight detectors. Indirect time-of-flight detectors measure the phase and amplitude for the detected signal without necessarily assigning timestamps to individual photons. This can be done using synchronously gated counters. Direct time-of-flight detectors on the other hand measure the time of arrival of individual photons and report timestamp values. Time-to-digital converter circuits are needed in these detectors.

The SwissSPAD sensor presented later can be used for indirect time-of-flight measurements even though it is not optimized for these kind of measurements, whereas the LinoSPAD sensor with FPGA TDCs is used for direct time-of-flight measurements.

1.5 Applications of single-photon cameras

Single-photon cameras are mainly used in applications where the timing of individual photons is important or when the number of photons is very low. Analogous to the advance of conventional CMOS image sensors in more and more areas that were once dominated by early analog sensors or CCDs, CMOS SPAD based sensors make their advance in more and more applications where non solid-state single-photon counting cameras were used before.

The main applications where CMOS SPADs have become increasingly used are the imaging of fast and repetitive phenomena in the basic sciences of physics, chemistry and biology. The applications where the cameras developed and presented in this work have been used are briefly introduced in the following subsections and more details and results will be given in separate sections.

1.5.1 Fluorescence lifetime imaging

Fluorescence lifetime imaging microscopy (FLIM) is a standard technique [52] used to distinguish fluorescent molecules either freely diffusing or attached as marker to other molecules. FLIM is not solely interested in the fluorescence intensity captured by the imaging system but also in the lifetime of the fluorophores after they have been excited. The excitation is commonly achieved using a pulsed laser and the fluorescence response of the observed sample is collected using a confocal imaging setup. The fluorescence lifetime depends on the used dye, but also on their environment characterized by pH or concentrations of other molecules. It can also depend on the state of the fluorophore marker, whether it is bound to an acceptor molecule or not.

The SwissSPAD sensor was used to create a wide-field fluorescence lifetime imaging system by using the global shutter circuit for time-gating the photo-sensitivity in synchronous operation. The chip was successfully used in super resolution microscopy, triggering improved data analysis algorithms [53] and enabling industry to develop advanced microlens fabrication processes. The initial methods and algorithms with experimental results of lifetime measurements on dye solutions with different short- and long-lived fluorophores are presented in section 5.1.1 and were the subject of several publications [54,55,56].

1.5.2 Quantum random number generation

Many algorithms of modern cryptography rely on random numbers to protect our digital secrets. Random numbers are used to generate passwords, cryptographic keys and signatures and to secure transactions in distributed currencies. Although for some applications like test pattern generation or Monte Carlo simulations, the use of pseudorandom numbers is sufficient, even desirable, true random numbers are increasingly used either due to security concerns or for regulatory reasons.

The increasing demand for applications requiring secure communication like e-banking, credit cards, cell phone encryption calls for high quality and at the same time high speed random numbers. A large network server with thousands of simultaneous connections might require a random number throughput of several Gb/s.

True random number generators (TRNGs) reaching high speeds have been proposed using a number of different mechanisms. Examples include thermal jitter in ring oscillators, RAM write collisions, flip-flop metastability in FPGAs [57] and ASICs [58]. In addition, optical effects based on LEDs and lasers have been proposed as source of entropy at high rates [59,60]. What is needed is a high-speed generator of true random numbers that can be realized at low cost.

To exploit the quantum nature of photons seems to be a good start to create an effective TRNG. Photons are available almost anywhere in abundant quantities and they carry inherent random properties if observed at the individual level.

The SwissSPAD sensor was used to explore the possibility of using SPADs interacting with photons as source of randomness. The high number of SPADs available and the high throughput of the sensor proved to be advantageous to that end. The main outcomes of the experimental work with SwissSPAD as TRNG is described in section 5.1.2 and was the subject of several publications [61,62,63].

1.5.3 3D time-of-flight

3D cameras experienced a popularity boost when Microsoft released the Kinect 3D camera for their game console in 2010. Intel was able to follow up with the release of the RealSense range of 3D cameras putting 3D cameras into the reach of a large public. These 3D cameras use the structured light approach for 3D scene reconstruction that work by projection a known pattern on the scene and analyzing the distortion seen by the camera.

With the LinoSPAD camera, we use the time-of-flight approach to 3D imaging that reconstructs depth information from the time it takes for a flash of photons to travel to an object and back to the sensor. Time-of-flight imaging requires precise timing of the arrival of photons on the camera. For this task, an array of time-to-digital converters has been implemented in the FPGA connected to the LinoSPAD sensor array. The TDCs are connected to histogram engines, which create and process full time-of-arrival histograms in the camera before sending them to a computer. The camera system has been presented for the first time at the SPIE Photonics Europe conference 2016 [64].

1.5.4 Other uses of SPAD imagers

This section lists very briefly other applications that are possible with SPAD based imagers and that have in part been realized with the cameras developed in the work of this thesis.

High-speed intensity

A SPAD based camera can be used as a normal intensity camera computing an intensity value for each pixel and producing an image on a computer screen. The high speed at which SPAD cameras can do this is one of the distinctive characteristics. The SwissSPAD camera is essentially limited in the frame-rate it can deliver by the capabilities of the I/O circuits of the sensor. The 128 lines can be operated reliable at frequencies up to 100 MHz and with that, the camera can reach frame rates of almost 200 kHz. Some image sequences are shown and the data rate limits discussed in section 2.5.9.

Fluorescence correlation spectroscopy

Fluorescence (cross-)correlation spectroscopy, F(C)CS is used to probe the mobility of fluorescing molecules in a small volume defined by the focus of the observation volume of a microscope. Usually confocal microscopes are used for FCCS. One records the fluorescence fluctuations from such a volume with high temporal resolution and then calculates their autocorrelations function. The autocorrelation function of the fluorescence intensity is characterized by the concentration (amplitude for $\tau = 0$) and diffusion coefficient (decay) of fluorophores in the observed volume. If several volumes are observed simultaneously (e.g. in a multi-confocal microscope or a TIRF or SPIM), the cross-correlation function between distant volumes can be used to measure the direction and velocity of flows in the sample. In a wide-field setup, the cross-correlation between neighboring pixels can be used to measure directed flow. The combination of single-plane illumination microscopy (SPIM) and FCS was used to build a wide-field FCS setup and SwissSPAD was used as sensor with high temporal resolution to measure short decay times.

Chapter 24 in [65] discusses the theory of FCS in detail. An overview of the SwissSPAD FCS setup and the measurement results is presented in section 5.1.3.

Super resolution microscopy

Super resolution microscopy is a collection of resolution enhancement methods for optical microscopy that allow optical instruments to break the diffraction limit. The technique was awarded the Nobel Prize in Chemistry 2014. SwissSPAD has been used in a GSDIM super resolution microscopy setup and has successfully captured super-resolved fluorescence images of labeled cells [66]. Some results are shown in section 5.1.4.

PET/SPECT, FRET, Raman spectroscopy

Other notable applications in which SPAD imagers have been shown to perform well include various imaging methods in nuclear science like PET and SPECT [67,68] where SPAD sensors are advantageous because they are insensitive to large magnetic fields present in MRI. This makes it possible to combine SPADs with MRI. There are more biological applications like FRET [69,70] and NIROT [71,72]. Last, but not least, SPAD based imagers are being extensively studied for space application where they will be used for example in Raman spectroscopy [73,74,75].

Chapter 2 SwissSPAD: a 512x128 time-gated single-photon imager

This chapter reports on the design and performance of SwissSPAD, a 512 x 128 SPAD pixel image sensor implemented in 0.35 µm CMOS technology. The sensor, which was published in [56] has pixels with integrated 1-bit counters and reaches a frame-rate of over 156 kHz. It has been used for high-speed imaging, fluorescence lifetime imaging and random number generation among other applications and is still being introduced to new applications.

Figure 2.1 illustrates the camera system, including the computer used to control the operation of the camera, the FPGA that provides the connection between computer and sensor, and finally the sensor itself. A LED is shown as it was used to evaluate the sensor for photo-response and random number generation. A second sensor is alluded to in the picture since SwissSPAD was made to be abuttable to double its resolution even though this was not further investigated due to fabrication yield issues.

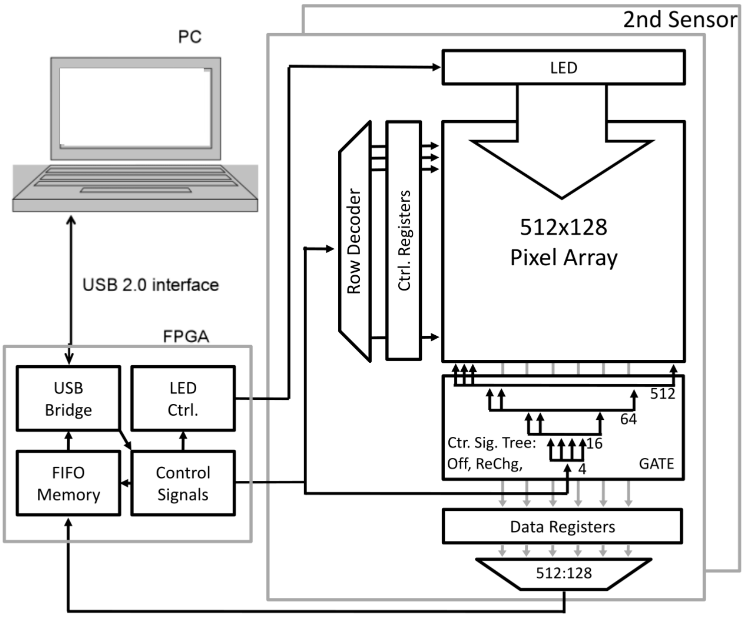


Figure 2.1: SwissSPAD camera architecture overview. The SwissSPAD sensor is connected to a FPGA implementing the control logic for sensor operation and the interface to a computer. A LED is shown as it was used for photo-response characterization and to generate random number sequences. A second sensor could be abutted to double the resolution of the camera to 512 x 256.

2.1 Chip architecture

The architecture of the SwissSPAD sensor is presented following a bottom-up approach from the detection of photons in the SPAD cell through the pixel circuit with the gating and memory functionality to the output registers of the chip and from there to the FPGA for further processing and transmission to a computer.

2.1.1 CMOS SPAD

The SPAD cell used for the SwissSPAD pixels is a basic design known from [76] using a p+ anode and a deep n-well cathode implanted in the p-type substrate of a 0.35 µm high-voltage CMOS process. A p-well guard ring is used to prevent premature edge breakdown of the high electric field used for the multiplication region. Using this design, as shown in Figure 2.2, has the advantage that both anode and cathode are individually isolated for each SPAD and their voltage can be chosen independently from other working voltages.

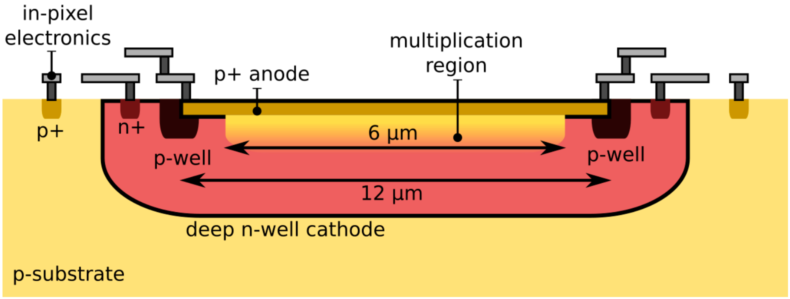


Figure 2.2: Cross-section of the SPAD structure used in SwissSPAD. It is fabricated in a 0.35 µm high-voltage CMOS process. The p+ - deep n-well junction defines the multiplication region and the p-well guard ring prevents premature edge breakdown. Anode and cathode voltage are isolated from other voltages in their own wells.

The SPAD structure is of circular shape with a total diameter of 12 µm and the active region (where photons are absorbed and induce avalanches) has a diameter of 6 µm. In the 24 µm x 24 µm pixel presented hereafter this results in a fill factor of 5%. An optimized shape of the SPAD could have increased the fill factor, but was not yet verified to work and thus not included in the large SwissSPAD sensor.

2.1.2 Pixel circuit

To be able to detect photons on a large array of SPAD pixels, minimal circuitry must be used per pixel to provide at least avalanche detection and quenching and a way to address the individual SPAD pixels. A photon-induced avalanche in a SPAD results in a macroscopic signal strong enough to activate an integrated transistor and any desired digital processing can be added to the pixel circuit from there. Driven by the need to have a small pixel in order to create a high-resolution area of reasonable size, the SwissSPAD pixels have been kept relatively simple. A gating circuit and a 1-bit memory is implemented in each pixel to reduce transistor count.

The SwissSPAD pixel is shown in Figure 2.3 and can be described in four parts: from left to right they are 1) the SPAD cell with passive quenching, 2) the gating and detection, 3) a 1-bit memory latch (counter), and 4) the address and reset circuit section.

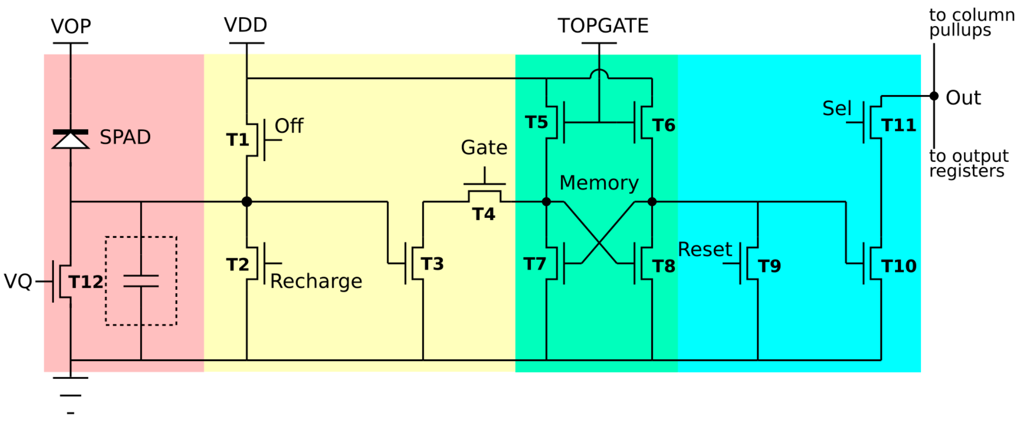


Figure 2.3: Transistor level schematic of the SwissSPAD pixel. From left to right the parts are 1) the SPAD with quenching transistor, 2) the gating circuit controlled by global signals *off*, *recharge* and *gate*, 3) the 1-bit memory latch (counter), and 4) the readout and reset circuit.

The first part of the circuit contains just the SPAD structure, connected to its operating voltage and a quenching transistor used as resistor to separate the anode from ground. The gate of the quenching transistor is driven with an analog voltage used to control the dead time of the SPAD separate from the gating part of the circuit.

The time gating and photon detection mechanisms consist of four transistors shown in the second section. Using T1, the SPAD operated below an excess bias of V_{DD} can be switched off by lowering its bias to V_{OP}-V_{DD}. Using transistor T2 the bias is restored to V_{OP} reactivating the SPAD. The wider-than-minimum-size transistor T2 provides a fast turn-on of the SPAD, but the asymmetric and larger load on the global signal has its issues as discussed later. Transistor T3 detects the voltage on the SPAD anode and T4 is used to disable photon detection and prevent the counter from incrementing when using T1 to turn off the SPAD. Note that the passive quenching through T12 and the anode recharge through T2 work in parallel and that short-circuits between V_{DD} and ground through T1 and T2 or T12 need to be prevented.

The third section of the pixel is a 1-bit (saturating) counter implemented as NMOS latch formed of transistors T7/T8, which are loaded by T5/T6. Activating T3 and T4 causes the latch to flip from its reset state to which it is restored by activating T9. Even though this design causes considerable static power consumption due to the NMOS design and the needed biasing, it was preferred to a fully CMOS solution as it permitted to create a smaller pixel cell without extra n-well for PMOS logic.

The fourth section of the pixel consists of the reset transistor T9 and transistor T11 used to address the pixel. The column line is pulled down with different strengths, depending on the latch state through transistor T10.

The pixel circuit needs to be laid out in a square such that the area of the SPAD is maximized relative to the total area of the pixel and such that a regular array can be constructed. From these considerations, it was decided that voltages are distributed along a row and global signals along the columns. The other signals follow common convention for imagers with control signals by row and output signals by column. This resulted in the layout as presented in Figure 2.4. Most of the area of 24 µm x 24 µm is occupied by the SPAD cell and guard ring and the long transistors to load the latch. Horizontal lines distribute the SPAD operating voltage V_{OP}, the logic voltage V_{DD} and the quenching bias V_Q as well as the reset and select signals. On vertical lines, we find the gating signals *off*, *recharge*, and *gate* and the pixel output signal. Two transistors enabling to bypass the memory are not marked and not further discussed.

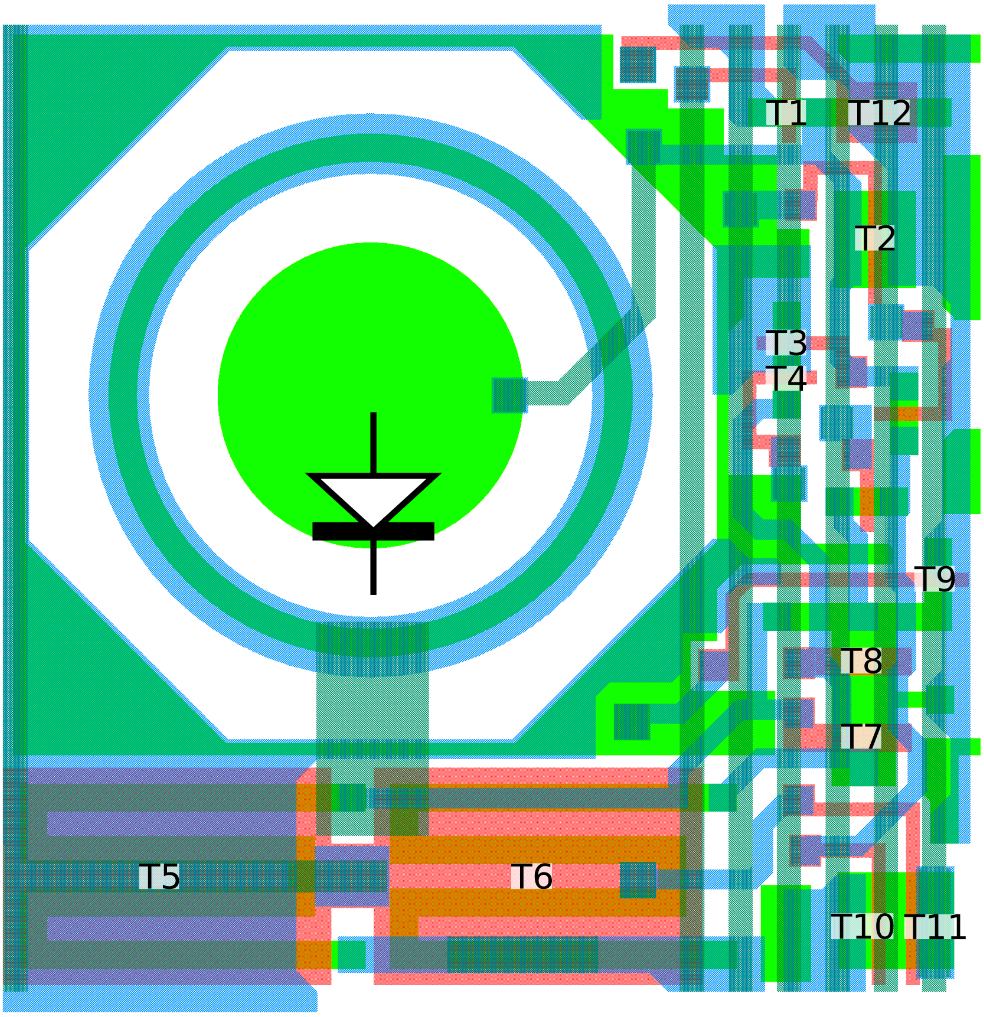


Figure 2.4: Layout of one 24 µm x 24 µm SwissSPAD pixel. The SPAD uses most of the area, yet the active area accounts for only 5 % of the total square. The long transistors T5 and T6 load the memory latch. The trained reader can spot two anonymous transistors that are not further discussed and could have been used to bypass the memory.

2.1.3 Sensor architecture

The SwissSPAD sensor is built by combining an array of 128 rows of 512 SPAD pixels detailed above with the electronics needed to control the image acquisition and readout. Figure 2.5 shows the block diagram of the SwissSPAD sensor with Figure 2.6 and Figure 2.7 showing the circuits used to drive a row of pixels and read a column respectively.

A single row of 512 pixels is selected by decoding the row address register and the addressed pixels pull down the column output lines depending on the state of their counters. The column lines are pulled up by the termination circuit shown in Figure 2.7 and feed 512 output data registers. The pull-up network is split in two transistors, one of which is stronger and is actively controlled. When a new row of pixels is selected, the pull-up transistor is pulsed initially to pull the output line up, where it is then held by the weak transistor or pulled down again by the pixel output transistor T10. The pixel output transistors strength is between the two pull-up transistors to maximize possible data rates.

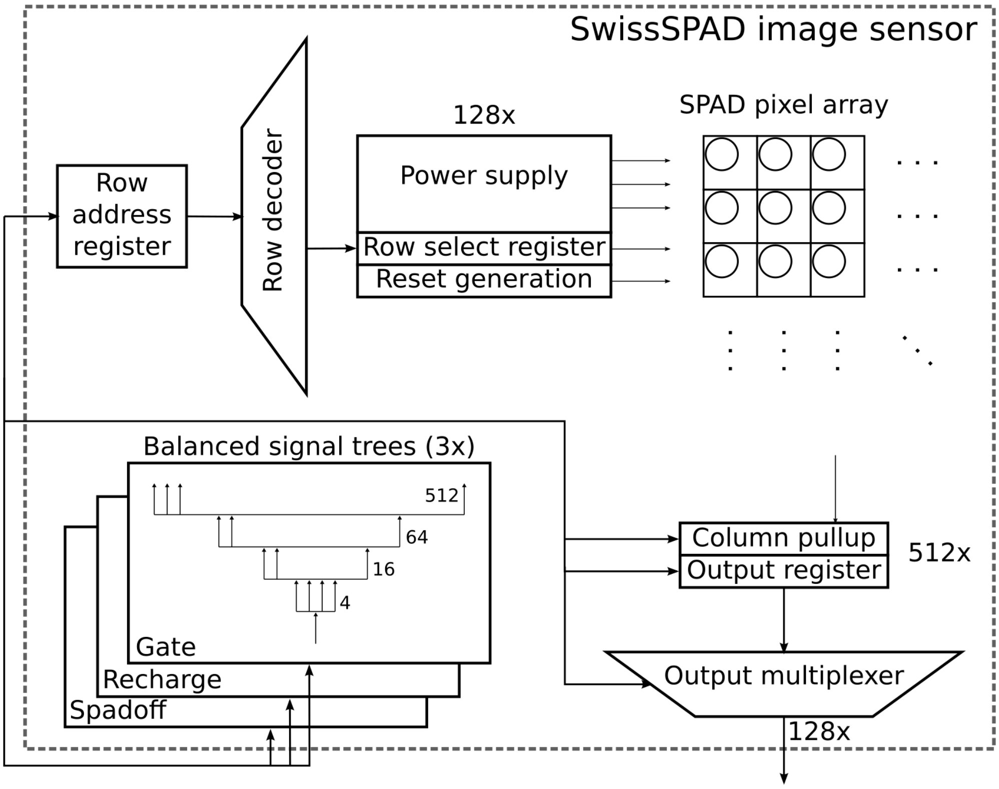


Figure 2.5: Block diagram of the SwissSPAD sensor. The signal trees for the global gating signals are on the long side below the pixel array. A row of 512 pixels is selected for readout and reset automatically when deselected once the outputs are registered. The output passes through 4 to 1 multiplexers and leaves the chip on the long side.

The architecture of the chip forms a pipeline with the first stage registering the row address and the second stage registering the row output data. The pipeline has been operated with 80 MHz for most characterization and measurements executed so far, but it is capable of operating at up to 100 MHz. The critical path spans from the row address register through the pixels to the output data registers. At a frequency of 100 MHz an output data rate of almost 12 Gbit/s (Gbps) using 128 lines is attained.

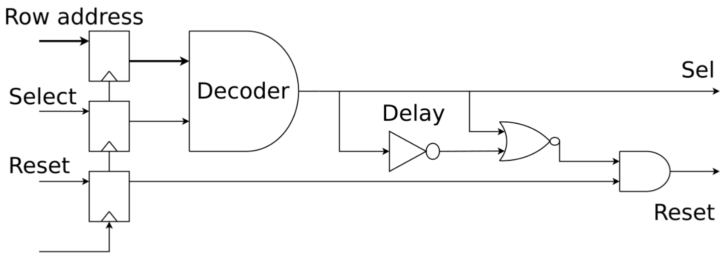


Figure 2.6: The SwissSPAD row decoder drives row selection and reset signals. A pulse on the reset signal is automatically generated when a row is deselected. The length of the pulse is set through the inverting circuit delay.

The three gating signals of the pixels are driven through matched signal trees that distribute the signals with low skew across the full array. These signals are asynchronous and thus completely independent from the signals used for the readout of pixel data. The only possible interaction between the two sets of signals is across the pixel memory when a photon hit is detected during the time when the reset is active.

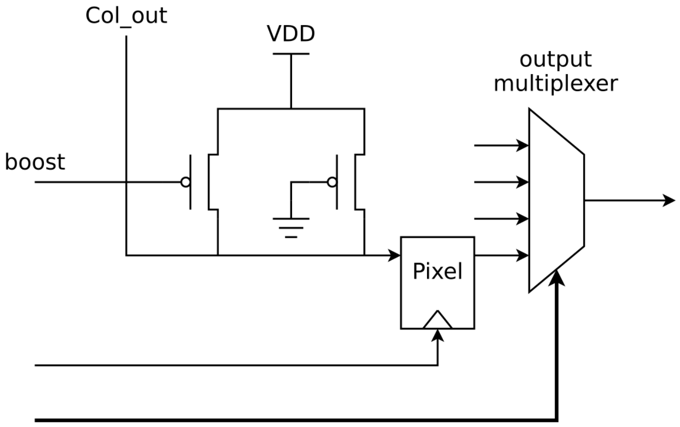


Figure 2.7: SwissSPAD pixel column termination. A column output line of the pixel array is terminated with two PMOS pull-up transistors and a data register connected to a 4 to 1 multiplexer. Four columns share a physical output line.

Figure 2.8 shows the micrograph of the SwissSPAD sensor. The central part is occupied by the array of pixels with the logic distributed on three sides of the array. To the left is the row selection and reset logic and on the bottom the column termination with output registers and multiplexers. Signal trees for the distribution of the gating signals are also at the bottom of the pixel array. On the left side are decoupling capacitors, regular routing and additional signal pads. As is standard in pixel sensor designs, the layout has a high regularity without which it would not be manageable.

The top side of the sensor has only minimal closing circuitry such that two sensors can be abutted with the loss of only six pixels approximately. Unfortunately, due to issues with fabrication yield this was not exploited as planned.

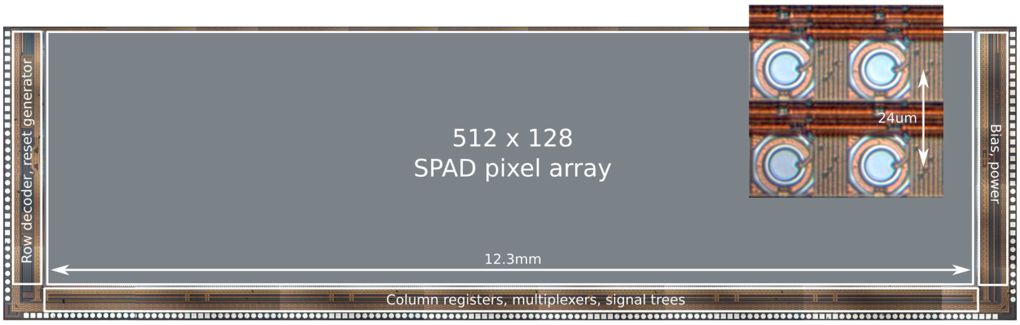

Row decoder, reset generator
512 x 128
SPAD pixel array
24um
Bias, power
12.3mm
Column registers, multiplexers, signal trees

Figure 2.8: Die micrograph of the SwissSPAD sensor with the SPAD-array occupying most of the area and control and readout logic on three sides, such that two sensors can be abutted with a gap of a few pixels. The inset is a detail of the SPAD cells with a pitch of 24 µm.

2.2 Chip operation

For operation of the SwissSPAD sensor, the pixel is split conceptually in two parts before and after the pixel's memory element. The first part before the memory contains the shutter or gating control that defines the sensitivity time window of the SPAD where avalanches from impinging photons are triggered and the memory can be written. This part operates in parallel for the whole sensor.

The second part after the memory is the readout part where pixel rows are selected, one after the other, and the memory state is transferred to the output registers. The line is then reset while the data is multiplexed on the output lines and the next row is selected.

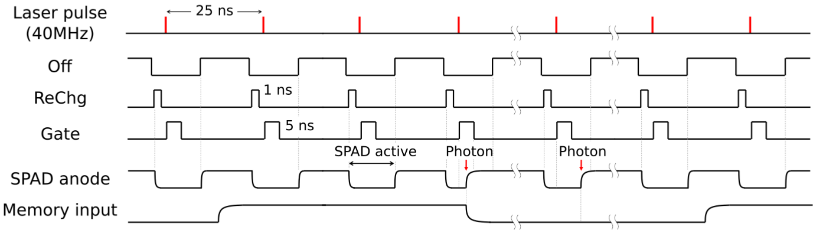


Figure 2.9: Timing diagram of the SwissSPAD global shutter. The shutter is shown here operating in synchronization with a pulsed laser for time-correlated photon acquisition. In a laser period of 25 ns the SPAD is turned on and made photosensitive for a window of 5 ns and turned off for the rest of the period.

The synchronization between shutter and readout differs, depending on whether global shutter or rolling shutter are used. SwissSPAD can be operated in both modes, but global shutter mode is not very photon efficient because only one detection can be stored in a pixel and the shutter must be closed for the duration of the readout. When operating in rolling shutter mode each line of pixels is enabled to detect photons impinging between successive readouts at a rate up to 156 kHz with a 80 MHz readout clock. When a row is read in rolling shutter, it is reset while the shutter continues to operate.

The global shutter circuit in SwissSPAD is controlled by the three signals driving the gates of transistors T1, T2 and T4 in each pixel. The *gate* signal on T4 is the most obvious as it directly controls the input to the memory latch. The *off* signal on T1 raises the SPAD bias by V_{DD} to possibly bring the SPAD below the Geiger region or at least greatly reduce its sensitivity. Later the *recharge* signal on T2 is used to restore the SPAD bias to its nominal voltage V_{OP} and make it sensitive to photon impacts. Transistors T1 and T2 are both needed to have better control of the onset of the photon sensitive time window by starting from below breakdown in the off state. The end of the photon sensitive window, on the other hand, is defined by one transistor. T4 cuts the connection of the SPAD to the memory and prevents accidental memory toggling from the *off* signal. The typical operation of the gating circuit when synchronized with a pulsed laser is shown in Figure 2.9.

The readout of SwissSPAD is controlled synchronously by the row address and output multiplexer signals and the associated register clocks. The clocks for the address registers and the output registers are separate, but must use the same frequency derived from the same time-base for successful operation. A readout of the pixel memory is initiated by clocking in address zero and giving a short pulse on the strong column pull-up. After one clock period, the output registers are clocked to sample the pixel data, which is then read during the next clock cycle by switching the output multiplexer. Figure 2.10 shows the pipelined operation of the sensor which runs with a row clock of 20 MHz to 25 MHz for a data clock rate of 80 MHz to 100 MHz respectively. A complete readout of the chip resulting in a 512 x 128 one-bit image takes between 5.12 µs and 6.4 µs.

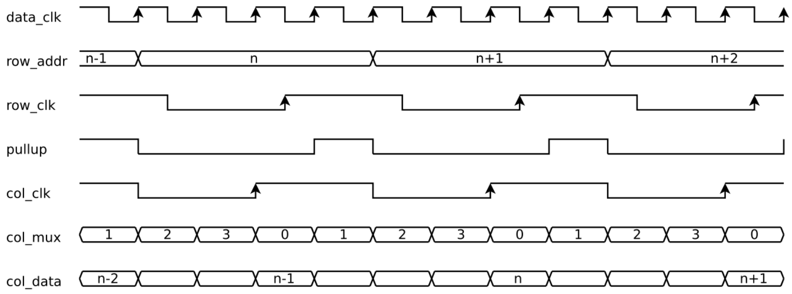


Figure 2.10: Timing diagram for the readout of SwissSPAD. The row and column clocks run at one 4th of the multiplexer and data rate output pin. A small delay between the clocks shown accounts for routing differences and ensures timing closure by respecting setup and hold timing for synchronous elements.

2.3 FGPA system design

To achieve a larger dynamic range using the one-bit digital pixel a number of measurements needs to be accumulated. This is done through the integration of SwissSPAD together with an FPGA that generates the control signals and performs multi-bit image accumulation.

The FPGA firmware as depicted in Figure 2.11 can be split in three major blocks. The first is the shutter control for the generation of the signals driving the three gating transistors T1, T2 and T4. This part is running at a high frequency synchronized to a master time reference to control the gating signals with high granularity. Using output serializers in the FPGA, the signals are driven with periods down to 1.25 ns, synchronous to the reference clock. The gating signals are related through the common clock driving the output registers, thereby defining the relative granularity. However, the exact phase of this output clock relative to the reference clock can be adjusted in steps of approximately 20 ps using phase shifting controls in the PLL function blocks in the FPGA. When used with an external reference the possible frequencies of the shutter clock are restricted by the minimum and maximum operating frequencies possible for the PLL and clock trees of the FPGA. The shutter clock must also be an integer multiple of the reference for best efficiency.

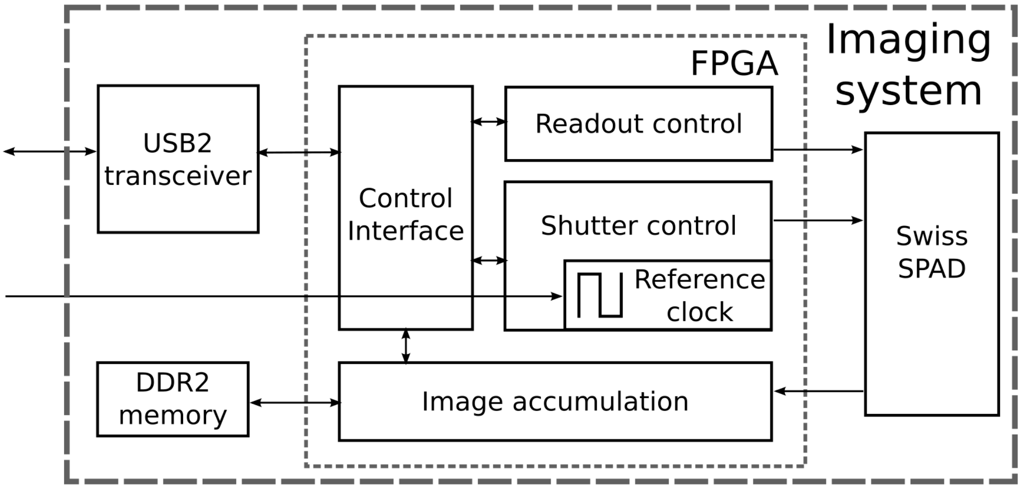


Figure 2.11: SwissSPAD FPGA block diagram showing the complete camera. The FPGA is used to generate the shutter and readout signal sequences and to perform accumulation of images. Longer image sequences can be stored in RAM before being transferred to a computer over USB.

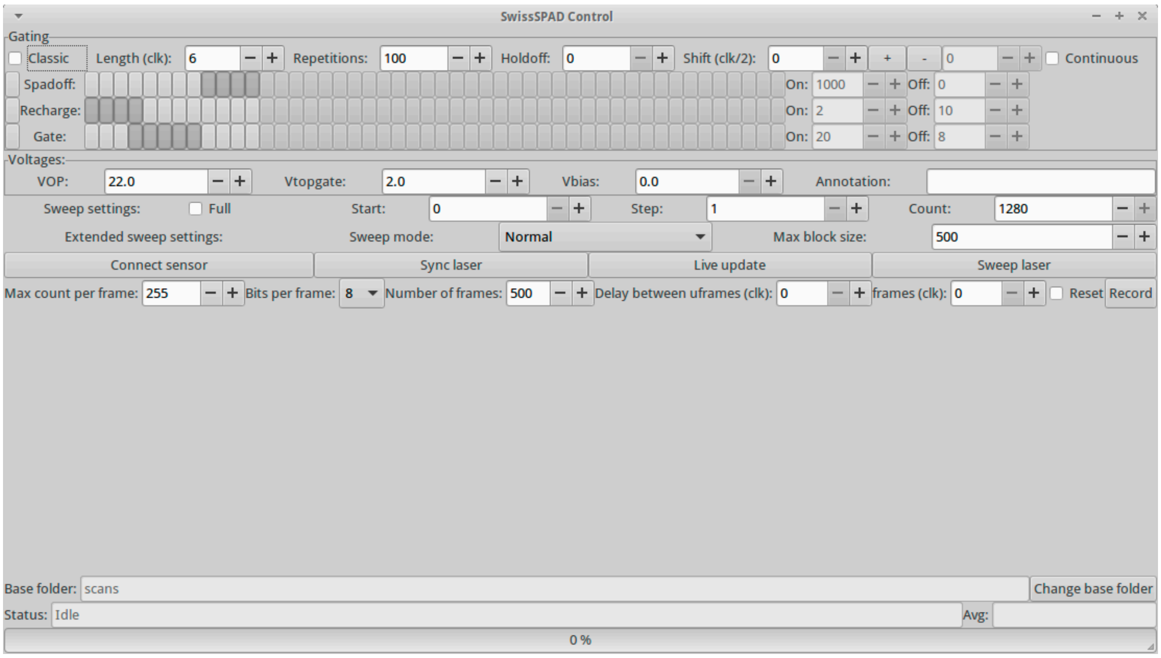
The second block of Figure 2.11 drives the readout part of SwissSPAD and runs at a fixed frequency of 80 MHz to 100 MHz independent from the shutter. This block drives the row address and the two clock signals as well as the output multiplexer select signal. In the readout sequence, it also synchronizes the image accumulation.

Finally, the third block in the FPGA firmware is the image accumulation buffer. Here a multi-bit image is accumulated by adding multiple one-bit frames read from SwissSPAD. The bit-depth is configurable in powers of two between 1 and 16 bit and a double buffering scheme is used to allow back-to-back image accumulation at full speed. After a complete image frame has been accumulated, it is stored in off-chip memory or transferred to a computer.

2.4 Computer software

The SwissSPAD camera is completed by a computer software written to control all aspects of the camera for image acquisition. It allows the user to define the shutter pattern and synchronization, the bit-depth and number of the images. It gives fine control on the number of shutter pattern used between readouts for global shutter mode and on the readout frequency for rolling shutter mode.

The SwissSPAD software has been written using free software libraries for the user interface and for USB communication. USB communication is abstracted to a stream of command and data words where the computer software manages the FPGA firmware. In the FPGA firmware the different modules (shutter, readout and memory) are individually addressable to configure them and read images.



□ Classic Length (clk): 6 Repetitions: 100 Holdoff: 0 Shift (clk/2): 0 0 □ Continuous

Spadoff: On: 1000 Off: 0

Recharge: On: 2 Off: 10

Gate: On: 20 Off: 8

VOP: 22.0 Vtopgate: 2.0 Vbias: 0.0 Annotation:

Sweep settings: □ Full Start: 0 Step: 1 Count: 1280

Extended sweep settings: Sweep mode: Normal Max block size: 500

Max count per frame: 255 Bits per frame: 8 Number of frames: 500 Delay between uframes (clk): 0 frames (clk): 0 □ Reset

Base folder: scans

Status: Idle Avg:

0 %

Figure 2.12: SwissSPAD control software. The software is used to configure all aspects of SwissSPAD operation. The top part configures the gating in a simple on/off mode or with a pattern synchronized to a reference clock. The part below the voltages controls the image accumulation and performs shutter pattern sweeps. An image preview area is located in the bottom half of the window.

Figure 2.12 shows the graphical user interface for the SwissSPAD in its latest evolution before it was replaced with the change to a new FPGA board as mentioned in section 2.6. The top part of the interface controls the shutter options with the choices between global and rolling shutter and between a synchronous high-speed repeated pattern and a simple on-off pattern. The part below the operating voltage controls the image accumulation and initiates pattern sweeps where the shutter is shifted repeatedly relative to the reference clock signal. Images are shown in the preview area and are stored to selected folders in the file system.

2.5 Performance evaluation

The performance evaluation for the SwissSPAD camera system as outlined in the previous sections can be divided in two parts. The first comprises the SPAD performance, which depends on the fabrication technology and layout of the SPAD itself. Here we measure the breakdown voltage, dark count rate and photon sensitivity under different operating conditions. The second comprises the pixel and array performance where the accuracy and precision of the gating and speed of the readout is measured.

2.5.1 Breakdown voltage

The breakdown voltage for each pixel in the SwissSPAD sensor is measured using the excess count rate (ECR) method outlined in [37] and is very similar to the excess DCR method used for the LinoSPAD sensor in section 3.5.1.

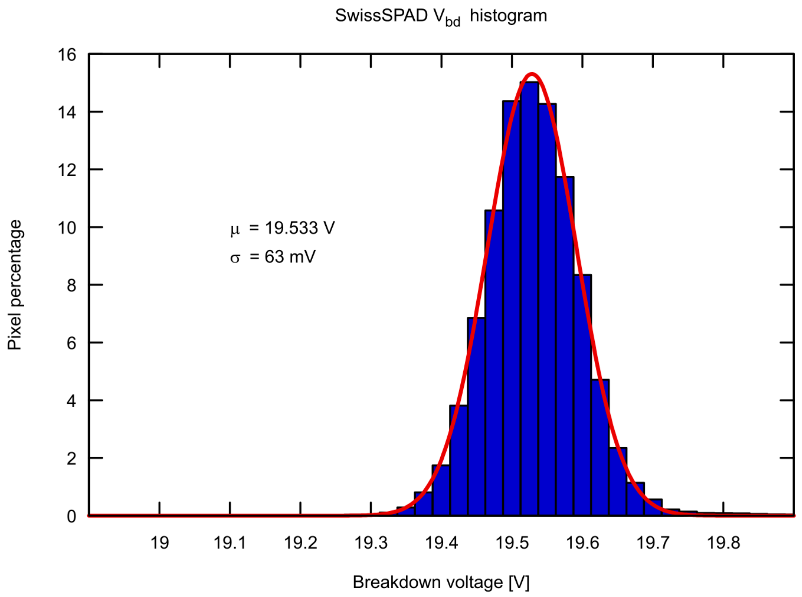


Figure 2.13: SwissSPAD breakdown voltage histogram. The breakdown voltage for each pixel is estimated using the excess count rate (ECR) method and subtracting 0.6 V for the threshold of T3. The histogram uses bins of 25 mV and a normal distribution to fit the values.

The sensor is illuminated with a uniform low intensity and the operating voltage is increased until the pixels begin to report photon counts. A linear fit is then performed on the reported count rate/voltage curve to find the value of the breakdown voltage. A fixed threshold of 0.6 V for transistor T3 is subtracted to find the breakdown value for the SPAD.

Figure 2.13 shows the histogram of breakdown voltages obtained for all pixels in a SwissSPAD array. The standard deviation of 63 mV for the measured values leads to a maximum change in count rate below 2% calculated from the PDP characteristic measured below.

2.5.2 DCR

Figure 2.14 shows the distribution of the dark count rate over the pixels of the array for different excess bias voltages at room temperature. As is typical for SPAD sensors most of the SPADs show a DCR in a narrow band around the median value with relatively few very cold or hot SPADs and even less completely broken SPADs. The percentage of pixels exhibiting DCR significantly above the median value depends mainly on the maturity of the fabrication process and the layout of the diode. A more mature fabrication with less defects leads to a reduced fraction of hot pixels. A more detailed analysis of the spatial distribution and DCR variation with temperature is presented in 3.5.9.

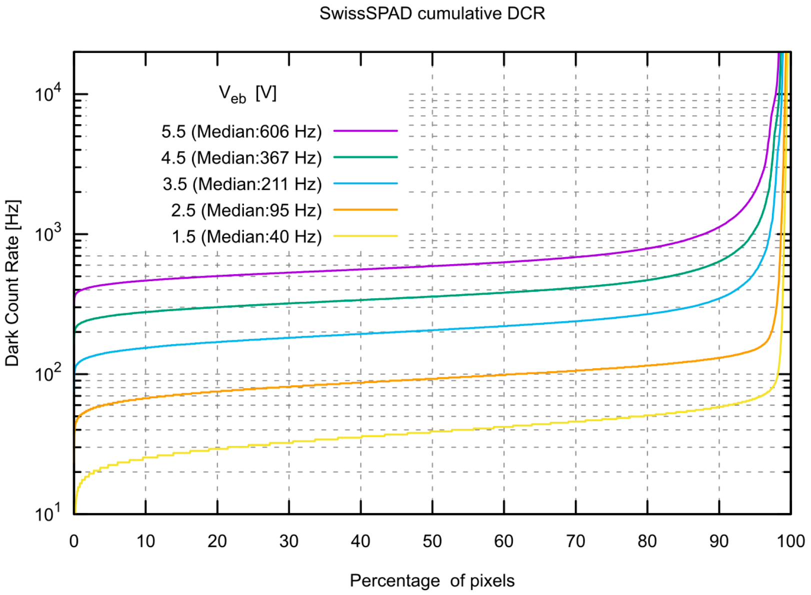


Figure 2.14: SwissSPAD cumulative DCR distribution plotted for different values of excess bias voltage at room temperature. The hottest 1% of the pixels reach a count rate up to 2 MHz.

2.5.3 PDP

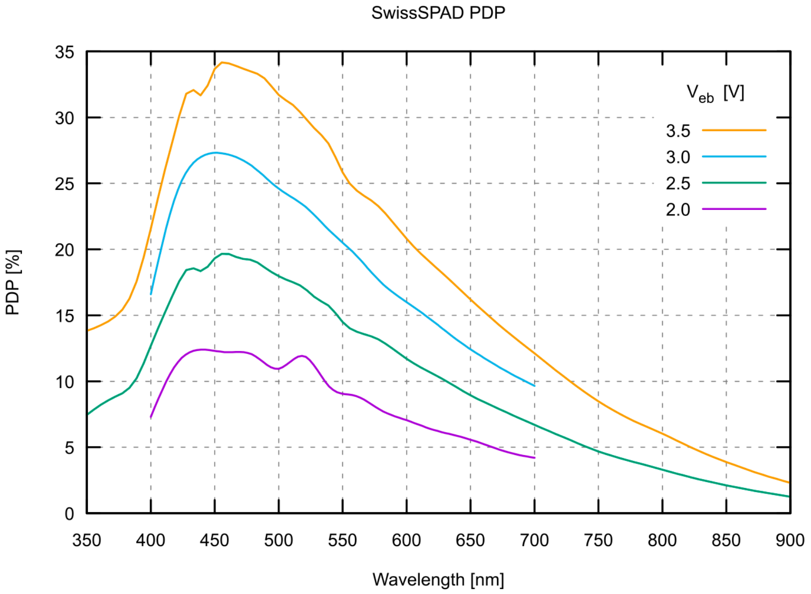


Figure 2.15: SwissSPAD photon detection probability (PDP) in the range of 350 nm to 900 nm for increasing excess bias voltages.

Figure 2.15 shows the photon detection probability of the SPADs for different excess bias voltages in the wavelength range from 350 nm to 900 nm. As is typical for shallow junction structure the absorption peak is in the low wavelengths around 450 nm with sensitivity dropping significantly for higher wavelengths, down to a value around 1% for 900 nm.

The photon efficiency of the sensor is computed from the PDP value by multiplying by the fill factor of 5%. In the case of SwissSPAD, one needs also to take into account the duty cycle of the global shutter that adds a further reduction to the photon efficiency. A part of the efficiency lost to fill factor can be recovered by using microlenses as discussed in section 4.2.

2.5.4 Photo response non-uniformity (PRNU)

The DCR and PDP as measured in the previous sections show some variation across a collection of SPADs that does not allow a direct calculation of the incident photon rate from the measured count rate. In addition, the readout scheme and speed add further deviations to the measured value. This section on photo response non-uniformity explains how to account for these variations on a pixel-by-pixel basis to find the best estimate of the incident photon rate from a measured count rate.

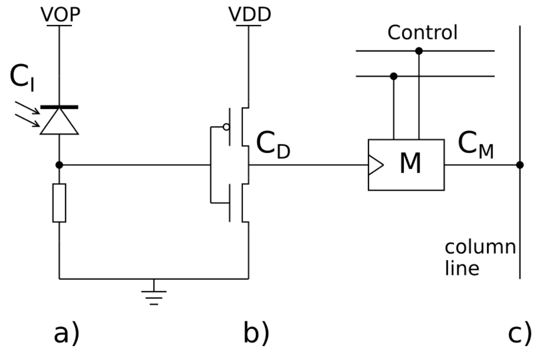


Figure 2.16: Generic 3-part CMOS SPAD pixel. Part a) is the SPAD detector with associated photon rate C_I, b) the discriminator at the output of which we observe the pulse rate C_D, and c) the processing or storage logic which reports the measured count rate C_M.

In order to be able to correct for (part of) the non-uniformities we introduce here a generic SPAD pixel model abstracting the relevant circuit parts to describe its operation. From this model, we derive the formulas to calculate the actual value of the incident photon-rate from the measured sensor count rate.

Figure 2.16 shows the generic pixel model used for the analysis published in [77] and presented hereafter. Part a) is the photon-sensitive SPAD connected to the discriminator labeled b) that is connected to the processing or storage logic c) from where data is read. The following symbols are used: C_I denotes the photon rate impinging on the SPAD sensitive area, C_D denotes the event rate detected by the discriminator and C_M denotes the count rate that is read from the sensor. We are not concerned with fill factor here. In the majority of the SPAD sensor systems, only C_M is ultimately available.

From the measurement of C_M, we are interested in knowing the true photon rate at the detector. However, C_M is not only a function of C_I, but also of other variables:

$$C_M = f(T_{\text{dead}}, T_{\text{readout}}, \text{PDE}, C_{\text{OFF}}, C_I), \tag{2.1}$$

where T_{dead} and T_{readout} are the SPAD dead time and pixel readout time respectively and C_{OFF} is a generic offset count rate for the pixel independent of impinging photons for which

$$\langle C_{\text{OFF}} \rangle = \text{DCR}. \tag{2.2}$$

The discriminator in Figure 2.16 with rate C_D will show a linear response to C_I of

$$C_D = C_I \times \text{PDE} + \text{DCR}, \tag{2.3}$$

when $C_D \ll \frac{1}{T_{\text{dead}}}$.

SwissSPAD can only differentiate between zero photons and more than zero photons. The probabilities for both outcomes must add to one such that we can say

$$p_{\text{counts}>0} = 1 - p_{\text{counts}=0} = 1 - e^{-\chi}, \tag{2.4}$$

from the assumption that photons are Poisson distributed. The expected number of photons for a readout period would be $\chi = C_D \times T_{\text{readout}}$ of which we only see one and thus for the measured count rate we expect

$$C_M = \frac{1 - e^{-C_D \times T_{\text{readout}}}}{T_{\text{readout}}}, \tag{2.5}$$

which is confirmed in the experiments shown in the next section.

2.5.5 Sensitivity and noise calibration

Inserting equation (2.3) in (2.5) and solving for the incident count rate C_I we find

$$C_I = \frac{\frac{-\ln(1 - C_M \times T_{\text{readout}})}{T_{\text{readout}}} - \text{DCR}}{\text{PDE}} \tag{2.6}$$

that needs to be applied to each pixel using the individual calibrated PDE and DCR in addition to the measured count rate C_M.

Equation (2.6) expresses what we term exponential count loss. It is illustrated in Figure 2.17. Exponential count loss results in a compression of the sensor response such that the maximum value is never reached. That in turn leads to a much larger dynamic range of the camera sensor than would be possible if the response were linear. The downside is that the resolution of the count rate becomes worse with increasing intensities. For the majority of practical applications that operate with low light levels, this will not be a problem, and for the application with the occasional very high level of incident light, the sensor is still able to resolve it and give a proportional response.

The count loss was verified experimentally on SwissSPAD and the results are presented in Figure 2.17. A monochrome light source was used to illuminate the chip and reference photodiode was used to measure the increasing impinging photon rates while measuring the reported count rates. The results show good agreement with the equations developed above.

Furthermore with the correction of equation (2.6) applied to the measurements the vast majority of the pixels of the 512 x 128 array can be used to measure intensity. Only 0.3% of the pixels with a very high noise-level and therefore operating in the highly compressed region of Figure 2.17 cannot be used due to insignificant response to incident light. The distinction between the clock-driven and event-driven model is given in the next section.

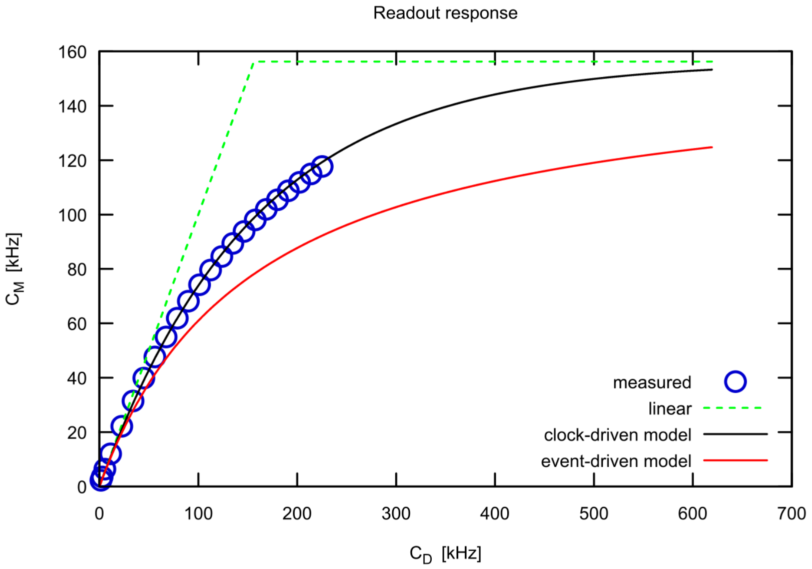


Figure 2.17: Illustration of exponential count loss. Exponential count compression for clock-driven and event-driven photon counting sensors lead to a larger useful dynamic range compared to a linear response at the expense of count rate resolution for high intensities that are often acceptable. The clock-driven model has been verified experimentally using SwissSPAD. The event-driven model is shown for comparison and has been evaluated in simulation for the same readout time.

2.5.6 Clock- and event-driven readout architectures

After the discriminator in the model pixel of Figure 2.16 with an event rate C_D the signal is processed and a count rate C_M is read out. There are two fundamentally different approaches for this part to consider.

The first approach is the *clock-driven* readout where a pixel is read periodically regardless of any events. This approach corresponds to SwissSPAD's mode of operation and as explained in the previous paragraph leads to

$$C_M = \frac{1 - e^{-C_D \times T_{\text{readout}}}}{T_{\text{readout}}}, \tag{2.7}$$

confirmed in the experiments.

Now consider the second approach that is the *event-driven* readout where a pixel is permanently connected to the output line that may be shared by other pixels and has a minimal dead time T_{readout} between successive signals. In that case, we can formulate the fraction of time where the readout electronic is blind as $N \times C_M \times T_{\text{readout}}$ with N the number of pixels sharing the line. Consequently, $C_D \times N \times C_M \times T_{\text{readout}} = C_D - C_M$ denotes the number of counts missed due to non-null T_{readout} and solving for C_M we get

$$C_M = \frac{C_D}{1 + N \times T_{\mathrm{readout}} \times C_D} \tag{2.8}$$

for which simulations are shown in Figure 2.17.

2.5.7 Gating

Many interesting applications for SwissSPAD are enabled by its precise gating circuit, used to define accurately the time-window for photon detection. The three shutter signals *off*, *recharge* and *gate* generated by the FPGA and distributed to all pixels in the chip are used to control the gating.

The three gating signals are driven from a common clock in the FPGA, which restricts their relative granularity. The clock itself, however, can be derived from an external reference and have a phase shift relative to that reference in increments of approximately 20 ps through the delay line in the FPGA PLL. This architecture permits the definition of a fixed window pattern that can be phase shifted using the PLL fine steps relative to the reference clock signal.

To measure the accuracy of the gating window in our system we used a 637 nm picosecond laser with 40 MHz repetition rate and 35 ps FWHM (Advanced Laser Diode Systems A. L. S. GmbH). From the laser synchronization signal, we derived the reference clock used to drive the gating. The laser is then made to illuminate the sensor and we define a shutter pattern over the 25 ns laser period that results in a sensitive window, which is significantly shorter.

This pattern is then shifted relative to the laser reference in steps of 20 ps using the PLL for fine shifting. Once a period of the fast clock is reached for the pattern delay a coarse shift is done by rotating the pattern. At each position in the reference period, an image of the intensity is acquired.

For each of the 1280 window positions in a 25 ns laser period we performed 255 readouts of the chip to accumulate an intensity value on 8 bits corresponding to each phase shift step. Figure 2.18 shows the resulting intensity response of the central pixel for illumination windows shifted between 0 and 25 ns. This waveform can be seen as the convolution between the signal from the laser and the instrument response of the gated SwissSPAD camera. Assuming an ideal pulsed illumination, we analyze this using a simple rectangle model for the peak, to determine rise and fall times and the position of the edges for all pixels of the array.

From the typical result in Figure 2.18, three points can be noted. The first is the very sharp second (falling) edge of the intensity signal. This edge corresponds to the end of the sensitive window that is defined by turning off transistor T4. This transistor blocks the memory such that no further events can toggle it regardless of the SPAD anode voltage or any photon activity. This transistor has minimal size, which leads to a small load on the signal tree and a very fast switching time.

The second point is the much slower first (rising) edge that corresponds to the beginning of the photon-sensitive window when the SPAD is turned on by restoring the anode voltage to ground using T2. Transistor T2 is larger compared to T4 and places a larger load on the driving signal tree leading to asymmetry in the switching time between the two transistors. Additionally the time it takes to restore the SPAD bias is not only dependent on the SPAD anode voltage at the outset of the operation but also on the photon flux during the time T2 is turned on. A photon arriving during the recharge period when the SPAD is already above breakdown can trigger an avalanche such that the bias is not fully restored to ground by the end of the recharge signal. Whether an event is recorded in this case depends on the final anode voltage of the SPAD and further photon activity. The ideal case would be that there is no photon activity during a recharge and the SPAD reaches its full operating excess bias before any events occur.

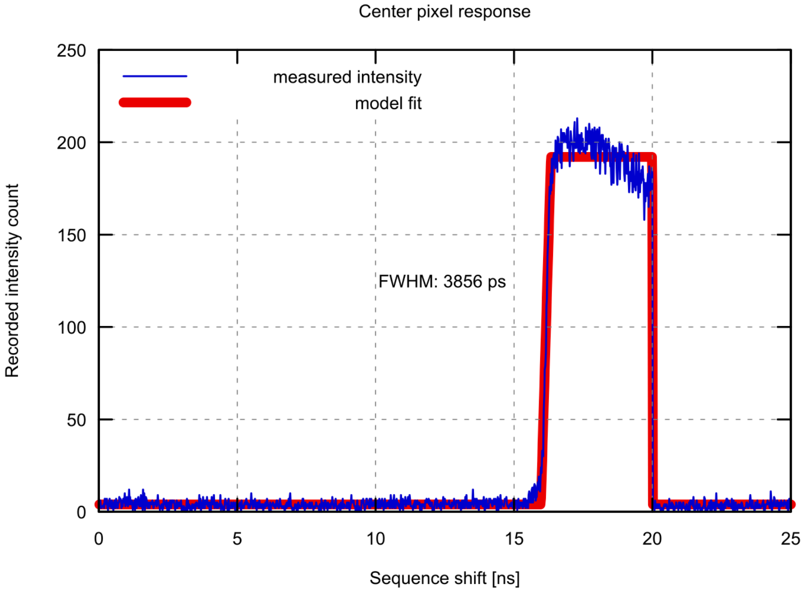


Figure 2.18: Pixel intensity response for sliding gate window. The gate window is shifted relative to the reference clock of the 40 MHz repetition rate laser used for the characterization. The first (rising) edge corresponds to the start of the window when the SPAD is turned on.

The third point to be noted about Figure 2.18 is that the active window does not have constant intensity. Rather the intensity is the highest right after the SPAD has been turned on and then it slowly decreases. This decrease, or non-constant window sensitivity is a typical characteristic for SwissSPAD, but none of the possible explanations given has been substantiated. One possibility is a slow increase in SPAD anode voltage through leakage current that makes the SPAD less sensitive over time but is insufficient to toggle the memory before the window ends or a regular (DCR-induced) avalanche occurs. On the other hand, the low intensity part of the response is constant for this window pattern.

The model peak used to evaluate the pattern uses six parameters to define the piecewise linear function shown behind the response in Figure 2.18. The parameters are: constant levels of floor and peak, position and width of the peak, and rise and fall times. A fit on the data is performed using Levenberg-Marquardt minimization to find the parameter values on each pixel.

In performing our simple fit we assume that the pixel responses are well behaved which turns out to be true for the vast majority of the pixels as indicated by the measurement results. The results are only indicative for the shutter pattern used here however and changing the pattern can lead to very different results that are no longer nicely approximated by a rectangle. Especially increasing the repetition frequency of the active window degrades results.

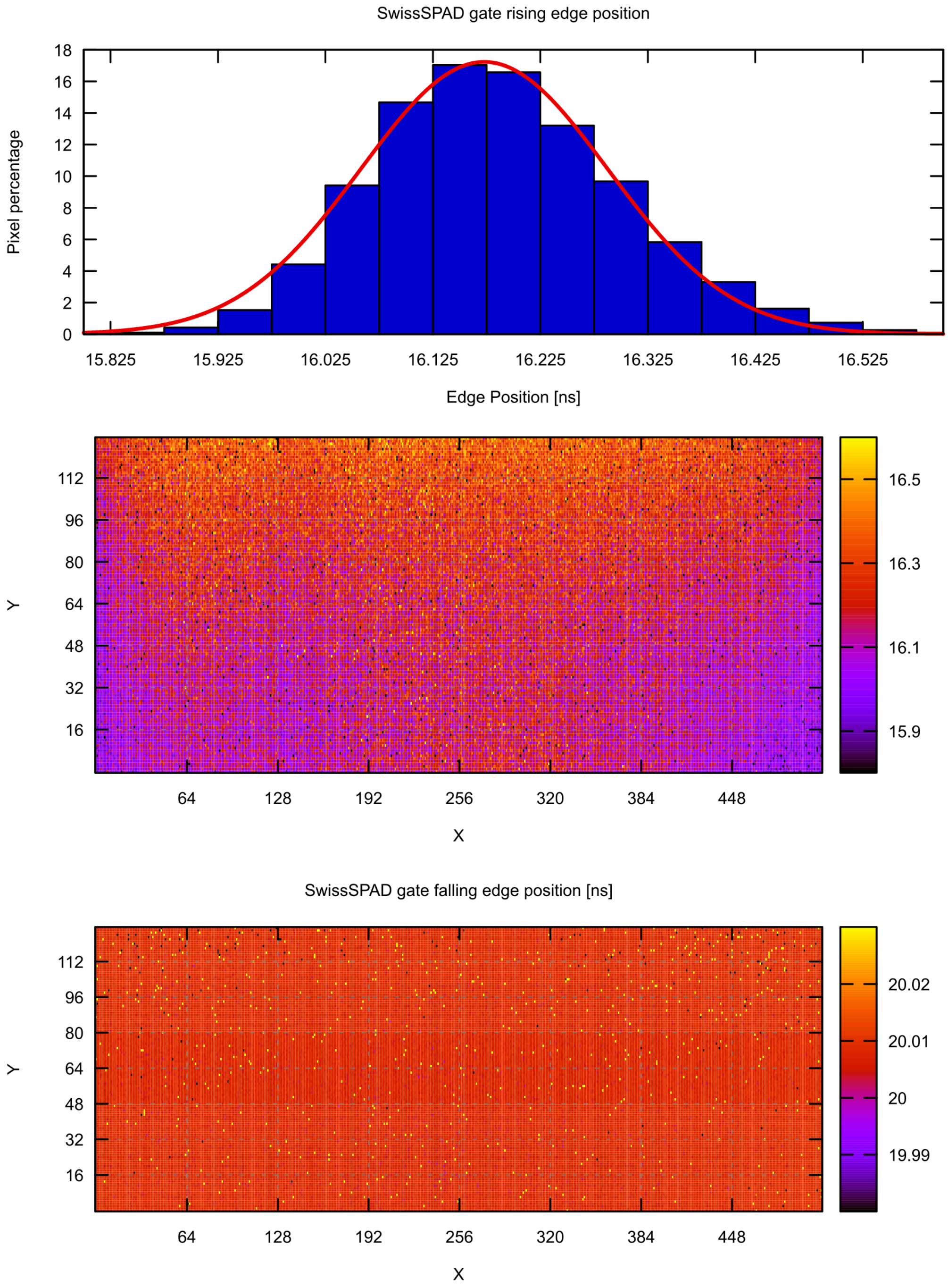


Figure 2.19: SwissSPAD shutter performance. The two top plots show the rising edge position as histogram and over the array. From the spatial distribution, the skew introduced by the large load on the recharge signal is evident. The histogram shows the extent with a FWHM of 271 ps. The position of the falling edge is shown in the bottom picture with the color bar spanning 50 ps. All pixels responses are contained in one PLL shift step and occur at the same time.

Figure 2.19 shows the timings for the rising and falling edges of the pixel responses over the full array. As noted for the center pixel, the falling edge is quasi instantaneous with a response time below 50 ps for well behaving pixels. The edge position shown for each pixel testifies additionally to the minimal skew. The color bar covers a range of 50 ps only and the uniform color indicates simultaneous edges over the full array.

The rising edge position shown spatially across the array in the middle plot of Figure 2.19 shows the systematic skew in the recharge due to the large load on the signal. The histogram at the top shows the spread of the edge positions, which has a FWHM of 271 ps, part of it systematic.

2.5.8 Readout

In the assessment of the readout performance for the system, we need to consider not only the rate at which the chip itself delivers the pixel data, but also the rate at which the camera system can absorb the data further down the processing chain. By having a configurable bit-depth in the accumulation module in the FPGA one level of data compression is introduced immediately after the chip. The accumulation module is capable of incrementing counters for 128 pixels in parallel and runs at the data rate of the chip. Once the configured number of accumulations is performed, the data has to be sent out during the time the next multi-bit frame is accumulated in a second buffer.

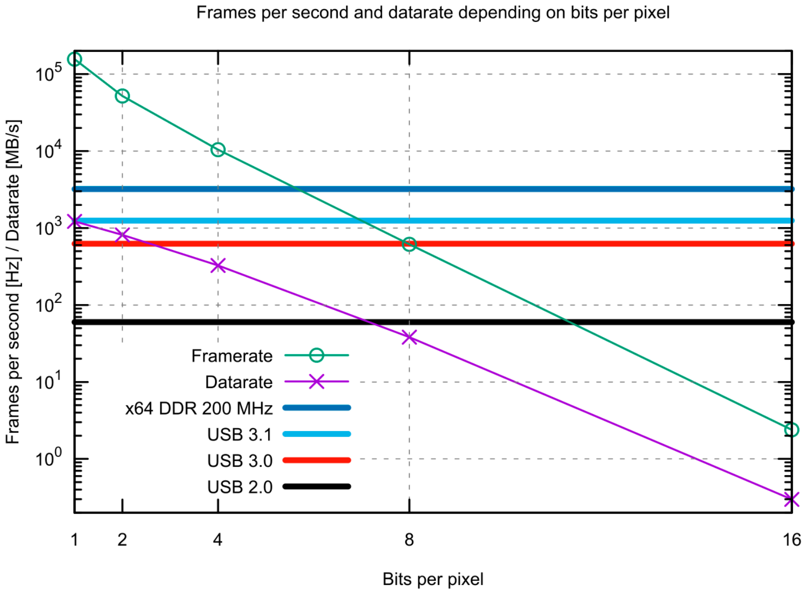


Figure 2.20: Relationship of frame-rate and data rate to image bit-depth. Only tried image accumulation bit-depths of power of two are shown. A custom firmware could allow intermediate levels and domain specific compression to lower the data rate.

The rate of accumulated frames delivered by the camera is calculated as:

$$f_{\text{ACCUM}} = \frac{f_{\text{SENSOR}}}{2^{\text{bpp}} - 1}\ [s^{-1}] \tag{2.9}$$

with f_{SENSOR} the frame rate for one bit images from the sensor and bpp the number of bits per pixel. The datarate is given by

$$r = f_{\text{ACCUM}} \times \text{bpp} \times N \left[\frac{bits}{s}\right] \tag{2.10}$$

with N the number of pixels in the image.

Figure 2.20 shows the relationship between the number of bits per pixel and the framerate and data rate assuming fully used dynamic range and resolution. The bandwidths of four typical back-end transmission technologies that have been used with SwissSPAD (except for USB 3.1) are indicated. For the USB connection, it has to be noted that raw data rate is given while usable data rate is typically about 20 % lower. The only technology fast enough to sustain the SwissSPAD data rate for longer sequences of images is DDR-type DRAM. Unfortunately, the size of DRAM in a camera is always limited and filled very fast with a maximum rate of approximately 1.3 GB/s.

2.5.9 High speed imaging

SwissSPAD can be used in a standard wide-field optical setup as high frame rate single photon sensitive camera. Image series with configurable intensity resolution can be captured using global or rolling shutter as discussed. The maximum frame rate that can be achieved using global shutter is determined by the shutter time plus the fixed readout time of 6.4 µs. In a rolling shutter, the minimum frame time of 6.4 µs separates successive readouts of the same line and 156,250 1-bit frames per second are read. The shutter duration in this case is a fraction of the full frame duration and determined by the photosensitive time between two successive readouts.

The FPGA accumulates the sensor output at a rate of 10.2 Gbps and sends the frames to the attached memory for intermediate storage. As shown in Figure 2.21, single frames with different bit depths can be extracted from a high-speed movie shot with SwissSPAD. Even at a resolution of 1-bit per pixel, the sine wave traced by the electron beam impacting the oscilloscope’s phosphorescent screen (as well as the screen’s persistence) can be clearly distinguished. The maximum recording length is only limited by the size of fast storage memory connected to the FPGA. For special applications, domain-specific data compression and triggering could be implemented using the reconfigurable control logic in order to increase recording length.

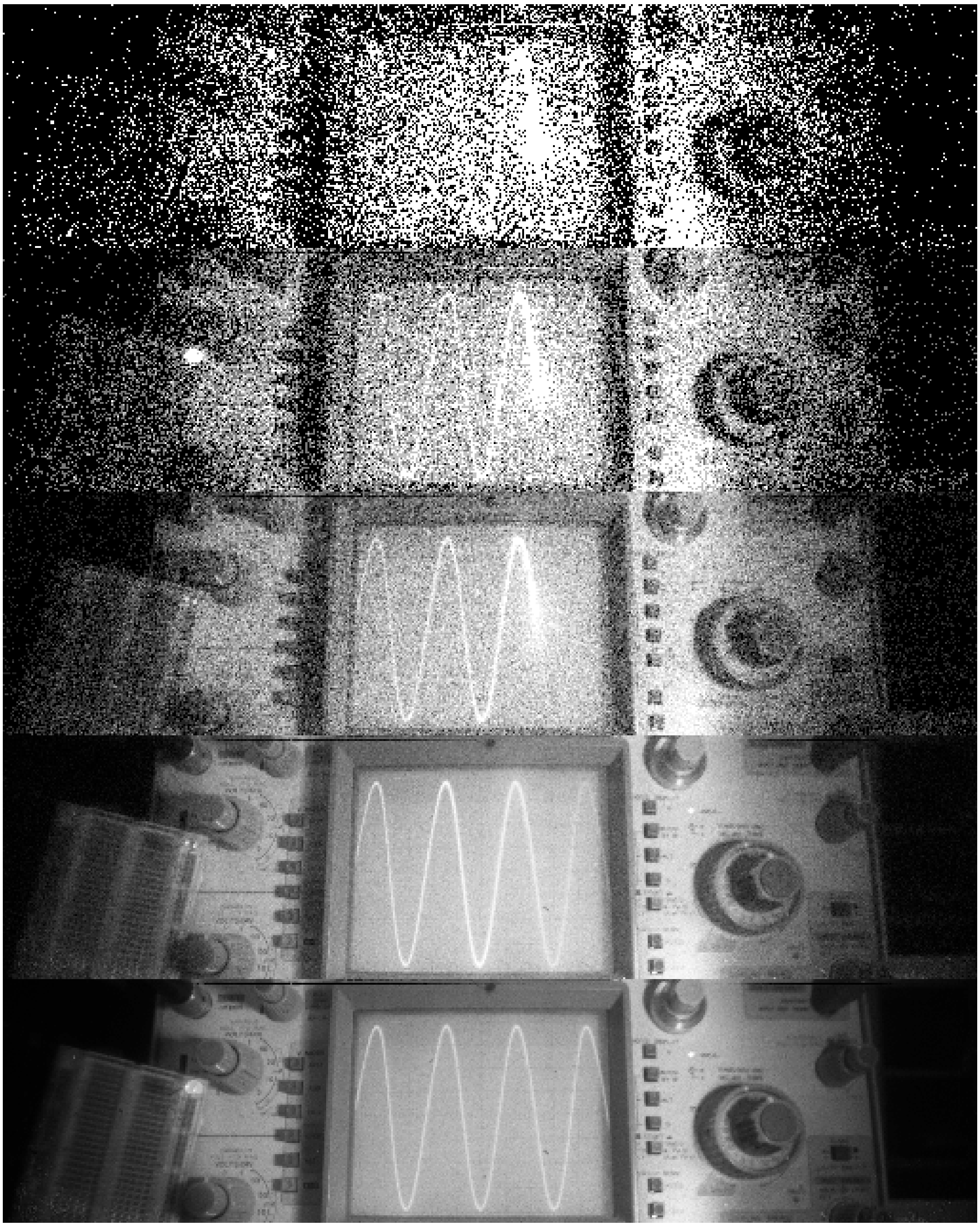


Figure 2.21: SwissSPAD high-speed intensity images with configurable bit-depth. Five intensity images of an analog oscilloscope screen acquired with 1-, 2-, 4-, 8- and 16-bit intensity resolution. The electron beam tracing the sine wave can be clearly distinguished at the highest frame-rate of 156 kHz with 1-bit intensity resolution.

2.5.10 Performance summary

The performance of the SwissSPAD system covers all aspects about the SPAD, the in-pixel gating and array readout performance. The FPGA interface allows for different operation scenarios as outlined in section 5.1 about applications of SwissSPAD. The main performance figures of the system are summarized in Table 2.1.

Parameter	typ.	Unit
Process	$0.35\mu m$	HV CMOS
Chip size	13.5 x 3.5	mm^2
Format	512 x 128	pixel
Pixel pitch	24	μm
Dead time	100	ns
Nominal fill factor	5	%
Fill factor with microlenses	30-50	%
Photon detection probability		
($V_e = 3V$, $\lambda = 450nm$)	27	%
Spectral range		
($V_e = 4V$, $PDP > 5\%$)	350-850	nm
Median dark count rate		
($V_e = 4.5V$)	367	Hz
Crosstalk, measured in [76]	<3.5	%
Minimal gating duration (FWHM)	4	ns
Clock frequency	80	MHz
I/O bandwidth	10.24	Gbps
Frame readout time	6.4	μs
Sensor power consumption		
(idle)	660	mW
(active)	1650	mW

Table 2.1 SwissSPAD performance parameters.

2.6 Transition to new FPGA board

A key lesson learned with SwissSPAD was that the bandwidth of the complete camera needed to be considered and adequately sized. The FPGA board used to control SwissSPAD was a legacy of older systems and did not include a convenient high-speed interface with a computer. The only option we had was to use DDR type memory modules to provide larger fast storage for the camera for longer high-speed sequences.

The issue is now being addressed on newest cameras using the SwissSPAD sensor that use the same base as the LinoSPAD camera presented in the next chapter. This includes a USB 3.0 interface and a new FPGA generation for increased data rates.

Chapter 3 LinoSPAD: a reprogrammable SPAD line sensor

The LinoSPAD camera presented in this chapter is quite different from SwissSPAD presented in the previous. LinoSPAD has only a single line of 256 pixels with the same pitch of 24 µm, but no integrated functionality beyond quenching and recharge. This leads to a remarkable fill factor of over 40%.

LinoSPAD is connected to a FPGA using one line per pixel to connect the digital SPAD output with a FPGA input. This makes it possible to reconfigure per-pixel circuits like counters or TDCs that would traditionally be implemented as fixed functions in the sensor. The reconfigurability of the pixel through the tight coupling of the sensor with a FPGA makes it easy to adapt to different applications where it can be used in proof-of-concept type systems for prototyping specialized cameras.

The chapter starts with a description of the sensor followed by a description of the FPGA architecture and the evaluation of the camera system comprised of both. Compared to SwissSPAD, the FPGA section is of greater importance here as it includes the logic that would have been in the sensor had we followed a more integrated approach.

3.1 Sensor

At the front-end of the LinoSPAD system is the SPAD line sensor – LinoSPAD – with a comparatively simple architecture consisting of a line of SPAD pixels without processing logic and with each pixel connected to one output pad. This sensor chip is bonded to an interface PCB, which connects it to the FPGA carrier board where all the processing takes place.

3.1.1 Chip architecture

The central design goal for LinoSPAD was to make a line of SPADs in a mature technology performing well in the fill factor, noise and photosensitivity metric and gaining its versatility from a FPGA tightly coupled to it. Because of this, only the minimal circuitry, needed to operate a line of SPADs, is integrated on-chip.

The fabrication of the sensor is in the same high-voltage 0.35 µm process as SwissSPAD and uses the same diode structure described in chapter 2. The fill factor could be increased to 40% by optimizing the shape of the diode and using a shared well for the cathodes of the SPAD line. The pixel circuit is comprised of the SPAD with a shape of a rounded square that is connected to a quenching transistor and a series of two inverters. The inverters are used to first digitize the anode voltage and secondly to drive an output pad. The pixel circuit is shown in Figure 3.1a) and its layout in Figure 3.1b). The full chip is an array of 256 pixels sharing common operating voltages.

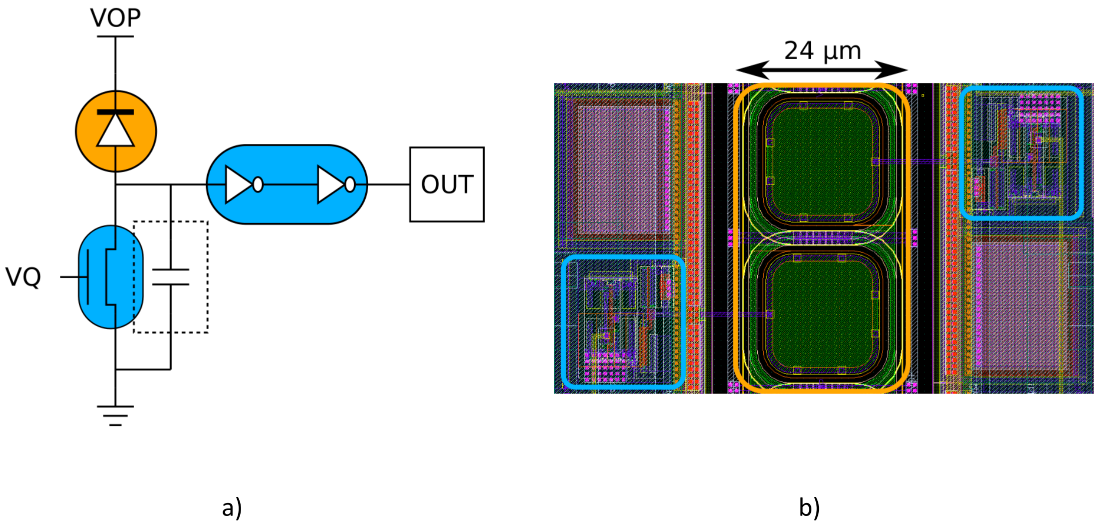


Figure 3.1: a) Circuit of one LinoSPAD pixel, b) Layout of a pair of pixels. The quenching and thresholding inverters are alternating between both sides of the line of SPADs.

Due to the high number of output pads required for a relatively small chip surface occupied by essential circuitry a staggered layout was chosen for the pad-ring. In addition to the usual pad-ring around the sensor with 192 pads, four blocks of 30 pads each were added inside the chip for 312 pads. The chip is 6.8 mm by 1.68 mm in size and is shown in Figure 3.2. The small size and conservative design in comparison with SwissSPAD has turned out to be very advantageous for the fabrication yield. No single pixel was found defective so far.

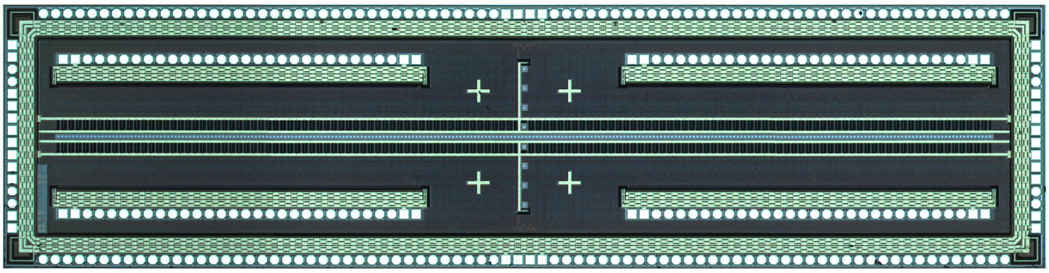


Figure 3.2: Micrograph of the LinoSPAD sensor. Vertically in the center is an array of eight auxiliary pixels and four alignment crosses surround the center. Additional to the ring of 192 bonding pads there are 120 bonding pads in four blocks inside the ring.

3.1.2 FPGA interface card

The LinoSPAD line sensor has 256 independent output lines for its pixels, which need to be connected to the processing logic. Since there was no suitable board to be found, we decided to design our own baseboard for chip evaluation purposes. The goal was to have separate PCBs for the chip and the processing logic with the sensor PCB being very light-weight to simplify assembly and changing of the chip. The baseboard on the other hand should have enough I/O to support the sensor, a reasonable sized FPGA for immediate interface logic and a fast connection to a computer. An expansion header provides capability to access further processing power with other (FPGA) cards.

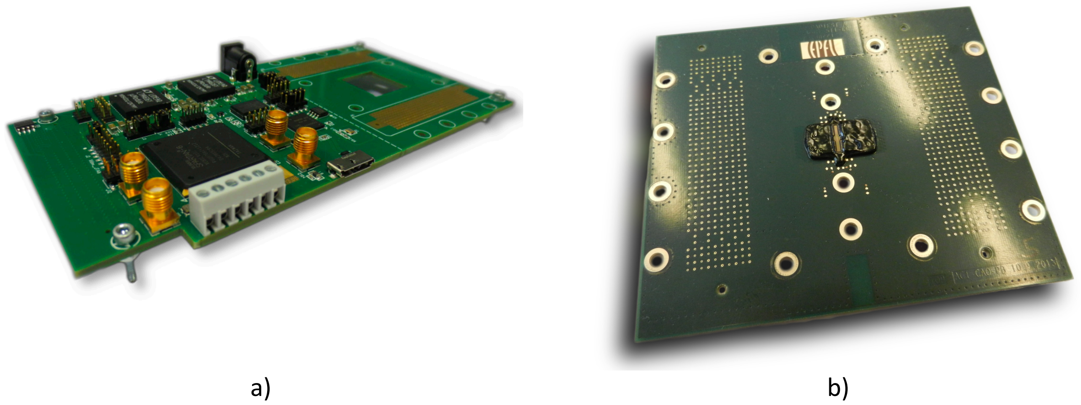
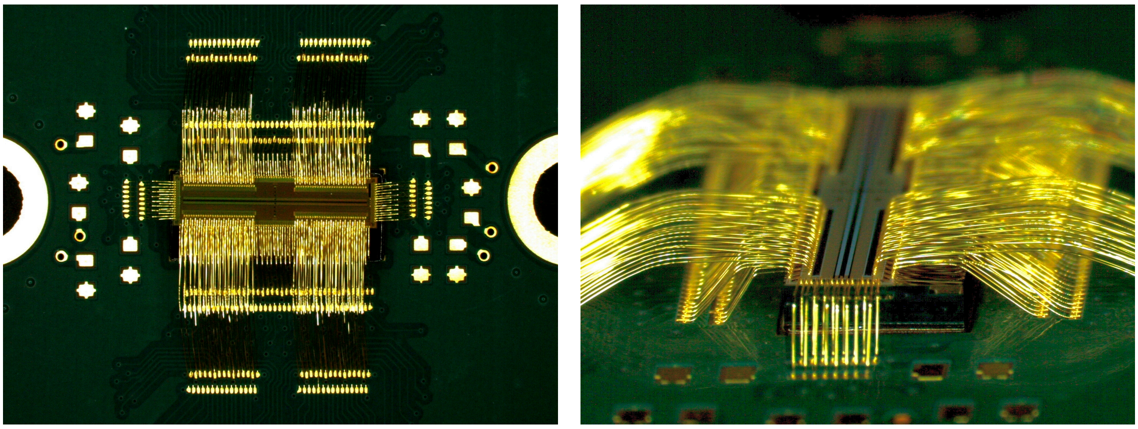


Figure 3.3: The mainboard a) and daughterboard b) for the LinoSPAD camera. The contact array in the background of the motherboard connects to the sensor card. The hole in the PCB gives access to the back of the sensor card for possible cooling. The LinoSPAD daughtercard contains only the sensor and pads for optional decoupling close to the chip to simplify assembly.

The sensor carrier PCB has a pad to bond the chip in its center and four rows of pads on the long sides to which the wires from the sensor are bonded. Starting from a pitch of 80 µm between the pads on the sensor this resulted in a pitch of 160 µm between pads on the PCB. This lead to PCB feature sizes of 50 µm, that are at the limits of current fabrication technology. A picture of the LinoSPAD daughterboard is shown in Figure 3.3b). Two pictures from the bonding of the chip to the PCB are shown in Figure 3.4.



Images courtesy of Microdul AG, Zürich.

Figure 3.4: LinoSPAD bonding images. The sensor is glued and wire-bonded to a carrier PCB. The PCB layout was made after specifications from the bonding company. The angles of the bondwires are controlled to give at least 90° aperture on the sensor. After bonding, the wires are protected using a resin as seen in Figure 3.3b).

The LinoSPAD motherboard shown in Figure 3.3a) contains the FPGA and USB 3.0 communication chip together with configuration memories and power supplies. A high-density spring connector array is used to connect the sensor cards without soldering for simplified assembly and maintenance. The hole in the PCB provides access to the back of the sensor card to integrate possible cooling.

For the FPGA we decided to use a Xilinx® Spartan™ 6, which offers a good trade off in performance and cost. It has enough user-defined I/O to connect with the SPAD sensors of this thesis and enough resources to absorb the data rate they produce. A high-performance FPGA would offer more integrated memory and faster switching logic, but at a prohibitive price point for system prototypes, which are realized as dedicated, application specific circuits if they prove to be useful.

3.2 FPGA architecture

The task of the FPGA connected to a LinoSPAD sensor is to detect the output signals generated by photons hitting the SPADs. Generally, we are interested in knowing the number of photons and their arrival times with respect to a reference clock. From the digital signal of the SPAD, a pulse with fast rising edges, we want to digitize the analog time information it contains with a reasonable precision, in the picosecond range. The subsequent sections detail how this is done in the FPGA.

3.2.1 Global architecture

The LinoSPAD architecture is a trade-off with respect to the capabilities of the FPGA regarding time-to-digital converters. FPGAs per se, are not made for the implementation of TDC circuits that require a form of analog processing to determine the arrival time of a signal. However, as the time-sensitivity of a SPAD sensor is one of its important advantages compared to other cameras we decided to implement a TDC array as large as the FPGA can reasonably handle. This may offer better time resolution than achievable through digital sampling.

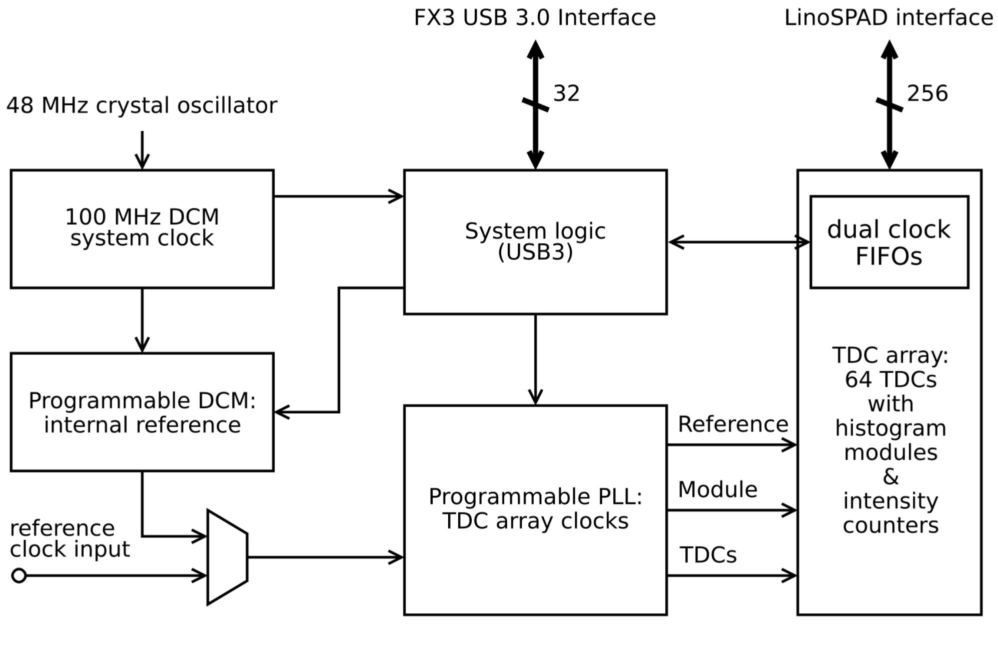

FX3 USB 3.0 Interface
LinoSPAD interface
32
256
48 MHz crystal oscillator
100 MHz DCM system clock
System logic (USB3)
dual clock FIFOs
TDC array: 64 TDCs with histogram modules & intensity counters
Programmable DCM: internal reference
Reference
Module
TDCs
Programmable PLL: TDC array clocks
reference clock input

Figure 3.5: Architecture of the LinoSPAD FPGA firmware. The LinoSPAD firmware contains two major parts. One is the clock management used to synchronize with an external time reference, the other is the array of time-to-digital converters connected with the SPAD sensor.

After initial design explorations, we fixed the number of TDC modules at 64 and decided to assign four pixels to each TDC and read them one at a time. To compensate for the lack of TDCs working concurrently, the histogram memory was made using a double buffering scheme such that minimal time is lost when switching the TDC to another pixel. Additionally a 32-bit intensity counter was connected to each pixel working in parallel with the TDC modules.

The FPGA firmware architecture is shown in Figure 3.5. The FPGA operates in response to requests received over the USB3 interface, by performing data acquisition and processing, and sending results back to the computer. The main components of the LinoSPAD design are the programmable clocks to synchronize the TDC modules to a wide range of reference signals, and the TDC array operated by a simple state machine. In the following sections, the implementation of the TDC array is explained and analyzed in detail.

3.2.2 Clock architecture

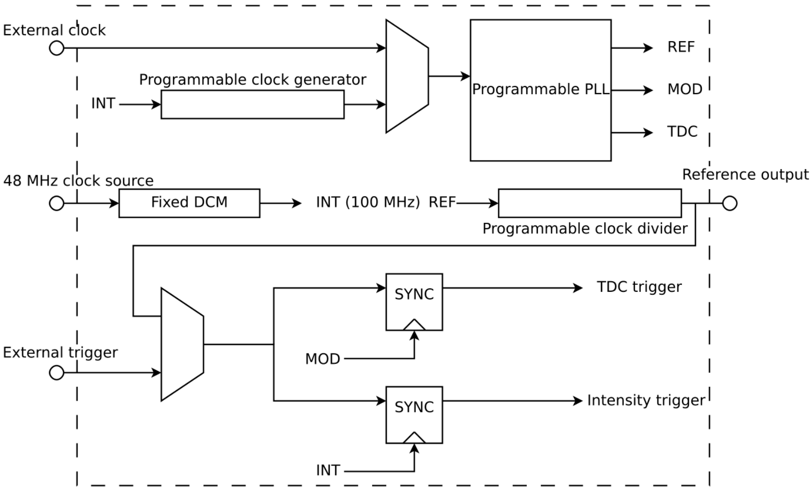


Figure 3.6: Schematic view of the LinoSPAD FPGA clock architecture. This is a functional representation of the clocking circuits in the LinoSPAD FPGA firmware shown Figure 3.5. The input to a programmable PLL can be switched between internal and external reference signals. The PLL generates related clocks for the time-to-digital conversion and processing logic. A clock output is available to use LinoSPAD as master reference.

The time-to-digital converters function by sampling a delay-line at high frequency and the time information that is extracted will be relative to the sampling clock. In order to relate the timestamps to an externally generated reference clock, or drive an external circuit coincident with the sampling, the systems must be synchronized. In Figure 3.6, we show the relevant parts of the FPGA firmware used for this purpose.

A programmable clock multiplier, a PLL, is driven either by an external clock source or by a clock generated internally from the main FPGA clock source. This allows the LinoSPAD system to act as master or slave when connected with an illumination system. From this base reference, the PLL derives two additional synchronous clocks. One is the sampling clock for the TDC delay-lines, that needs to run at 400 MHz, and the other is the processing clock at 1/3 of this frequency. The input reference is also synchronously regenerated to serve as time-base in the measurements.

3.2.3 TDC core

The TDC core is the central hardware module of the LinoSPAD design, where the timing information from a pixel signal is extracted. Four pixels share one TDC core in an opportunistic sharing scheme. As long as there are no concurrent events on multiple pixels, events are recorded for all pixels. In case there are simultaneous events, the user can decide to ignore the pixel address and record the events nevertheless. Another possibility is to mask 3 out of the 4 pixels and to activate the set of pixels one by one. When a valid event is detected in the delay line, the sampled state of the line is encoded to a time code and forwarded to the processing module, where it will be accumulated in a histogram or recorded as raw timestamp. Figure 3.7 shows the core logic modules, with delay lines, encoder and further processing. The main modules are described in the following subsections.

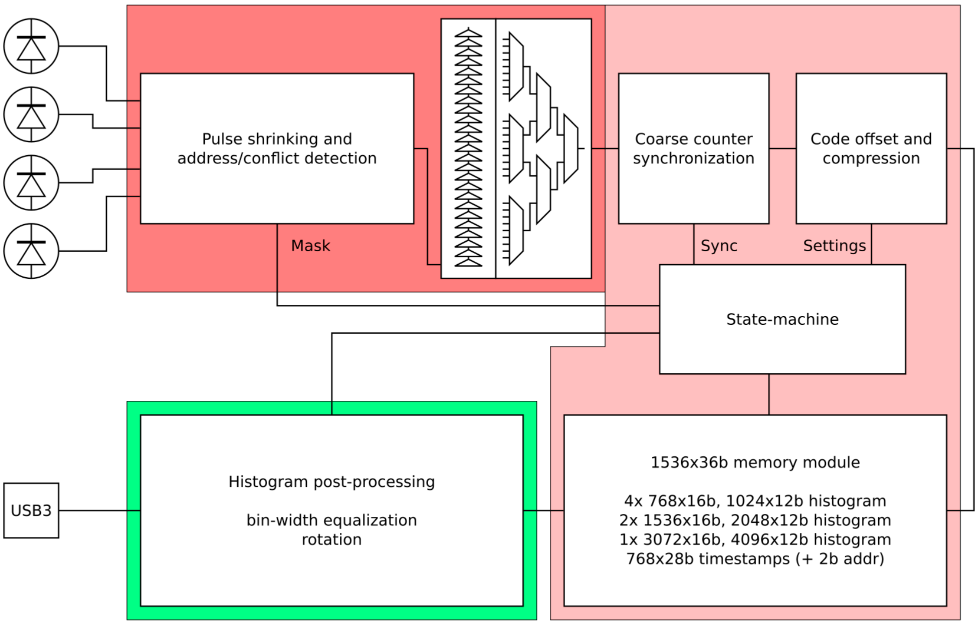

Pulse shrinking and address/conflict detection
Mask
Coarse counter synchronization
Code offset and compression
Sync
Settings
State-machine
USB3
Histogram post-processing
bin-width equalization
rotation
1536x36b memory module
4x 768x16b, 1024x12b histogram
2x 1536x16b, 2048x12b histogram
1x 3072x16b, 4096x12b histogram
768x28b timestamps (+ 2b addr)

Figure 3.7: FPGA core logic of the LinoSPAD system. This block diagram expands the TDC array of Figure 3.5. The red part is replicated 64 times with each part processing four pixels. The input and delay line part (dark red) are running at 400 MHz, the remaining logic at 133 MHz. The post-processing (green) is shared among all 64 TDCs.

Shared input stage

The input stage is connected to the FPGA input buffers for the four signals sharing one delay line, and decodes the address of the currently active line and determines whether there was a conflict. While doing this, the timing of the signal must be retained, and crosstalk between the four signals prevented. This requires careful design of this block, taking into account the details of physical mapping inside the FPGA.

The first part of the input stage consists of a pulse shrink circuit for each signal as shown in Figure 3.8. This pulse shrinking is employed to minimize the time a signal occupies the shared delay line. The circuit takes into account the limitations on clock signals of many FPGAs and avoids routing the input signal to a clock load. Instead, the same global clock signal is used to achieve the pulse shrinking for all input signals. An input pulse is cut short as soon as it has been sampled by two registers clocked at 400 MHz. Depending on the arrival of the signal relative to the clock, this results in an output pulse between 2.5 ns and 5 ns neglecting FPGA path timing. For its physical implementation, the circuit is constrained such that the final AND gate is realized by the LUT through which the signal passes to reach the first register. The mask register and second delay register are constrained to have a worst-case output pulse length of 7.5 ns.

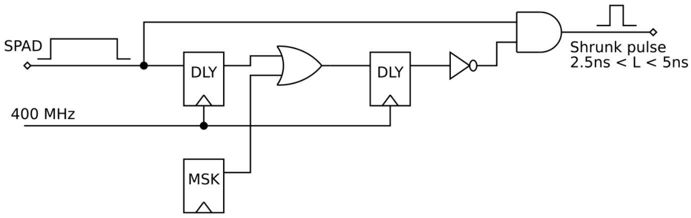


Figure 3.8: FPGA pulse shrinking circuit. This circuit is replicated 256 times, for each input signal from the SPAD sensor. A long pulse on the SPAD line is cut off after two periods of the 400 MHz clock when the second flipflop in the chain registers it. This circuit preserves the timing of the signal in the output pulse that is connected to the delay line trigger block.

The second part of the input stage generates the signal that is injected in the delay line, and decodes and verifies the address of the input signal. The circuit with the FPGA mapping is shown in Figure 3.9. This circuit occupies one slice [78] on the FPGA, in order to have the smallest possible timing uncertainties. Four registers are used, two for the address, one to indicate a valid address[1] and one to indicate the presence of a signal in the delay line. The registers are clocked with the same 400 MHz clock used for the pulse shrinking circuit and the delay line described in the next section. The delay from the address validation to the start of the delay line ensures that no crosstalk between the four contending signals occurs.

Delay line

Registers inside the FPGA are limited in the maximum clock frequency at which they operate reliably. Sampling an input signal with a register will give a time precision limited by the fastest possible clock period, i.e. in the nanosecond range. To determine the arrival time of a signal with a better precision than the clock frequency, the signal is delayed through a long carry chain, a dedicated structure usually employed to implement fast arithmetic circuits. The registers connected to the chain are used to sample its state, in order to determine how many registers a signal reached at the instant when the clock signal arrives. Carry-chain based TDC architectures appear increasingly in diverse fields, often related to nuclear science and high-energy physics, where they are used to measure arrival times of photons or other particles [79,80].

The core element of the whole FPGA TDC logic is thus the delay line constructed from the dedicated carry logic inside the programmable logic elements. The connections through the carry logic provide the fastest paths from one register input to the next and consequently give the best timing resolution possible. In Spartan 6 FPGAs, the carry logic can be accessed in primitives of four-bit length, which map to the corresponding hardware blocks. The carry chains are the only, and coincidentally fastest, routing resources we

[1] Where exactly one line is active.

have direct control over in the Spartan 6 architecture. For all other paths, the delay depends on the implementation done by the place-and-route tools from Xilinx. While the functionality is that of a (ripple) carry chain, the actual implementation is done using look-ahead to reduce worst-case delay in arithmetic circuits. This has the adverse effect of causing very non-uniform differential delays from one output to the next, and occasionally bubbles, where a conceptually later output activates before the earlier one. An additional source of unpredictable delay in long chains is the crossing between local clock domains in the FPGA, where the regularity of the chain is broken. Also, variations from one FPGA to another need to be considered, as shown later. Statistically however, the mean delay between the outputs is of the order of 20 ps.

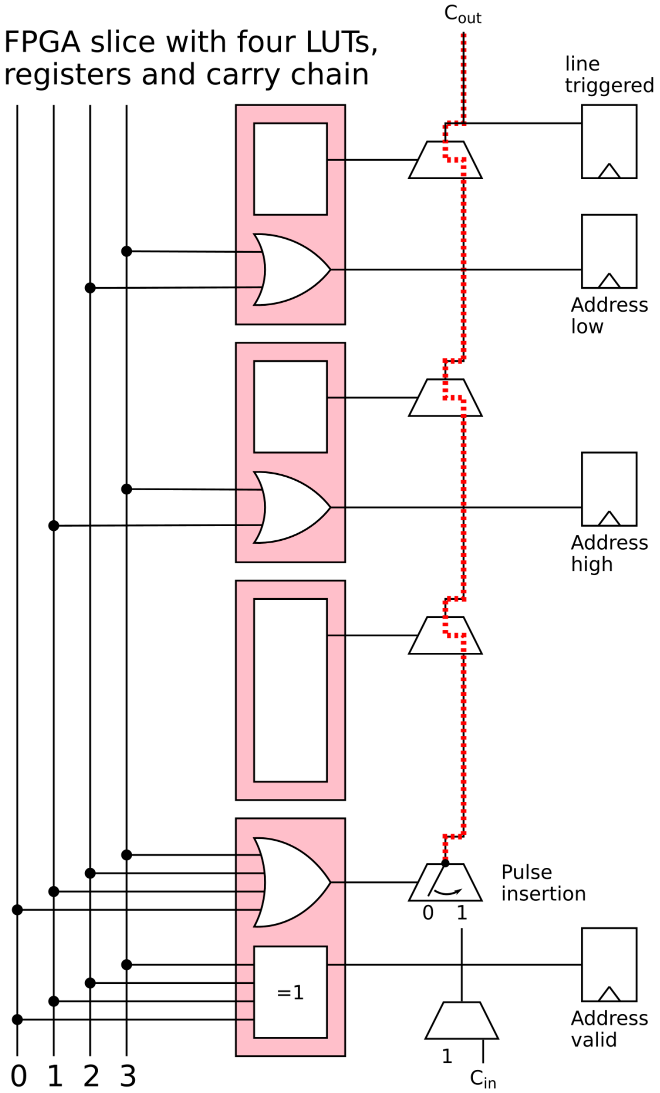


Figure 3.9: Delay line pulse insertion and address decoding. The (shortened) pulses from four SPADs are combined to insert the signal in the carry chain and decode the active line. The circuit is constrained to the slice (physical location) at the start of the delay line on the FPGA.

The delay line outputs are sampled with a 400 MHz clock, the maximum allowed frequency for this architecture, to have the shortest possible delay line. The shortest possible delay line needs to be longer than the sampling period in order to achieve the highest precision and not lose any events. We found that 35 carry primitives reaching 140 registers cover the 2.5 ns clock period for all operating conditions. A shortened delay line and its operation are shown in Figure 3.10.

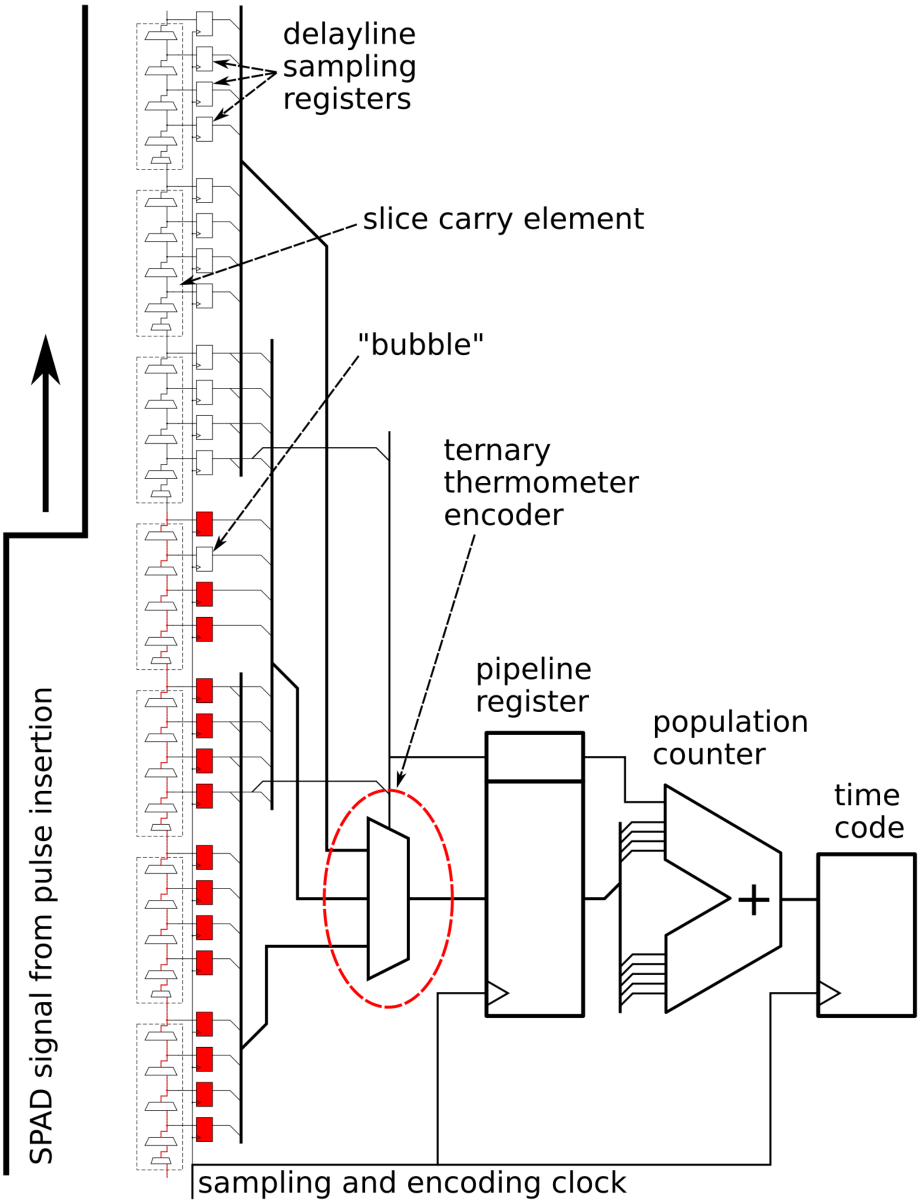

delayline sampling registers
slice carry element
"bubble"
ternary thermometer encoder
pipeline register
population counter
+
time code
SPAD signal from pulse insertion
sampling and encoding clock

Figure 3.10: LinoSPAD FPGA TDC delay line and encoder schematic. This schematic details the TDC delay line implementation and operation as a pulse from the input stage shown in Figure 3.9 propagates through the line. The circuit is fully pipelined for 400 MHz operation. In the encoder, a ternary thermometer coder reduces the bit-count before the bits are summed up to find the number of active elements corresponding to the edge position in the line.

Encoder

From an initial state of a zero input and all zero outputs of the delay line, a one input will cause outputs to switch one by one after the propagation delay. Regardless of any bubbles, the number of ones will increase over time to reach an all one output after more than 2.5 ns. The term *thermometer code* is used for the output of the delay line where the information is contained in the position of the most advanced one similar to the tip of the liquid column in a mercury thermometer. The purpose of the encoder is to generate a binary code representation of the 140 bit thermometer output of the delay-line.

The goal in the design of the encoder was to keep the logic running at 400 MHz while making it as compact as possible and retaining the full resolution. The design is a mixture of a thermometer to binary encoder for efficiency and a population counter to be able to cope with possible bubbles. The problem of bubbles in thermometer encoders is studied for non-FPGA designs as well [81]. A simplified design of our encoder is shown in Figure 3.10.

The design is fully pipelined over six stages to output an 8-bit count of the number of active bits in the input. The first two stages implement a ternary search for the delay line segment containing the event transition. The bits at one third and two thirds of the input length are used to select the correct segment. This architecture is more efficient than the naive binary search for the Spartan 6 architecture as it makes better use of the 6-input look-up-tables. A binary search maps to 3-input functions (1 selection bit, 2 segments) and has an output size of half the input size. The ternary search on the other hand maps to 5-input functions (2 selection bits, 3 segments) and has an output size of one third of the input size. From the observation that bubbles do not span more than 2 slices (8 bits), the segments are overlapped such that a bubble in the selection bit does not give a wrong result.

The stages after the ternary reduction implement a direct population count for the remaining 24-bit segment and add up the values for the final output. Careful pipelining in this part ensures that each stage corresponds to only one level of look-up tables. Since a population count is inherently tolerant of bubbles, no special modification to this part was needed.

To determine the validity of the encoded data, the input of the delay line is sampled separately. The line is in a valid state only, when it was fully empty before being triggered (again). The line is empty when two zeroes have been sampled at the input, since it is not longer than 5 ns. Therefore, if we sample a one at the input, we know that all outputs of the line are determined from this event exclusively. This information on the code validity, together with the information on the address validity, is used to decide if the event is further processed.

Rate reduction

Memory blocks in the FPGA cannot run at the 400 MHz frequency we use for the delay line sampling. In addition, keeping the high frequency for further processing would overly complicate the design, due to the heavy pipelining necessary and most likely still not allow us to implement many TDC blocks, due to limited availability of fast routing resources. A rate reduction module reduces the TDC code rate after the encoder from 400 MHz to 133 MHz. For one 133 MHz clock period of 7.5 ns, 3 TDC samples are evaluated to see if there is a valid one. A valid sample is selected and tagged with a 2-bit clock offset. The resulting 10-bit code is forwarded to the processing module. Figure 3.11 illustrates the rate reduction circuit. Since at most one of three values can contain a valid code, no events are lost due to this rate reduction.

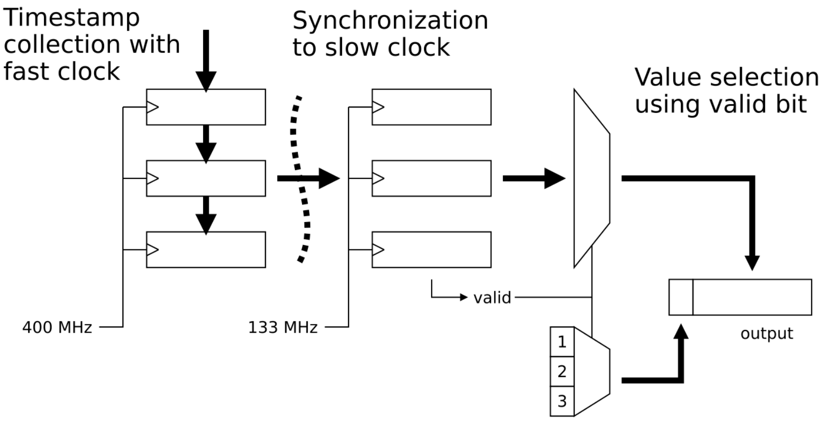

Timestamp collection with fast clock
Synchronization to slow clock
Value selection using valid bit
valid
400 MHz
133 MHz
1
2
3
output

Figure 3.11: LinoSPAD FPGA rate reduction circuit. Three steps reduce the processing rate in the TDC from 400 MHz to 133 MHz. 1) Timestamps are collected in the fast clock domain. 2) Collected values are synchronized to the slow clock. 3) One of three values is selected based on the valid bits.

3.2.4 Postprocessing

A time-to-digital conversion result from the rate reduction module is a code between zero and 3 × 140 representing a span of 7.5 ns without any link to another time reference. The first step in the post-processing expands the range of 7.5 ns using a coarse counter running at 133 MHz. This coarse counter uses 28 bits; such that a maximum resolvable time range of $2^{28}/140 \times 2.5\ \mathrm{ns} \approx 4.8\ \ \mathrm{ms}$ is reached. The delay line precision is not affected by this range expansion. The coarse counter is reset synchronous to the reference clock (from the illumination) of the system and has a period of a multiple of the TDC period of 2.5 ns.

Histogramming

The LinoSPAD firmware creates time-of-arrival histograms from the timestamps using internal memory of the FPGA. The available memory capacity imposes limits on the maximum number of histograms, their length and resolution. To make the most of the limited memory in different applications, we take advantage of the flexibility of the FPGA and make memory related trade-offs configurable. Three dedicated block-RAM per TDC are used to create histograms of the data coming from the 4 connected pixels. One memory block is 512 × 36 bit large, which is sufficient to store 4 histograms of 768 bins at 16 bit resolution, which in turn cover a full period of 80 MHz or faster. For slower reference periods, part of a histogram can be stored. 16-bit resolution per bin is sufficient for long integration times and enables a simple readout using 32-bit words. As the memory contains parity bits, we found a way to use a part of those as well to store 1024 bins per histogram at 12-bit resolution. Upon readout of 12-bit bins, they are expanded to 16 bit to use the same processing logic for both modes. A simple extension is the possibility to share the memory among multiple pixels, if the position information is not needed, or if some pixels are masked in order to acquire histograms with up to 3072 bins at 16 bit or 4096 bins at 12 bit.

In an alternative mode, useful for very low intensity applications, the memory module can be configured in a timestamp recording mode, where the memory blocks are used to store the timestamps instead of accumulating them in a histogram.

Once the acquisition is completed, the histograms are read out and the memory is reset. During readout, the histograms can be processed in real-time to reduce static DNL/INL caused by delay line irregularities. For timestamp readout, the processing is configured in a transparent mode.

Histogram processing

To characterize the underlying delay line, in order to be able to reduce static non-linearity, we collect raw histograms from the sensor illuminated with uncorrelated light or using dark counts. The delay line is sampled every 2.5 ns and we know from the TDC architecture that an event will produce only one valid code between 0 and 140. Since the delay line is always longer than 2.5 ns, not all codes are used. After a large number of events, a TDC has produced a histogram, which is converted into a density plot as shown in Figure 3.12. The size of a bin is obtained by multiplying its fraction of counts with the 2.5 ns total period of the histogram.

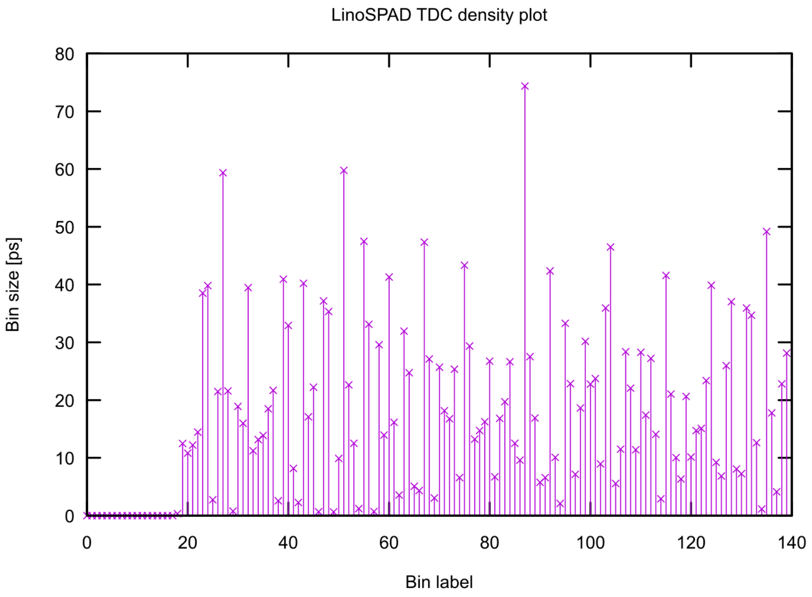


Figure 3.12: LinoSPAD raw TDC density plot. This distribution of bin sizes is typical for a FPGA TDC using carry chain delay lines. In our implementation, the first few codes do not occur, as they do not contain valid events. When the event edge reaches their position, it has already been sampled in the previous period. The number of unused codes is different for each TDC.

Each TDC is treated individually for the correction, with a method similar to the one presented in [82]. Assuming that the input histogram H_{in} with $k = 140$ bins covers a 2.5 ns period (τ_{TDC}), each raw bin is assigned a size $S_{\text{in,i}}$ and position $P_{\text{in,i}}$, calculated from the histogram counts $C_{\text{in,i}}$, where

$$S_{\text{in},i} = \tau_{\text{TDC}} \times \frac{C_{\text{in},i}}{\sum_{j=0}^{k} C_{\text{in},j}} \tag{3.1}$$

and

$$P_{\mathrm{in},i} = \sum_{j=0}^{i} S_{\mathrm{in},j} \tag{3.2}$$

for i from 0 to k raw input bins.

In order to obtain a corrected histogram of N bins with sizes $S_{\mathrm{out},i} = S_{\mathrm{out}} = \frac{\tau_{\mathrm{TDC}}}{N}$ and positions $P_{\mathrm{out,i}} = i \times \frac{\tau_{\mathrm{TDC}}}{N}$, we calculate the (sparse) $N \times k$ correction matrix M as

$$m_{i,j} = \max\left(0, \frac{\min\left(P_{\mathrm{in},j} + S_{\mathrm{in},j}, P_{\mathrm{out},i} + S_{\mathrm{out}}\right) - \max\left(P_{\mathrm{in},j}, P_{\mathrm{out},j}\right)}{S_{\mathrm{in},i}}\right), \tag{3.3}$$

where the elements of M correspond to the overlap between the output and input histogram bins. The final histogram H_{out} sent to the computer is then written:

$$H_{\mathrm{out}} = M \times H_{\mathrm{in}}. \tag{3.4}$$

Multiplying the raw histogram of size k by M, maps it to the corrected histogram of size N, by redistributing the counts. This operation can be done on the fly using very little memory, by exploiting the construction of M and the fact that it is sparse. Conceptually, the output histogram can be constructed by summing up the input histogram and producing output bins as they become full. A column of M containing the distribution for one input bin is stored very efficiently using at most two values, if the particular input bin is to be spread between more than two output bins. In cases where the input bin is entirely absorbed in an output bin, no storage is used, and in cases where one output bin is filled, only one fraction is stored. Since the sum of the column values must add up to 1, the missing values can be calculated on the fly. The implementation of this correction process is shown in Figure 3.13.

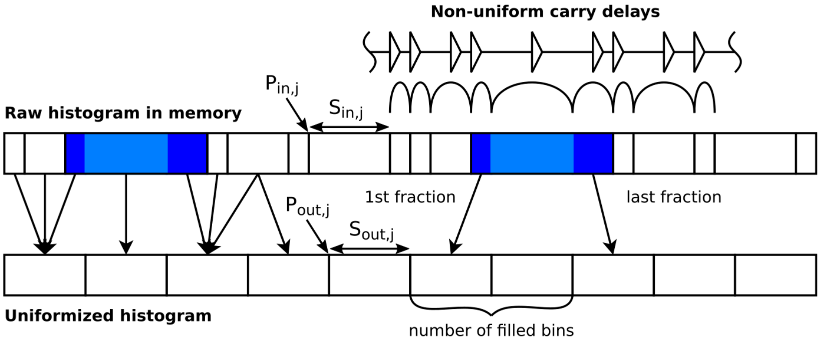


Figure 3.13: Illustration of the LinoSPAD histogram post-processing. A raw histogram as shown in Figure 3.12 is processed by multiplying it with the correction matrix from (3.4) to reduce static INL/DNL from TDC irregularities.

Each input bin of the histogram is mapped to the output histogram using three control values. These values define the column in the correction matrix M in equation (3.4) associated with the bin. The first is the number of output bins it fills, the second is the fraction of the counts that belongs to the next bin and the third value is the fraction of counts belonging to the last affected bin. As explained above, the fractions are only stored when needed. The implementation encodes the fractions as 8 bit fixed comma values and the completed bin count can range from 0 to 4. This limitation in bin count imposes a lower limit for the output bin size of one fifth of the largest input bin.

The histogram correction outlined above does not correct for another individual characteristic of the TDCs, or rather the SPAD signals. The signals have different delays from the routing that connects the FPGA input buffer with the pulse shrinking circuit and possibly as well from the pulse shrinking to the input of the delay line. This leads to the issue that even for a signal transition arriving at the same time on all FPGA inputs, different codes are produced for each pixel in every TDC. To account for this difference in signal delay, a histogram can be rotated after it has been processed to reduce DNL and INL. This only works correctly in all cases, when the recorded histograms cover the full reference period. Otherwise, they can only be partially aligned, but rotation is a trivial operation and partial histograms need to be treated separately in most applications anyway.

3.3 FPGA Implementation

On the way to a working FPGA implementation of the firmware with 64 TDCs and histogram modules, a few obstacles needed to be overcome. At some point in the development, simply throwing all the VHDL source files at the Xilinx synthesis tools would no longer work to produce a working bitstream. The tools would take a very long time to synthesize a routed design and the design would not meet the timing constraints for operation at the specified frequencies of 400 MHz and 133 MHz.

The synthesis for large FPGAs is a heuristic process and (deterministic) random numbers are used to control certain aspects when placing the design in the FPGA. In the synthesis tools, the seed value used for random number generation used in the place and route algorithms can be changed. SmartXplorer is the vendor-provided software to exhaustively try possible seed values and analyze the quality of result [83]. SmartXplorer is used to try the hundred allowed seed values for the synthesis process with the hope that one of them will produce a result that meets timing.

The LinoSPAD firmware reached a point where it would no longer meet timing for any of the placement seed settings. Despite the high regularity of the design, the synthesis tools either abort after a very long runtime or produce a poor result. Manual placement of a design in FPGA, even with high regularity, is not well supported by available design tools. It is also increasingly unlikely that manual placement achieves better results than software tools optimized spending many man-hours. The conclusion from the trials to synthesize the full TDC design without giving hints beyond the hardware description and timing constraints leads to failure. It looks like the tools end up trying to optimize at the wrong ends or in the wrong way, for example exchanging the placement of identical modules for each other.

The most critical parts for the place and route in the LinoSPAD firmware are the modules that need to run at 400 MHz. This includes the TDC delay-lines and encoders, which need to be placed very carefully, such that they can use the fastest routing available in the FPGA to meet the required clock period. Once they are in place and there is enough free logic for the remaining parts of the firmware, there should also be enough routing available to connect them. Unfortunately, the synthesis tools do not provide a way to express these requirements and they are unable to derive them from the VHDL sources of the design. There is no direct control over the routing in most FPGAs, including the Spartan 6 used for LinoSPAD.

When the two parts of the design, the front-end running at 400 MHz and the back-end at 133 MHz, are implemented separately by replacing one with minimal dummy logic the tools are able to find a solution in reasonable time. For this case, the dummy logic must be such that it prevents the part to be implemented from being optimized away, but at the same time, it should not compete for critical resources. A way to achieve this is by providing an external access to the output signals of the part about to be implemented, and ignore the timing in this logic. In this way we are able to produce a firmware where either the 64 TDCs with encoders are functional without additional processing or a firmware where the histogram accumulation logic, memory and processing are functional, but working with dummy values.

At this point, both parts of the design can be tested and improved on individually. The process outlined above also saves time when exploring optimizations or tracking down implementation issues. On a larger or faster FPGA, the two parts could probably be implemented by the synthesis tools from the hardware source and timing constraints alone. For the Spartan 6 FPGA this was not possible and to combine the two parts of the design the use of partitioning [84,85] was essential.

Design partitions for FPGAs allow part of a hardware design to be reused and thereby to prevent the implementation tools from spending additional time to place and route it. Reusing a partition forbids further optimizations that move circuit parts to other locations. Optionally also the routing of the partition can be preserved. Only the parts that have changed need to be re-implemented and they are placed in locations not used by the preserved partitions. For the LinoSPAD firmware the natural way to partition it, is at the clock crossing between the parts running at 400 MHz and the parts running at 133 MHz, and to implement and optimize the fast part separately before adding all the processing logic. This also permits modifications in the processing without having to optimize the 400 MHz part again.

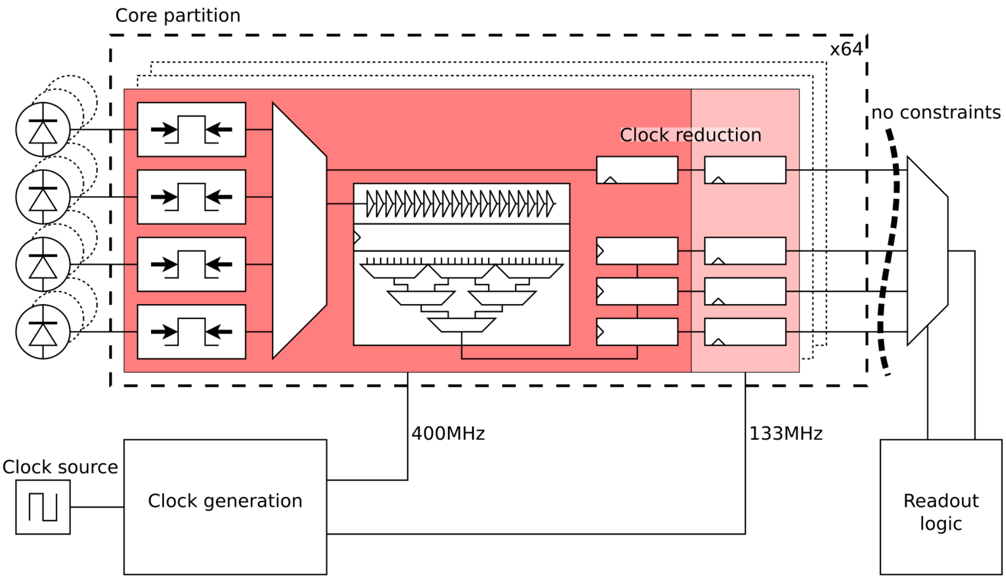


Figure 3.14: FPGA core used for timing closure. This simplified FPGA firmware was synthetized to achieve timing closure on the parts running at 400 MHz. These modules are partitioned separately and constrained to locations between the memory blocks of the FPGA to leave unconstrained locations where the processing logic can be added.

Figure 3.14 shows a simplified schematic of the firmware implemented to optimize the placement of the TDC core logic running at 400 MHz. The logic with timing constraints based on the 400 MHz period is placed in its own hierarchy and then defined as design partition. The partition includes the complete logic from the SPAD inputs of the FPGA to the first register clocked at the slower frequency of 133 MHz.

Minimal logic is added to this core design to prevent optimization algorithms from removing everything because they determine no effect on FPGA outputs. The simple readout and command logic uses the USB modules also used for the full firmware to access all input and output signals of the core block. The timing for these accesses is ignored to prevent optimizations from considering these connections.

Placement constraints on the 400 MHz core logic are used to forbid placement in FPGA locations close to memory blocks, to reserve these blocks for the processing logic that read and writes memory to construct histograms or store timestamps. The design was synthesized using SmartXplorer for all available placement seed values, to find the implementation that meets timing using the least amount of resources. This partition is used to implement the full LinoSPAD firmware, of which an FPGA utilization map is shown in Figure 3.15.

The firmware uses 66307 or 71% of the FPGAs look-up tables occupying 20950 or 90% of the slices. A slice contains 4 look-up tables and 8 registers sharing common control signals. Out of 268 memory blocks, 252 are used: 192 for histogram or timestamp storage, 48 for the post-processing information, and the remaining in readout FIFOs and for clock-crossing. The implementation results of the firmware described here and of a reduced variant used in many measurements are given in Table 3.1.

Firmware	Concurrently shared TDCs	Sequential access TDCs
Number of TDCs (N)	64	64
Pixel to TDC assignment	4 neighboring	every 64th
TDC sample rate	400 MHz	400 MHz
Processing rate	133 MHz	80 MHz to 133 MHz
Histogram length	4×768 to 1×3072	1×1024
Timestamp memory	1536 shared	512
Memory organization	single buffered	double buffered
Readout scheme	acquisition stopped or rolling	from inactive buffer
Intensity counters	No	32 bit per pixel
LUT utilization	71%	71%
Register utilization	43%	44%
Slice utilization	90%	90%
Memory utilization	94%	73%

Table 3.1: LinoSPAD TDC firmware comparison. The main difference is in the way the memory blocks are used to store histograms or timestamps. The shared TDC firmware can record 4 shorter, per-pixel histograms, or 1 longer, combined histogram for the 4 pixels. Resource utilization, except for the memory, is comparable because of intensity counters in the sequential access firmware.

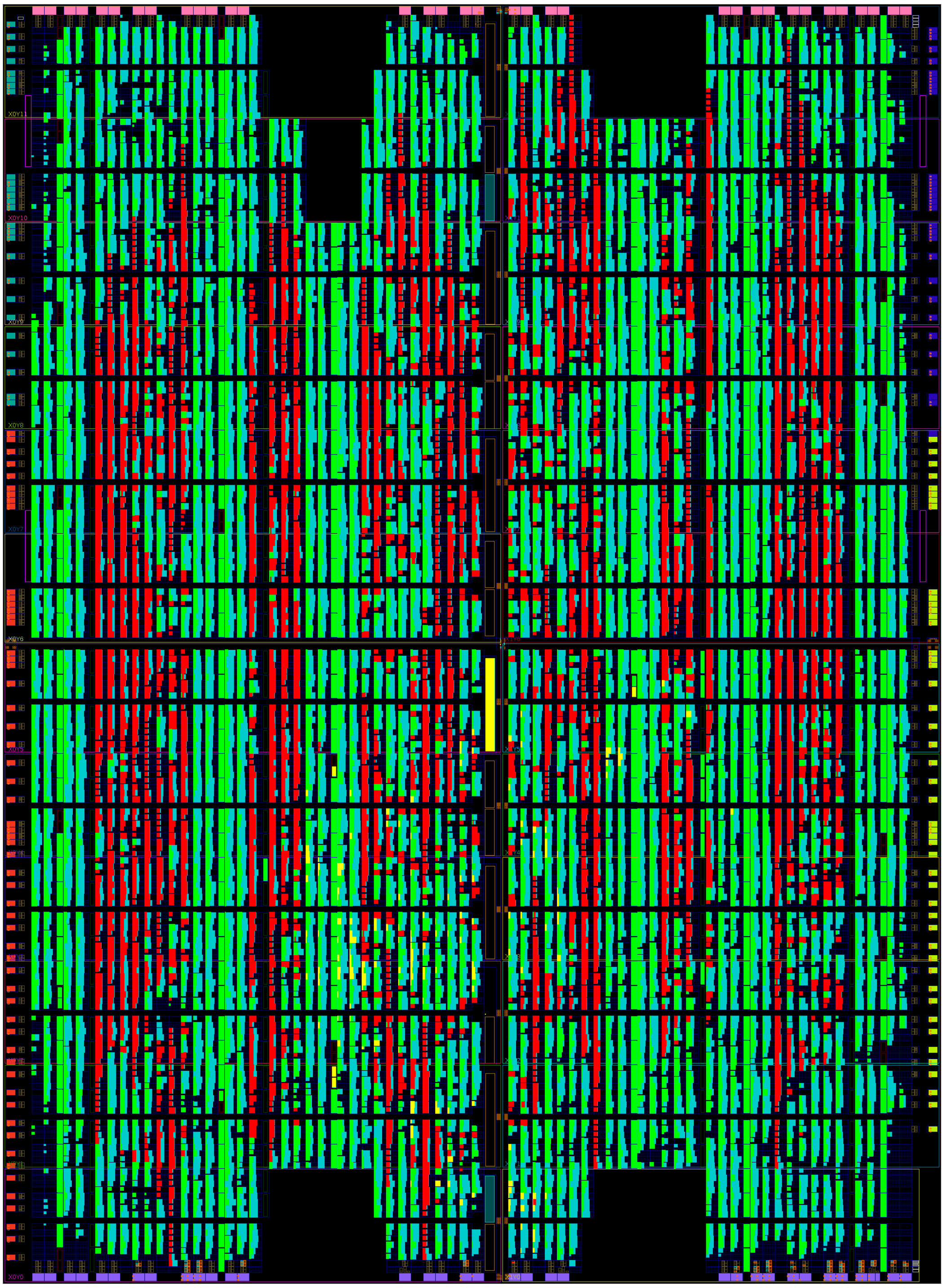
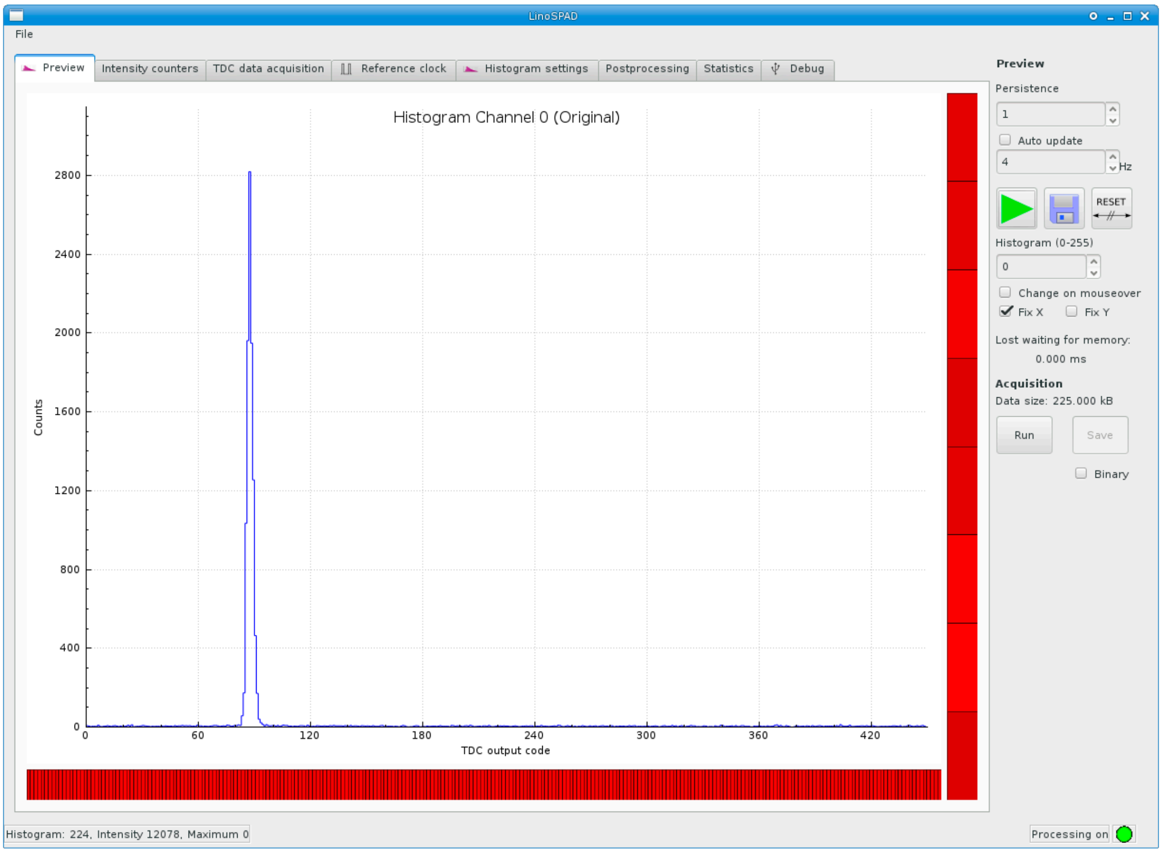


Figure 3.15: FPGA utilization of the TDC firmware. Over 90% of the memory elements and 70% of the look-up tables in the largest Spartan 6 FPGA are utilized. The registers are color coded with red parts running at 400 MHz, green parts at 133 MHz and the few yellow specks at 100 MHz.

3.4 Computer software communication

The LinoSPAD camera is used together with a computer to perform measurements. A Cypress FX3 USB 3.0 transceiver is used to handle the communication between the FPGA and the computer. The FX3 has an integrated ARM processor core running its own firmware to configure the chip. In order to reach highest transfer speeds possible with USB 3.0, the FX3 can be configured such that the communication takes place without intervention of the processor.

The communication between the FPGA and the FX3 uses a bidirectional 32-bit wide bus with handshake channels to signal availability of data or buffer space to the FPGA. The data bus running at 100 MHz can, theoretically, transfer a maximum rate of 400 MB/s between the FX3 and the FPGA. Data rates over 200 MB/s have been measured with the configuration used during this thesis, where the FX3 is used to create a bidirectional FIFO between the FPGA and the computer.



Preview

Persistence: 1

☐ Auto update: 4 Hz

Histogram (0-255): 0

☐ Change on mouseover

☑ Fix X ☐ Fix Y

Lost waiting for memory: 0.000 ms

Acquisition

Data size: 225.000 kB

☐ Binary

Histogram: 224, Intensity 12078, Maximum 0

Processing on

Figure 3.16: LinoSPAD software screenshot showing the graphical user interface with the preview pane. The red bars on the sides of the histogram of a pixel show the measured intensity for all pixels. The tabs provide access to all configuration options of the firmware. Data is stored as text or binary files that can be imported in analysis programs.

In the FPGA firmware, an interface circuit connects to the FX3 to handle the communication and split the bidirectional external bus into unidirectional “send” and “receive” FIFOs. On the USB side, the circuit prioritizes data reception from the computer over data transmission from the FPGA and uses two FPGA-FIFOs to synchronize the data-streams between the FX3 and FPGA clock domains. On the FPGA side, a simple state-machine handles the connection to different modules in the firmware.

The protocol from the computer to the FPGA firmware uses a simple module-addressing scheme, where the first 32-bit word in a transfer indicates the destination address and length of the transfer, and the following words contain the data for the module. The first word is used by the state machine to update the address output and an internal counter register, and the following words are sent to the FPGA modules. All modules listen to the data from the computer, but only respond when the current address matches their own.

Sending data to the computer is prioritized based on the module addresses by giving modules with lower addresses higher priority. Handshaking signals are used for both directions such that the FPGA modules or the FX3 can interrupt the transfers when internal memories are full or operations are ongoing.

The LinoSPAD software follows the communication principles outlined above and acts as master of all operations. It is important that the software is always aware of the state of the camera to perform its operations. The software then sends configuration data and measurement commands to the camera firmware and reads and displays or stores the results. An image of the software main screen is shown in Figure 3.16.

3.5 Results

The characterization of LinoSPAD covers two main aspects. The first is the performance of the SPAD sensor connected to the FPGA and the second is the performance of the FPGA-based TDCs in the firmware.

First, we present the results of the sensor performance measurements. For these measurements, the sequential access firmware of Table 3.1 was used. In this firmware, the four pixels are connected to the delay line with a multiplexer, such that only one pixel's timing information is measured at a time and there is less memory for histogram accumulation. On the other hand, the memory is double buffered for continuous acquisition and readout over the four pixels, and there are additional 32-bit intensity counters connected to every pixel and operated fully in parallel.

The intensity counters are clocked at 100 MHz and increment based on the sampled state of the pixel input. One of the counters per pixel increments when the pixel input is seen active and the second counter increments when a rising edge is detected in the sampled state. These counters will reliably register SPAD events of 10 ns or longer. By comparing the counter for the active time and the edges we can estimate the average output active time (related to the dead time) of the sensor which is of the order of 40 ns as is confirmed by measurements with the oscilloscope. By observing the counter for the active state, it is also possible to detect saturation of the sensor when the signal stays active for a long time without any transitions.

The measurements concerning the sensor performance presented in the following sub-sections have been carried out using the intensity counters. Unless otherwise stated the quenching voltage used was 1V and the excess bias voltage given is referenced to the mean breakdown voltage of the observed chip.

3.5.1 Breakdown voltage

The first measurement performed on a LinoSPAD sensor is to find the breakdown voltage. We estimate the breakdown voltage for each pixel by using the excess noise method [37]. From a bias voltage where no DCR is observed the bias is raised in steps of 5 mV up to a voltage 200 mV above the point where all pixels show noise. At each voltage point, the intensity counters are used to count five periods of 1 second to have good fidelity also for very low noise values.

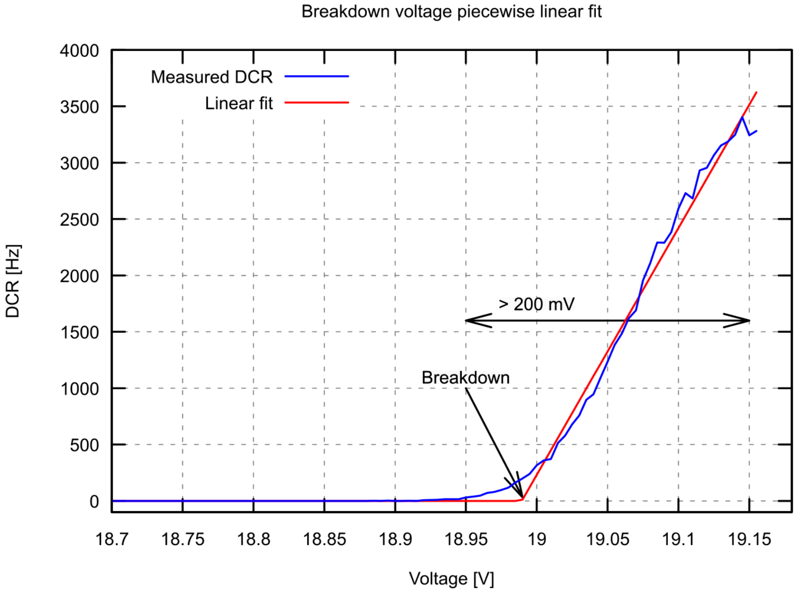


Figure 3.17: Illustration of the approach used to estimate the breakdown voltage. To find the breakdown voltage, the DCR is measured while the operating voltage is increased in steps of 5 mV to a voltage at least 200 mV over the first non-zero output. A two-piece function is then fitted and the voltage at the intersection, minus 0.6 V to account for the inverter threshold, gives the estimated breakdown voltage.

For each pixel, the breakdown voltage is estimated by fitting a piecewise linear function to the measured noise vs. voltage curve. The two-piece function is equal to zero during the first part and linear during the second part and fitted using a least-squares approach. The intersection of both regimes defines the breakdown voltage after the thresholding inverters. Assuming a 0.6V threshold for the first inverter in the pixel circuit, we subtract this value to get the final estimate for the breakdown voltage.

Figure 3.17 shows a linear fit to find the breakdown value on a single pixel and Figure 3.18 shows typical breakdown values for all pixels on a sensor. The variation of the breakdown on a chip is below 100 mV standard deviation, comparable to values from SwissSPAD. Variations from one chip to another are analyzed in the next chapter.

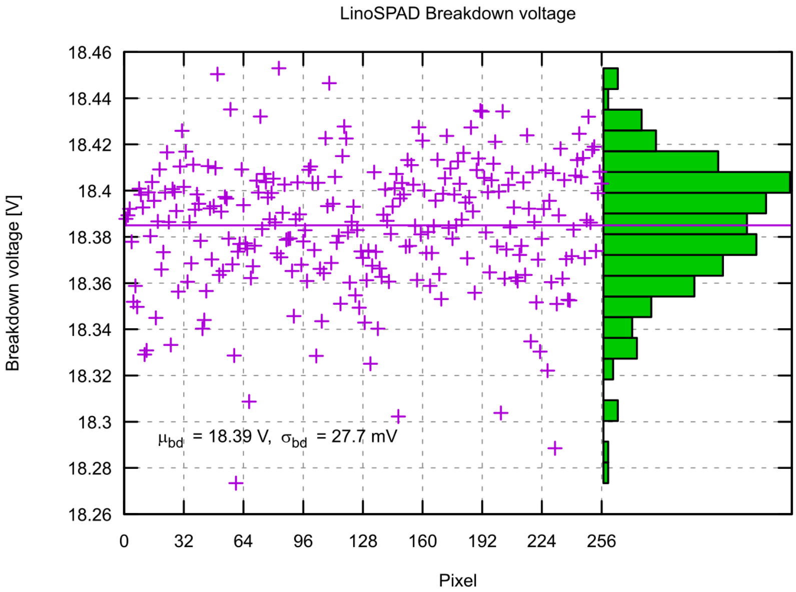


Figure 3.18: LinoSPAD breakdown voltages per pixel. The mean value is 18.39 V with a standard deviation of 27.7 mV.

3.5.2 DCR

After the breakdown voltage measurements to establish the excess bias reference, the DCR of the sensor was characterized. Two different characteristics were measured. The first one was the temperature dependence of the DCR and the second the dependence on excess bias voltage. With the second measurement, the distribution of the DCR across the pixels is also provided.

Figure 3.19 shows the evolution of the median DCR value with increasing temperature from -40° C to 80° C. While the noise does not increase from -40° C to -30° C, it increases exponentially for temperatures above about 20° C. Clearly, cooling the sensor, or at least controlling its temperature, is very beneficial to control its noise.

Figure 3.20 shows the cumulative DCR for all pixels in one sensor, for excess bias voltages between 2 V and 4 V. For this plot, the noise values are sorted in increasing order, which allows a quick appreciation of the median noise value and the percentage of noisy pixels. In comparison with SwissSPAD, which has only a few percent of pixels with noise values greatly exceeding (by at least 2 decades) the median, LinoSPAD has about 25 % of hot pixels indicated by the typical characteristics shown in Figure 3.20. This is thought to be due to the larger size of the SPADs leading to a higher rate of defects in the active area of the SPAD. The hot pixel of LinoSPAD have still a useable dynamic range as indicated by the DCR corrected photo response in Figure 3.25.

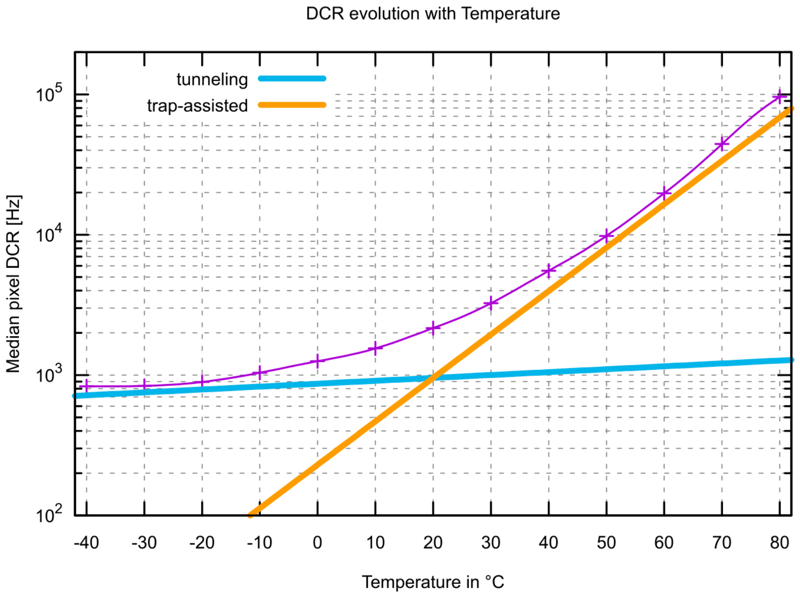


Figure 3.19: LinoSPAD DCR for temperatures of -40° C to 80° C. The values were measured using 2 V excess bias voltage for the sensor.

Figure 3.21 shows the spatial distribution of the noise across the sensor. The median noise can also be estimated from this representation whereas the fraction of hot pixels is more difficult to see. Good fabrication quality is determined by the result, that no clustering of high or low noise pixels is visible.

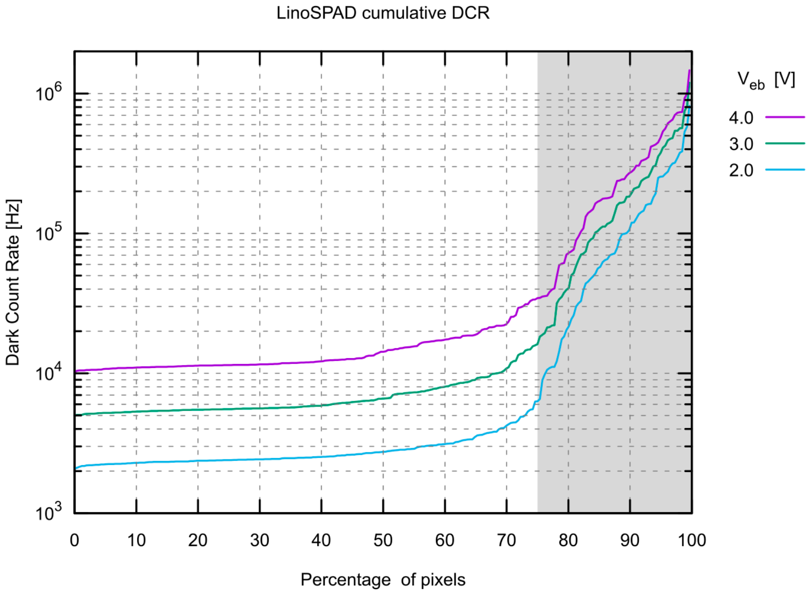


Figure 3.20: LinoSPAD cumulative DCR at room temperature. Shown are three curves for excess bias voltages between 2 V and 4 V, the typical operation range. The shaded part indicates the typical fraction of hot (pixel DCR > 10*median DCR) pixels. The measurements were done at room temperature.

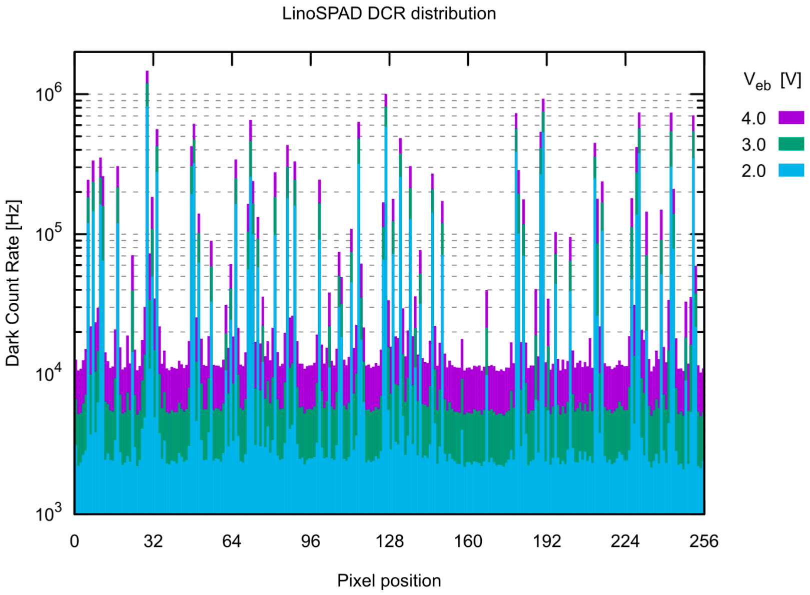


Figure 3.21: LinoSPAD spatial DCR. This plot shows the spatial distribution of the DCR corresponding to Figure 3.20. No particular pattern could be observed for any chip.

3.5.3 PDP

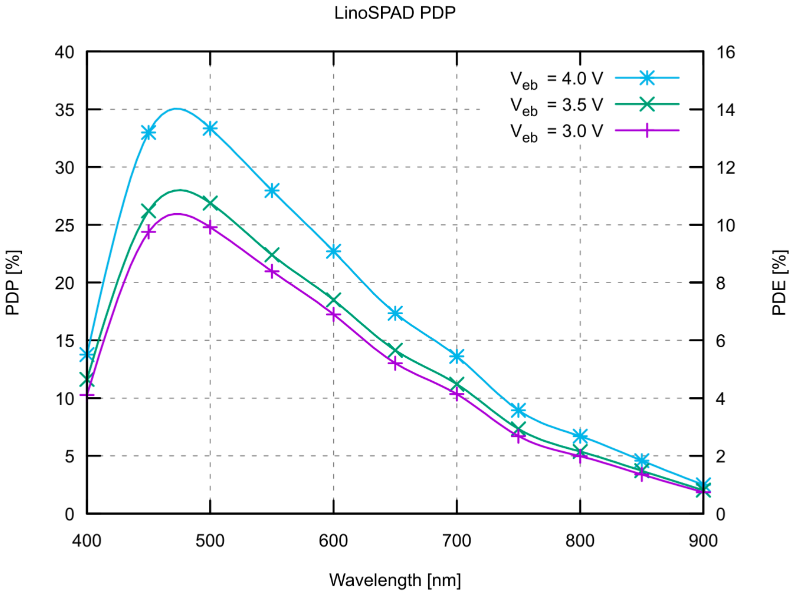


Figure 3.22: PDP and PDE for a LinoSPAD sensor. The values were measured for different excess bias voltage settings with an integrating sphere and a reference photodiode.

The photo-detection probability of LinoSPAD sensors was measured using two approaches. The first measurements were carried out using a monochromator and integrating sphere with reference photodiode to measure the PDP for wavelengths ranging from 400 nm to 900 nm. In these measurements, a low number of pixels with noise values around the median were used to obtain good linearity. From the measured count rates, the DCR was subtracted and the resulting photon rate compared with the rate estimated from the reference diode current. Figure 3.22 shows the resulting PDP vs. wavelength graph. Also shown is the photon detection efficiency on the right y-axis, which takes into account the fill factor of 40 %.

The second measurements were carried out with a diffused LED illuminator that was calibrated using a photodiode power meter. Using two LEDs with center wavelengths of 465 nm and 640 nm respectively and spectral width around 25 nm the PDP for each pixel was estimated at these wavelengths. The sensor was illuminated with a reference intensity of 1 µW/cm^2 corresponding to photon rates of 5.4 MHz and 7.4 MHz per pixel, respectively. The high irradiance reduces uncertainties in the reference measurement, but can lead to a lower measured sensitivity as the LinoSPAD sensor saturates. The measurements are still useful to judge the PDP linearity of the pixel array and give another reference point for the PDP.

For Figure 3.23 we calculated a PDP value for each pixel for rising excess bias voltages between 1 V and 5 V and an illumination of 1 µW/cm^2 at 465 nm. To calculate the PDP, the count rate of each pixel was recorded at each point, corrected for DCR and then divided by the incident photon rate on the active area of the pixel. The same measurement was performed using a LED with 640 nm wavelength and we show the PDP at 2 V excess bias voltage for each pixel at the two wavelengths in Figure 3.24.

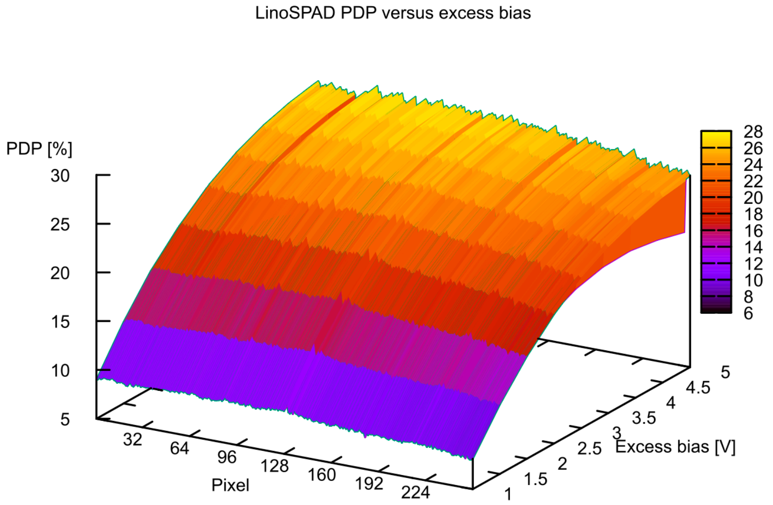


Figure 3.23: LinoSPAD PDP at 465nm for each pixel. With constant illumination, the excess bias voltage was increased from 0.5 V to 5V where PDP compression can be observed. For higher voltages, the hottest pixels reach saturation, where their output stays active for longer periods resulting in lower count rate represented in the plot as lower PDP.

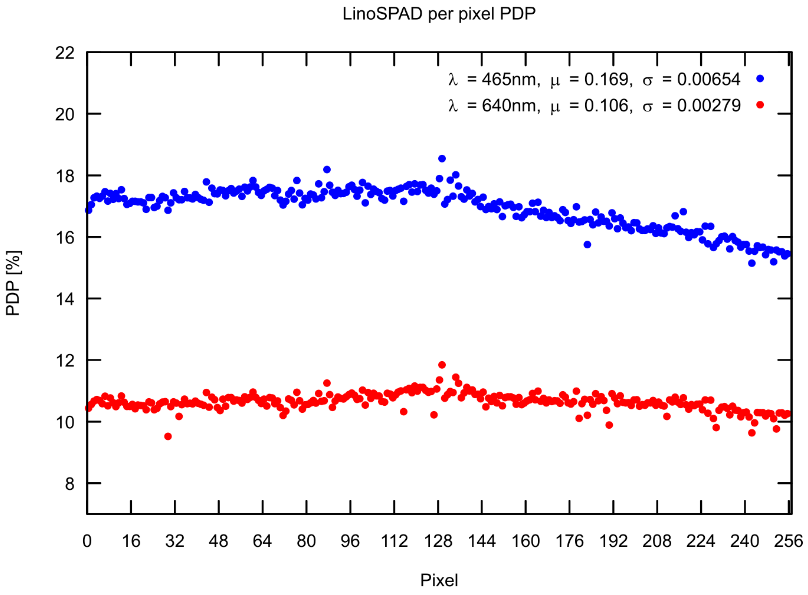


Figure 3.24: LinoSPAD per-pixel PDP with 2 V excess bias voltage. The values were obtained using an irradiance of 1 μW/cm².

3.5.4 Photo response

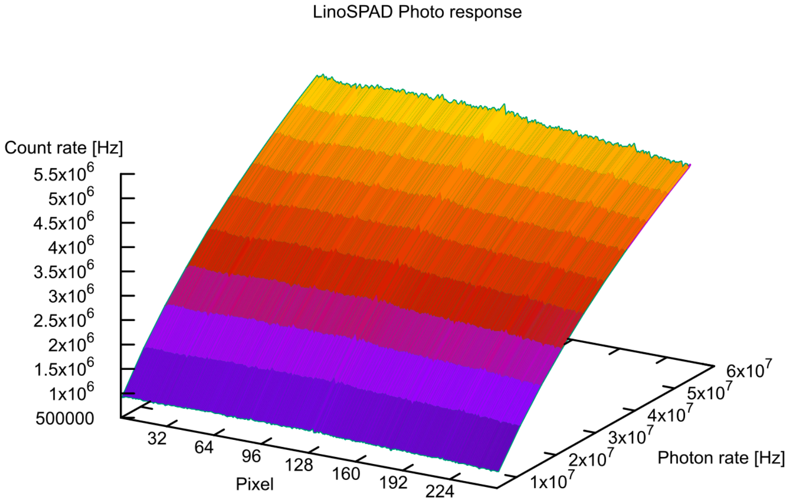


Figure 3.25: LinoSPAD per-pixel photo response. The values are measured using 465 nm illumination and 2 V excess bias voltage. Irradiance ranges from 1 μW/cm² to 10 μW/cm². The measured count rates are individually corrected for DCR and show good linearity. Count compression is observed for higher photon rates.

To measure the photo response of the LinoSPAD sensor, the excess bias voltage was kept constant at 2 V while the irradiance was increased from 1 µW/cm^2 to 10 µW/cm^2. Again, the measurements were performed at the wavelengths of 465 nm and 640 nm. Photon rates on each pixels were measured over 5 seconds and corrected for individual DCR values. We observe a high uniformity across the SPAD array, which shows that even highly noisy pixels have a good linearity range. Due to the high photon rates a compression of the response towards higher intensities appears. Figure 3.25 shows the photo response on a typical sensor.

3.5.5 Power consumption

With the measurement of the power consumption of the LinoSPAD camera, we cross the border between measurements focused on the sensor itself and measurements involving the FPGA. Power measurements were performed via current measurements on the sensor bias and logic supplies, and on the main FPGA system supply. The sensor is supplied with a bias voltage corresponding to 2 V excess bias and 3.3 V for the logic.

The logic supply powers the pixel inverters and the output pads, which largely dominate current consumption. The FPGA is powered by a 5 V supply feeding two dual switching supplies to provide internal voltages of 1.2 V, 1.8 V, 2.5 V and 3.3 V. The FPGA uses 1.2 V for internal logic elements, 2.5 V for configuration logic and on the interface with the FX3 USB transceiver and 3.3 V on the interface with the sensor. The FX3 uses 2.5 V for configuration and interface with the FPGA and 1.2 V as internal supply. Typical efficiency of the voltage regulators is between 80% and 85%.

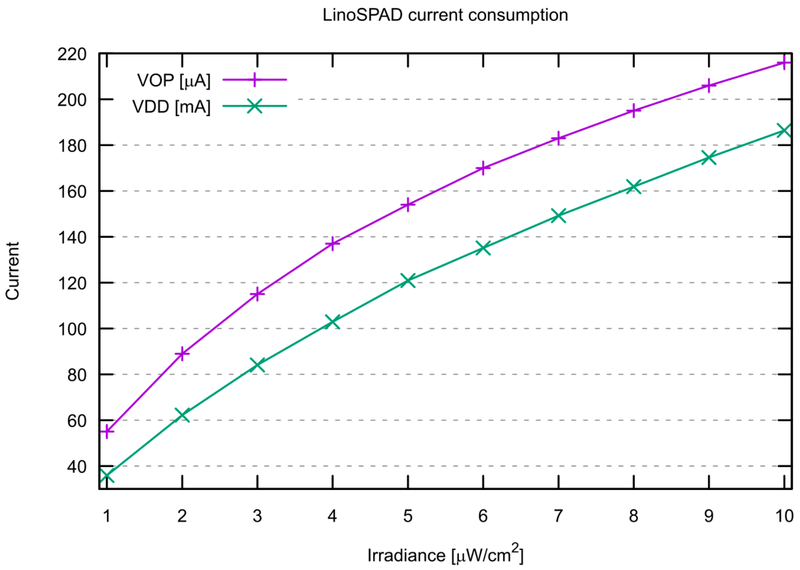


Figure 3.26: LinoSPAD sensor current consumption with increasing switching activity. The switching activity was increased with rising irradiance from 1 µW/cm^2 to 10 µW/cm^2. A LED emitting at 640 nm was used.

Table 3.2 lists the boundary conditions for power consumption measurement. The current on the SPAD bias increases with the irradiance on the sensor while the current for the logic increases with the activity of the output pads. The limit of the logic current is around 600 mA where the pixels start to go into saturation. For increasing light irradiance beyond that point, the current decreases, as switching activity becomes lower due to post-saturation compression (Figure 1.12).

The current of the FPGA supply reflects the activity of the firmware used for these measurements. It has a high baseline power consumption that does not increase much with sensor activity.

	V_{OP} (2 V_{ex})	V_{DD} (3.3 V)	FPGA (5 V)
Darkness	7 µA	2.0 mA	928 mA
Intense ambient light	~425 µA	~600 mA	1063 mA
Saturation	~750 µA	~0.3 mA	944 mA

Table 3.2: Current consumption for a LinoSPAD camera. To measure the current consumption the excess bias voltage was set to 2 V and the quenching voltage to 1 V.

Figure 3.26 shows the sensor current increase under irradiance between 1 µW/cm^2 to 10 µW/cm^2 from a 640 nm LED. The SPAD operating current is measured in µA and the logic current in mA.

3.5.6 TDC response

In the evaluation of the TDC response, we start from the unprocessed results reproducing the statistics of the delay line, which is using the carry chains in the FPGA. The first characteristics concern the differential non-linearity (DNL) and the integral non-linearity (INL).

The characteristics of DNL and INL are typically used to describe the performance of analog-to-digital converters (ADCs, [86]). In our case, instead of the analog value being a voltage, it is a time and the same principles apply for the evaluation. Differential non-linearity measures the deviation between two (analog) input values corresponding to successive output values. For TDCs, it is commonly defined as

$$\mathrm{DNL}_i = \frac{t_{in,i+1} - t_{in,i}}{\mathrm{IDEAL\ LSB}} - 1, \qquad (3.5)$$

where $t_{in,i}$ and $t_{in,i+1}$ are the times of successive code steps with an ideal difference of one LSB.

The INL measures the absolute deviation of the actual response of the TDC from the ideal linear response. The INL of a measurement corresponds to the integral of the DNL up to that point and is defined as

$$\mathrm{INL}_n = \sum_{i=0}^{n} \mathrm{DNL}_i. \qquad (3.6)$$

The INL starts at zero and ends at zero when the full range of input values is covered. The TDC design with unused bins guarantees that the minimum and maximum code values correspond to the minimum and maximum of the possible measurement range.

As discussed in section 3.2, on the FPGA architecture, the delay line has very non-uniform time delays from one output to the next. With the careful design of the TDC, we ensured that the output is monotonically increasing with the time-of-arrival of events. This is important, to facilitate calibration.

For our TDCs, the DNL is directly related to the density characteristics [87,88] as shown in Figure 3.12. Under non-correlated illumination, all output codes of the TDC should occur with the same frequency and all bins having an identical size given by

$$\mathrm{s}_n = \frac{C_n}{\sum_{i=0}^{N_\mathrm{bins}} C_i}, \tag{3.7}$$

where C_n is the number of counts in bin *n* and N_bins the total number of bins in the histogram.

Figure 3.27 shows the DNL for one TDC in a camera. A negative value indicates a bin that is too small and a positive value a bin that is too large. Entirely unused bins have been removed, such that the total length is just over 120 bins. As there are no negative bins the DNL is bounded in the negative range by -1 when the bin is of zero length. The DNL is directly related to the density of Figure 3.12, which is obtained by adding 1 LSB to every DNL value.

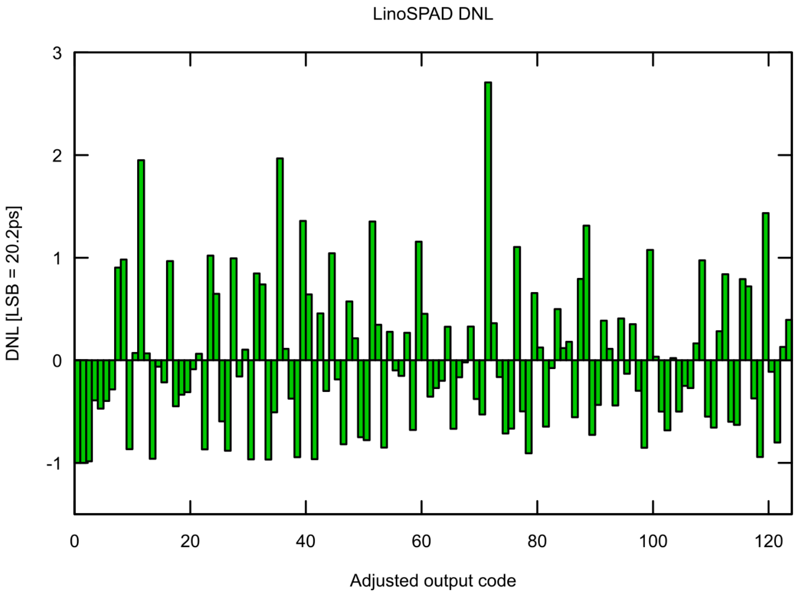


Figure 3.27: DNL characteristics of one LinoSPAD TDC. The values correspond to the difference from the ideal value of one LSB for this TDC, which measures 20.2 ps. The LSB is calculated without unused output codes.

Figure 3.28 shows the INL characteristics of the transfer function of the first TDC in the test camera. The stretch of completely zero bins has been removed from the DNL so that it does not artificially lead the INL into the negative range. Still the first bins have a large influence on the accuracy for the lower half of the possible output codes.

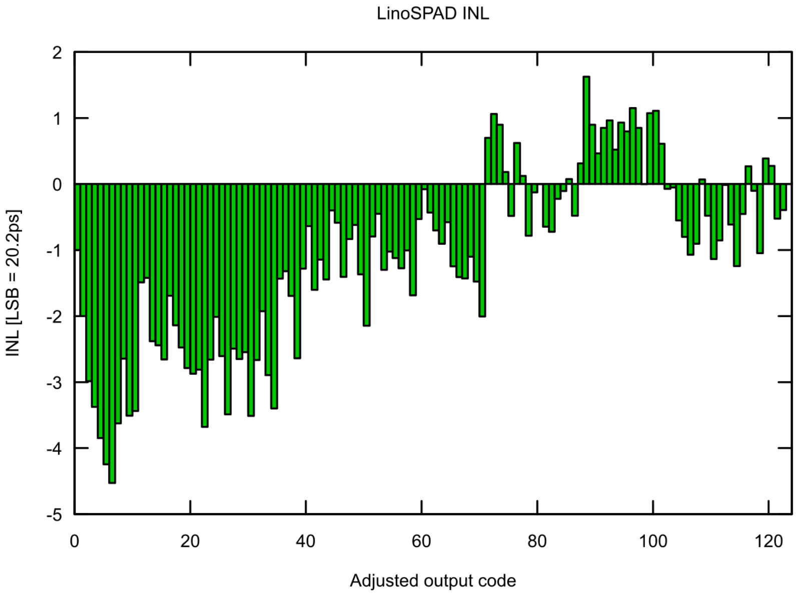


Figure 3.28: INL characteristics of one LinoSPAD TDC. Including unused codes would lead to the impression of much larger negative excursion.

3.5.7 Post-processing

The real-time post-processing is evaluated in the same manner as the TDC response by computing apparent DNL and INL after the processing has been applied. First, a number of histograms is measured to calculate the density distribution of each TDC and program the histogram processing logic with the correction matrix given in equation (3.4). Afterwards, the measurement is repeated and the remaining DNL and INL in the processed density analyzed.

As this is a statistical correction with limited precision and working on integer values, the histograms need to contain a sufficient number of events for any effect to be observed. For the following results, we programmed the correction to reduce a histogram covering 12.5 ns from 700 bins to 450 bins. The corrected histogram has then an ideal LSB of 27.7 ps. The correction is applied in the FPGA using fixed-point calculations with 8-bit precision for the values of the correction matrix and rounding so that no count is lost.

From a sample of 100 histograms with around 100k events each, the DNL and INL values for each code were calculated. The plots display the minimum, average and maximum value of DNL and INL at each TDC code. Figure 3.29 shows the DNL values, which have a mean close to zero at each position and spread between -0.1 and 0.1 approximately. Figure 3.30 shows the INL results spreading over a range between -0.5 and 0.5 LSB showing that the correction is working and producing histograms with good linearity.

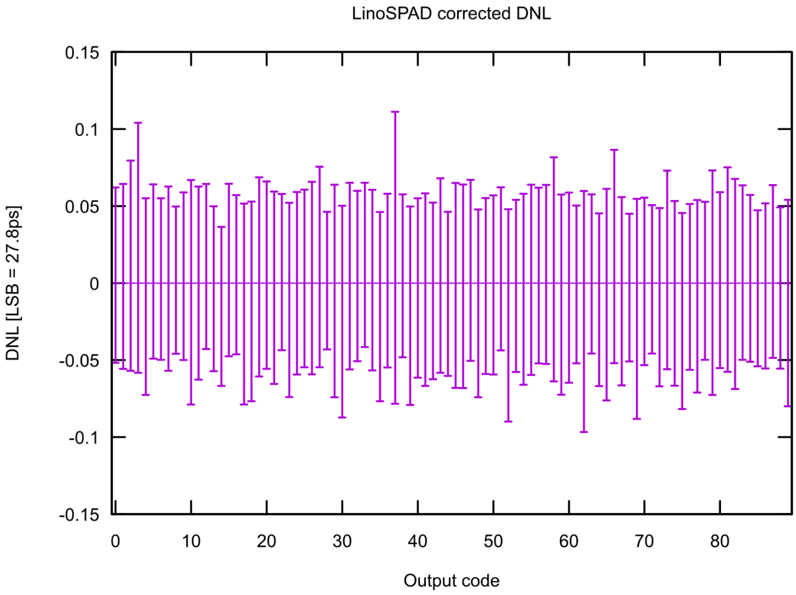


Figure 3.29: FPGA corrected DNL of LinoSPAD TDC. From the DNL analysis of 100 processed histograms, minimum and maximum values for each bin are shown.

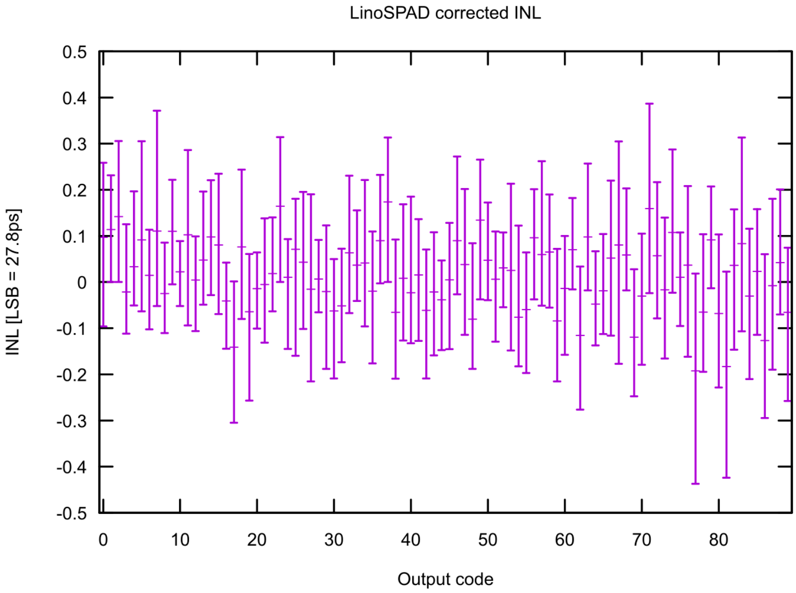


Figure 3.30: FPGA corrected INL of LinoSPAD TDC. The plot shows mean, minimum and maximum values for the INL computed from 100 sample histograms. The linearity error smaller than 0.5 LSB in magnitude indicates efficient correction, able to resolve one LSB.

The outcome of the correction depends on the quality of the calibration data and the stability of the environment between calibration and measurement. Temperature changes, also caused by changes in light intensity, degrade the quality of the correction and make recalibration necessary. The main cause for degradation is the change in length of the delay lines and it is expressed first as excursion in the INL for low codes. The variations between the TDCs are analyzed in 0.

3.5.8 Histograms

The last results to be presented in this chapter cover the complete LinoSPAD camera in combination with a synchronous illumination to evaluate the timing resolution obtainable with from the camera.

For the illumination, we use a custom laser diode driving circuit based on bipolar transistors [89]. From a trigger signal, the circuit generates a sub-nanosecond current pulse, which excites the laser diode to emit a picosecond light pulse.

For the characterization measurement where we point the laser diode directly at the sensor from a short distance, we use a laser diode emitting at 650 nm with a power rating of 5 mW (US-Lasers). The laser diode is placed directly facing the sensor at a distance sufficient to illuminate all pixels. Figure 3.31 depicts the characterization setup. No optics are used between the laser diode and the sensor.

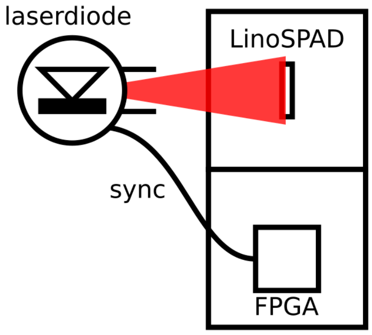


Figure 3.31: LinoSPAD timing characterization setup. A low-power laser diode is placed close to the sensor. Light pulses with a duration well below the SPAD dead time are used to measure the SPAD and TDC timing response.

The firmware is configured to output a synchronization signal at 80 MHz and to measure time-of-arrival histograms synchronous to the 12.5 ns period of the illumination. With post-processing disabled, the camera sends histograms with a length of 700 bins over the illumination period. One-hundred histograms, each with an acquisition window of 50 ms, were acquired. Figure 3.32 shows the result of the experiment, after manual alignment of the pixel delays to bring the signals around a center time value. The manual histogram rotation was based on the scaled rotation values measured for the corrected histograms.

To program the correction module, the camera was illuminated from an LED at the same intensity as produced from the pulsed laser diode to take into account temperature effects due to changing circuit activity. From a measurement of 1000 histograms, the TDC densities and corresponding correction values were computed and written to the correction module.

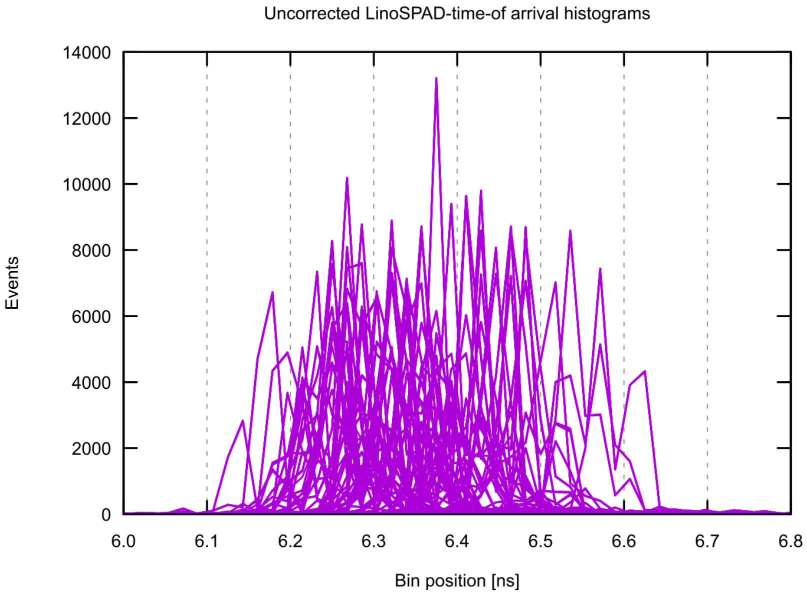


Figure 3.32: LinoSPAD response histograms from 500 ms acquisition. The pixel histograms in this figure are read unprocessed from the camera. They were manually aligned to a position around 6.25 ns using the same alignment used for processed histograms.

Afterwards, from a measurement of 10 density corrected histograms, the histograms were delay corrected to align the peak values around the center arrival time of 6.25 ns. Figure 3.33 shows an overlay of all histograms after correction.

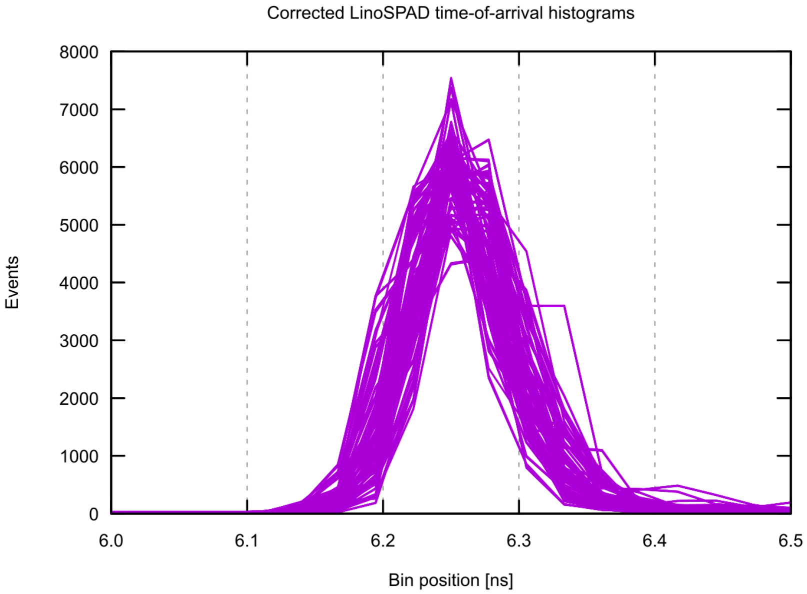


Figure 3.33: Processed LinoSPAD histograms from 256 pixels. The histograms are measured using the same settings as before, but with FPGA correction enabled. The mean FWHM of individual histograms is below 100 ps.

The measurements concerning the correction efficiency for the LinoSPAD firmware conclude the basic characterization of the sensor and the firmware. The next chapter extends the characterization with the discussion of non-linearities in the sensor, variations in a FPGA, and between multiple cameras.

3.5.9 Performance summary and comparison to other FPGA TDCs

Table 3.3 lists the main performance parameters of the LinoSPAD camera. The parameters of the FPGA TDC are included to illustrate the capabilities of the current FPGA pixel circuit implementation. Other circuits could add different parameters like number and bit-widths of counters and phase resolution for the implementation of synchronous photon counting.

Parameter	Value
Chip size	6.8 × 1.68 mm^2
Technology	AMS HV 0.35 µm 4M
Resolution	256 × 1 (+8)
Pixel pitch	24 µm
Fill factor	40%
Dead time (V_Q = 0.9 V)	<100 ns
Median DCR at 20°C	2.5 kHz
Spectral range (V_{ex} = 3 V, PDP > 5%)	400-850 nm
Light incidence	45° from normal
Number of TDCs	64
Maximum TDC event rate	133 MHz/TDC
Average TDC resolution (LSB)	< 25 ps
TDC range	28 bit (4.5 ms)
TDC DNL range, uncorrected	4 LSB
TDC DNL range, corrected	0.2 LSB
TDC INL range, uncorrected	7 LSB
TDC INL range, corrected	0.5 LSB
Data transfer rate	200 MB/s

Table 3.3: LinoSPAD performance summary.

Table 3.4 compares the system with other FPGA based TDCs.

		LinoSPAD	Favi [82]	Fishburn [90]	CERN [91]
FPGA		Spartan 6	Virtex 5	Virtex 6	Spartan 6
Process	[nm]	45	65	40	45
Number of TDCs		64	1	160	1
Number of channels		256	1	160	1
LSB Resolution	[ps]	25	17	10	26
Event rate	[MHz]	133	300	300	125
Carry blocks	[TDC^{-1}]	35	50	40	124
Clock frequency	[MHz]	400	300	600	125
DNL_{pk-pk}	[LSB]	4	4.6	3	n/a
INL_{pk-pk}	[LSB]	7	5.6	6	n/a

Table 3.4: LinoSPAD TDC comparison. The stated INL/DNL peak-to-peak ranges are uncorrected values.

Chapter 4 Challenges of large SPAD array imagers

In this chapter, we discuss in more detail the challenges associated with large numbers of SPADs integrated in an imaging system. In systems with single pixels, the exact values of common metrics such as breakdown voltage, dark count rate, photon detection probability and dead time are not overly important, as they can be factored in when analyzing the results. When thousands of SPADs are integrated as in SwissSPAD or even a smaller number as in LinoSPAD, however, it becomes important that there are not important parameter variations from one SPAD to the next. At the very least, one should quantify the differences in order to account for them.

With each additional level of integration, there appear further sources of non-uniformity in the integrated elements, which need to be analyzed and quantified, in order to be taken into account in current data analysis and for the conception of future systems. The two categories of non-uniformity to analyze at the sensor level are pixel-to-pixel variations and chip-to-chip variations. Microlens deposition on the SwissSPAD sensor adds additional non-uniformity and the inclusion of multiple TDCs in LinoSPAD does so as well. Furthermore, with LinoSPAD an opportunity to analyze the combined characteristics of the FPGA and the sensor over a small series of fabricated systems was offered.

4.1 SwissSPAD parameter non-uniformity

As CMOS SPAD based sensors became more and more powerful and suitable for a wide range of applications, their integration into larger arrays makes a thorough uniformity analysis necessary in order to identify deficits related to non-uniformity and further improve the sensor integration. So far, the small number of pixels in SPAD sensors allowed to neglect the issues related to array uniformity, but with the implementation of the 512 x 128 pixel SwissSPAD sensor and the goal to further increase the number of pixels and close the resolution gap with other imager technologies, uniformity analysis becomes unavoidable.

Non-uniformity in SwissSPAD is mainly composed of non-uniformity in PDE, DCR, gate timing and microlenses. Other sources of noise, which lower the uniformity of the sensor, include afterpulsing and crosstalk. These non-uniformities are in turn caused by non-uniformity of breakdown voltage, temperature and more commonly, any fabrication uncertainties predominantly in the SPAD devices and their immediate interfacing electronics. On the positive side, it can be noted that SPAD based sensors have the big advantage, compared to other image sensors, of an early digitization of the signal after which no further noise is introduced.

4.1.1 Breakdown voltage

The excess bias voltage of a SPAD operating in Geiger mode is the most important parameter for the photon detection efficiency and is defined (see 1.3.4) as the difference between the operation voltage and the breakdown voltage of a SPAD. Assuming a common operating voltage across all SPADs in an array, we measure the differences in excess bias voltage as differences in breakdown voltage as outlined in [37], using the excess count rate (ECR) method. Once the excess bias voltage reaches a level that is sufficient to switch the transistor T3, counts can be measured and the ECR versus VOP curve interpolated to find the voltage where ECR = 0. This voltage then corresponds to the breakdown voltage V_{bd} plus the threshold of T3.

Figure 4.1 shows a typical breakdown voltage distribution for a SwissSPAD chip. The average breakdown voltage over all pixels is 19.53 V assuming a detection threshold voltage of 0.6 V for T3. The standard deviation is 67 mV, which results in a relative count change below 2%. The yellow lines in the image point to defective columns of pixels or connections between the sensor and the FPGA. The irregular distribution of the breakdown voltage over the sensor makes it impossible to account for it in a systematic way without doing so on a pixel-by-pixel basis.

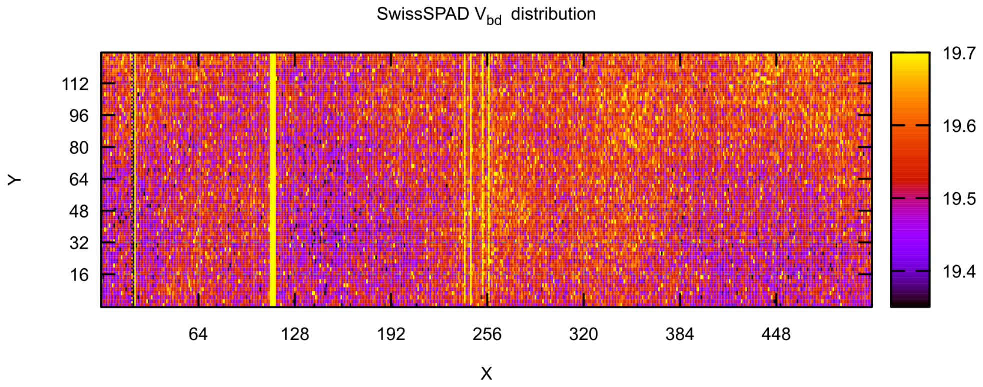


Figure 4.1: SwissSPAD spatial breakdown voltage distribution. Similar values of breakdown cluster together while extreme values are spread over the whole array. Yellow vertical lines are defective pixel columns or bad connections.

4.1.2 Dark count rate

Variations in the noise values between pixels are the most obvious in SPAD imagers, especially for hot pixels, which exhibit noise rates an order of magnitude larger than neighboring pixels. Dark counts are usually trap-assisted or tunneling-assisted with trap-assisted noise dominating for higher temperatures.

The dark count rate in SPADs follows a Poisson distribution, where the probability to measure k counts in a given time period is

$$p_{counts=k} = \frac{\chi^k e^{-\chi}}{k!}, \tag{4.1}$$

where χ is the expected value of counts.

The DCR distribution is generally modelled with a normal distribution assuming that hot pixels fall clearly outside the curve profile and introduce a right-skew. A gamma distribution can be used as a model to compare different values of skew. Lower temperatures reduce the skew and width of the distribution [77].

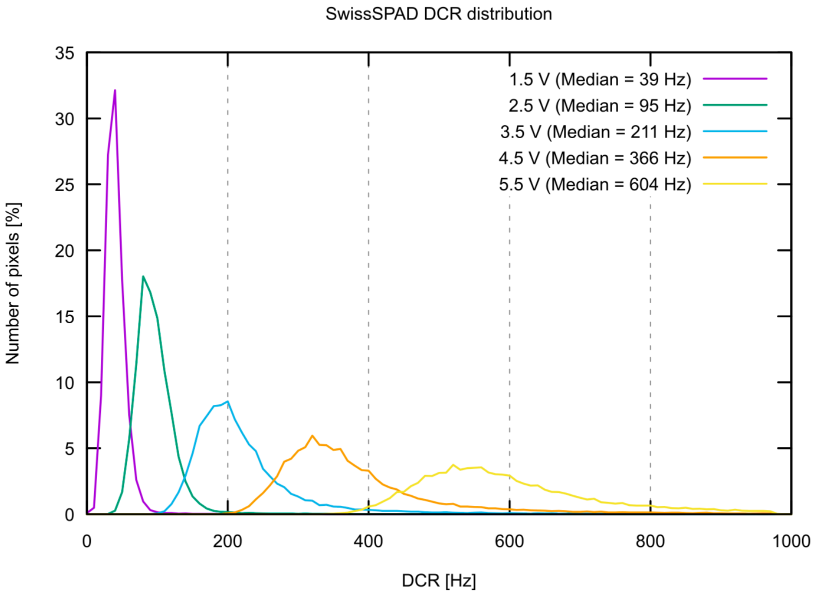


Figure 4.2: SwissSPAD DCR histogram for rising excess bias voltage. DCR distribution is shown for excess bias voltage increasing from 1.5 V to 5.5 V. The measurements were performed at room temperature with bin sizes of 4 Hz.

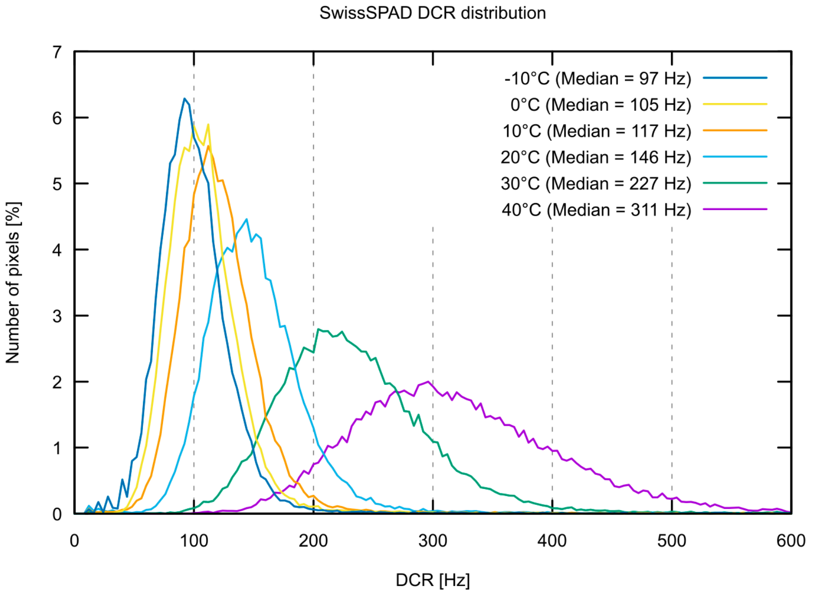


Figure 4.3: SwissSPAD DCR distribution for increasing temperatures. In the lower temperature range, where noise is dominated by tunneling, the relative change in count rate is small, whereas for higher temperatures, where thermal noise is dominant, the relative change is larger. The distributions were recorded with 3.5 V_{ex}.

While the root cause of dark counts are found in the physical realities of SPADs and cannot be eliminated, the dark count rate is influenced by the operating conditions of the SPAD sensor as well. The two main conditions that influence the dark count rate are the excess bias voltage of the SPAD and the temperature. DCR increases with both.

Figure 4.2 shows the evolution of the DCR histogram when excess bias voltages are increased. These histograms were measured at room temperature. Higher voltages lead not only to an increase in the mean count rate, but in the standard deviation as well, shown by the widening of the distribution. The skew increases due to particularly hot pixels and exponentially increasing count rates.

The DCR behavior for rising temperature is very similar to the behavior for rising excess bias voltage and is shown in Figure 4.3 over a temperature range from -10°C to 40°C for a fixed excess bias of 3.5 V.

There is a difference, however, if one looks not only at the evolution of the average count rate over temperature, but at the evolution of the median value. In a semi-logarithmic graph, we notice that the slope of the median values changes around a certain temperature.

Figure 4.4 shows the evolution of the average and median DCR values from Figure 4.3 and two straight lines indicate the change in slope for the median value. This change, which occurs around the 10° C mark for SwissSPAD, indicates the point where the noise shifts from being largely due to band-to-band tunneling to being predominantly trap-assisted.

The spatial distribution of DCR exhibits no particular pattern. Figure 4.5 shows the distribution at 20°C ambient temperature. Like in the image of the breakdown voltage above, there are a few faulty column connections and other non-working pixel columns.

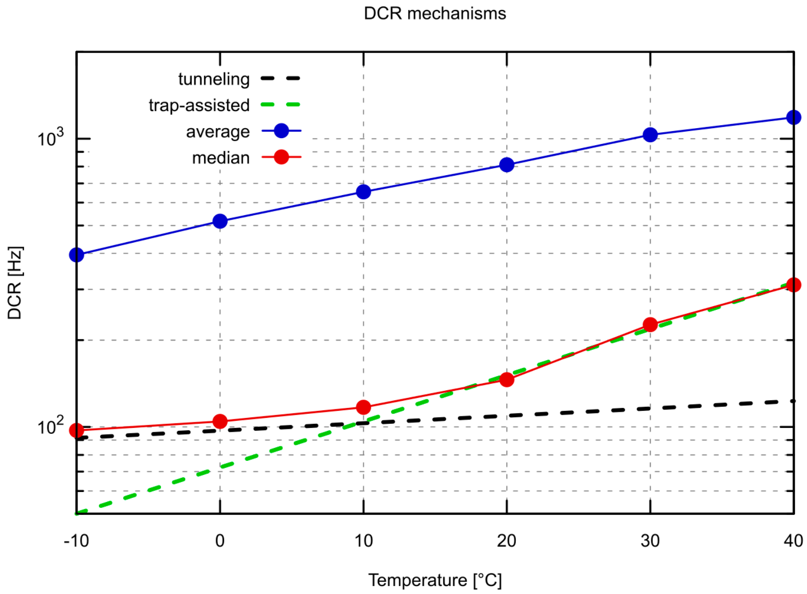


Figure 4.4: SPAD sensor DCR behavior with temperature. The median DCR value indicates the cutoff between predominantly tunneling-assisted and trap-assisted noise by a change in slope. Cooling below the cutoff has reduced influence on the median DCR. The average value, however, is largely dominated by hot pixels and still benefits from further temperature reduction.

As for the breakdown voltage, it is not possible to correct for DCR in a systematic manner. Depending on the application, a simple solution like returning a fixed value or replacing hot pixels with neighboring pixels can give good results, for example in FCS measurements [92]. The other solution is to apply count loss correction as discussed in section 2.5.5.

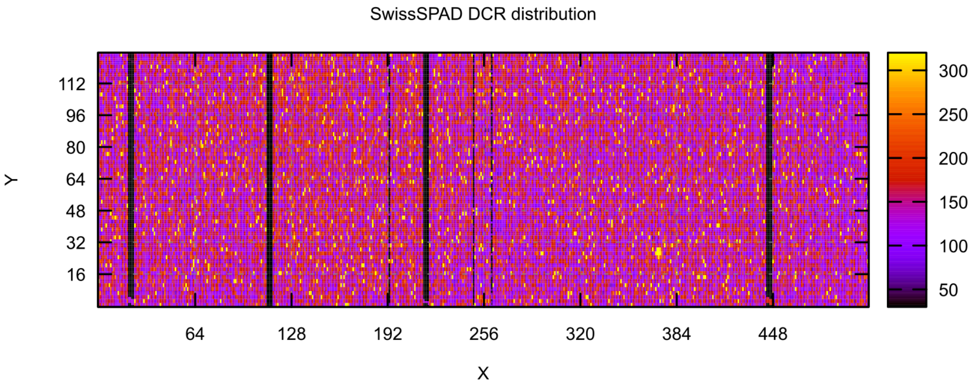


Figure 4.5: SwissSPAD spatial DCR distribution at 20°C. Comparatively few hot pixels are present in SwissSPAD. Their distribution follows no particular pattern so that they must be treated individually. The black bars show bad connections between sensor and array.

The cooling of SPAD sensors is important to minimize noise and operate as much of the pixels as possible with acceptable dynamic range. Beyond the cutoff temperature between mainly trap-assisted noise and tunneling noise, cooling effectiveness diminishes rapidly. A good operating point for SPAD sensors with a low fraction of hot pixels is therefore this cutoff temperature.

4.1.3 Afterpulsing and crosstalk

Afterpulsing relates to false events in SPAD detectors caused by the release of an electron or hole trapped during a previous avalanche event. Crosstalk events are triggered by events in neighboring pixels either electrically due to charge diffusion, or by photon emission from carrier recombination. Emitted photons can hit a neighboring pixel directly or after reflection, possibly on microlens structures. Both, afterpulsing and crosstalk are correlated noise sources and can be measured using auto- and cross-correlation respectively. Another method to measure them is from the observation of photon interarrival times from a source with known photon distribution [37]. The interarrival times on a single pixel and between neighboring pixels are used to estimate afterpulsing and crosstalk respectively. For a photon source with uniform distribution in time, the interarrival times follow an exponential distribution if the SPAD is free from correlated noise.

The histograms in Figure 4.6 and Figure 4.7 show the distributions of interarrival times for a single pixel and two neighboring pixels respectively. Due to the frame time of 6.4 µs between successive readouts, when not more than one photon can be measured, we observe no significant afterpulsing, indicated by the red exponential fit showing good agreement with the data from the start of the distribution at 6.4 µs. The fact that the SPADs in SwissSPAD can be turned off and actively recharged helps to reduce afterpulsing as well.

For crosstalk evaluation, the interarrival time distribution begins at time zero since both observed pixels could avalanche in the same frame. Other than that, the data analysis for crosstalk is not different from the afterpulsing analysis. The measurements result in less than 0.3% afterpulsing and crosstalk. This is too small to introduce significant non-uniformity requiring to be taken into account.

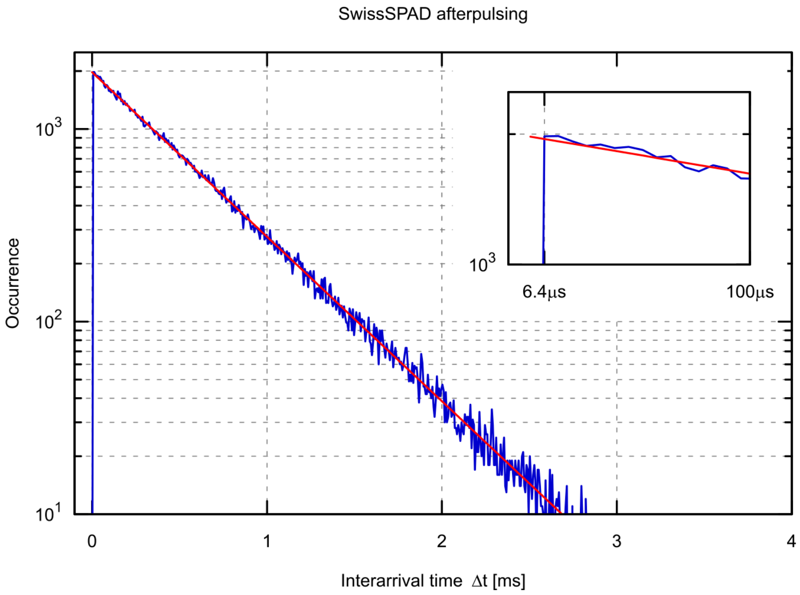


Figure 4.6: SwissSPAD afterpulsing on a pixel with exponential fit, indicating negligible afterpulsing due to the frame dead time of 6.4 μs.

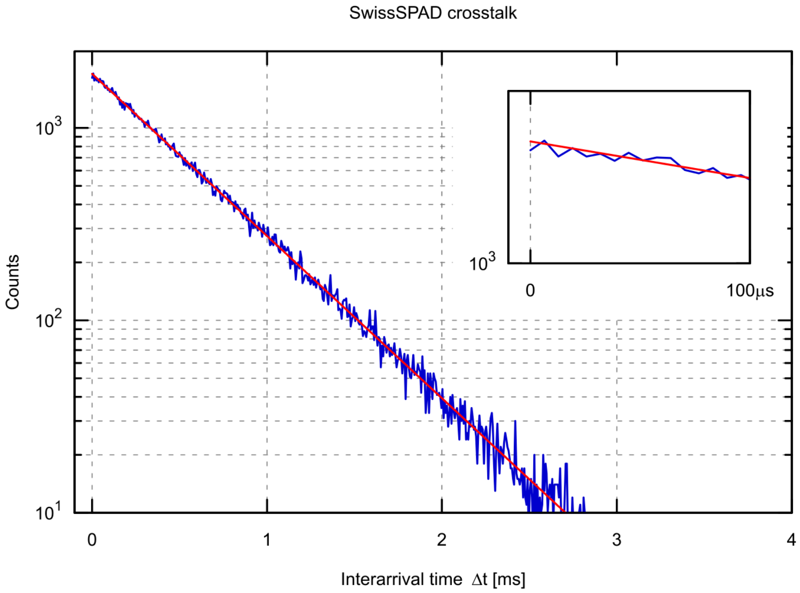


Figure 4.7: SwissSPAD crosstalk with exponential fit. Note the graph starting at time 0.

4.2 Optical enhancement for SwissSPAD

A microlens array can be used to increase the photo efficiency for a sensor with a low fill factor. The option of using microlenses to recover fill factor makes it possible to include more in-pixel functionality while maintaining reasonable efficiency. An added benefit for SPAD sensors is an important reduction in DCR when smaller SPADs are used.

A quartz mold to produce a 512 × 128 array for microlenses was designed for a SPAD sensor similar in dimension to SwissSPAD. Unfortunately this sensor was not working. In order to use the work carried out for the microlens design, SwissSPAD needed to be compatible with lenses made from the existing mold. This requirement mandated the inclusion of two alignment marks in the form of crosses with a diameter of 150 µm on the short side of the pixel array. The crosses are implemented in the top-most metal layer and are visible under a microscope used for alignment of the microlens mold during fabrication of a lens array. To use some of the space occupied by the crosses, they were partly integrated into supply voltage lines on the sides of the pixel array. Figure 4.8 shows a close-up micrograph of the two marks.

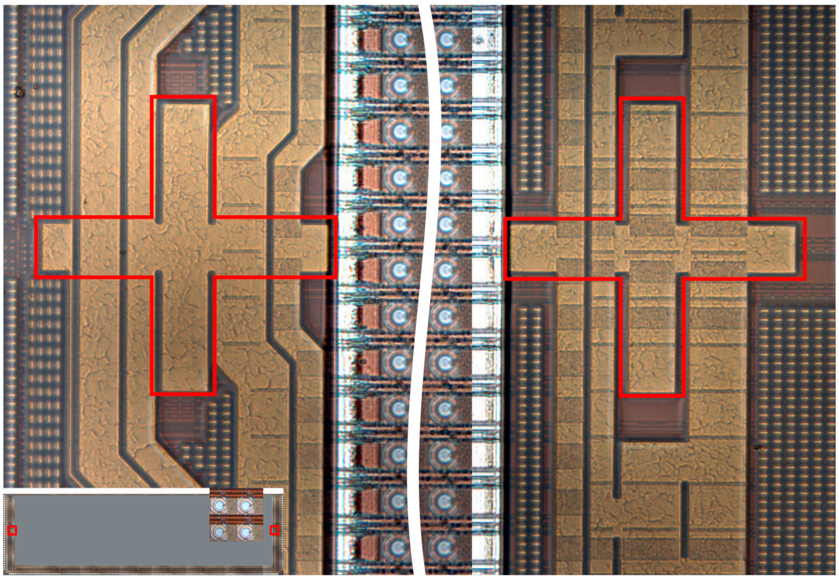


Figure 4.8: Detail image of microlens alignment marks. The image shows the left and right alignment marks for the microlens mold on the SwissSPAD chip. The crosses have a size of 150 µm and are placed at 100 µm distance on the short side of the pixel array. To save space, the crosses on the top metal layer were partly embedded in the power traces around the array.

The microlens array for SwissSPAD was specified for incoming light that is largely collimated, with a maximum angular distribution of 5.6 mrad after an optical system with a focal length of 300 mm. Accordingly, the height of the microlenses would have to be between 50 µm and 55 µm for an optimal concentration gain in this setup. An accuracy of the fabricated height to within ±5 µm, tolerance given by the manufacturer, would have negligible impact on lens performance. As we show later, this tolerance could not always be respected for microlens replications on SwissSPAD. Another factor leading to reduced microlens performance in SwissSPAD, are differences from the initially specified optical setup to the actual setups, leading to larger angular distribution of light on the sensor.

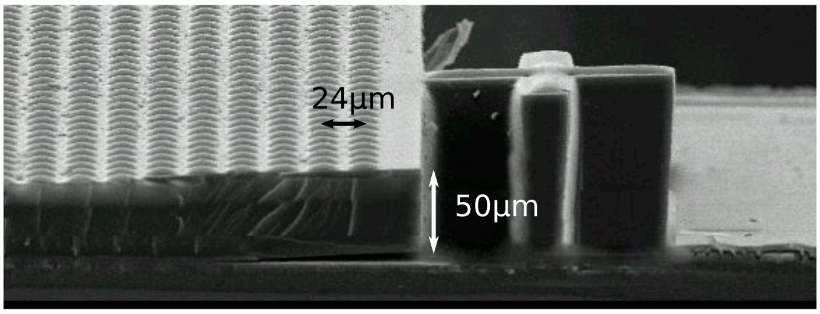


Figure 4.9: SEM picture of the microlens array. At the edge of the microlens array on a SwissSPAD, the imprinted alignment mark is clearly visible.

Fabrication of a microlens array is performed in three steps. First, a positive reference surface is fabricated using a fused silica wafer and standard lithography processes. Next, a reusable negative stamp is made from the reference using a sol-gel and including the alignment marks of the sensor. Finally, this stamp is used to replicate the microlens in a sol-gel on the chip. The polymer used for the microlenses is ORMOCER® and the height of the lenses is controlled by the distance between the stamp and the sensor. The refractive index of the material is 1.55. After microlens replication, the polymer needs to be UV cured and thermally stabilized with multiple cycles in an oven.

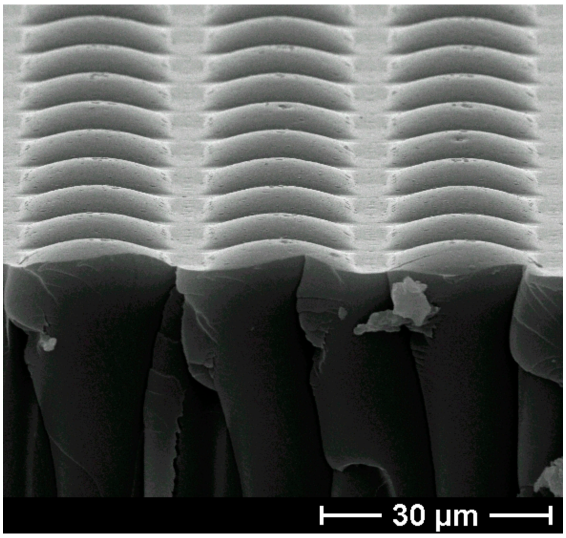


Figure 4.10: Close-up SEM image of SwissSPAD microlenses. (Image courtesy of CSEM SA.)

Microlens arrays were fabricated on SwissSPAD chips prior to chip bonding and the fabrication results were analyzed with a profilometer to verify their height, and with optical microscopy to verify alignment between the marks. Figure 4.9 and Figure 4.10 show images from a microlens array obtained with a scanning electron microscope. In the first image, the alignment mark can be clearly seen on the edge of the array.

The performance of the microlenses when illuminated over a wider range of angles than initially foreseen has been studied using simulation and observed experimentally. Using the simulator developed in [93], the difference in concentration factor caused by the change of angle in light over the long axis of the sensor has been estimated. The results in Figure 4.11 show the change in intensity for pixels from the center to the edge of the chip. Increasing the focal length of the imaging lens in the simulator reduces the angle of incidence of the chief ray at the edge of the chip, which leads to better lens performance.

To achieve as uniform concentration as possible for a sensor as large as SwissSPAD requires microlenses tailored to the optical setup intended to be used with the chip. Economic constraints often forbid changes to already fabricated lens arrays due to the high initial costs of the fabrication of the initial lens master and stamp.

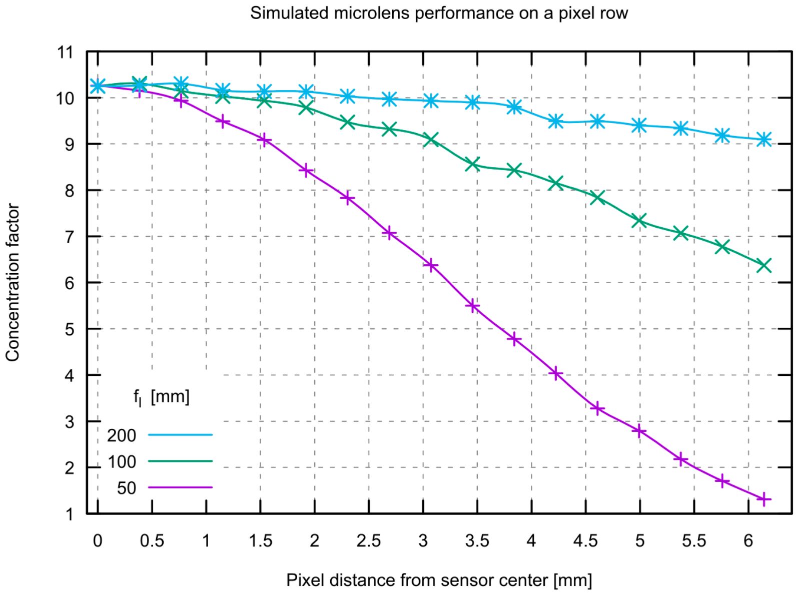


Figure 4.11: Microlens concentration factor for a line of pixels. The plot shows the simulated performance for a row of pixels on SwissSPAD from the center to the edge of the sensor. The simulations used $f/8$ resulting in a NA of 0.062. A shorter focal length lens results in a larger performance difference, because the angle of incidence on the edge of the sensor increases. The angle of incidence for the chief ray at the edge of the sensor increases from 1.8° for $f_l = 50\text{ mm}$ to 7° for $f_l = 200\text{ mm}$.

4.2.1 Fill factor recovery using microlenses

The sensitivity improvement obtained with microlenses is given as concentration factor [94,95], defined as ratio between measured light intensity with and without microlenses:

$$CF = \frac{I_{\text{lenses}}}{I_{\text{no lenses}}}. \tag{4.2}$$

Since the lenses on SwissSPAD are optimized for collimated light and do not improve sensitivity when used with uncollimated light, a lens system with variable f-number in the range 1.8 to 22 was employed to measure the concentration factor of the lenses for variable collimation. With increasing f-number, less light passes through the lens, but at the same time, the remaining light becomes more collimated. The concentration factor at rising f-numbers is calculated based on the reduction of light intensity and observed change in count rate. The f-number is defined as

$$N_f = \frac{f_l}{D} \tag{4.3}$$

where N_f is the f-number, f_l the focal length of the lens and $D \propto \sqrt{I_l}$ the diameter of the lens diaphragm opening that is proportional to the square root of the intensity. We use this relation to measure relative changes in concentration factor at different f-numbers.

Figure 4.12 shows measurement results obtained on a central spot on the chip when the f-number of the lens was increased from 1.8 to 22. Also shown are simulation results for microlenses with the same height. The simulations were used to define the reference concentration factor at f/1.8.

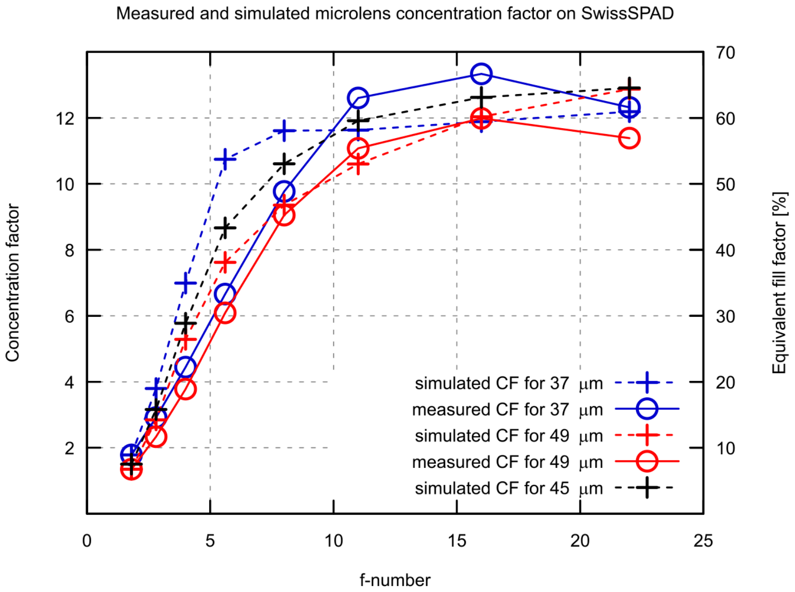


Figure 4.12: Microlens performance with array height. The results show simulated and measured performance of microlenses in a central spot for two different heights of the lens array. Based on these results the target fabrication height was optimized.

4.2.2 Non-uniformity in microlens fabrication and performance

When microlenses are placed on a SwissSPAD sensor, they add their own non-uniformities to the variations already present from one pixel to the next. The most important factor for the performance of the microlens array is the alignment of the individual lenses. Using the alignment marks, this is well controlled.

The next most important aspect about the microlens array is the uniformity in height above the sensitive part of the detector. The microlenses are designed for a focal length between 50 µm and 55 µm. The goal in the fabrication is thus to have them at this distance from the multiplication region of the SPAD. When the lenses are formed on top of the sensor, it is therefore important to know at what distance from the surface the multiplication region is, to adjust the height of the lens polymer. This is given by the optical stack.

In a CMOS fabrication process, the optical stack defines the different layers of a chip the light travels through until it hits a certain structure. From the top, there are usually a passivation layer followed by multiple pairs of silicon oxide insulation and metal layers. Substrate implantations that define the circuit devices constitute the layers at the bottom. For optically sensitive devices, it is important that the resulting stack above them is as transparent as possible.

To ensure that a maximum of photons reach the sensitive area of SwissSPAD, the pixels were carefully designed to avoid as much as possible metal on top of the multiplication region. Only a thin finger made in the lowest metal layer connects to the anode of the SPAD. The pixel is designed without the use of the top-most thick metal layer to have the largest angles of incidence possible for incoming photons. Furthermore, a pad opening was drawn over the pixel array to remove the light-absorbing passivation layer.

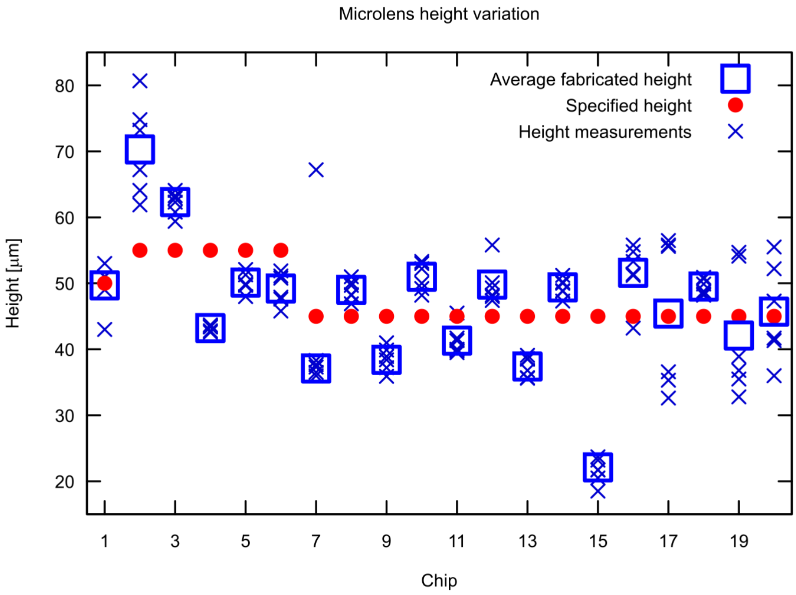


Figure 4.13: Microlens fabrication non-uniformity. This plot aggregates profilometry height measurements for microlenses fabricated on SwissSPAD. Six measurement points on the short sides of the chip were used after the first production run. The data points for index 1 correspond to individual chips in the first fabrication batch.

The first batch of microlenses that was fabricated was made with a target height of 50 µm, assuming that the height of the optical stack would be a few microns only. For the next batch, the target height was even increased to 55 µm, after performing simulations with increased precision on the microlens mold. However, when we found that lenses with reduced height were performing better we examined the optical stack again and had to conclude that it was in fact higher than initially assumed and we were using too low a baseline for the microlenses. As a result, the target height for subsequent batches of microlenses was adjusted to 45 µm.

Figure 4.13 shows the fabrication variations for a total of 25 chips that had microlenses replicated. The first dataset corresponds to 6 chips where we do not have multiple height measurements on the array, but only an average. Not all of these chips were functional. The measurement marks indicate the height at six points of the chips along the short sides. The variation in a single chip can be as large as 20 µm, which will result in a significant change in concentration factor due to distance and tilt error.

The variations in height lead to errors in the form of offset from the desired height and tilt from the desired flatness of the lens array. Using the simulator, we can evaluate the impact of these fabrication uncertainties observed on the limited number of fabricated samples. Simulations to estimate the variations in the concentration factor from the height variations result in a variation of 2.4 at a collimation of f/8.

Figure 4.14 and Figure 4.15 show microlens simulation results obtained with the simulator also used in [93] for the lenses on SwissSPAD. The first figure shows concentration factors with increasing f-number simulated for different heights of the microlenses. It is used to estimate the height of the optical stack. The second figure shows the concentration factor for a fixed f-number with increasing microlens height. It is used to find the optimal height for new arrays.

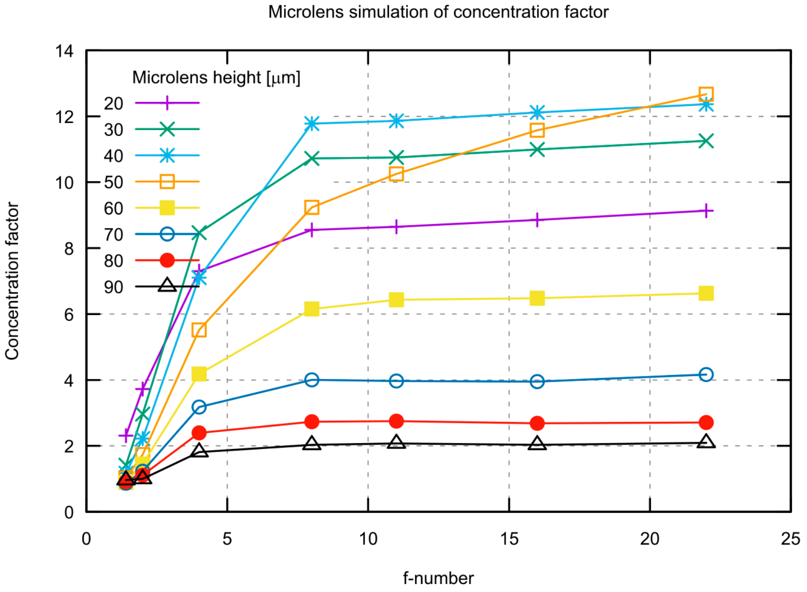


Figure 4.14: Microlens simulations for fixed heights and increasing f-number. The simulations show the behavior of concentration factor for different heights of the lenses. The results are used to estimate the height of the optical stack on a sensor.

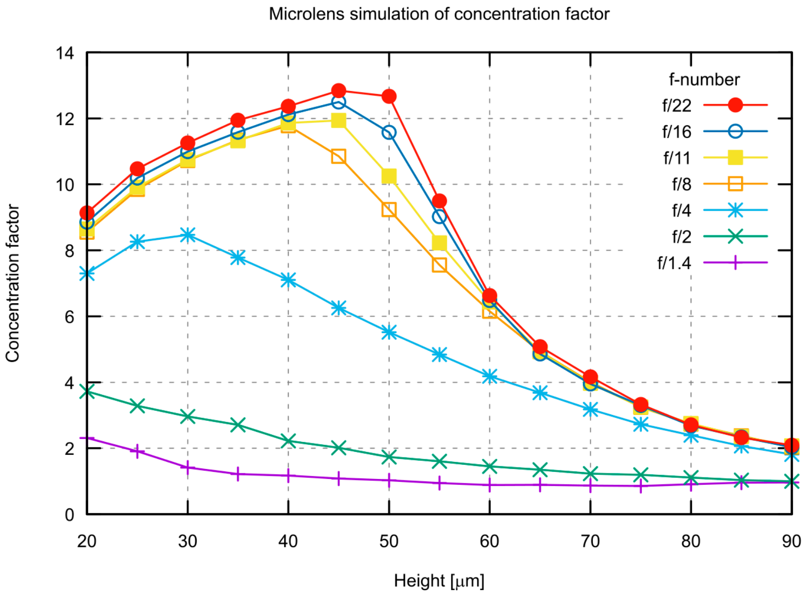


Figure 4.15: Microlens simulations for fixed f-numbers and increasing height. This representation is used to optimize the height of newly fabricated microlens arrays.

4.3 Temperature effects in SPAD sensors

An important factor for non-uniformity to consider with large format detectors is temperature. Temperature changes naturally become more important as the sensor grows in size due to increased power consumption.

In SwissSPAD, the design of the pixels with an NMOS-latch leads to a considerable static power dissipation, in addition to the dynamic power consumed when the SPADs, readout logic and I/O are operating. In LinoSPAD, with simpler pixels, most power is consumed by the output buffers. With twice the amount of output buffers compared to SwissSPAD, LinoSPAD loses the advantage of the fully CMOS design and consumes as much power as SwissSPAD.

4.3.1 SwissSPAD turn-on temperature changes

The first effect related to temperature, which we want to illustrate is the heating after power is applied to the sensor. Figure 4.17 shows how, after turning the camera on and starting to acquire images, the average count rate recorded rises over time until it stabilizes after the chip reaches thermal equilibrium. This effect is due to the heating of the sensor, which leads to increased DCR. It could be partly mitigated by cooling. Cooling the sensor leads to a faster temperature stabilization. Figure 4.17 shows the same behavior when the sensor is in a temperature controlled environment where it stabilizes faster after a small increase in relative count rate.

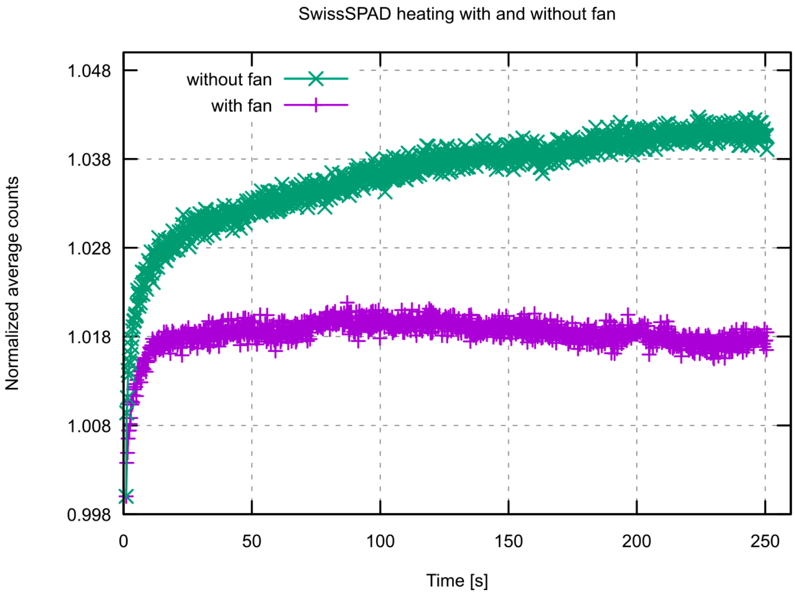


Figure 4.16: SwissSPAD heating effects after applying power to the sensor. Adding a fan reduces the time it takes to reach the temperature equilibrium of the chip.

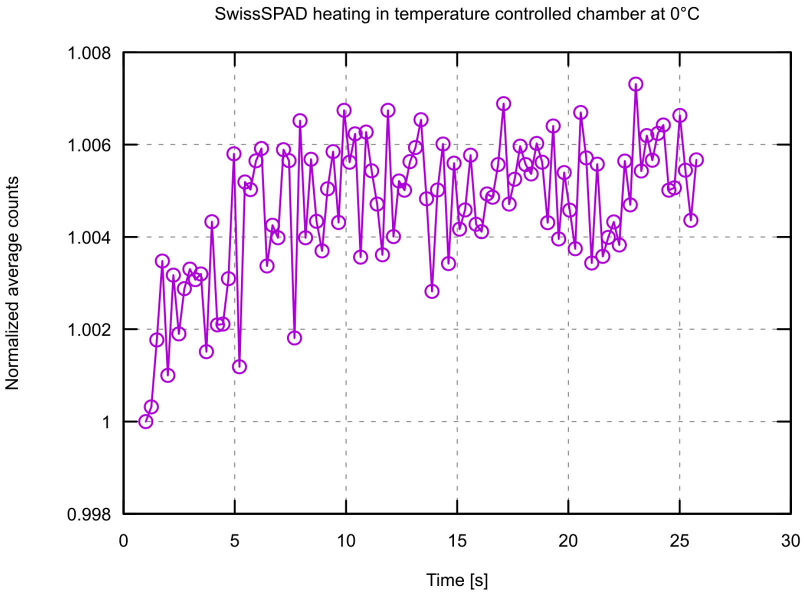


Figure 4.17: SwissSPAD heating in a temperature controlled environment. When SwissSPAD is used in a temperature-controlled chamber at 0°C there is very little relative change after applying power and thermal equilibrium is quickly reached.

As temperature induced variations take place on a relatively long time-scale compared to the measurements carried out with SPAD sensors, it is in most cases sufficient to wait after turning on a sensor until it has reached its thermal equilibrium to ensure stable operating conditions during measurements. A similar procedure does actually apply to the use of almost every test and measurement equipment.

A temperature-controlled environment, especially a cooled one is, of course, beneficial in lowering the DCR as shown in section 4.1.2.

4.3.2 LinoSPAD TDC resolution variation with temperature

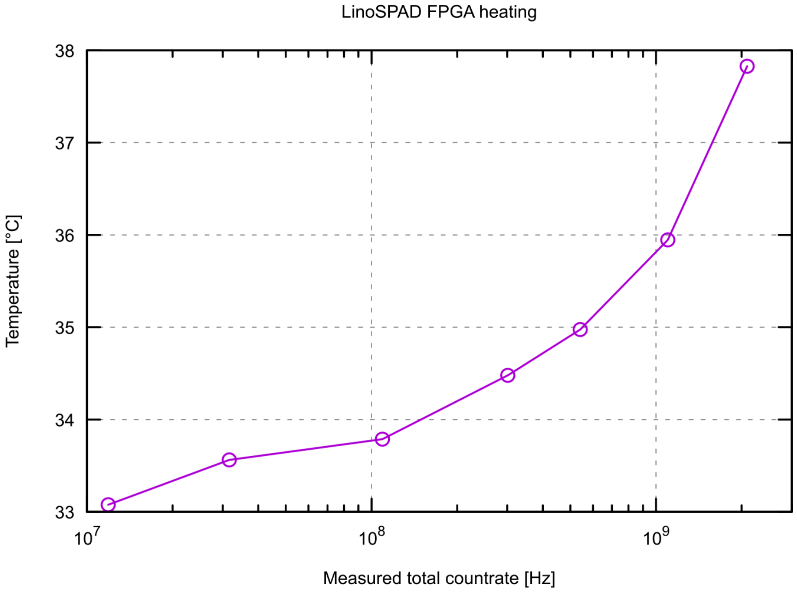


Figure 4.18: FPGA heating on the LinoSPAD camera.

In the LinoSPAD camera, it is the delay lines in the FPGA, which are most visibly affected by changing temperatures. The temperature of the FPGA changes depending on its activity, and mainly due to the high switching rates in the TDC front-end clocked at 400 MHz. Figure 4.18 shows the temperature changes measured next to the FPGA when the illumination increases. The FPGA was cooled using a fan and aluminium radiator during these measurements carried out over 30 seconds for each illumination setting. At the exterior of the FPGA, the temperature changes slowly relative to typical measurement times, due to the thermal mass of the package. Locally in the FPGA, temperature changes are expected to take place on a shorter time scale.

In Figure 4.19, we show the average delay line length of the TDC array as the temperature increased. The code span ranges from 120 for the lines with the longest propagation delay, to 135 for the lines with the shortest delays. As the temperature increases, the average length becomes shorter, indicating that the signal propagates slower and the average resolution is therefore reduced. For best performance of the TDC correction, it is necessary to stabilize the FPGA temperature as much as possible and recalibrate after important changes.

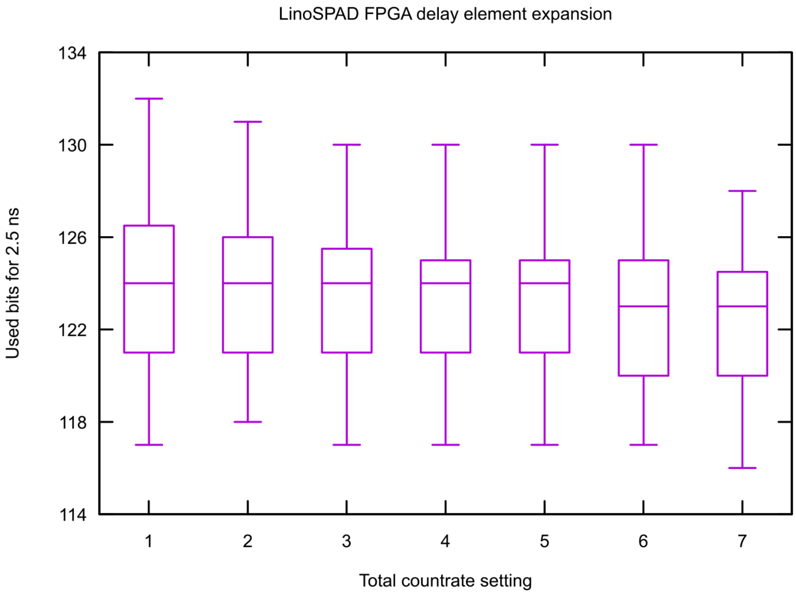


Figure 4.19: TDC delay variation with average count rate. This plot shows the number of delay elements that are used to cover the 2.5 ns sampling period. Increasing delay in the elements is shown by the decreasing number of used elements with rising temperature.

4.4 LinoSPAD non-uniformities

Many of the non-uniformity challenges discussed on SwissSPAD apply to LinoSPAD as well. With the comparably lower count of pixels, the amount of statistics and its significance are reduced. On the other hand, the LinoSPAD system adds another category of non-uniformities related to the pixel circuits implemented in the FPGA. We also had the chance to analyze the LinoSPAD performance by comparing a small series of identically fabricated systems to assess the range of natural non-linearity for such systems.

This subsection discusses non-uniformities at the pixel level in terms of dead time and afterpulsing before discussing chip-to-chip variations. Variation between TDC modules in one FPGA and between different FPGAs are briefly characterized as well.

4.4.1 Dead time and afterpulsing

The LinoSPAD sensor is passively quenched using the CMOS transistor connected to the SPAD cathode. The bias voltage of the transistors controls its equivalent resistance and hence the current that flows through the SPAD when an avalanche occurs. After an avalanche event, the SPAD bias is restored through the quenching transistor before the diode can detect another photon. The time it takes for the SPAD to be recharged is defined as the dead time of the diode, since during this time-period it is unable to generate any events.

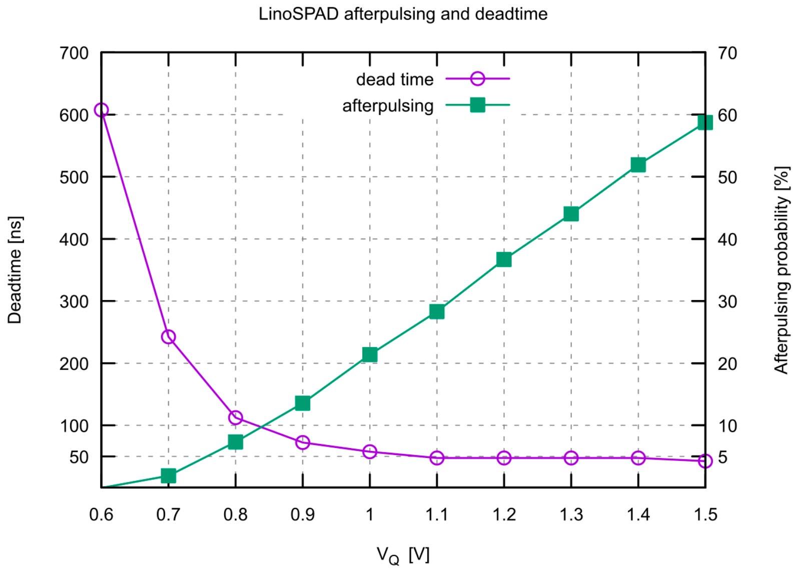


Figure 4.20: LinoSPAD dead time and afterpulsing. Dead time and afterpulsing in LinoSPAD depend on the quenching bias set by the gate voltage of the bias transistor. The dead time reported corresponds to the peak position in the interarrival time histogram.

Sometimes the dead time is defined as the time window during which the SPAD output remains active, which is arguably not precise, as it does not take into account the digitization threshold. For LinoSPAD, we define the dead time as the shortest inter-arrival time observed between two consecutive events, which is also a more practical definition for measurements [37].

Figure 4.20 shows the measurement of the dead time as a function of increasing quenching voltage. A voltage of 0.6 V on the gate of the quenching transistor is barely enough to recharge the SPAD, and, as a result, the smallest interarrival time that can be observed is roughly 600 ns. As the voltage increases, more current is allowed to flow through the SPAD and the SPAD bias restores faster. The smallest interarrival times decrease quickly to roughly 250 ns for 0.7 V quenching voltage and further to 50 ns, the minimum time needed for a full detection cycle.

Because of lower dead times and rising quenching voltage, the afterpulsing probability, also shown in Figure 4.20, rises considerably. From non-measurable values for the lowest quenching voltage, it increases up to almost 60% for a quenching voltage of 1.5 V.

Afterpulsing for LinoSPAD is measured in the same way as for SwissSPAD in section 4.1.3. Using a constant current driven LED illuminator, the sensor measures time-of-arrival timestamps that should be uniformly distributed in time. From these timestamps, we can easily obtain the interarrival times and display a histogram as shown in Figure 4.21. The histogram starts at the value of the dead time.

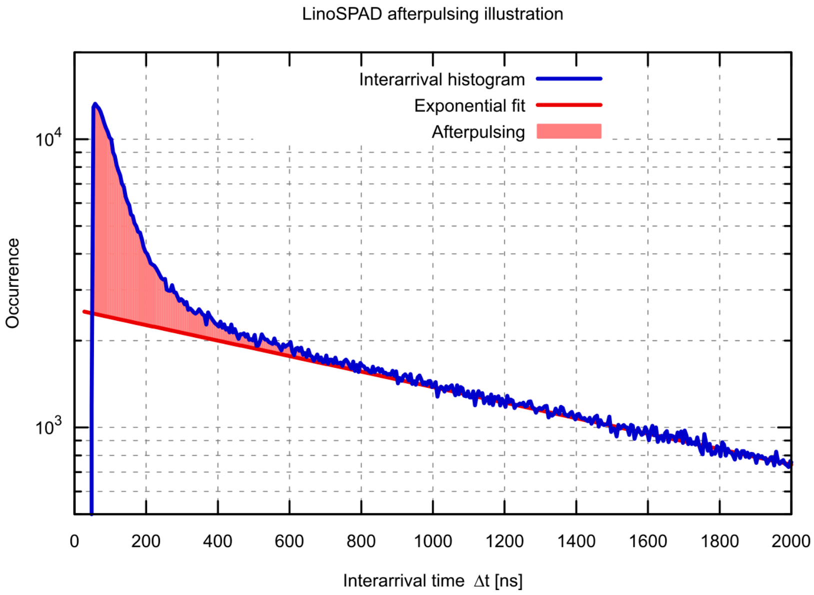


Figure 4.21: Illustration of afterpulsing in LinoSPAD. The blue curve follows the histogram of interarrival times from a million data points. The red fit is calculated on the part after 1 µs.

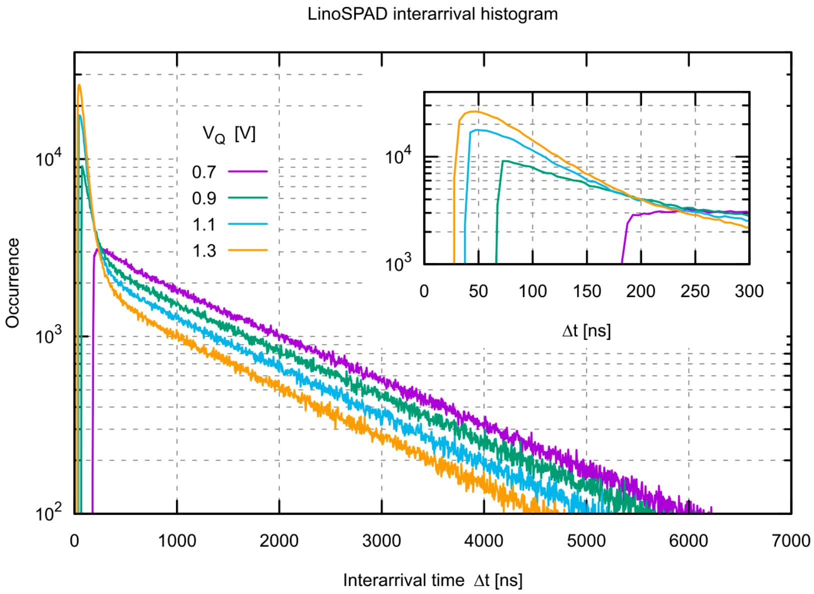


Figure 4.22: LinoSPAD interarrival time histograms. The curves correspond to different voltages of VQ between 0.7 and 1.3V with increasing afterpulsing occurring at short interarrival times.

The interarrival times of uniformly distributed photons (and DCR) should follow a decaying exponential in the absence of correlated noise. To find the afterpulsing, we fit an exponential to the histogram for interarrival times greater than 1 µs where we assume negligible afterpulsing. The exponential fit is also shown in Figure 4.21 as is the area between the histogram and the fit. This shaded area divided by the total area below the histogram gives the afterpulsing probability [37].

Figure 4.22 shows histograms of interarrival times for quenching voltages between 0.7 V and 1.3 V. In the semi-logarithmic presentation, the increasing peaks for short interarrival times clearly indicate the presence of afterpulsing.

Crosstalk was also estimated for LinoSPAD based on histograms of interarrival times between events on two neighboring pixels. The resulting histograms look very similar to the ones of Figure 4.22 with the difference that the decay starts at time zero. No significant crosstalk could be measured for LinoSPAD using this method.

4.4.2 TDC-to-TDC variation

From the non-uniformities in a single LinoSPAD sensor, we now shift focus to non-uniformities inside the FPGA that is used to make measurements with the sensor, and, together with the sensor, constitutes the full camera system. Common boxplots are used to show variation for these measurements. The box in the boxplot spans the range between the first and third quartile of the values. A bar is placed at the median. The whiskers in our plots extend to the maximum value within a range of 1.5 times the interquartile distance from first or third quartile. Outliers are shown individually.

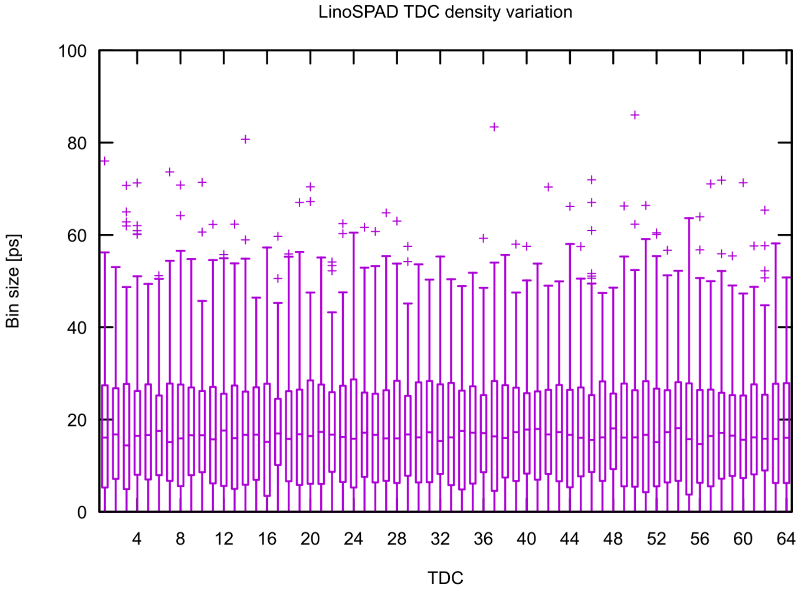


Figure 4.23: LinoSPAD TDC-to-TDC bin size variation. The average bin size for the 64 TDCs on a FPGA show little variation. However, almost every TDC has a few outliers bin with significantly larger delay. No relation between the delay length and the location in the FPGA could be observed.

In our firmware, we implement 64 time-to-digital converters in a low-cost and relatively low performance FPGA as described in Chapter 3. The basic building block of these TDCs is a delay line built using dedicated carry blocks to delay a signal. The delay from one output of a carry block to the next is variable and the only boundary quoted from the FPGA manufacturer is the maximum delay through one complete carry block with four outputs.

The datasheet [96] lists the maximum delay from the carry input CIN to the carry output COUT, with 80 ps for the fastest speed-grade, which is the one we are using. This sets a baseline for the delay across 4 bits of the delay line. The observed delay from one bit to the next is not simply the 80 ps divided by 4, because the delay line uses fast adders, and clock skew affects the observation. However, the delays listed from the CIN input to the outputs of the slice are given as 210 ps, 300 ps, 290 ps and 310 ps, significantly more than the 80 ps needed to traverse the slice from CIN to COUT. This is due to the additional multiplexers involved in routing these signals to the outputs of the slice. Similar delays are assumed to exist between the delay line and the register inputs within the same slice.

Summarizing the previous paragraph, we can say that on average the delay in the TDCs should be no more than 80 ps per block of 4 bits. However, the observation of these 80 ps is skewed due to different paths from the carry block to the registers, each with individual skew of the clock.

Figure 4.23 shows the variation of the delay between all TDCs implemented in a FPGA. The estimated bin-sizes were obtained from a density test explained in section 3.2.4. We find that 75% of the bins are below 30 ps in size, but almost every TDC has a few outliers, where the delays reach up to 90 ps. In contrast with the findings in [97] for a Virtex 6 FPGA, the clock regions in Spartan 6 FPGAs appear homogeneous. No correlation between the large delay bins and location in the FPGA could be found.

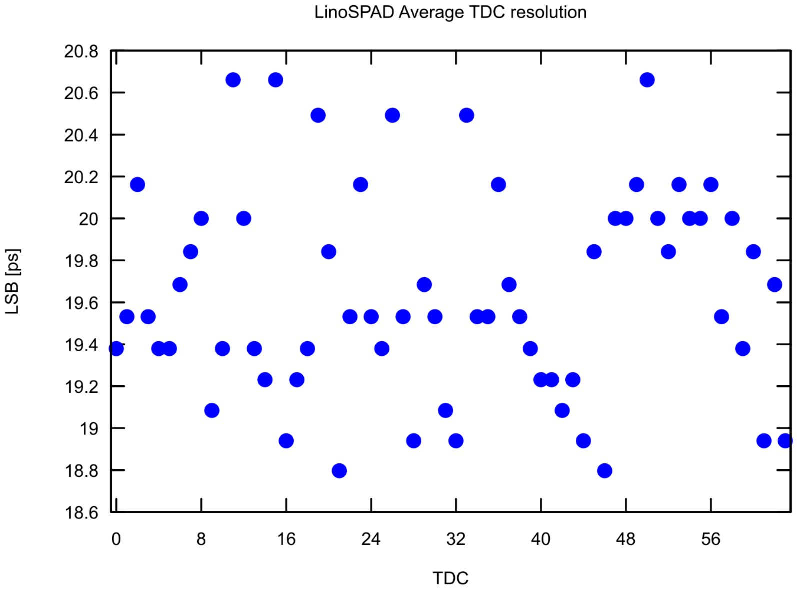


Figure 4.24 Average FPGA TDC resolution. The average resolution of a TDC in the FPGA is calculated by dividing the sampling period by the number of delay line bits used in a density test.

From the density test, we also extracted the average delay per element when unused elements are removed. The results are shown in Figure 4.24. The delay lines offering the highest resolution use 133 elements to resolve the 2.5 ns sampling period, while the lines with the lowest resolution use only 121 elements.

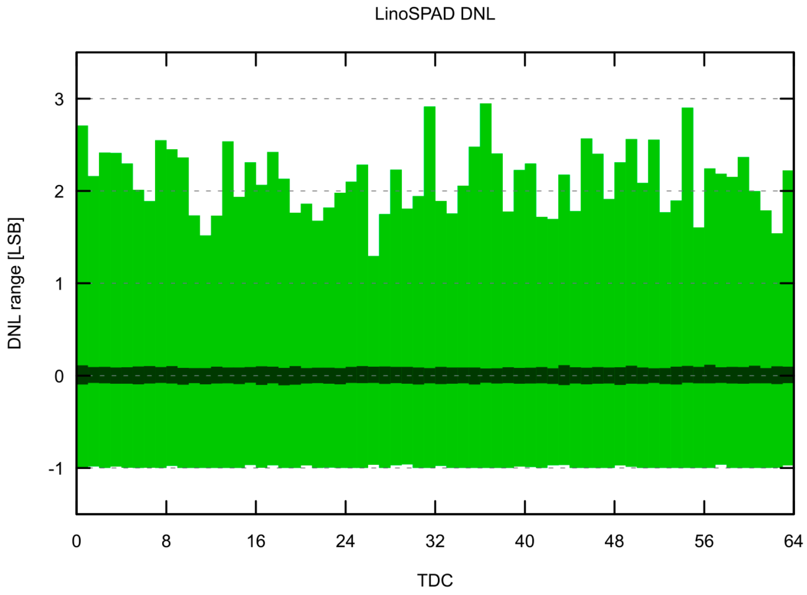


Figure 4.25: TDC DNL correction performance. The plot shows the TDC DNL range before (light) and after (dark) correction. DNL is bounded by -1 from the TDC construction.

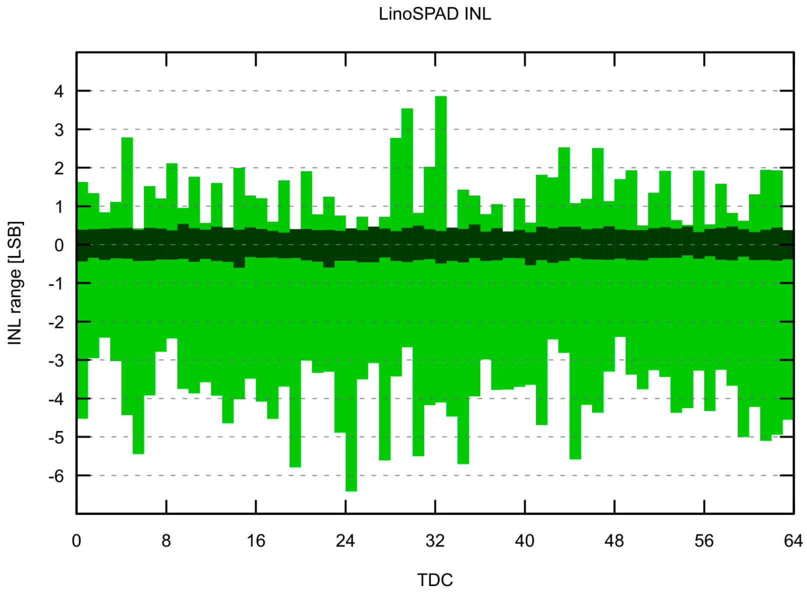


Figure 4.26 TDC INL correction performance. The plot shows the TDC INL range before (light) and after (dark) correction. The larger negative range before correction is caused by not fully used bins at the end of the delay line.

The density test of the delay lines also gives us the range of the differential and integral non-linearity for each TDC. The performance of the real-time correction for individual TDCs is shown using density measurements with and without correction and displaying the results in the same graphs.

Figure 4.25 and Figure 4.26 show the range of DNL and INL, respectively for each TDC with and without real-time correction applied. For the range without correction, the unused bins have been removed not to distort the (minimum) INL excessively. The unused bins of the delay line tend to introduce overly excessive negative excursion in the first bins of the INL because of their very low occurrence.

4.4.3 Sensor-to-Sensor variation

Next, we want to quantify the differences in characteristics and performance between different sensors. A set of ten LinoSPAD cameras has been manufactured using sensors from one production run and identical components for the FPGA board. In the following measurements, we quantify the differences in characteristics and performance across these systems and show the results.

The first characteristic to compare is the breakdown voltage of the sensors. The breakdown voltage of each sensor is measured on a pixel-by-pixel basis using the DCR-fit method described in 3.5.1.

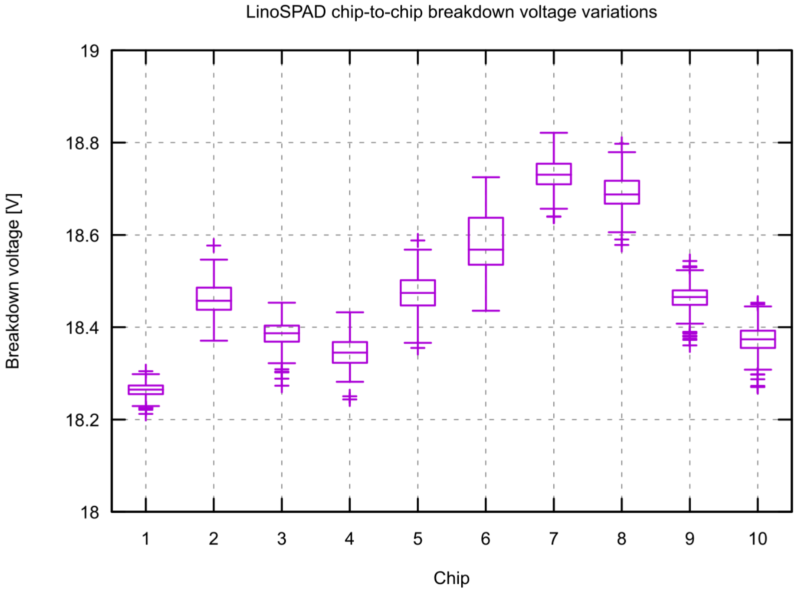


Figure 4.27: LinoSPAD chip-to-chip breakdown voltage variation. The breakdown voltage on a sensor shows good uniformity, below the peak-to-peak difference in breakdown value encountered over multiple chips.

Figure 4.27 shows the variation in breakdown voltages between different LinoSPAD sensors fabricated at the same time in a multi-project wafer production run. As the SPAD structures use process properties that are not guaranteed by the foundry for the multiplication junction, their behavior can vary. Even for the small sample size analyzed here, we look at a span of about 0.5 V difference in average breakdown voltage. The span of breakdown voltages on a single chip is around 200 mV, with a standard deviation below 50 mV.

The next characteristics to compare was the dark count rate. For the comparison, the excess bias voltage was fixed to 2 V on each sensor and the resulting DCR distributions are summarized in Figure 4.28. The median count rate for most sensors is between 1 kHz and 2 kHz and, due to the nature of the DCR distribution discussed in section 3.5.2, there are many outliers in the boxplot representation.

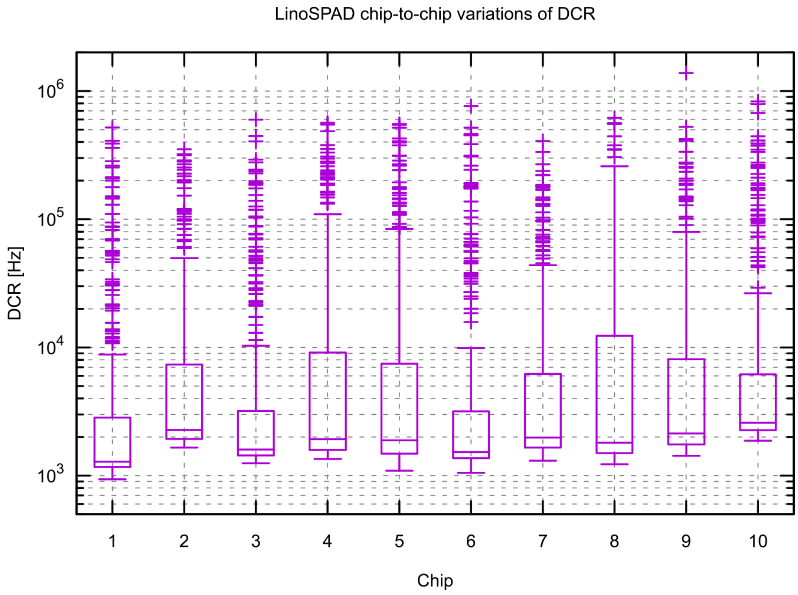


Figure 4.28: LinoSPAD chip-to-chip DCR variation. The values were measured for V_{bd} = 2 V. The DCR shows little variation from one sensor to another compared to the DCR range on a single sensor.

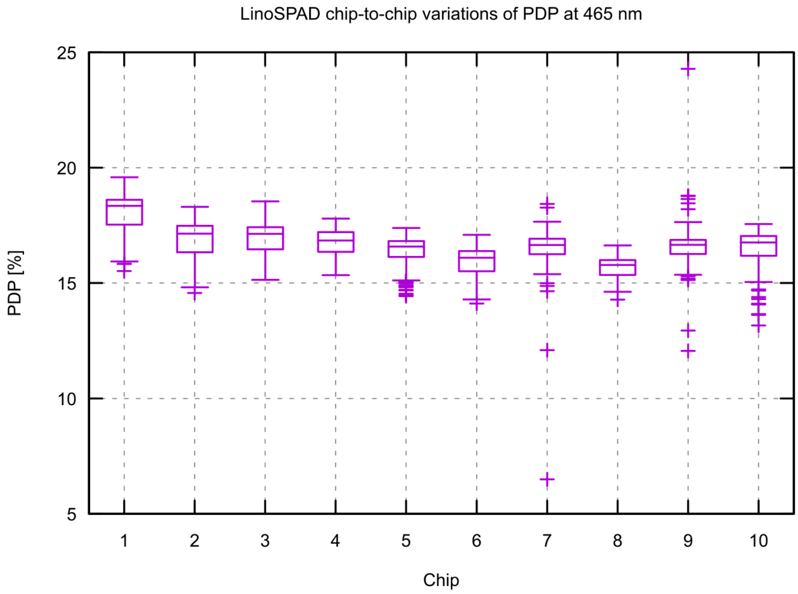


Figure 4.29: LinoSPAD Chip-to-chip PDP variation at 465 nm. LinoSPAD chip-to-chip variation of PDP has been measured at 2 V excess bias. The identical excess bias on each chip offsets the variation in breakdown voltage shown in Figure 4.27.

The last performance characteristics compared between the sensors is the photon detection probability at two different wavelengths, one close to the peak sensitivity at 465 nm and the other at around 50% of the peak sensitivity at 640 nm. The PDP measurements reported in Figure 4.29 and Figure 4.30 were obtained using 2 V excess bias voltage and 1 μW/cm^2 illumination. The PDP behavior with rising excess bias and rising illumination intensity is discussed in more detail in section 3.5.3. In general, we observe that the lower PDP brings the chips responses, but also the pixels on a single chip, closer together.

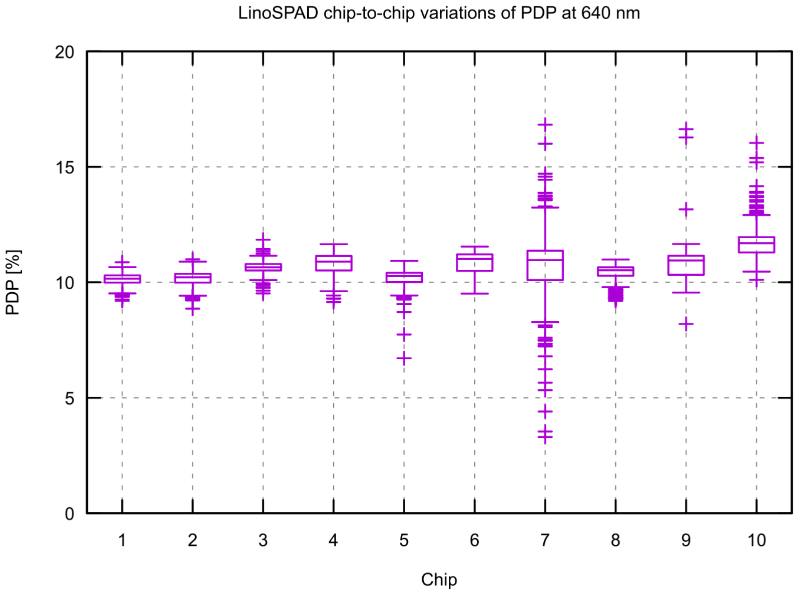


Figure 4.30: LinoSPAD Chip-to-chip PDP variation at 660 nm. PDP at 660 nm has been recorded using 2 V excess bias. The reduced PDP compared to the measurement at 465 nm resulted in likewise reduced variation.

From the PDP measurements, we conclude that there are, as of yet, unidentified problems with chip #7 showing very high spread of the 640 nm PDP values. Another problem was identified in chip #9 where two pairs of pixels with neighboring bonding pads each have a short-circuit of a few ohms between their bond wires. These short-circuits lead to these pixels reporting a count rate close to twice the expected value due to induced crosstalk.

The measurements of breakdown voltage, DCR and PDP over a set of 10 fabricated sensors indicate good uniformity values compared with SwissSPAD. Variations of breakdown voltage and PDP within a sensor are less than variations between different sensors. LinoSPAD also proved to be a good design when looking at fabrication yield. No defective pixels were found so far.

4.4.4 FPGA-to-FPGA

The last category of non-uniformities we want to discuss briefly in this chapter, are those from one FPGA to another. Since we do not have any control over the FPGA manufacturing, we can only characterize the non-uniformities we observe on the circuits we use. Furthermore, only some of the specifications we rely on in the TDCs are guaranteed by the manufacturer in the datasheet.

Figure 4.31, Figure 4.32 and Figure 4.33 show variations over the 10 FPGAs analyzed using the characterization presented in Chapter 3. The trend visible in the variation of the delay line length necessary to measure a period of 2.5 ns is indicative of the spread of logic delay in a single speed-grade of the FPGAs. Clearly, there are FPGAs with, on average, smaller logic delays than others.

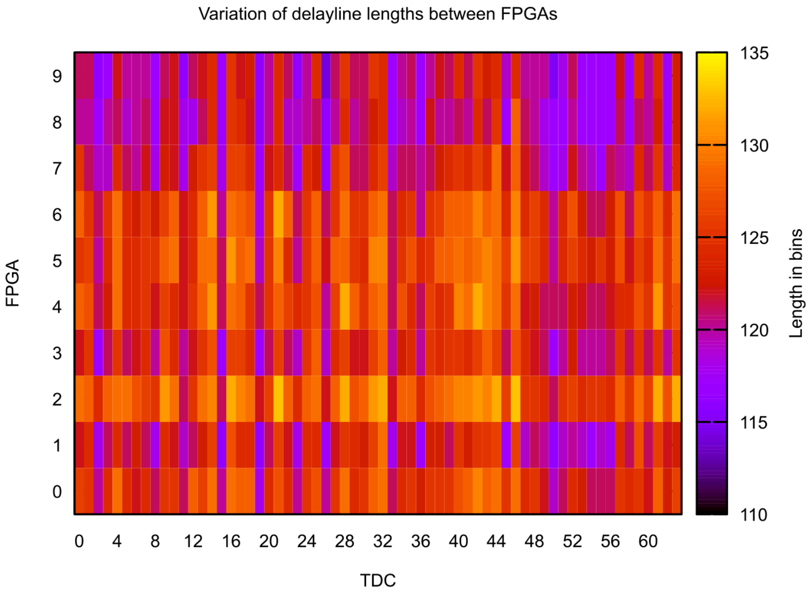


Figure 4.31: FPGA-to-FPGA variation of delay line lengths. The similar color over full rows indicate that some FPGAs have consistently smaller logic delays. A known characteristic of CMOS fabrication that results in speed binning [98,99]. All 10 FPGAs are of identical speed-grade.

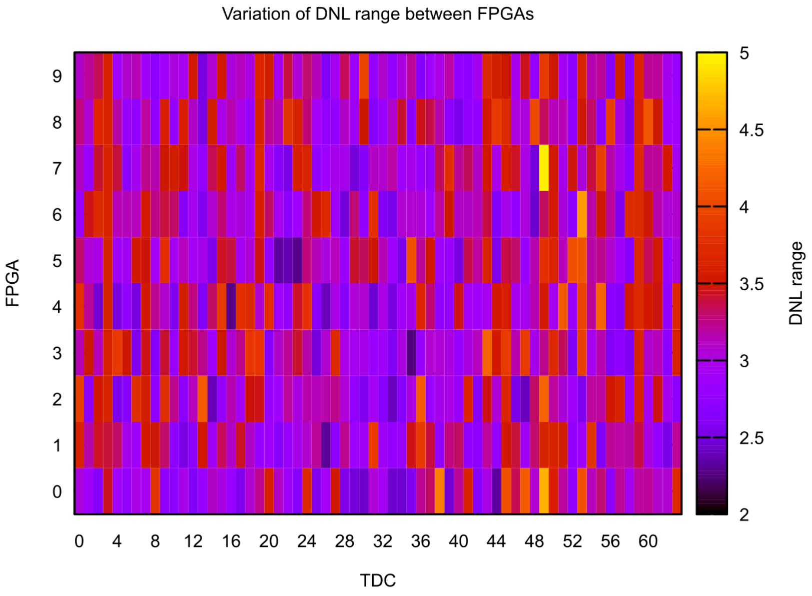


Figure 4.32: FPGA-to-FPGA variation of DNL range in the TDCs. Improvements could be made by placing the TDCs that show consistently higher than average DNL range over all FPGAs in other locations.

The recognition of vertical bands of similar values in all figures hints at possible improvements by finding better locations on the FPGA for consistently sub-average performing TDCs. It may be possible to forbid particularly bad placements, but more work will clearly be needed to account for these issues, as evidenced by the difficulty in design synthesis reported in section 3.3.

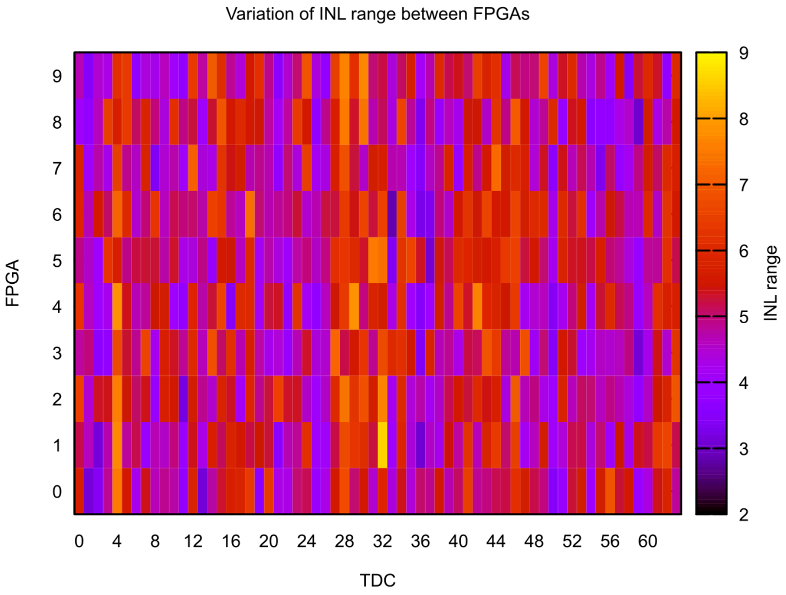


Figure 4.33: FPGA-to-FPGA variation of TDC INL range. The INL range hints at possible improvements similar to the DNL range, though it is difficult to read them directly from this image because of the TDC architecture with unused bins that leads to biased results.

This chapter analyzed and described many sources of non-uniformity in SPAD based imagers. We analyzed non-uniformity of the sensors alone and in combination with a FPGA. The results indicate that there are some avenues for systematic improvement of the parameters. The most demanding applications however, require individual compensation of the non-uniformity on a pixel-by-pixel and TDC-by-TDC basis.

Chapter 5 Results

In this chapter, we present the measurement results and insights from the main applications of the two SPAD camera systems presented over the previous chapters. SwissSPAD has been primarily used in microscopy applications of which fluorescence lifetime imaging microscopy (FLIM) was the first one attempted. Later, the camera has also been used in fluorescence correlation spectroscopy (FCS) and lately in super resolution microscopy. The focus here is on the FLIM results measured by the author, while other results are briefly summarized and references given.

The generation of true random numbers has become another important application for SwissSPAD and the results obtained are presented in this chapter as well. This work led to the development of other specialized circuits and further evaluation is still ongoing. Furthermore, SwissSPAD can be used as generic high-speed camera with extremely fast shutter and can be used for synchronous imaging of fast repetitive phenomena with high accuracy.

LinoSPAD being the more recent camera has not yet seen a spectrum of applications as broad as SwissSPAD. The key benefit of the LinoSPAD camera is the close coupling between the SPAD array and a large FPGA. This combination makes it ideally suited to prototype novel sensor architectures and to implement proof-of-concept applications. Here we present 3D time-of-flight measurements obtained with the LinoSPAD camera using the reference FPGA firmware described in Chapter 3.

Table 5.1 tries to compare the quite unlike systems on their key parameters.

Parameter	SwissSPAD	LinoSPAD	Unit
Process	0.35µm	0.35µm	HV CMOS
Chip size	13.5 x 3.5	6.8 x 1.68	mm^2
Format	512 x 128	256 x 1	pixel
Pixel pitch	24	24	µm
Dead time (typical)	100	100	ns
Nominal fill factor	5	40	%
Peak PDP (V_e = 3.5V)	30 @ 460	28 @ 470	% @ nm
Crosstalk	0	0	%
Afterpulsing	0	<10	%
Data lines	128	256	
I/O bandwidth	10.24	2.56	Gbps
Frame readout time	6.4	0.1	µs
Idle power consumption	660	0	mW
Active power consumption	1500	2000	mW

Table 5.1 SPAD camera comparison.

5.1 SwissSPAD

5.1.1 FLIM experiments

FLIM is a powerful imaging technique used in a wide range of biological measurements [100]; it distinguishes itself from other fluorescence imaging techniques by measuring fluorescence decay behavior and not only the fluorescence intensity and spectrum. The typical lifetimes of commonly used fluorophores have lifetimes in the nanosecond range, but some, like ICG, are in the picosecond range [101]. The observed lifetimes and lifetime changes of fluorophores reveal information about their type and their environment.

Fluorescence lifetime measurements are typically carried out using fast-pulsed excitation illumination and time-correlated single-photon counting methods to extract decay lifetimes from time-of-arrival statistics of photons or using precisely modulated illumination with synchronous detection to extract phase distortion that is related to the fluorescence lifetime.

The SwissSPAD camera has a global shutter that can generate light sensitive windows of a few nanoseconds and high uniformity across the pixel array. The camera firmware can generate this acquisition window at a specific point in time relative to a reference signal. Additionally it is possible that the position of these windows relative to the reference signal is shifted using the PLL inside the FPGA by steps of a few tens of picoseconds. Details of the mechanism are explained in Chapter 2.

Using the global shutter of SwissSPAD, we acquired fluorescent decays with a procedure very much similar to the one used to characterize the gating circuit in Chapter 2. The first series of measurements and analysis of the results were carried out in parallel to measurements with another system using the single-photon synchronous detection method [36], with results published in [53]. The second series of FLIM measurements were acquired in cooperation with Dr. X. Michalet and Prof. S. Weiss in the Department of Chemistry & Biochemistry at UCLA.

While the results in the following sections do not show lifetime images with spatially separated features, but rather an image of the laser excitation spot, the proposed methods result in lifetime measurements for each pixel. A full imaging system can be constructed by changing the illumination from a spot to an area, as it was actually the case for the measurements at UCLA. However, the efficiency of SwissSPAD was, without microlenses at the time, not sufficient to acquire lifetime images of structured targets.

FLIM measurements using ICG

For the first measurements of fluorescence lifetime, we used Indocyanine Green (ICG), a fluorescent marker approved by the US Food and Drug Administration (FDA). ICG was diluted in water and milk (3.5% fat) with concentrations from 2 µM up to 160 µM, to be able to compare the measurements with those in [102]. The lifetimes of ICG in these settings is below 800 ps, a very short lifetime compared to the shortest gate possible with SwissSPAD. The setup used for the measurements is shown in Figure 5.1. A picosecond laser with 100 MHz repetition rate and 55 ps pulse width emits light at 790 nm. The light is reflected by a dichroic mirror and focused in the fluorescent solution. Fluorescence light is collected through the dichroic mirror and a wavelength filter and focused on the camera. Figure 5.2 shows a typical intensity profile from a bright sample seen by the camera.

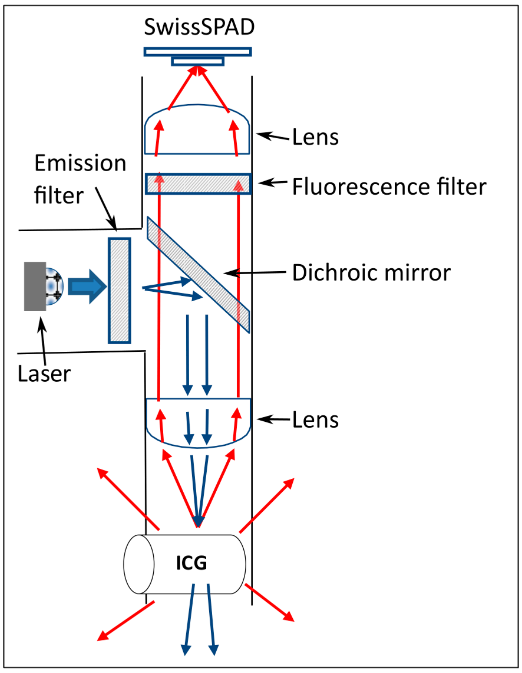


Figure 5.1: Point detection setup for fluorescence lifetime measurement of ICG.

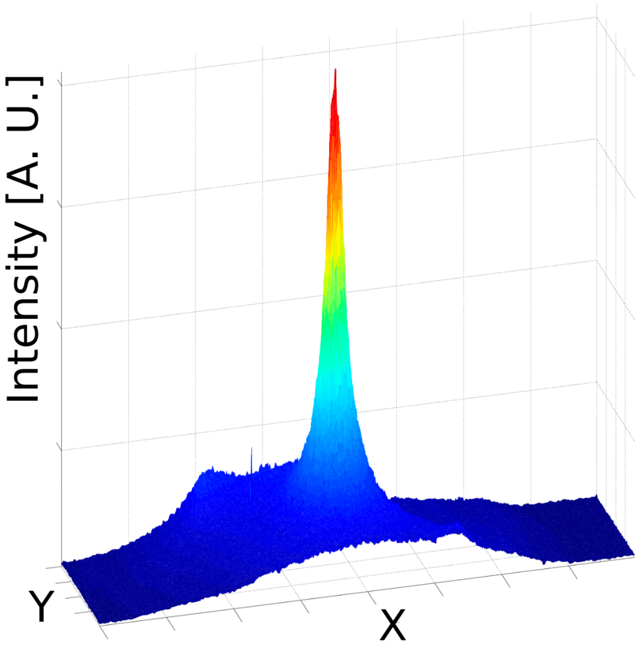


Figure 5.2: Intensity profile for the point detection FLIM setup with a target of 40 µM ICG in milk. The spot size for a diffuse medium like milk is significantly larger than that for a non-diffuse medium like water.

The first step in measuring time-resolved fluorescence signals consists in finding suitable operation parameters for SwissSPAD. We want to have a gate window as close to a rectangular window as possible, which works at the repetition frequency of the laser. In order to get a good response signal, we opted to use a shutter period longer than the laser period of 10 ns for our test measurements. Figure 5.3 shows the pattern finally used with a duration of 40 ns. The 10 ns gate active time at the FPGA leads to an approximately 6 ns photon-sensitive window of the sensor. The long period where the SPADs are turned off helps to ensure a quick restoration of the SPAD bias to obtain a fast turn-on time.

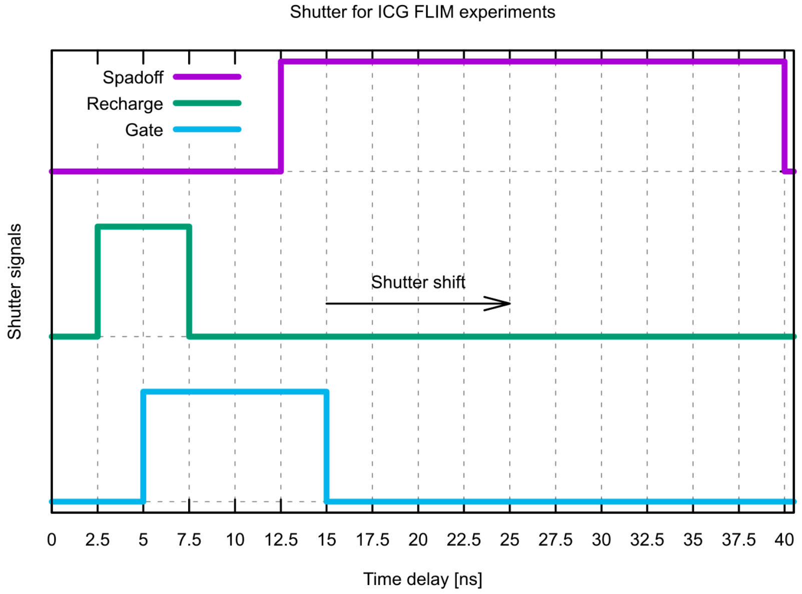


Figure 5.3: Pattern of the shutter signals used for fluorescence lifetime measurements on ICG. The pattern is defined with a period of 40 ns and is used with a laser period of 10 ns.

In Figure 5.4, we show the instrument response function (IRF) that serves as reference for measurements with fluorescence. This measurement was obtained by imaging the laser without filters and sliding the gate window over a period of 10 ns. As in the characterization measurements, we note the disparity between the two edges of the waveform, where one edge changes much faster than the other does. The resolution of the measurement is 512 steps for the window of 10 ns.

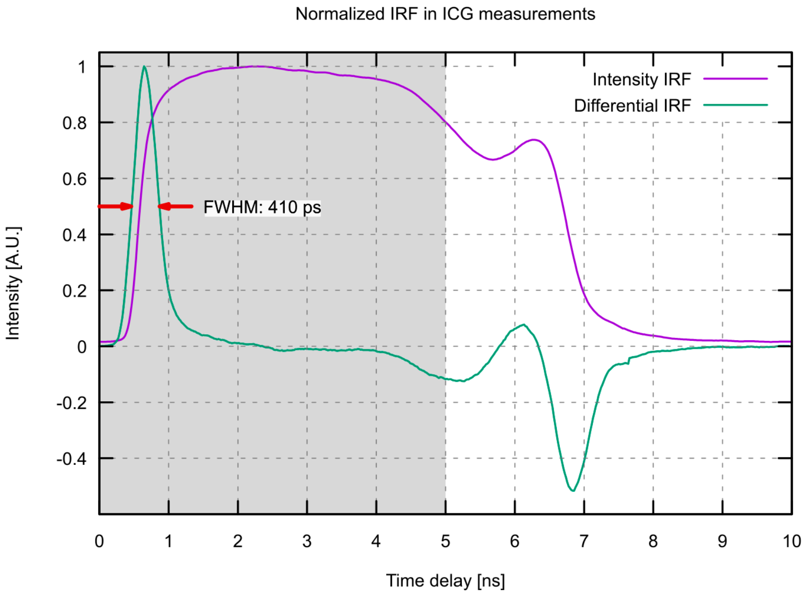


Figure 5.4: Instrument response function of the FLIM measurements for a shutter pattern shifted over a laser period of 10 ns. The sensitive window is approximately 6 ns long and the FWHM of the differential for the faster edge is 410 ps. The first half of the differential signal was used for the indirect method.

A set of response models is built from the IRF, assuming a single exponential decay with a lifetime τ between 10 ps and 2000 ps. The model is effectively the convolution of the IRF with the exponential decay as shown in Figure 5.5 for values of τ between 100 ps and 1100 ps.

Two approaches for lifetime extraction from the intensity data were evaluated. The first approach aims to extract the laser impulse from the IRF and the impulse response from the fluorescence data. To extract the shape of the laser pulse from the measured intensity window we take the derivative based on the assumption that the measured window is the convolution of the sensitive window and the laser impulse. This would give us a positive and a negative pulse from the moment when the laser enters the window and when it exits again. We use only the first positive pulse with a measured FWHM of 410 ps.

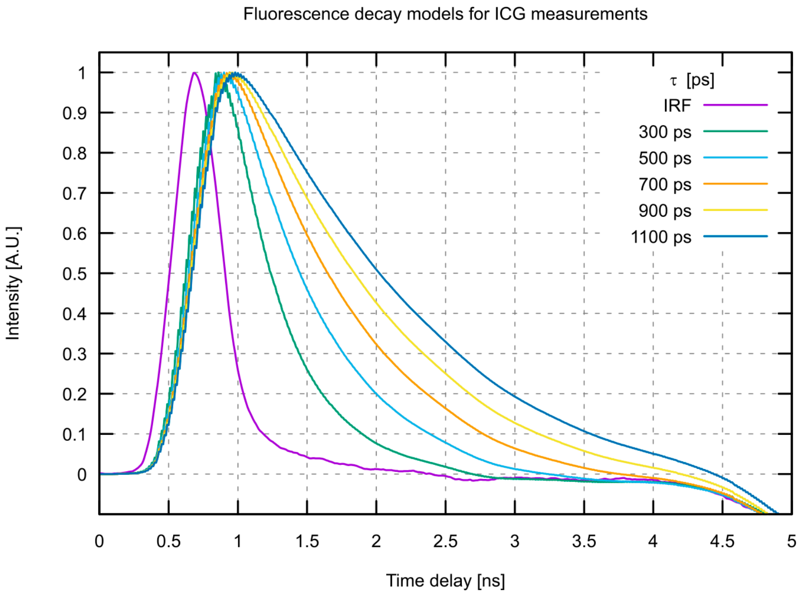


Figure 5.5: Precomputed fluorescence decay models based on the laser excitation extracted using the derivative from the measured intensity window.

For our measurements, we used ICG diluted in milk and water as in [102] to have a reference. The concentration of ICG was between 2 µM and 160 µM. For a diffuse medium like milk, the fluorescence signal was scattered across a larger volume, resulting in a better visible peak on the sensor and a lifetime value could be extracted down to a concentration of 2 µM.

Figure 5.6 shows the results of the measurements of ICG in milk. A median filter with radius 2 has been applied to the intensity and lifetime images to remove defect and hot pixels and generally increase the quality of the result. From the lifetime distributions, we notice that the lifetime is overestimated in most cases, but the absolute deviation is small and we need to consider that we work with very small lifetime values. The error is below 200 ps for most measurements, which is acceptable for many applications where different fluorophores need to be distinguished. However, the error is too important for applications where absolute lifetimes or small relative differences need to be detected. This is for example the case in applications where a fluorophore's lifetime changes in response to its environment.

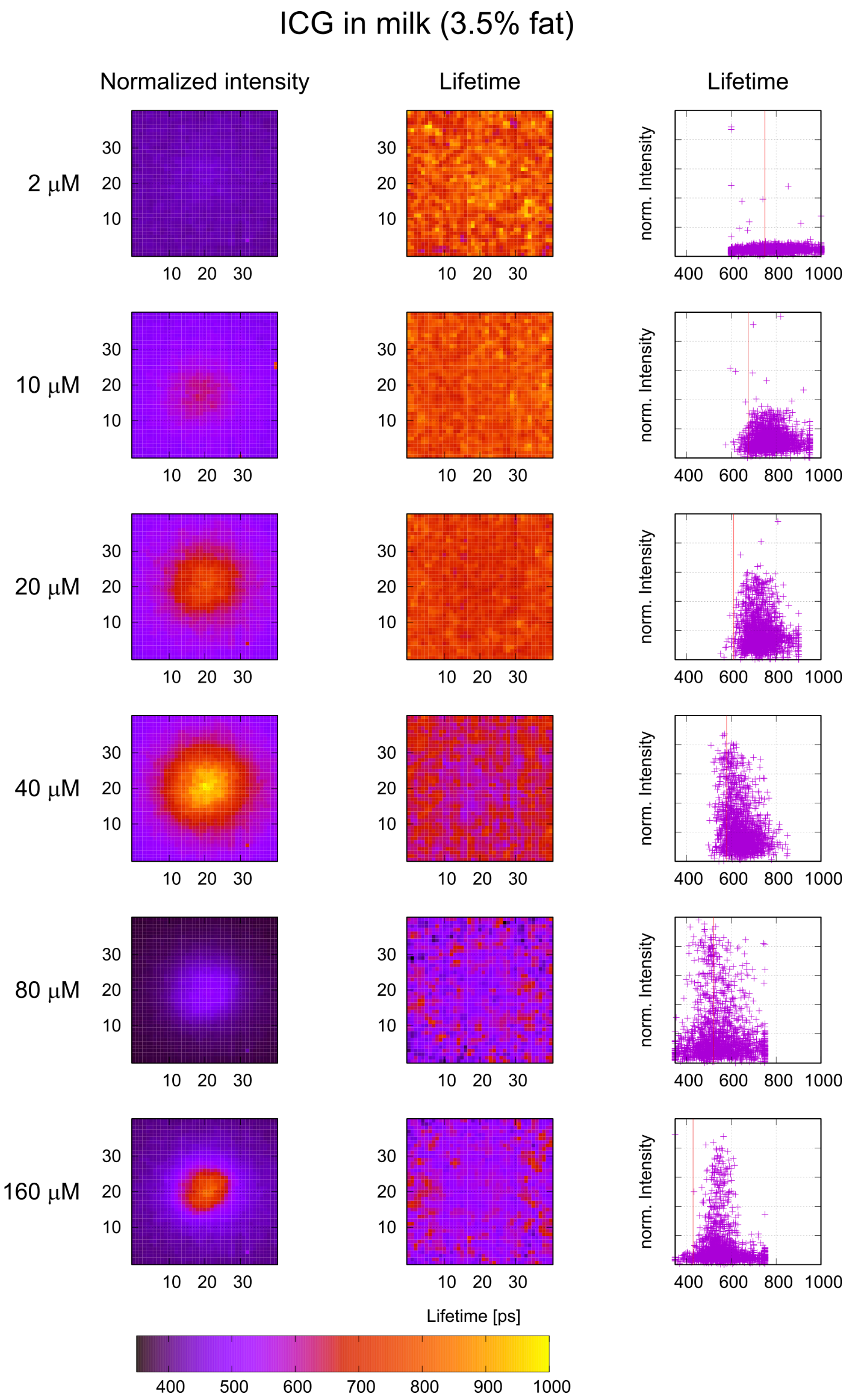


Figure 5.6: Fluorescence lifetime results of ICG in milk. Concentrations between 2 µM and 160 µM were studied and compared against literature reference. The reference lifetime value are shown as red line. The low intensity for 80 µM concentration is due to reduced measurement time.

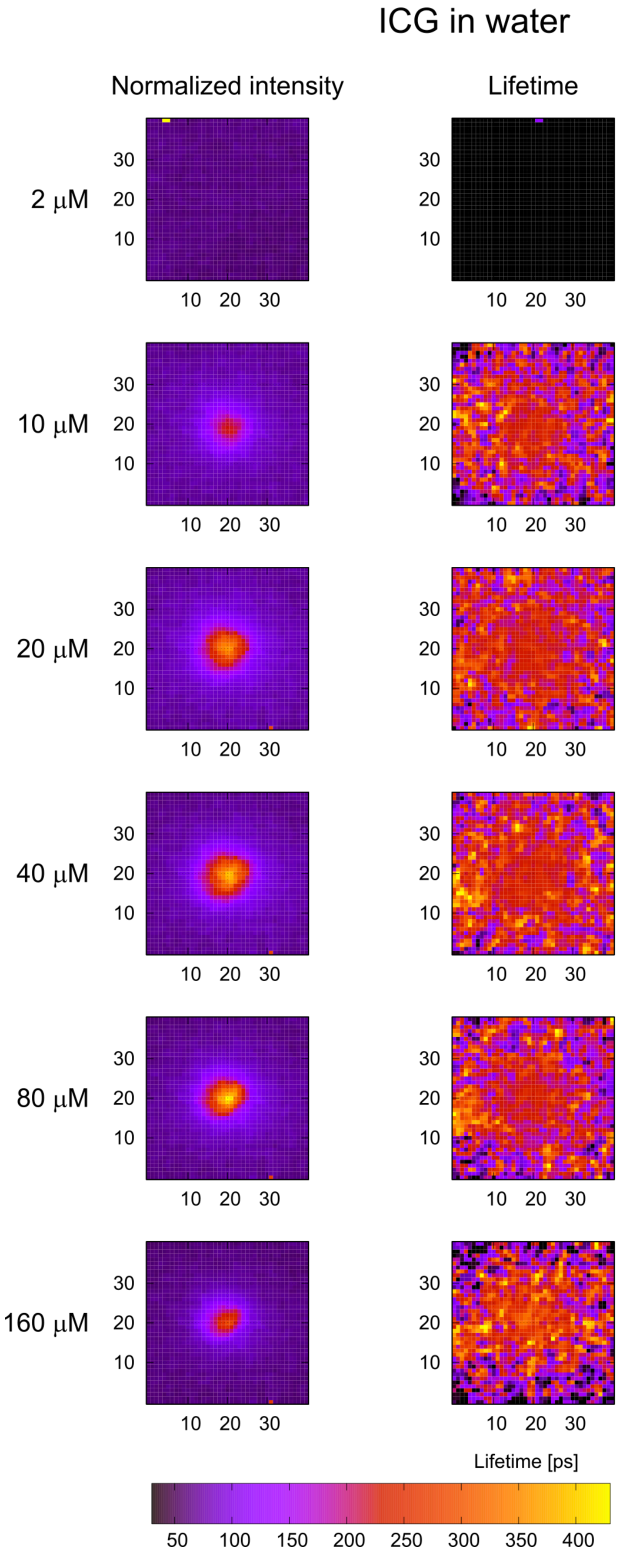


Figure 5.7: Fluorescence lifetime results of ICG in water. Due to the reduced signal intensity the lifetime distribution is spread out more. For 2 μM concentration the signal was too weak to extract lifetime values.

Figure 5.7 shows the measurements of ICG in water with the same treatment as the measurements with milk. The lifetime in water is much smaller at around 160 ps and the lifetime error is comparable to that with milk. The main difference is the lower signal intensity, which leads to a spread in the distribution of the extracted lifetime values.

In the second approach that was evaluated, we used the measured IRF intensity response directly as excitation signal and convolved it with a single-exponential decay to obtain model responses for different lifetimes. In that way, we compute from the periodic excitation signal an equally periodic expected result, with which we compare the measured response. Compared to the first method, we save the derivation of the response curves and we use the full period window for curve fitting. A series of fluorescence response models together with the IRF are shown in Figure 5.8.

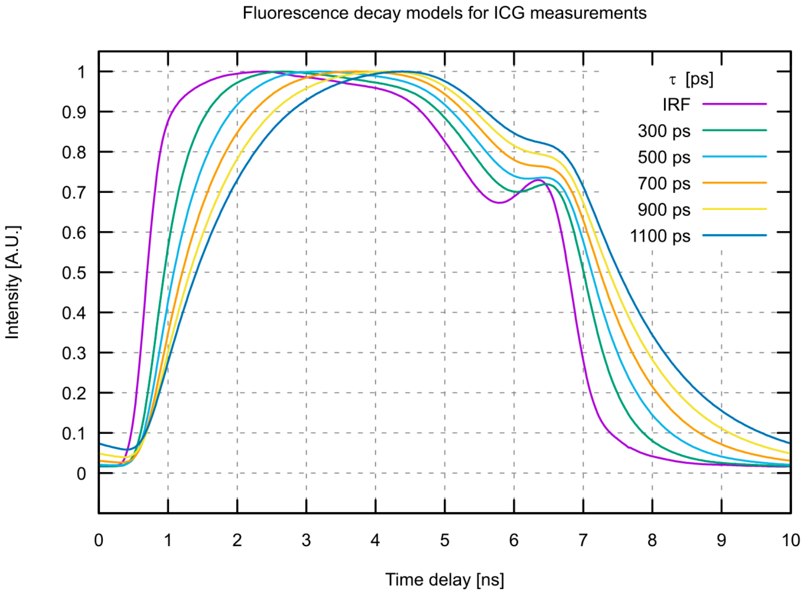


Figure 5.8: Single-exponential fluorescence decay models for full IRF window excitation.

Figure 5.9 and Figure 5.10 show the results of the measurements of ICG diluted in milk and water using the full period method. Using this method, the lifetime is rather under-estimated in most measurements. In [53], H. Homulle, F. Powolny, et al. studied the first method of lifetime extraction in more detail and used simulation to find the measurement error and found that the method will overestimate short lifetimes. Knowledge of this methodical error can be added to the processing in order to correct for this.

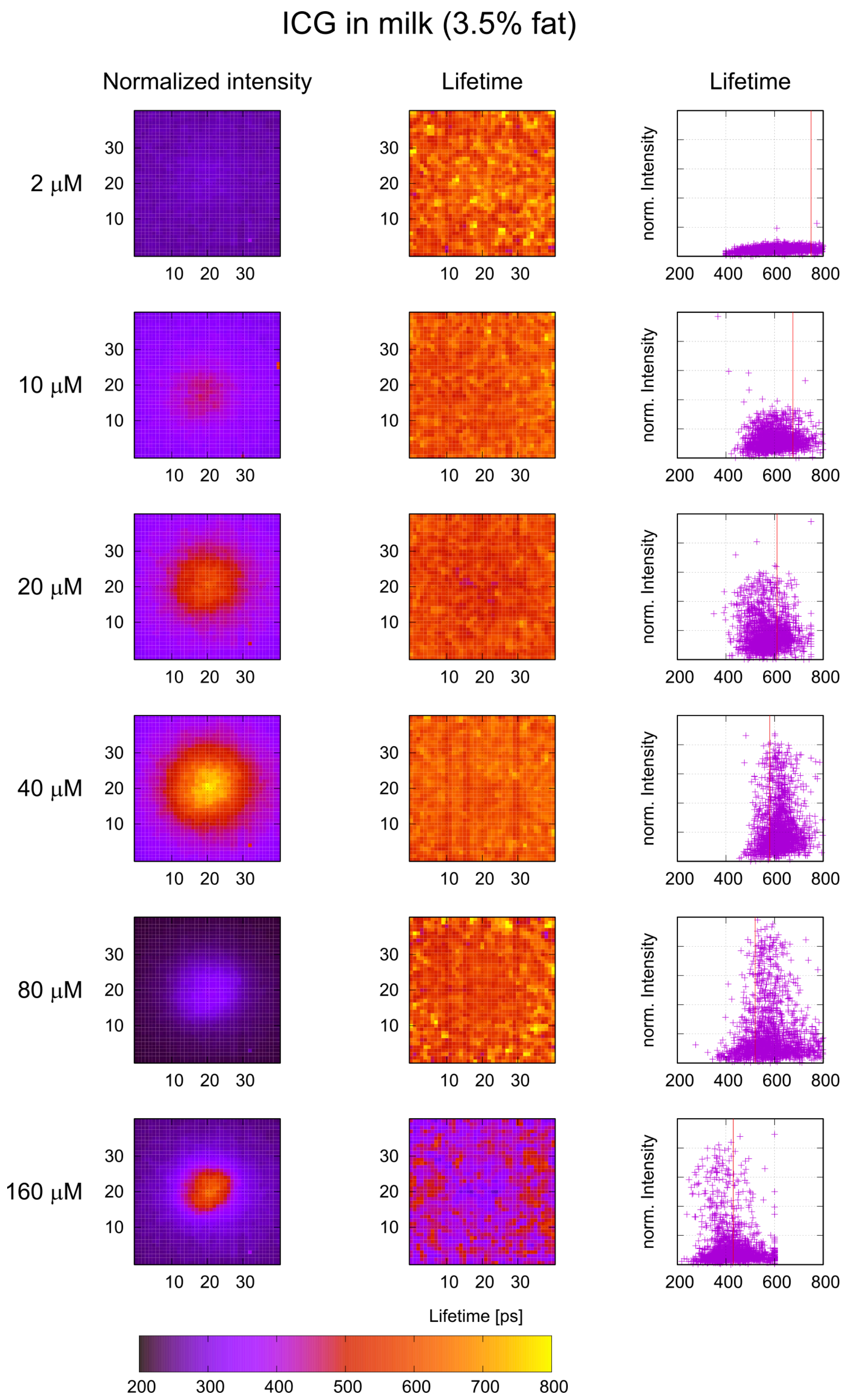


Figure 5.9: Fluorescence lifetime results of ICG in milk using the full period response.

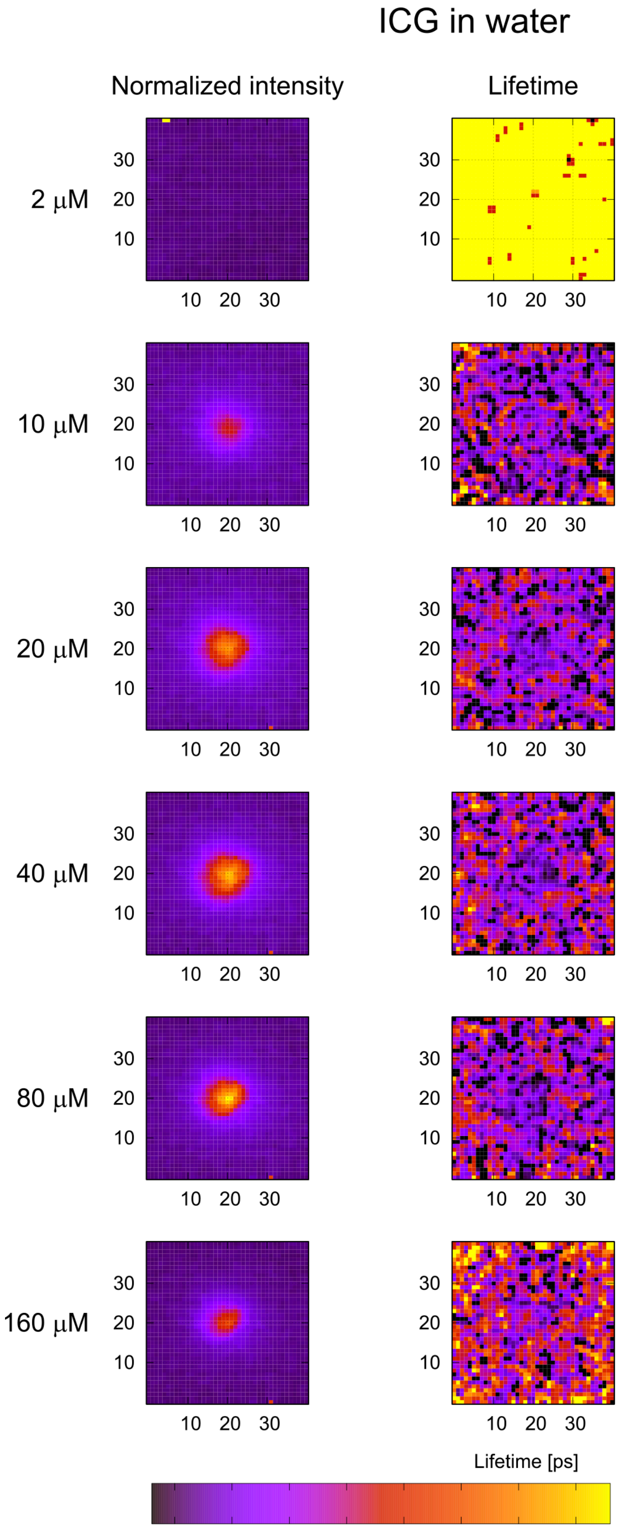


Figure 5.10: Fluorescence lifetime results of ICG in water using the full period response.

FLIM measurements with longer lifetimes

After the characterization of SwissSPAD using ICG diluted in milk and water, we set out to apply the methods to other fluorophores commonly used in biochemistry laboratories. The aim was to characterize the camera further by using conditions closer to current applications of fluorescence lifetime imaging and eventually to obtain wide-field FLIM images of fluorophores bound to other molecules in-vitro or even in living species (in-vivo). The dyes used in these measurements have considerably longer lifetimes than ICG used in our initial setup.

The setup we used for these measurements at the University of California, Los Angeles, differed significantly from the simpler one used for the ICG measurements. The excitation laser, a PicoTrain High Q laser with 532 nm wavelength, 68 MHz repetition rate and 8 ps pulse width, was focused via a dichroic mirror in the back focal plane of a microscope objective lens. The emitted fluorescence of the molecules was redirected through the dichroic mirror and a band pass filter to a camera port where SwissSPAD was attached.

The different frequency of the excitation laser and necessary synchronization with the processing FPGA lead to changed operating frequencies for the shutter subsystem of the FPGA firmware. Instead of running at 200 MHz the system was adapted to run at 3 × 68 MHz resulting in a 204 MHz base clock. To shift the signal pattern over a full excitation period of 14.7 ns we used 3 × 256 steps of 19.15 ps.

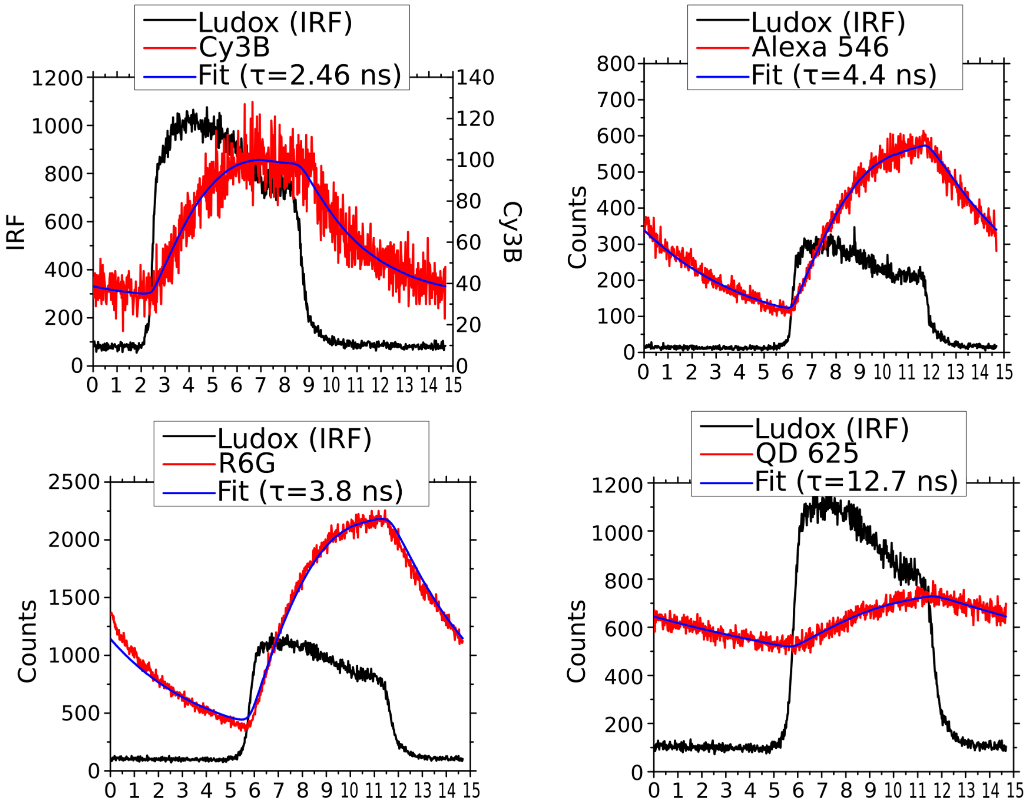


Figure 5.11: FLIM recording of different fluorescent samples using a 68 MHz (period: 14.6 ns), 532 nm wavelength laser with an 8 ps pulse duration. [56]

The signal intensity was recorded for each position of the shutter signals within the excitation period resulting in a signal corresponding to the convolution of the excitation pulse, the fluorescence decay and the sensitivity response of the SwissSPAD sensor. The instrument response function (response without fluorescence) was acquired using a scattering sample. Next, the convolution of the IRF with an exponential decay function was fitted to recorded response curves in order to find the lifetime. As opposed to the lifetime “search” used by the author for ICG measurements, X. Michalet implemented a downhill simplex minimization [103] in LabVIEW to fit a convolution of the IRF and decay function to the measured response and extract the pixel lifetime.

Figure 5.11 shows typical response waveforms from different fluorophore solutions together with the IRF used to calculate the lifetime fit. The results of these measurements are summarized in Table 5.2. Partial fluorophore quenching due to the high concentrations of the solutions [65] can explain the differences between the extracted lifetimes and the reference values.

Fluorophore	Reference* lifetime [ns]	Extracted lifetime [ns]
Rhodamine 6G	4.08	3.8
CY3B	2.8	2.4-2.5
Alexa Fluor 546	4.1	4.2-4.4
QD 625	N/A	12.7

Table 5.2: Reference and extracted FLIM lifetime data for solutions of different fluorophores. * [101]

While measuring the results of Table 5.2, we realized that we had to use very high concentrations of fluorophores and increase laser power in order to measure a useful signal in a measurement duration of roughly one minute. Such a long measurement time and high laser power lead to photo-bleaching in many fluorophores, even more so when power needs to be further increased because of smaller fluorophore concentrations in samples. Even if refined analysis algorithms would have permitted to reduce the acquisition time or excitation intensity, the photon efficiency combined with the shutter non-ideality of SwissSPAD (without microlenses at the time) was not sufficient to go further and measure fluorophores in-vitro, let alone in-vivo.

We concluded the experiments at this point, and decided to wait until we could sufficiently characterize microlenses on SwissSPAD to start another attempt to record widefield FLIM images using SwissSPAD.

5.1.2 Quantum random number generation

While the detection of single photons in SPADs is a classical event, their measurement is recognized as quantum process [104,105] and consequently is of true random nature and can be used as entropy source in a true random number generator (TRNG). The quantum nature of photons as entropy source has already been used in past works to build TRNGs [59,60], but the integration level did not permit a high power efficiency or widespread applicability. There was also no evidence of the scalability of such TRNGs using an array of photon detectors working in parallel.

Using SwissSPAD, we recorded large sequences of single-bit images and showed that they exhibit excellent random properties, thus demonstrating the scalability of the single-photon approach and its energy efficiency. We built the TRNG using SwissSPAD by placing a FPGA controlled illumination source, a LED, on top of the sensor, in order to control the average photon influx. The firmware was enhanced with a possibility to post process a bit-stream from single-bit measurements produced by the sensor through a von Neumann filter for de-biasing. The setup corresponds to the one shown at the beginning of Chapter 2.

A von Neumann filter treats pairs of bits and discards the bit pairs, whose elements are identical. A single output bit is generated from 1-0 and 0-1 sequences. Applied to an ideal random input, the output rate of the filter is 25% of the input rate, as half of the bit pairs produces no output and the other half results in one bit. We therefore define the random bit efficiency (RBE) of the filter as:

$$\mathrm{RBE} = \frac{\text{Bitrate after von Neumann filter}}{\text{Bitrate from random bit generator}} \qquad \mathrm{RBE}_{\max} = 25\% \tag{5.1}$$

The pixels of SwissSPAD are well suited for the generation of random bit sequences with the integrated 1-bit memory cell. To generate a sequence of random bits, the sensor needs to be operated in such a way that the probability of reading a 0 or a 1 from a pixel is as close to 50% as possible. With SwissSPAD, this is achieved by carefully choosing the operating conditions to reach an average probability of 50% for a photon detection on each readout of the chip.

The RBE of SwissSPAD with the LED illuminator will depend on the operating voltage of the SPADs (by influencing the PDE) and the illumination and shutter signals used (defining the photon flux during the acquisition time window). Further dominant influence on single pixels is their DCR and, to a lesser extent, the temperature. To acquire random bit sequences, we used a shutter sequence where the SPADs are only charged immediately preceding the photon sensitive window and are turned off for the remainder of the cycle of illumination and readout.

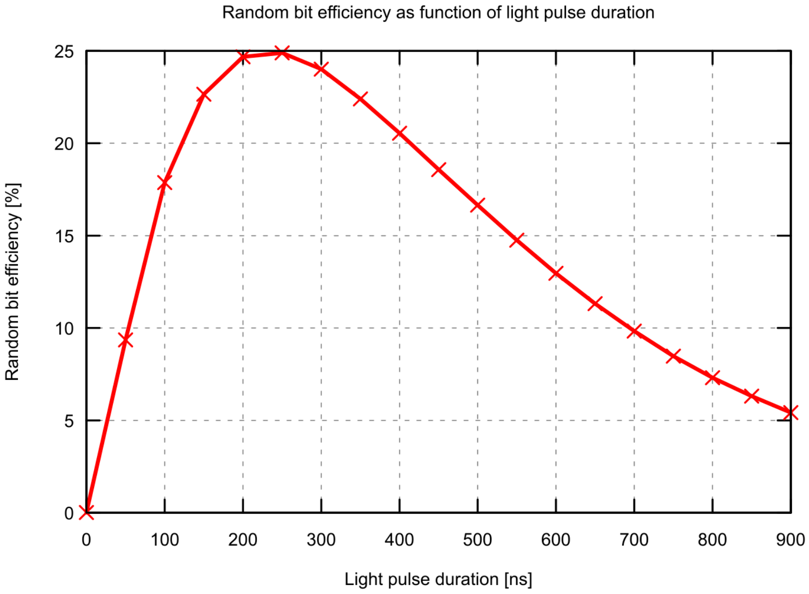


Figure 5.12: SwissSPAD random bit efficiency as function of illumination LED pulse length.

We measured the RBE of the bit stream produced at different conditions of temperature, SPAD bias voltage, and length of the LED pulse. Figure 5.12 and Figure 5.13 show the measured RBE for changing levels of illumination and excess bias voltage with the other parameter fixed. The measurements used a fixed acquisition window. We see that the RBE approaches the maximum of 25% for the optimal combination of excess voltage and pulse duration. High DCR pixels that generate very biased bit-streams were filtered out in these tests.

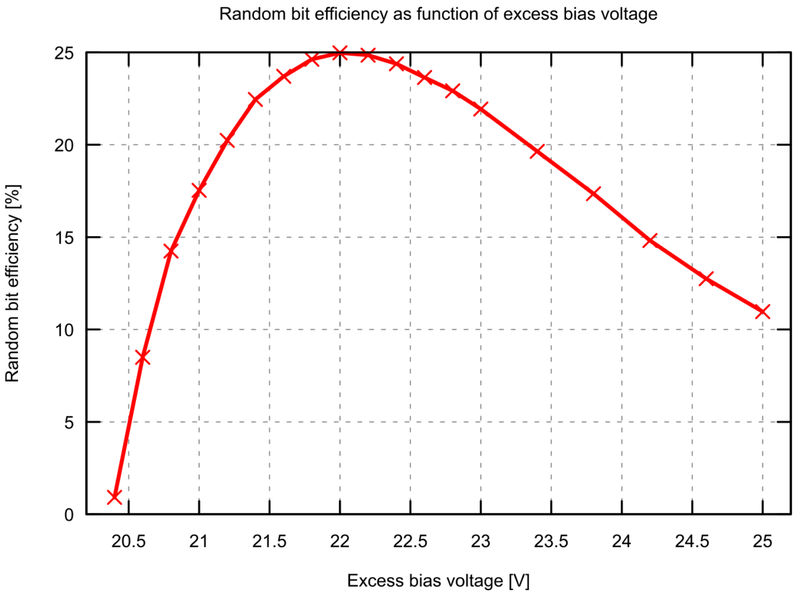


Figure 5.13: SwissSPAD random bit efficiency as function of SPAD excess bias voltage.

We also assessed the number of pixels that are needed to produce the maximum bit-rate possible with SwissSPAD corresponding to its native data rate of 10.24 Gbps. This data rate is limited by the speed of the I/O logic of SwissSPAD. An individual SPAD pixel can be used to generate random number at a frequency of 2 MHz, including a safety margin for afterpulsing. To find the minimal number of pixels needed for a given data rate, the maximum random bit rate for a single pixel needs to be found. This value is mainly limited by the dead time and the fact that we need to minimize afterpulsing, which would introduce correlation into the random bit stream. Figure 5.14 shows the relation between the (desired) QRNG throughput and the number of pixels needed to reach it.

The quality of the stream of random numbers was finally evaluated by applying to the produced streams the battery of tests proposed by the NIST test suite for the validation of random number generators [106]. We applied the test to both raw data as well as de-biased ones, and we repeated the test applying it to several sequences having different length. In particular, we applied the tests to sequences having lengths ranging from 10^4 to 10^6 of bits. As required by the NIST procedure, the test parameters were adjusted accordingly to the sequence length.

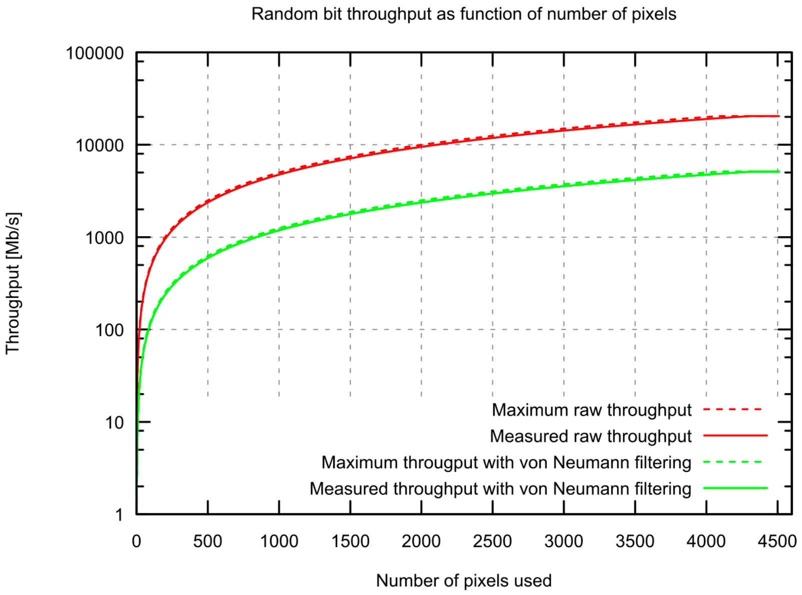


Figure 5.14: Number of pixels needed to reach a certain throughput using SwissSPAD.

Test	Accept Threshold	Von Neumann	Pass / No Pass
Frequency	0.951464	0.9833	Y
BlockFrequency	0.951464	0.9833	Y
CumulativeSum	0.951464	0.9833	Y
Runs	0.951464	1.0000	Y
LongestRun	0.951464	1.0000	Y
Rank	0.951464	1.0000	Y
FFT	0.951464	0.9833	Y
NonOverlappingTemplate	0.951464	0.9667	Y
Universal	0.951464	1.0000	Y
ApproximateEntropy	0.951464	1.0000	Y
RandomExcursion	0.951464	0.9744	Y
RandomExcursionVariant	0.942202	0.9744	Y
Serial	0.951464	1.0000	Y
LinearComplexity	0.951464	1.0000	Y

Table 5.3: Results of the NIST randomness tests [106] applied to a sequence generated with a LED pulse length of 100 ns and an excess bias voltage of 2.8 V. The tests were run on the data from the de-biasing filter.

Our results show that all the de-biased sequences are passing the statistical tests. As example, the test results of one sequence are reported in Table 5.3. In addition, the raw sequences performed well on the NIST tests, as most of them passed before any post-processing. Thus, we showed that a SwissSPAD sensor can reach random bit rates up to 2.5 Gbps, proving the scalability and performance for random number generators based on SPADs. Based on the chip power consumption for the data acquisition, we estimated the power needed to produce one random bit at 25 pJ/bit, the lowest power consumption to date. Table

5.4 compares the SwissSPAD TRNG with other state-of-the-art TRNGs in terms of throughput and power efficiency illustrating the competitive performance of SPAD-based TRNGs.

Measure								Unit
Reference	This work	[107]	[108]	[57]	[58]	[60]	[59]	
Throughput	10	6.4	0.02	0.3	0.04	280	300	Gbit/s
Power	500	N/A	1.9	N/A	29	N/A	N/A	mW
Core area	7.7	N/A	0.012	N/A	0.752	N/A	N/A	mm^2
Energy/bit	25	N/A	950	N/A	725	N/A	N/A	pJ/bit
Technology	0.35µm CMOS	Custom (InGaAs)	SiN MOSFET	CMOS (FPGA)	0.35µm CMOS	Custom	Custom	

Table 5.4: Comparison between the SwissSPAD TRNG and the state-of-the-art.

Based on our findings after evaluating of SwissSPAD as true random number generator, new SPAD-based specialized circuits were implemented focusing on high throughput and high RBE while consuming minimal power. Evaluation of these circuits is still ongoing.

5.1.3 Fluorescence correlation spectroscopy (FCS)

This section briefly introduces and mentions additional experimental work successfully carried out with SwissSPAD, evaluating SPAD-based sensors in novel applications. The images in this section were published by J. Krieger [109,110] who executed the measurements [111] on the SPIM-FCS together with J. Buchholz [92]. The initial setup and part of the characterization and following discussions regarding optimizations were a collaborative effort.

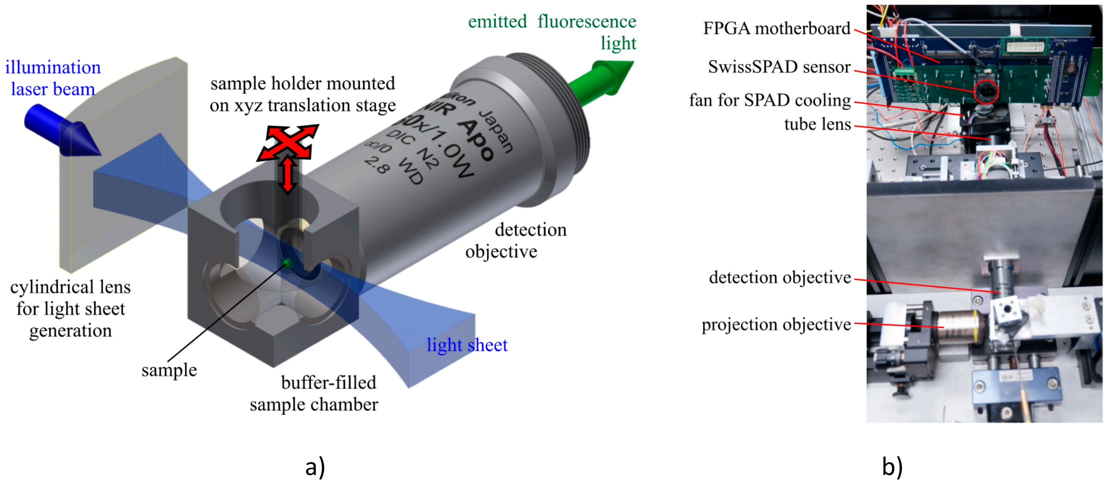


Figure 5.15: a) Principle of SPIM FCS and b) Lab setup for SPIM FCS. A laser light sheet is generated to illuminate selectively a single z-plane of a sample in the focus of a microscope objective to enhance the z-resolution compared to other microscopy approaches. SwissSPAD was placed into the SPIM FCS setup at DKFZ Heidelberg, Germany and used successfully to measure concentrations and diffusion coefficients of different samples. [109]

Fluorescence correlation spectroscopy (FCS) studies rapid intensity fluctuations of fluorescence signals detected from the (single-pixel) observation volumes of a microscope. Two observables are measured: The average signal intensity relates to the number of molecules present in the observation volume and the (auto-) correlation of the signal relates to the diffusion time of molecules through the volume. To measure the diffusion coefficients of small molecules, a high timing resolution is needed to resolve intensity fluctuations because of the high speed of small molecules.

Fluorescence correlation spectroscopy using a confocal microscopy approach has been developed since the 70s [112,113,114]. However, in order to analyze molecular processes across a larger space, in more detail a wide-field measurement technique needs to be used. This was achieved by the combination of selective plane illumination microscopy (SPIM) and FCS [115]. Using SwissSPAD as detector allowed the analysis of smaller molecules in-vitro and in-vivo thanks to its high frame rate and good sensitivity.

Figure 5.15 a) shows the principle of lightsheet microscopy. A laser beam is formed into a lightsheet by focusing it in one direction only using a cylindrical lens. The thin sheet of light traverses the sample, which is immersed in a liquid-filled sample chamber and excites fluorescence only in a very thin (1-3 µm) slice of the sample. A second objective is used to image the emitted fluorescence from the sample onto an imaging detector (e.g. an EMCCD camera or the SwissSPAD). Figure 5.15 b) shows a photograph of the laboratory setup with SwissSPAD.

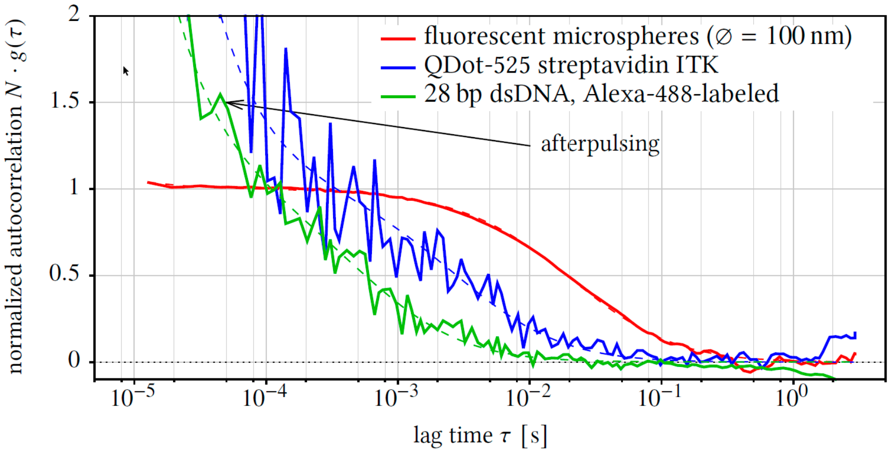


Figure 5.16: FCS autocorrelation functions acquired with SwissSPAD. The dashed lines are one-component fits to the measured data. [109]

Sample	SPIM-FCS: SwissSPAD $D_{20°C,W}$ [µm²/s]	Confocal FCS $D_{20°C,W}$ [µm²/s]	theoretical $D^{(theo)}_{20°C,W}$ [µm²/s]
dsDNA 28 bp	(83 ± 34)	(84 ± 4)	83
QDot-525 streptavidin ITK	(21 ± 12)	(22 ± 3)	20 – 40
QDot-565 ITK	(22 ± 9)	(33.1 ± 2.6)	20 – 40
Green µspheres, ø = 100 nm	(3.3 ± 0.4)	(3.0 ± 0.2)	4.3

Table 5.5: SwissSPAD FCS diffusion coefficients for different samples. The values are given at 20°C with water as solvent. For comparison, measurements of the same samples with FCS on a confocal microscope are given.

A selection of different fluorescent dyes and labeled molecules were observed using the SPIM-FCS and diffusion coefficients were extracted from the autocorrelation functions measured using SwissSPAD at the maximum frame rate of 156 kHz. Figure 5.16 shows these autocorrelations and Table 5.5 summarizes the results of the measurements and compares them to values from a confocal FCS setup.

Recently SwissSPAD has also been used successfully to acquire in vivo widefield FCS measurements. Figure 5.17 shows images for fluorescence intensity, diffusion coefficients and dye concentration for HeLa cells expressing three different oligomers of eGFP. The normalized auto-correlation is shown in the last row of Figure 5.17 and histograms of the intensity values are shown in Figure 5.18. As can be seen in the first and last row of Figure 5.17, the differences in concentration that can be expected from the intensity variation over a single cell were well recovered by the measurements. In addition, the diffusion coefficient differences between the three samples (2nd row) were recovered properly. Still, the absolute values of D did not conform to reference measurements with a confocal setup in the same type of cells. This is partly due to the lower ACF quality in SPIM-FCS, as compared to confocal FCS, which is caused by the lower detection efficiency of SwissSPAD, as compared to the single SPADs used in confocal microscopy, and the overall lower fluorescence intensity, expected in lightsheet microscopy. A significant improvement could be gained with higher detection efficiencies for the SPAD arrays and larger SPADs in them. Still, these results are the first successful applications of SPAD arrays as detectors in imaging FCS on a true widefield microscopic setup and therefore demonstrate the wide applicability of SwissSPAD.

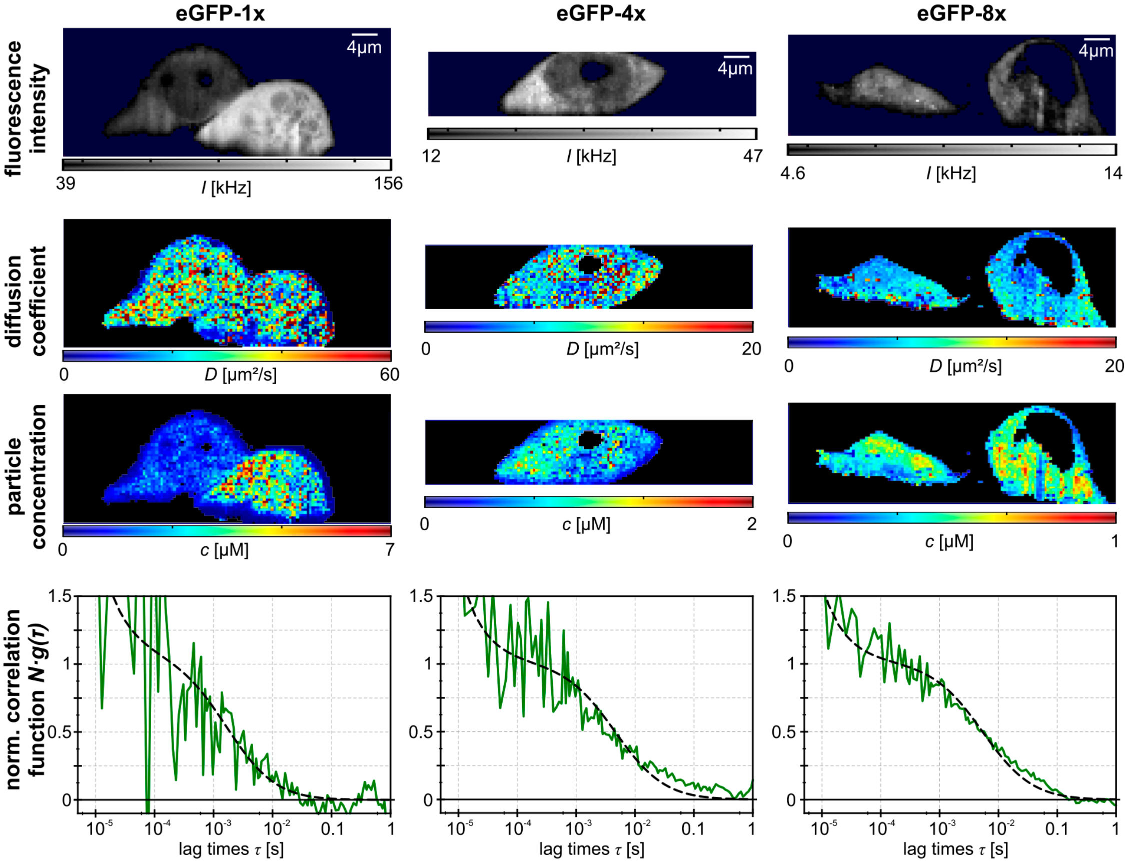


Figure 5.17: FCS images of eGFP oligomers in HeLa cells imaged by SwissSPAD. Shown are the intensity, diffusion coefficient and dye concentration for eGFP monomers, tetramers and octamers. The width of the images is 51.2 µm; the height depends on the region of interest. A global fit algorithm [110] was used to extract the data. [110]

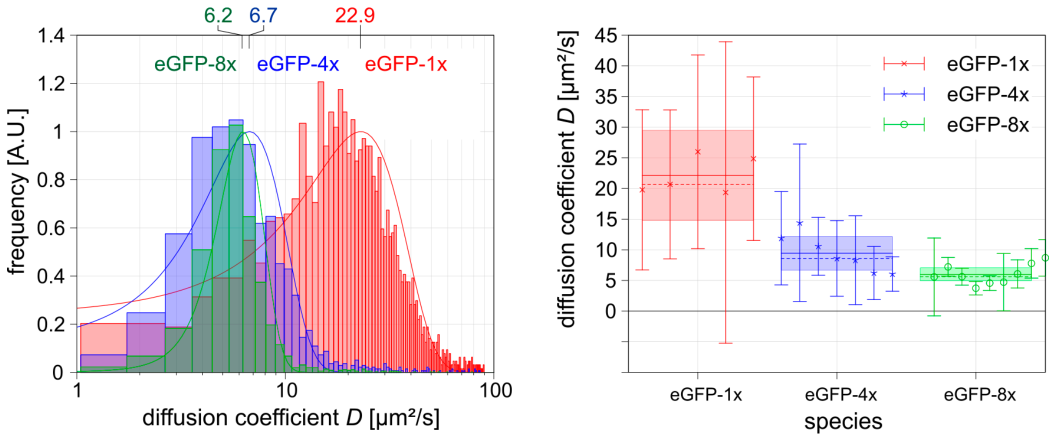


Figure 5.18: FCS diffusion coefficients for eGFP expressed in HeLA cells. Shown are histograms and error estimates of the diffusion coefficients in the cells in Figure 5.17 expressing three different oligomers of eGFP. [110]

5.1.4 Super resolution microscopy

Super resolution microscopy is another application where SwissSPAD has recently been introduced. Super resolution microscopy techniques enable imaging with a higher resolution than the diffraction limit. The 2014 Nobel Prize in Chemistry was awarded to E. Betzig, W. E. Moerner and S. Hell for the development of super-resolved fluorescence microscopy [116,117,118,119,120,121,122].

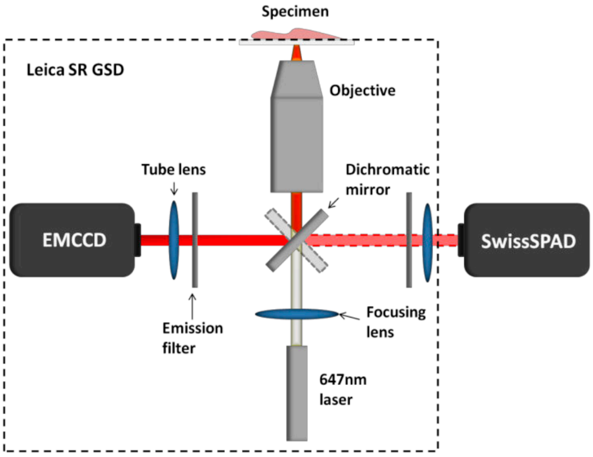


Figure 5.19: SwissSPAD microscopy setup for GSDIM super resolution. The image shows the configuration used in the recording of super resolution images with SwissSPAD. An EMCCD connected in parallel was used as reference and for comparisons.

A SwissSPAD camera with microlenses has been successfully used to record images using ground state depletion (GSD) super-resolution microscopy, the first localization super resolution images recorded with a SPAD based camera. The setup is shown in Figure 5.19. In addition, the fast frame rate permitted detailed analysis of the fluorophore blinking characteristics that can be used in the future to optimize the fluorophores or the acquisition regime to achieve higher frame-rates or lower acquisition times with the same image quality.

Figure 5.20 and Figure 5.21 show the first super resolution images made using a SPAD based imagers [123]. To calculate the images, the formula for PRNU correction [77] was applied to the raw data before feeding it to super resolution algorithms [124]. The localization uncertainty was 30 nm using about 200 photons per localization, compared to the EMCCD, which reached an uncertainty of about 15 nm with 1800 photons. Using FRC [125] the resolution of obtained images was estimated to be close to 100 nm in the samples shown. Even though ten times more photons were collected with the EMCCD, the localization results are not better by a factor $\sqrt{10}$ because of excess readout noise, a factor not present in SwissSPAD.

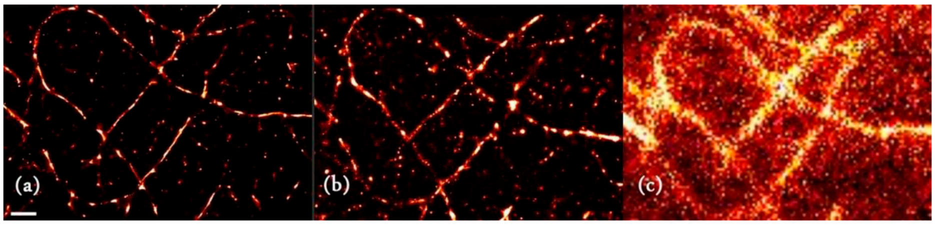


Figure 5.20: (a) EMCCD and (b) SwissSPAD super resolution images of microtubule in MEA buffer.(c) shows a widefield image. Using FRC, the resolution is estimated at 112.7 nm for SwissSPAD and 81.1 nm for the EMCCD respectively. The bar measures 1 µm. The U2OS cells were stained with Alexa Fluor 647. (Courtesy: I. M. Antolovic, submitted paper)

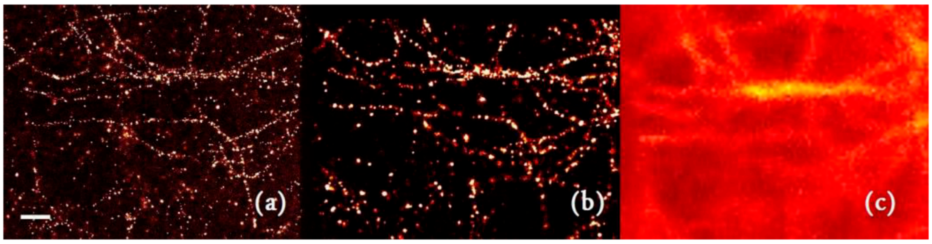


Figure 5.21: (a) EMCCD and (b) SwissSPAD super resolution images of microtubule in Vectashield buffer. (c) is a widefield image of the same region. The resolution was estimated at 98.4 nm for SwissSPAD and 64.4 nm for the EMCCD. The reference bar measures 1 µm. (Courtesy: I. M. Antolovic, submitted paper)

5.2 LinoSPAD

This section presents the first application results from the LinoSPAD camera beyond the characterization presented in Chapter 3 and 0. For this, a 3D time-of-flight system was built by combining the camera with a pulsed light source.

The pulsed light source used in 3D range measurements was built with the same driving circuit used for basic characterization, but with a stronger laser diode. The diode used was a Hitachi opnext HL6545MG rated for 120 mW CW optical output power at a wavelength of 660 nm.

5.2.1 Quantitative range measurements

The first application used LinoSPAD as a laser range finder using the setup shown schematically in Figure 5.22. Two sets of measurements were acquired, one covering round-trip distances between 150 mm and 1200 mm and the other covering distances from 3 m to 17 m.

Short range measurements

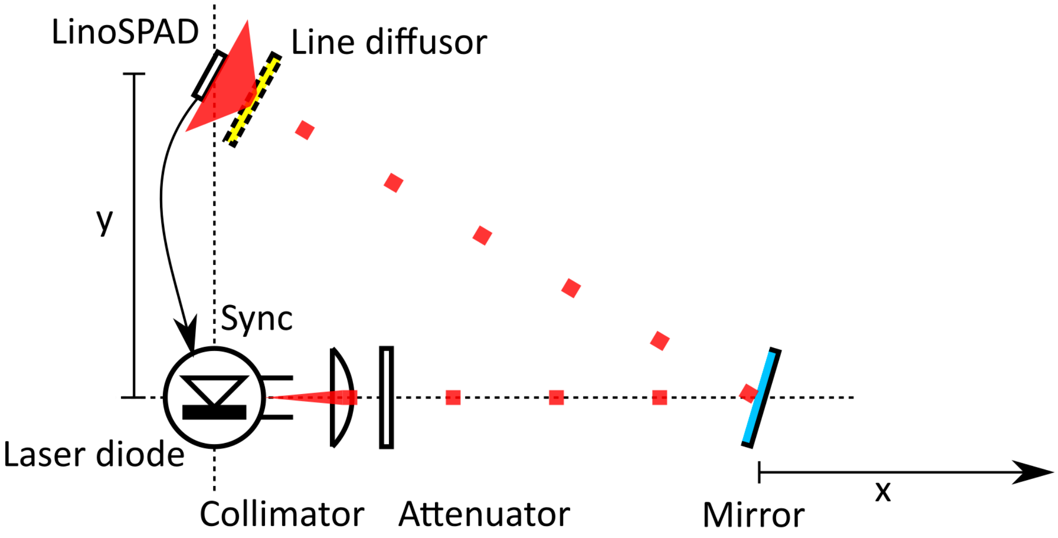


Figure 5.22: 3D measurement setup for laser range finding application. The light pulses are collimated from the laser diode and only diffused a short distance in front of the sensor to have good signal to noise ratio.

For the short-distance measurements, the LinoSPAD camera drives the laser pulse generator at 66.7 MHz or a period of 15 ns. The light pulse from the laser diode is collimated to a beam of approximately 3 mm diameter and attenuated by a neutral density filter (OD 1). From there, it travels in a straight line to a mirror from where it is reflected in the direction of the sensor. In front of the sensor is a Thorlabs engineered diffusor that shapes the light beam into a line. This arrangement ensures sufficient signal-to-noise ratio on all pixels over the whole measurement range.

The system is calibrated initially to produce histograms with 25 ps bin size. With the mirror at the maximum distance of 575 mm from the laser, the histogram peaks are aligned to code 300, the middle of the full range of 600, which covers the 15 ns laser period. One-hundred histograms over 30 ms are recorded before moving the mirror to the next position 25 mm closer to the laser.

A typical histogram shown in Figure 5.23 contains 100k events and has a FWHM between 300 ps and 400 ps. This FWHM is considerably larger than that reported in section 3.5.8. This can be explained on one hand by the lower laser intensity in these measurements, such that the laser pulse width becomes more important due to multi-photon distortions as discussed in [37]. On the other hand, the laser diode is different in the distance measurements. The higher power diode works at a higher current such that the relative changes in current injected from the pulse generation circuit are smaller compared to the low power diode. The smaller relative current difference together with the larger diode can lead to a wider optical pulse.

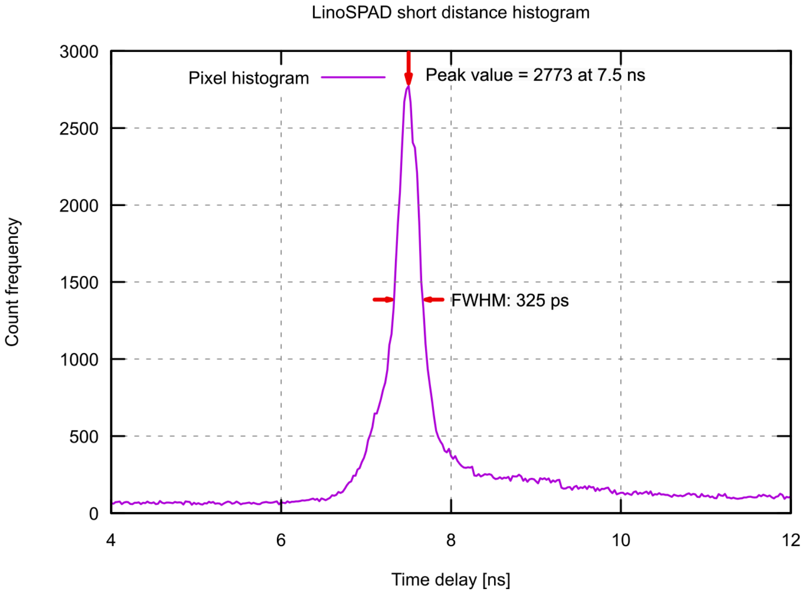


Figure 5.23: LinoSPAD time-of-flight histogram The histogram was recorded over an acquisition period of 30 ms referenced to a 15 ns window. Only the region of the peak is shown.

The interesting information in the distance measurements lies of course in the position of the histogram peak. Additionally the FWHM can be used as a measure of precision of the measurements. Different methods can be used to extract the peak position and FWHM of a histogram like the one of Figure 5.23. We discuss two methods for parameter extraction, which are based on the assumption of an ideal pulse with only one span of values that exceed the half maximum. The first is a simple and efficient method easily amenable to real-time implementation in FPGA, and the second method uses curve fitting. For both methods, we assume a well-defined histogram with a single peak extending over the half maximum.

The first method, which we refer to as the 'quick' method, starts by finding the histogram bin with the highest intensity. We denote our histogram $H = \{h_i\}$ with $i \in [0, N)$ for N denoting the number of bins so that $h_{\max} = \max_i(h_i)$ is the maximum intensity at position $t_{\max} = i|h_i = h_{\max}$. Then the histogram is split into two sets of bins:

$$\begin{aligned} H_{sig} &= \{h_i | h_i \geq 0.5 h_{\max}\} \\ H_{\text{noise}} &= \{h_i | h_i < 0.5 h_{\max}\} \end{aligned} \tag{5.2}$$

From the signal set, we calculate a sample mean to find the peak position, measured in LSB, as follows:

$$t_{\text{peak}} = \frac{\sum_{H_{\text{sig}}} i \times h_i}{\sum_{H_{\text{sig}}} h_i} \tag{5.3}$$

This peak is well defined only if H_{sig} is not split over the histogram borders. If that were the case, the histogram would have to be rotated before peak calculation.

Furthermore, we use $\text{FWHM} = |H_{\text{sig}}|$ LSB and approximate:

$$\text{SNR} = \frac{h_m}{\sqrt{h_m + \frac{\sum_{H_{\text{noise}}} h_i}{|H|}}} \tag{5.4}$$

SNR and FWHM can be used to determine if the signal is sufficient for a reliable extraction of the peak position.

To evaluate our measurements of peak positions, we calculate reference peak positions from the distance steps of our setup in Figure 5.22. The round trip distance is given by:

$$d_s = x_0 + x_s + \sqrt{(x_0 + x_s)^2 + y^2} \tag{5.5}$$

The reference temporal position for the peak is then

$$t_s = \frac{d_s}{c} \tag{5.6}$$

with c the speed of light. The reference positions need to be adjusted for the measurement calibration such that the furthest distance d_{cal} has position t_{cal} = 7.5 ns or 300 LSB.

Measurement results of the short distance measurements between 150 mm and 1200 mm path length are shown in Figure 5.24 where the mean, minimum and maximum t_{peak} for each pixel for 11 steps are drawn. Reference lines are drawn at t_s for each step. This corresponds to the result as it could come from the FPGA in real-time. One limitation appears readily. There are individual systematic errors for each pixel that the histogram processing in the FPGA cannot fully eliminate, because the alignment is limited to full LSB. However, the correction could be added in a further step after parameter extraction as outlined above.

From the error plot in Figure 5.25, we can see that even with the existing pixel-to-pixel systematic errors, the error is limited to a range of approximately 5 LSB with 50% of the pixels in a range of just 1 LSB. The evolution of the mean values exposes another systematic error in the measurement setup, as evidenced by only minute changes from one measurement to the next and few larger jumps when components of the setup were moved inadvertently.

Histograms for the FWHM and SNR over all measurements extracted using the quick method are shown together in Figure 5.26. The FWHM is restricted to integer value by the method and is between 10 and 19 LSB.

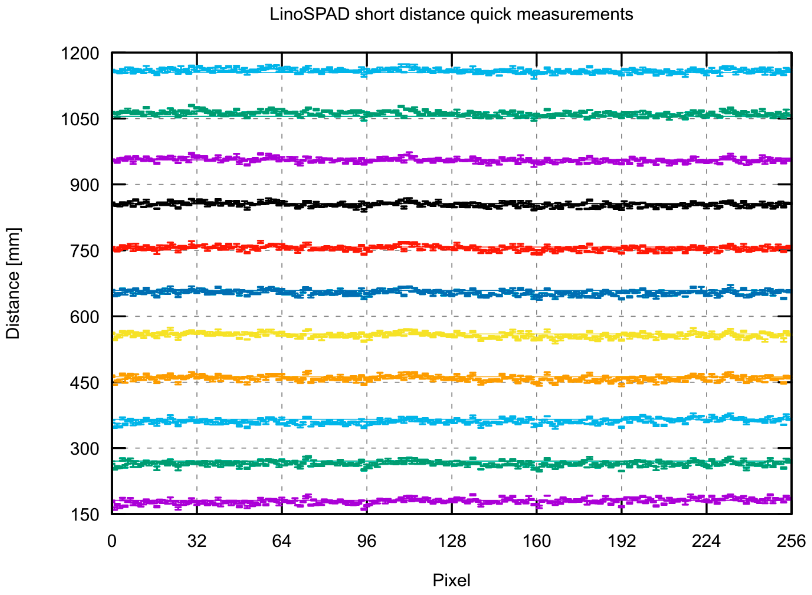


Figure 5.24: LinoSPAD short distance measurements. The mirror was moved in increments of 25 mm from the laser diode increasing the optical path by approximately 50 mm for each step. Only every second step is shown. Plotted values are the minimum, mean and maximum peak position over 100 measurements for each pixel.

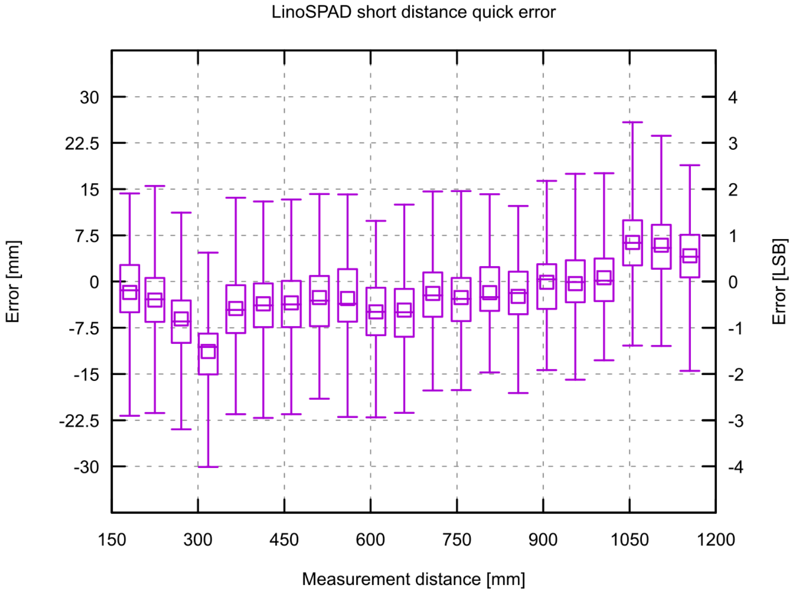


Figure 5.25: Overall, distance error for LinoSPAD short distance measurements. The box shows first to third quartile with the bar denoting the median. Whiskers extend to the minimum and maximum value and the square is the mean value. The evolution of the mean is best explained by absolute errors in the reference value calculation.

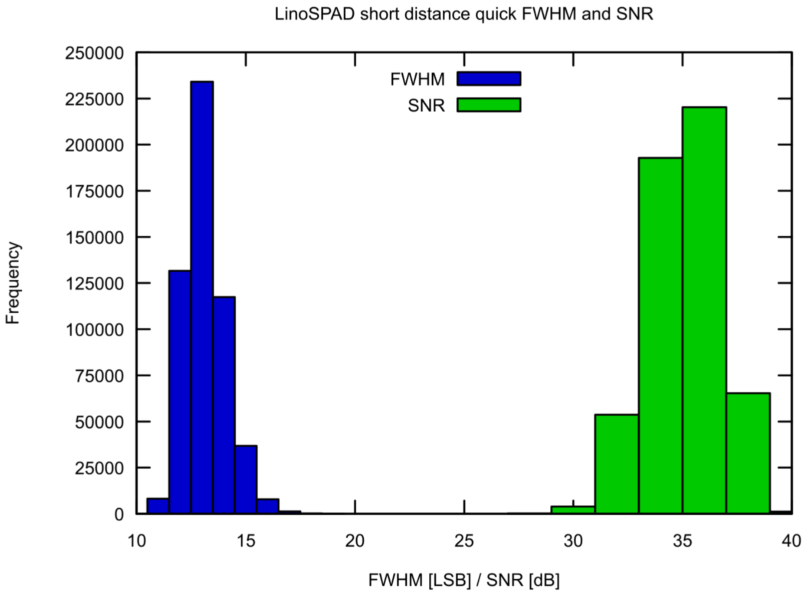


Figure 5.26: Histograms for the FHWM and SNR as estimated by the quick extraction method from short distance measurements. The FWHM is always an integer value using this method.

The second method to extract timing data from histograms can be used when higher precision is needed at the expense of computation speed. This method extracts the chief parameters of the histogram shown in Figure 5.23 using curve fitting of a model function.

The model maps the temporal axis to the intensity using:

$$I(t) = s \times (\mathrm{rise}(t) + \mathrm{puls}(t) + \mathrm{fall}(t)) + n \tag{5.7}$$

with

$$\begin{aligned} \mathrm{rise}(t) &= \left(\frac{t}{c-w}\right)^r \Theta\big((c-w)-t\big) \\ \mathrm{puls(t)} &= \Theta\big(t-(c-w)\big)\Theta\big((c+w)-t\big) \\ \mathrm{fall}(t) &= \exp\left(\frac{(c+w)-t}{f}\right)\Theta\big(t-(c+w)\big) \end{aligned} \tag{5.8}$$

where $\Theta(t)$ is the Heaviside Theta or unit step function used to separate the three parts. In (5.7), s is the signal amplitude and n the noise amplitude. Additional parameters in (5.8) are c for the center, w for the half width, r for the steepness in the rising part and f for the time constant in the falling part.

Figure 5.27 shows a model histogram with the defining parameters. From the model parameters, we define:

$$\begin{aligned} t_{\mathrm{peak,fit}} &= c \\ \mathrm{FWHM_{fit}} &= 2 \times w \end{aligned} \tag{5.9}$$

$$\mathrm{SNR_{fit}} = \frac{s}{\sqrt{s+n}}$$

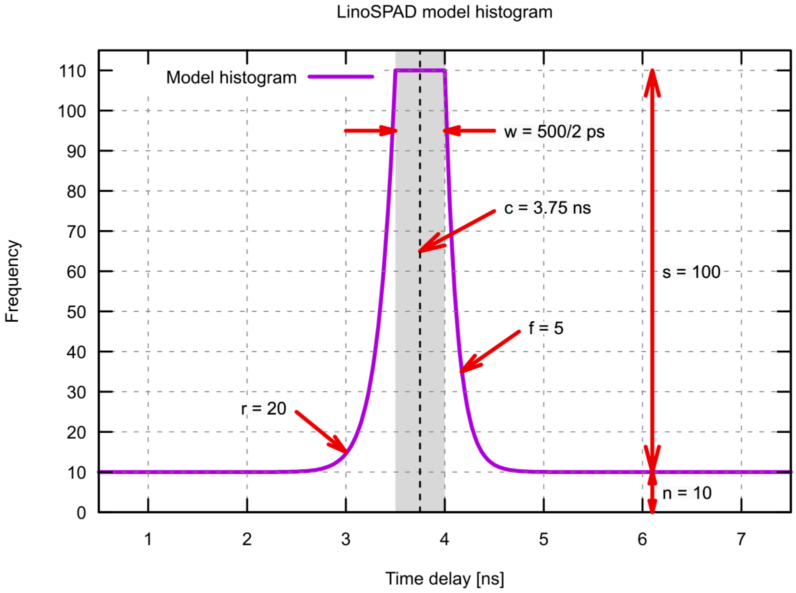


Figure 5.27: Histogram modelling illustration.

To find the model parameters for a histogram, we use the Levenberg-Marquardt algorithm (LMA, [126,127]) also known as damped least-squares (DLS) method. For the fit, we use $a = (s+n)$ and $b = \frac{n}{s+n}$ and transform equation (5.7) into:

$$I(t) = a \times \big((1-b) \times (\mathrm{rise}(t) + \mathrm{puls}(t) + \mathrm{fall}(t)) + b\big) \tag{5.10}$$

Then, the following steps are performed:

- The histogram is shifted to satisfy $t_{\max} = \frac{|H|}{4}$.
- The histogram is interpolated using a piecewise cubic Hermite polynomial for an upsampling by a factor of 5.
- Initial guesses $a = h_{\max}$ and $c = t_{\max}$ are extracted from the histogram.

The fit on the interpolated histogram is then computed using the bounds:

$$\begin{gathered} a \in [0.9 \times h_{\max}, 1.1 \times h_{\max}] \\ b \in [0, 1] \\ c \in [10, |H| - 10] \\ w \in [0.1, 100] \\ r \in [1, 1000] \\ f \in [0.1, 50] \end{gathered} \tag{5.11}$$

From the fitted parameters the distance, FWHM and SNR are calculated as defined in (5.9) with $\mathrm{SNR_{fit}} = \sqrt{a}(1-b)$. In the fitted measurement results presented here, the systematic pixel-to-pixel error is removed by adding an individual offset to every pixel calculated from the variations over all measurements.

Figure 5.28 shows the error values for all pixels over the 21 distance steps measured. The box representing 50% of the measurements is half of a LSB on average and the full span is roughly 2 LSB whereas using the quick extraction it is around 1 LSB and 4 to 5 LSB respectively as shown in Figure 5.25. This proved the efficiency of the curve fitting using our peak model to increase greatly the accuracy when high precision is needed. In cases where highest precision is not needed, the quick method provides robust results available in real time.

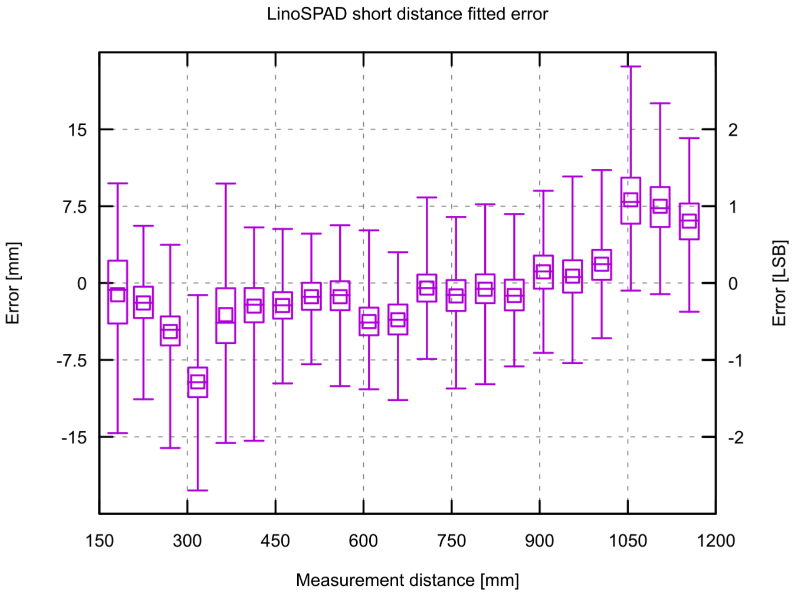


Figure 5.28: Error values for the curve fitting distance extraction for short distances. Compared to the quick extraction the interquartile range and the full range of distance errors is roughly halved. 50% of the pixels report a measurement within a range of ½ LSB or 3.25 mm on the round-trip distance.

The histograms for the FWHM and SNR calculated from the results of the fit according to (5.9) are shown in Figure 5.29. The comparison with Figure 5.26 shows that the SNR calculated from the fit is close to the value approximated by the quick method. The FWHM on the other hand is much better, but the definition used for the fit does not take into account the rising and falling curve component, but only the central peak width. This gives a better estimate of the precision of the measurements as the true peak value for a continuous histogram should fall into this range.

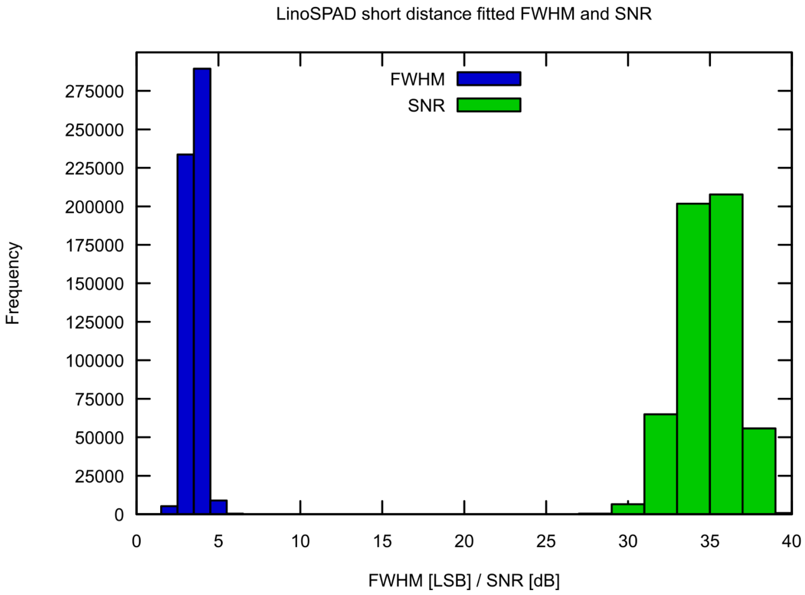


Figure 5.29: Histograms for the FHWM and SNR extracted from the fit parameters. The SNR is comparable to the quick method values even though the quick value is only an approximation. The FHWM however is greatly reduced using the fitted results. However, part of it is due to the method since rising and falling edges are not taken into account by our definition.

Long range measurements

After measuring short distances, and proving the viability of our histogram fitting method to increase the precision, compared to the results of the quick analysis, we move on to measure time-of-arrival histograms over longer distances.

The firmware used in these measurements, which is the same used in the characterization presented in Chapter 3, limits the length of a histogram to 1024 bins. Using the full resolution of the TDC, this corresponds to approximately 18 ns total time-of-flight or a 5.4 m total distance, which can be unambiguously resolved. Therefore, we used the option of 4-to-1 binning in the firmware to extend the histogram measurement range at the expense of histogram resolution.

Using again the setup shown schematically in Figure 5.22, the laser is now driven at 16.7 MHz or ¼ of the frequency used for short-range measurements. This change results in a period of 60 ns and it is recorded in raw histograms of 840 bins, which are corrected in real time and transmitted as 600 bin histograms with an LSB of 100 ps or 30 mm.

The same analysis was performed on the results as was done for the short distance measurements, with the difference in this case, that one LSB measures 100 ps and the round-trip distances cover a range from approximately 3 m to 17 m with steps of 2 m. The histogram was aligned in FPGA firmware for $t_{\mathrm{peak}} = 65\ \mathrm{LSB}$ at a distance of 3 m.

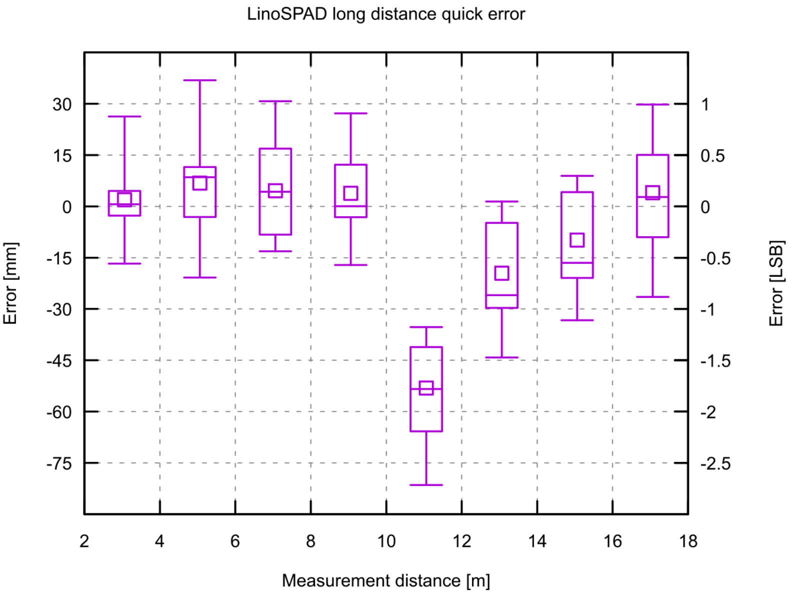


Figure 5.30: Error values from long distance measurement histograms analyzed using the quick method. Additional mirrors were inserted beyond 10 m, which explains the large deviation from the reference values. The range of 50% of the values is about 1 LSB with and the full range about 3 LSB over all distances, considering the four times larger LSB compared to the short distance measurements.

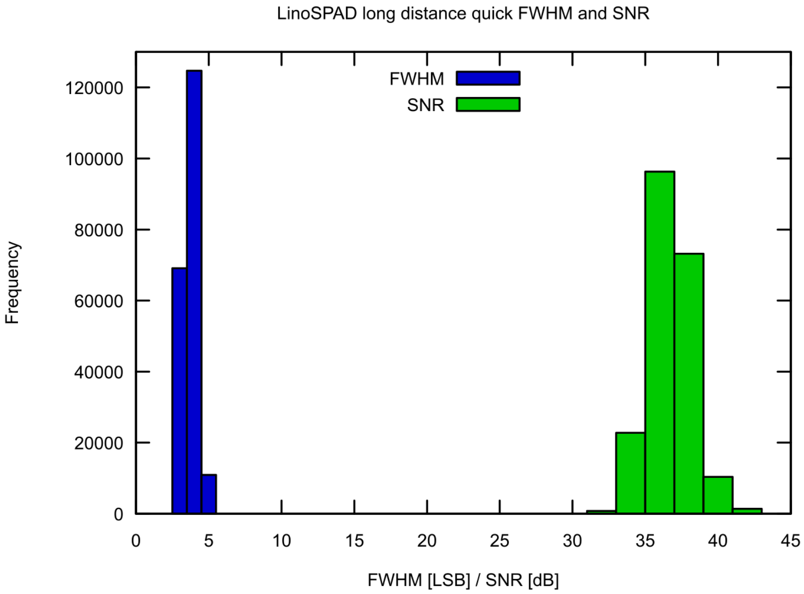


Figure 5.31: FHWM and SNR histograms from the quick extraction method for long distance measurements. Due to the larger LSB, the FWHM is smaller than for short distances and the SNR is larger because the attenuator for the laser was removed for the long distance measurement.

Figure 5.30 shows the range of measured errors when compared to reference distances. The larger setup that no longer fits on an optical breadboard introduced larger systematic errors than in the short-distance measurements, with a clear jump for the measurement at 11 m where a set of additional mirrors was added in the system to extend the range. This systematic error could be easily corrected, but in a real case it would not be present, thus we did not invest the effort to perform the correction in real time. 50% of the values are in a range of 1 LSB and the total range is 1.5 LSB wide. This indicates that the quick analysis already reaches good precision compared to the LSB size, which is 100 ps in these measurements.

Figure 5.31 showing the FWHM and SNR from the quick method shows that the estimated FWHM is comparable to the one from short-distance measurements with the peak around 300-400 ps. The SNR is higher because the laser attenuator was removed for the longer distances.

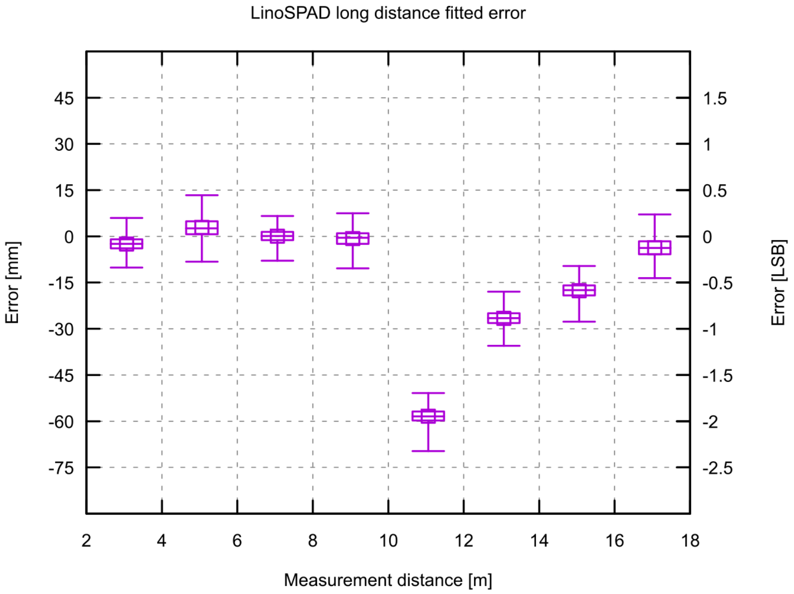


Figure 5.32: Error values for fitted histograms over long distances. The histogram fitting method performs really well for long distance measurement with larger LSBs and brings the interquartile range below ¼ LSB, the LSB of the short distance measurements. The full measurement range of all pixels is now clearly within 1 LSB of the system.

The results from the histogram fitting presented in Figure 5.32 and Figure 5.33 show improvements similar to the ones obtained with shorter distances. The performance of the fit for the distance results as evidenced by the measurement error results is very high. The interquartile of the error drops to ¼ LSB corresponding to 25 ps or 1 bin of the uncompressed histogram and the full range is below ½ LSB. FWHM and SNR show a characteristic comparable to that obtained over short distances.

With these distance measurements and the histogram fitting, we have shown how accurate time-of-arrival measurements can be with a low-cost FPGA and without implementing resource-intensive techniques like wave union [128] or multiple TDCs for a single signal. With the advancement of FPGA integration, the techniques presented here can be further refined and more powerful analysis than the quick method presented in this section can be implemented in the FPGA.

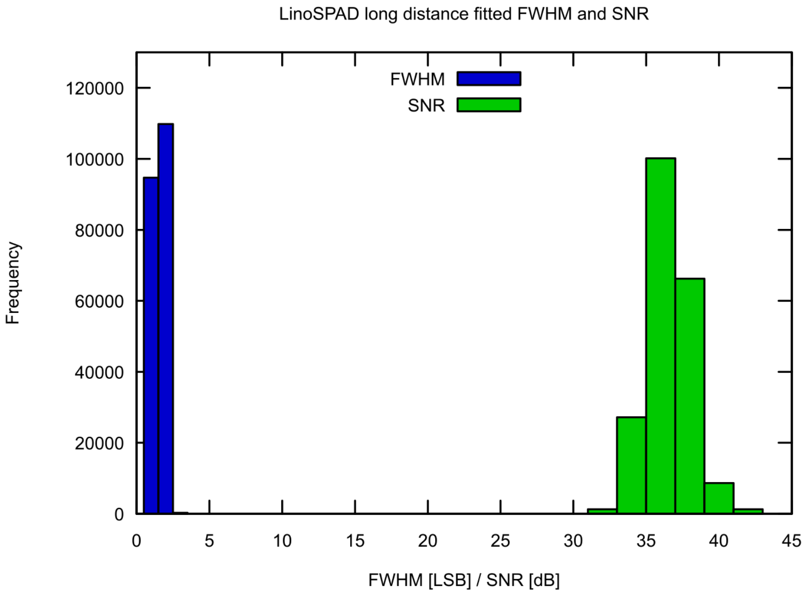


Figure 5.33: FWHM and SNR histograms from fit parameters for long distance measurements. The FWHM approximation for the fitted histograms is below 3 LSB for almost all measurements and the SNR is almost identical to the one obtained with the quick approximation. These values should be taken into account for the decision whether a fit is valid or not.

5.2.2 Scanning 3D depth images

To go from the point or spot distance measurements in the previous section, to true 3D images using the LinoSPAD camera, requires a few changes. First, we have to add the necessary optical element in form of an objective to form an image on the sensor. Then, we need to add a scanning mechanism to image the 2D projection of the field of view on the line sensor. This can be done by moving the camera, or equivalently, by moving the object in front of the camera. Another possibility would be to use a special lens or fiber optic bundle to transform light coming from an area into a line. The third step needed is the illumination, which now needs to cover the full field of view of the camera instead of a spot only.

Figure 5.34 shows the scanning platform constructed to address the aforementioned points and to provide a system to 3D scan small objects. A ceramic cup, 45 mm tall and 55 mm wide at the top, was fixed to the moving platform using a magnet. Through a 25 mm objective (Thorlabs MVL25M23), a slice of the cup is imaged on the sensor. For the illumination, we use a Hitachi opnext HL6445MG 660 nm laser diode, the same as in the range measurements in the previous section, driven at 66 MHz.

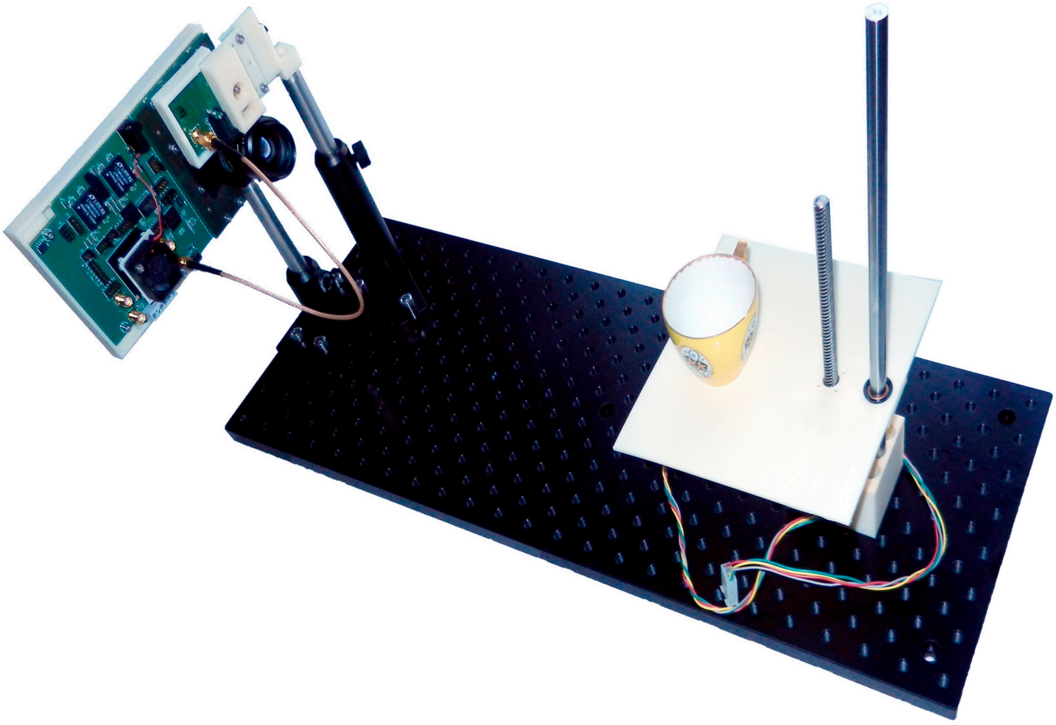


Figure 5.34: Moving stage 3D scanning setup to scan small objects with the LinoSPAD camera. The camera is at a distance of 35 cm from the object platform, which is mounted on a drive screw driven by a stepper motor. The illumination module is mounted over the objective and the laser diode decollimated to illuminate the whole scene.

To capture the shape of the cup, the platform is moved in small steps, until the full cup has passed by, and has been imaged on the sensor. For this scan, we used a histogram acquisition time of 45 ms and captured 10 histograms at each position.

From the total intensity seen by the sensor, the image shown in Figure 5.35 is calculated. The intensity is scaled to the 85th percentile of all point intensities, which leads to the noisiest pixels appearing very bright. A pixel with unreliable electrical connection can be distinguished from its intensity pattern around pixel 216. The gold paint on the handle and the flowers lead to highlights depending on their angle relative to the illumination and the camera. The handle projects a clear shadow due to the displacement of the illumination from the optical axis. Between the drive screw and sliding axis, we placed a sheet of cardboard to provide a background where depth information can be extracted as well.

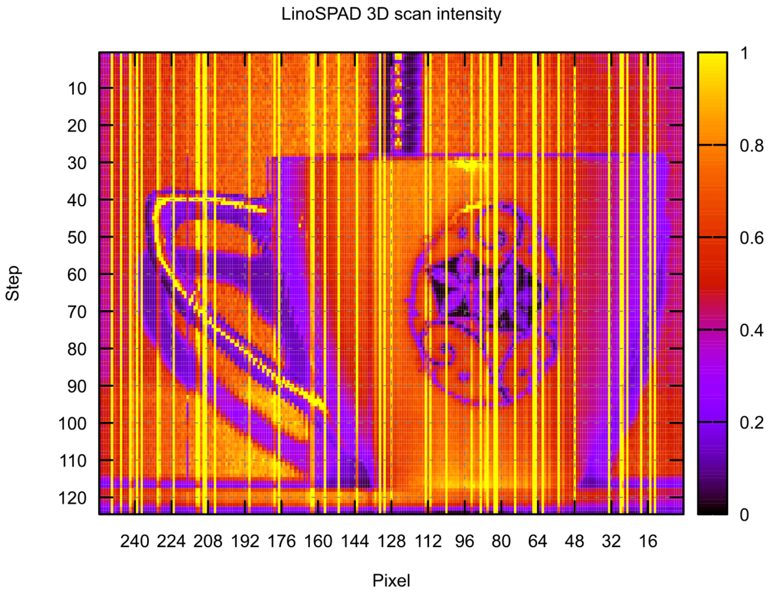


Figure 5.35: View from the sensor on the object platform during a 3D scan. The axes have been inverted to reproduce the view from the camera. This intensity image is calculated from the total counts seen by the sensor by pixel and position. The image has been contrast-enhanced such that hot pixels appear as yellow lines. The cup measures 55 mm in diameter at the top and 28 mm at the bottom. It is 45 mm high. The shadow of the handle is from a sheet of cardboard behind the cup.

Despite the fact that hot pixels are very bright, they produce useful histograms where time-of-flight information can be extracted. To analyze the histograms from the sensor, we used the quick and curve fitting approaches presented in the previous section. Systematic pixel-to-pixel errors have been minimized in both approaches using calibration on a reference line.

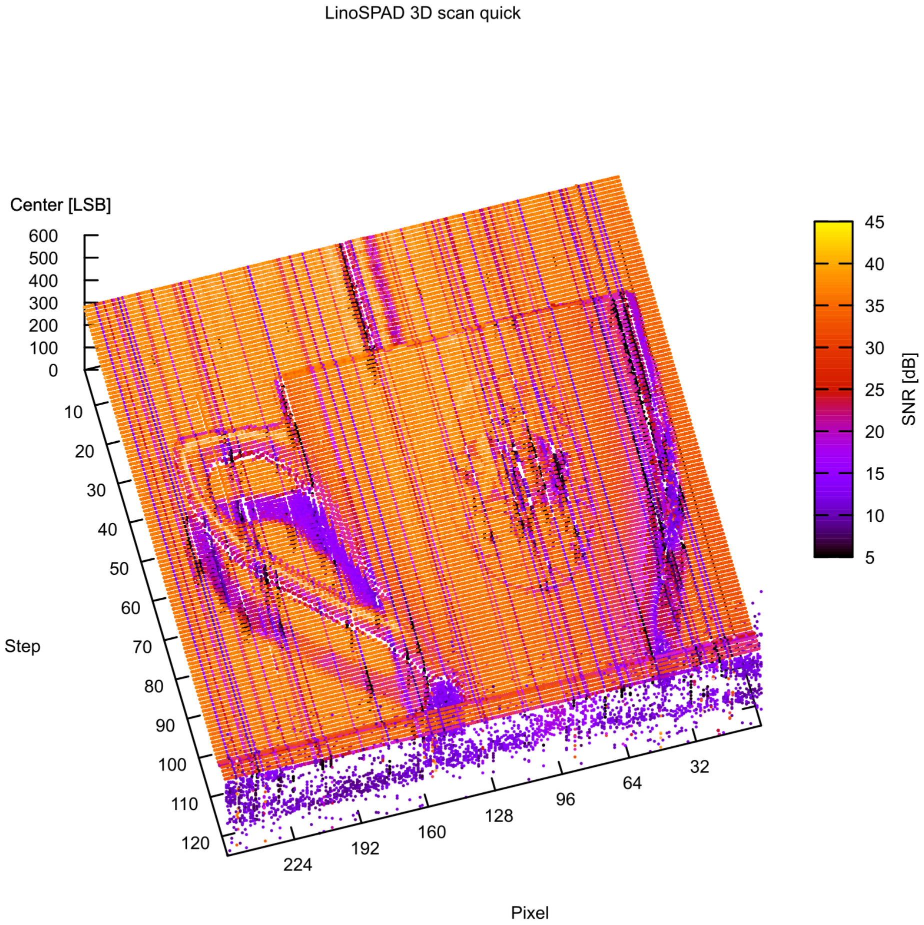


Figure 5.36: 3D image of the cup with color-coded SNR for the full scan. At each point, all 10 measurements are drawn with a color corresponding to their SNR given by the quick estimation method. The noisy pixel produce darker lines due to their lower SNR. Points in the shadows also have lower SNR, but due to reduced reflected illumination. At the bottom, there are few steps with low SNR because nothing reflective is present below the platform.

The results given using the quick analysis are presented in Figure 5.36, with a close up on the face of the cup shown in Figure 5.37. For each pixel and step, 10 points corresponding to the extracted depth are drawn. The noisier pixels appear as low SNR lines, as do the shadows. In the close-up, we can see the curved front of the cup with a radius of 27.5 mm or 7.33 LSB only.

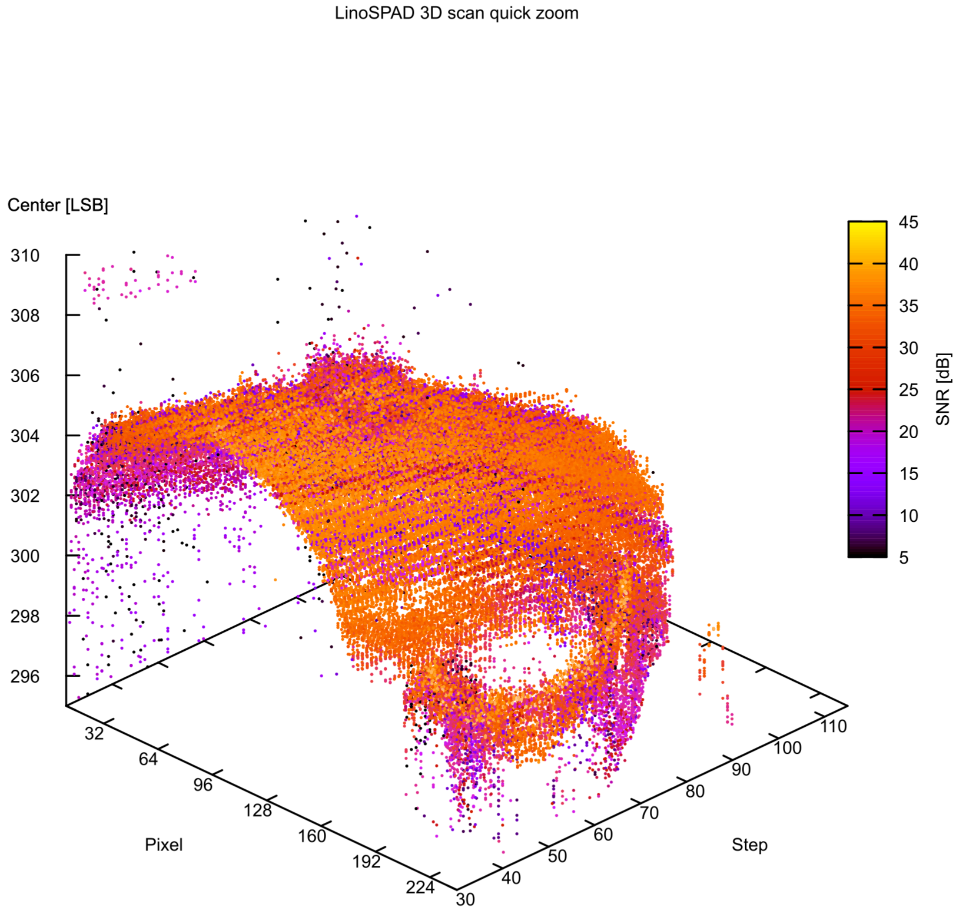


Figure 5.37: A close-up on the cup border extracted by the quick method provides an idea of the accuracy and resolution of the depth measurement. The top of the cup has a radius of 27.5 mm or 7.33 LSB and is at a distance of approximately 35 cm from the sensor.

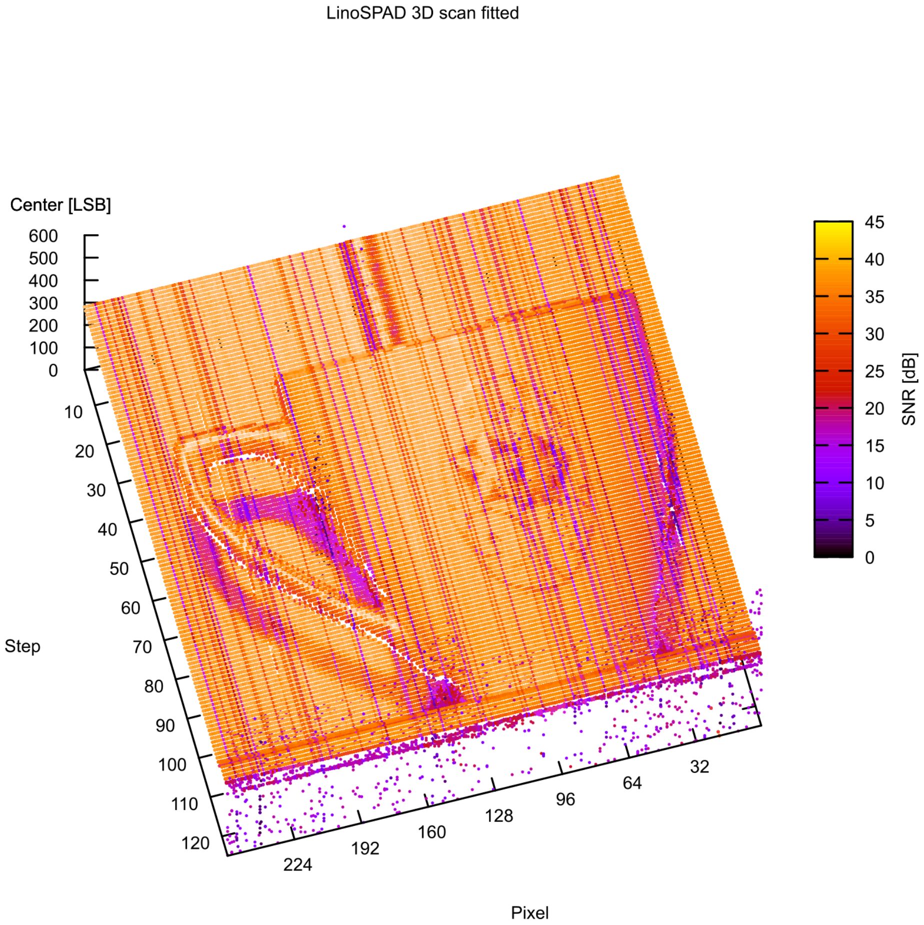


Figure 5.38: 3D image of the cup using depth extracted from the curve fitting. Only 0.1% of the histograms failed to fit. Histograms with a low signal above a high noise floor provided good distance results. The image is more uniform than its quick counterpart is, especially in low SNR areas.

The results from the curve fitting are presented in the same way, as those from the quick method in Figure 5.38 and Figure 5.39. Points where the fit failed to converge, primarily due to very low SNR, were removed. The fit failed to converge for 357 out of 320,000 histograms (0.1%) especially on the last step below the moving platform. The visual impression of the fitted result is much cleaner compared to the quick method, especially in areas of low SNR.

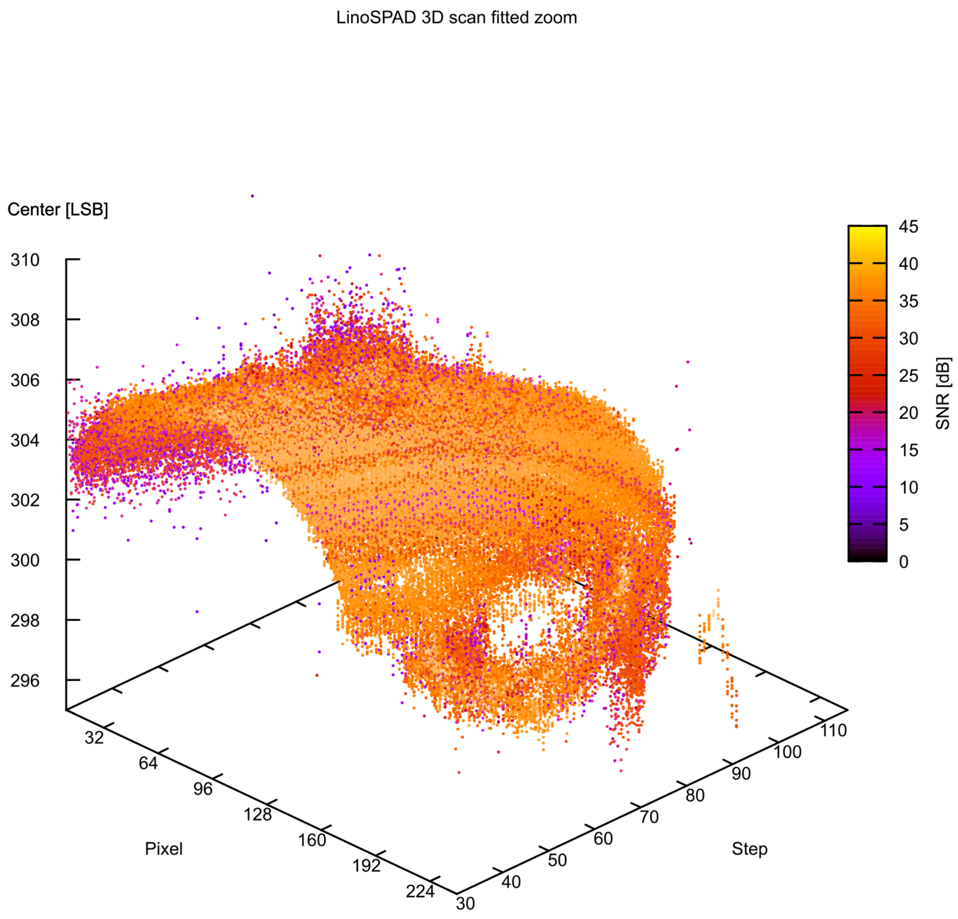


Figure 5.39: The close-up of the fit results on the cup is quite comparable to the results from the quick method. One noticeable difference is that there is less noise at the edges of the cup, at low and high pixel indices.

The direct time-of-flight 3D images presented in this section demonstrate the performance of the LinoSPAD camera TDC architecture and processing chain. The 64 TDCs provide a precision around 25 ps for repetitive measurements, with minimal processing, and enable the rapid prototyping of time-of-arrival applications and algorithms.

These measurements conclude the first body of work establishing the LinoSPAD platform, based around a SPAD line sensor, with 256 pixels connected individually to a FPGA implementing flexible time-of-arrival detection.

Chapter 6 Conclusions

6.1 Summary of results

This dissertation introduced the largest SPAD array imager to date, SwissSPAD. SwissSPAD was designed as a generic high-speed image sensor, striking a balance between the circuit features integrated in a single pixel and the photon detection efficiency. The 512 × 128 pixels fabricated in a standard CMOS process have gating functionality and 1 bit of memory in a square of 24 µm × 24 µm, with a fill factor of 5%.

A fast signal distribution for the gate signals controlled externally is integrated in the sensor to drive the in-pixel transistors over the whole sensor with minimal skew. The gate allows for photon sensitive integration windows as short as 4 ns, synchronized to almost arbitrary signals, thanks to the flexibility of FPGAs. The FPGA is also the link for the data from the sensor to the outside world, as it commands the readout of the sensor.

SwissSPAD is connected to the readout logic in a FPGA over 128 signal lines, each operating at a rate of up to 80 MHz, for a combined data rate of 10.24 Gbps. The handling of the large data rate was one of the major challenges in the work related to SwissSPAD and later LinoSPAD as well. The data from the sensor needs to be received and processed at the full rate if one does not want to lose data or increase the time needed to complete a measurement. In SwissSPAD, the data rate is reduced using accumulation counters for each pixel in the FPGA, where multiple one-bit frames can be summed up, thereby reducing the data rate for each additional bit in the output by a further factor of 2. However, the challenge does not end at the FPGA most of the time. Instead, the data needs to be transmitted for further analysis or display and storage. Reducing the data rate of SwissSPAD to the data rate of high-speed USB means that only 8-bit bitmaps can be transferred, thereby loosing much of the possible temporal resolution beneficial to applications like FCS. The solution found for SwissSPAD was the usage of a RAM buffer connected to the FPGA to store an image sequence before transmitting the data to the computer at a lower rate over USB2. This is a quite restrictive solution, however, as the memory fills up at a rate of over 1 GB/s. The best solution still might be to process the data in the FPGA or even on a future sensor as much as possible such that it can be offloaded in real time.

After SwissSPAD saw the light of day, it was time to assess the characteristics of large numbers of SPADs operated in parallel. With 512 × 128 pixels being operated in parallel, it is important that they also respond in parallel. This lead to the analysis carried out together with I. M. Antolovic regarding the non-uniformity over a range of parameters, and to the development of the generic readout model for SPAD-based imagers, that enables the linearization of measurements. The most important outcome is the correction formula to convert measured count rates to the correct impinging photon rates on each pixel.

During the work on this dissertation, a number of applications have used of SwissSPAD, further exploring its limits and guiding the work on future sensors. This work presents the results of SwissSPAD covering three applications. A high-speed imaging camera with frame rates up to 156,000 frames per second, fluorescence lifetime imaging using a spot imaging system and the generation of quantum random numbers. The author assisted other researchers in the acquisition of fluorescence correlation and fluorescence super resolution measurements.

Quantum random numbers appear quite naturally in SPAD based image sensors, as the generation of an avalanche from a photon in a SPAD has random properties in time and space. Using the accurate gating on SwissSPAD, the sensor is operated such that the chance of seeing a photon is ½ for a single measurement. After further de-biasing, the stream of bits from consecutive measurements pass the most important statistical test-suite for randomness assessment. True random numbers play an important role in cryptography used in practically all communication nowadays.

Fluorescence lifetime imaging using a time-gated camera was explored using SwissSPAD, and algorithms for lifetime extraction from a pixels response to changing gate signals were developed and analyzed on a number of fluorescent samples. The analysis showed that the fill factor of SwissSPAD is too often not sufficient for applications such as FLIM where every photon counts. This led to important work on a microlens array for SwissSPAD. Using miniature lenses formed directly on the sensor surface, the effective fill factor is increased to capture a higher number of photons.

The development of LinoSPAD was driven by a premise quite distinct from SwissSPAD. Where SwissSPAD aimed for a large format and had to trade-off fill factor and pixel functionality, LinoSPAD targeted a small format with high fill factor and flexible pixel circuitry at the same time. LinoSPAD is a 256 × 1 SPAD line sensor with a fill factor over 40%. Each pixel only has quenching circuitry integrated on the sensor and a direct connection to a FPGA.

As in SwissSPAD, a FPGA plays a pivotal role in the LinoSPAD system, where the FPGA is even closer to the pixel. The LinoSPAD FPGA receives not only the information whether a photon was seen or not, but also the photon timing. The ability to measure the time-of-arrival of single photons with high precision is what differentiates a SPAD camera from other scientific cameras. Thus, the first circuit that was needed in the FPGA of LinoSPAD was a time-to-digital converter. Moreover, not only one, but optimally 256, one per pixel.

With 64 TDCs in a Spartan 6 FPGA, each capable of timing photons at a rate of 133 MHz and with a precision of 25 ps over a range of a few milliseconds, the LinoSPAD firmware outperforms many commercially available TDC systems in both the number of channels and the precision and range of the measurements. The data rate was a challenge again, like in SwissSPAD, but was easier to handle thanks to USB3, which offers ten times the data rate of USB2.

LinoSPAD reduces the data rate to be sent to the computer by building histograms of arrival times. The data rate can be even further reduced by analyzing the resulting histograms on the FPGA and only forwarding the relevant statistics to the PC. To this end, real time histogram processing has been added to the FPGA firmware, and simple algorithms to extract the relevant parameters have been proposed in this work. Building on these blocks, many interesting applications can be prototyped using LinoSPAD and promising ones can be implemented on the FPGA and later in dedicated circuits. LinoSPAD provides a flexible base for a high number of possible applications using photon time-of-arrival.

6.2 Current status

This thesis has shown the current state in large format SPAD sensor performance through the analysis of many factors of non-uniformity. The LinoSPAD system presented in Chapter 3 presents a good platform to prototype various alternatives of in-pixel or per-pixel digital processing.

Many of the non-uniformities presented can be successfully accounted for in post-processing algorithms. Leveraging the processing speeds of today's microprocessors or graphic accelerators, which have continued to increase over the course of this work, we believe that it is possible to realize a megapixel SPAD sensor and integrate many of the uniformity corrections as is done for other camera technologies.

The performance of cameras competing with SPAD sensors, especially of the current CMOS-based sensors, has improved, and begins to rival even the readout speed of SPAD based sensors. In terms of timing resolution though, especially regarding the arrival time of single photons, SPADs are and will very likely remain, clearly ahead of CMOS.

6.3 Future development

Many avenues for future development can be imagined, starting from the work presented here. A few of them have already been mentioned and some are actively pursued already.

SwissSPAD provided an interesting base for fluorescence-based measurements on one hand, and for quantum random numbers on the other. Regarding fluorescence-based measurements, the most important point in the future is increasing the fill factor for array sensors, either by further development of microlenses or by architectural changes such as 3D integration and backside illumination for SPADs. First steps in these directions have been made, but more are needed to bring the solutions all together in the next SPAD based cameras.

For quantum random numbers, the fill factor or efficiency is of less concern, but minimal power consumption and low noise are desired parameters to optimize. Regardless of the application, the high data rate produced by SPAD-based sensors where each photon generates a data point with an increasing number of bits, remains an important challenge.

New approaches to parallel algorithms for correcting non-uniformity, capable of handling billions of photons per second, will be needed to keep up with the growing size of sensors. In addition, even if many of these algorithms can be implemented close to the source, down to the individual SPAD pixel, the rate of data to be transferred will keep growing. It is important, that future sensor systems be designed with attention to bandwidth at different levels in the design and employ appropriate transmission protocols. The power used at the I/O of the sensors of this work would be better spent elsewhere or avoided completely in exchange for a cooler sensor with less noise.

There remain many challenges in the future of SPAD sensors as the number of applications, which become susceptible of using SPAD sensors, grows. New, unforeseen applications making use of high time resolutions will certainly appear in the future.

References

[1] H. Hertz, "Ueber einen Einfluss des ultravioletten Lichtes auf die electrische Entladung," *Annalen der Physik*, vol. 267, no. 8, pp. 983-1000, 1887.

[2] Julius Elster and Hans Geitel, "Ueber die Entladung negativ electrischer Körper durch das Sonnen- und Tageslicht," *Annalen der Physik*, vol. 274, no. 12, pp. 497-514, 1889.

[3] A. Einstein, "Über einen die Erzeugung und Verwandlung des Lichtes betreffenden heuristischen Gesichtspunkt," *Annalen der Physik*, vol. 322, no. 6, pp. 132-148, 1905.

[4] V. K. Zworykin, G. A. Morton, and L. Malter, "The Secondary Emission Multiplier-A New Electronic Device," *Proceedings of the Institute of Radio Engineers*, vol. 24, no. 3, pp. 351-375, Mar. 1936.

[5] B. K. Lubsandorzhiev, "On the history of photomultiplier tube invention," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 567, no. 1, pp. 236-238, 2006.

[6] Hamamatsu Photonics, *Photomultiplier Tubes: Basics and Applications*., 2007.

[7] "Bull's-eyes in the Night.," *Popular Science*, p. 73, 1946.

[8] Armasight. (2016, May) Night Vision History. [Online]. http://www.armasight.com/night-vision-academy/night-vision-history

[9] Michael Lampton, "The Microchannel Image Intensifier," *Scientific American*, vol. 245, no. 5, pp. 62-71, Nov. 1981.

[10] X. Michalet, O. H. W. Siegmund, J. V. Vallerga, P. Jelinsky, J. E. Millaud, and S. Weiss, "Photon-Counting H33D Detector for Biological Fluorescence Imaging.," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 567, no. 1, p. 133, Nov. 2006.

[11] Gert Nützel, "Single-Photon Imaging Using Electron Multiplication in Vacuum," in *Single-Photon Imaging*, P. Seitz and A. Theuwissen, Eds. Heidelberg: Springer, Sep. 2011, ch. 5, pp. 73-102, ISBN 978-3-642-18442-0.

[12] W. S. Boyle and G. E. Smith, "Charge Coupled Semiconductor Devices," *Bell System Technical Journal*, vol. 49, no. 4, pp. 587-593, 1970.

[13] Sony Corporation. (2015, January) Sony CCD Discontinuation Plan. [Online]. https://www.framos.com/fileadmin/media/pdf/news/2015/SONY-CCD_Final-Discontinuation-Plan.pdf

[14] Ian S. McLean, "Charge-coupled Devices," in *Electronic Imaging in Astronomy*.: Springer, 2008, ch. 7.

[15] James R. Janesick, *Scientific charge-coupled devices*.: SPIE - The International Society for Optical Engineering, 2001.

[16] John V. Vallerga, Patrick N. Jelinsky, and Oswald H. W. Siegmund, "Initial results from a photon-counting intensified CCD development," in *Proc. SPIE*, vol. 2518, 1995, pp. 410-421.

[17] Donal J. Denvir and Emer Conroy, "Electron Multiplying CCD: The new ICCD," in *Proc. SPIE*, vol. 4796, 2003, pp. 164-174.

[18] Eric R. Fossum, "Active pixel sensors: are CCDs dinosaurs?," in *Proc. SPIE*, vol. 1900, 1993, pp. 2-14.

[19] E. R. Fossum, "CMOS image sensors: electronic camera on a chip," in *Proceedings of International Electron Devices Meeting*, 1995, pp. 17-25.

[20] N. Teranishi, A. Kohono, Y. Ishihara, E. Oda, and K. Arai, "No image lag photodiode structure in the interline CCD image sensor," in *International Electron Devices Meeting*, 1982, pp. 324-327.

[21] Eric R. Fossum and Donald B. Hondongwa, "A Review of the Pinned Photodiode for CCD and CMOS Image Sensors," *IEEE J. Electron Devices Soc.*, vol. 2, no. 3, pp. 33-43, May 2014.

[22] Colin Coates, Boyd Fowler, and Gerhard Holst, "sCMOS Scientific CMOS Technology A High-Performance Imaging Breakthrough," Andor Technology, Fairchild Imaging, PCO AG, Tech. rep. 2009. [Online]. www.scmos.com

[23] Xinqiao Liu, Boyd A. Fowler, Steve K. Onishi, Paul Vu, David D. Wen, Hung Do, and Stuart Horn, "CCD/CMOS hybrid FPA for low light level imaging," in *Proc. SPIE*, vol. 5881, 2005.

[24] P. R. Rao, X. Wang, and A. J. P. Theuwissen, "CCD structures implemented in standard 0.18 um CMOS technology," *Electronics Letters*, vol. 44, no. 8, pp. 548-549, Apr. 2008.

[25] E. Charbon and M. W. Fishburn, "Monolithic Single-Photon Avalanche Diodes: SPADs," in *Single-Photon Imaging*, Peter Seitz and Albert J.P. Theuwissen, Eds. Heidelberg, Germany: Springer, Sep. 2011, ch. 7, pp. 123-157.

[26] R. J. McIntyre, "Recent developments in silicon avalanche photodiodes," *Measurement*, vol. 3, no. 4, pp. 146-152, 1985.

[27] S. Cova, A. Longoni, A. Andreoni, and R. Cubeddu, "A semiconductor detector for measuring ultraweak fluorescence decays with 70ps FWHM resolution," *IEEE Journal of Quantum Electronics*, vol. 19, no. 4, pp. 630-634, Apr. 1983.

[28] S. Cova, A. Longoni, and A. Andreoni, "Towards picosecond resolution with single-photon avalanche diodes," *Review of Scientific Instruments*, vol. 52, no. 3, pp. 408-412, 1981.

[29] Roland H. Haitz, "Mechanisms Contributing to the Noise Pulse Rate of Avalanche Diodes," *Journal of Applied Physics*, vol. 36, no. 10, pp. 3123-3131, 1965.

[30] Alexis Rochas, Gregoire Ribordy, B. Furrer, P. A. Besse, and R. S. Popovic, "First passively-quenched single photon counting avalanche photodiode element integrated in a conventional CMOS process with 32ns dead time," in *Proc. SPIE*, vol. 4833, 2002, pp. 107-115.

[31] D. Mosconi, D. Stoppa, L. Pancheri, L. Gonzo, and A. Simoni, "CMOS Single-Photon Avalanche Diode Array for Time-Resolved Fluorescence Detection," in *Proceedings of the 32nd European Solid-State Circuits Conference*, Sep. 2006, pp. 564-567.

[32] H. Finkelstein, M. J. Hsu, and S. C. Esener, "STI-Bounded Single-Photon Avalanche Diode in a Deep-Submicrometer CMOS Technology," *IEEE Electron Device Letters*, vol. 27, no. 11, pp. 887-889, Nov. 2006.

[33] C. Niclass, M. Gersbach, R. Henderson, L. Grant, and E. Charbon, "A Single Photon Avalanche Diode Implemented in 130-nm CMOS Technology," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 13, no. 4, pp. 863-869, July 2007.

[34] M. A. Karami, M. Gersbach, H. J. Yoon, and E. Charbon, "A New Single-photon Avalanche Diode in 90nm Standard CMOS Technology," *Optics Express*, vol. 18, no. 21, pp. 22158-22166, Oct. 2010.

[35] E. Charbon, H. J. Yoon, and Y. Maruyama, "A Geiger Mode APD fabricated in Standard 65nm CMOS Technology," in *Proceedings IEEE International Electron Device Meeting (IEDM)*, Dec. 2013, pp. 2751-2754.

[36] C. Niclass, C. Favi, T. Kluter, F. Monnier, and E. Charbon, "Single-Photon Synchronous Detection," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 7, pp. 1977-1989, July 2009, 00189200.

[37] M. Fishburn, *Fundamentals of CMOS Single-Photon Avalanche Diodes*. The Netherlands: TU Delft, Sep. 2012, PhD thesis.

[38] Chockalingam Veerappan and Edoardo Charbon, "A Low Dark Count p-i-n Diode Based SPAD in CMOS Technology," *IEEE Trans. Electron Devices*, vol. 63, no. 1, pp. 65-71, Jan. 2016.

[39] Chokalingam Veerappan, *Single Photon Avalanche Diodes for Cancer Diagnosis*. The Netherlands: TU Delft, 2016, PhD thesis.

[40] A. Eisele, R. Henderson, B. Schmidtke, T. Funk, L. A. Grant, J. A. Richardson, and W. Freude, "185 MHz Count Rate, 139 dB Dynamic Range Single-Photon Avalanche Diode with Active Quenching Circuit in 130nm CMOS Technology," in *Proceedings International Image Sensor Workshop (IISW)*, June 2011, pp. 278-281.

[41] Ivan Rech, Antonino Ingargiola, Roberto Spinelli, Ivan Labanca, Stefano Marangoni, Massimo Ghioni, and Sergio Cova, "Optical crosstalk in single photon avalanche diode arrays: a new complete model," *Opt. Express*, vol. 16, no. 12, pp. 8381-8394, May 2008.

[42] A. Rochas, M. Gösch, A. Serov, P. A. Besse, R. S. Popovic, T. Lasser, and R. Rigler, "First Fully Integrated 2-D Array of Single-Photon Detectors in Standard CMOS Technology," *IEEE Photonics Technology Letters*, vol. 15, no. 7, pp. 963-965, July 2003.

[43] M. Gersbach, R. Trimananda, Y. Maruyama, M. Fishburn, D. Stoppa, J. Richardson, R. Walker, R. K. Henderson, and E. Charbon, "High frame-rate TCSPC-FLIM readout system using a SPAD-based image sensor," in *Proc. SPIE*, vol. 7780, Aug. 2010.

[44] C. Niclass, C. Favi, T. Kluter, M. Gersbach, and E. Charbon, "A 128x128Single-Photon Image Sensor with Column-Level 10-bit Time-to-Digital Converter Array," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 2977-2989, Dec. 2008.

[45] C. Niclass, M. Sergio, and E. Charbon, "A Single Photon Avalanche Diode Array Fabricated in Deep-Submicron CMOS Technology," in *Proceedings of the Design Automation Test in Europe Conference*, vol. 1, Mar. 2006, pp. 1-6.

[46] J. Mata Pavia, M. Scandini, S. Lindner, M. Wolf, and E. Charbon, "A 1x400 Backside-Illuminated SPAD Sensor With 49.7ps Resolution, 30pJ/Sample TDCs Fabricated in 3D CMOS Technology for Near-Infrared Optical Tomography," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 10, pp. 2406-2418, Oct. 2015.

[47] Samuel Burri and Edoardo Charbon, "SPAD Image Sensors: From Architectures to Applications," in *Imaging and Applied Optics Technical Papers*, 2012.

[48] C. Niclass, A. Rochas, P.-A. Besse, and E. Charbon, "Design and characterization of a CMOS 3-D image sensor based on single photon avalanche diodes," *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1847-1854, Sep. 2005.

[49] Maximilian Sergio, Cristiano Niclass, and Edoardo Charbon, "A 128x2 CMOS Single-Photon Streak Camera with Timing-Preserving Latchless Pipeline Readout," in *IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, Feb. 2007.

[50] M. Gersbach, Y. Maruyama, R. Trimananda, M. W. Fishburn, D. Stoppa, J. A. Richardson, R. Walker, R.K.Henderson, and E. Charbon, "A Time-Resolved, Low-Noise Single-Photon Image Sensor Fabricated in Deep-Submicron CMOS Technology," *Journal of Solid-State Circuits*, vol. 47, no. 6, pp. 1394-1407, June 2012.

[51] E. Charbon, M. Scandini, J. Mata Pavia, and M. Wolf, "A dual backside-illuminated 800-cell multi-channel digital SiPM with 100 TDCs in 130nm 3D IC technology," in *2014 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Nov. 2014, pp. 1-4.

[52] K. Suhling, "Fluorescence lifetime Imaging," in *Cell Imaging*, D. Stephens, Ed. Bloxham: Scion Publishing, 2006.

[53] H. A. R. Homulle, F. Powolny, P. L. Stegehuis, J. Dijkstra, D.-U. Li, K. Homicsko, D. Rimoldi, K. Muehlethaler, J. O. Prior, R. Sinisi, E. Dubikovskaya, E. Charbon, and C. Bruschini, "Compact solid-state CMOS single-photon detector array for in vivo NIR fluorescence lifetime oncology measurements," *Biomedical Optics Express*, vol. 7, no. 5, p. 1797, Apr. 2016.

[54] F. Powolny, S. Burri, C. Bruschini, F. Regazzoni, X. Michalet, and E. Charbon, "Comparison of Two Cameras based on Single Photon Avalanche Diodes (SPADs) for Fluorescence Lifetime Imaging Application with Picosecond Resolution," in *Proceedings International Image Sensor Workshop (IISW)*, June 2013.

[55] S. Burri, F. Powolny, C. Bruschini, X. Michalet, F. Regazzoni, and E. Charbon, "A 65k pixel, 150k frames-per-second camera with global gating and micro-lenses suitable for fluorescence lifetime imaging," in *Proc. SPIE*, vol. 9141, May 2014.

[56] S. Burri, Y. Maruyama, X. Michalet, F. Regazzoni, C. Bruschini, and E. Charbon, "Architecture and applications of a high resolution gated SPAD image sensor," *Optics Express*, vol. 22, no. 14, pp. 17573-17589, July 2014.

[57] K. Wold and S. Petrovic, "Optimizing Speed of a True Random Number Generator in FPGA by Spectral Analysis," in *ICCIT*, Nov. 2009, pp. 1105-1110.

[58] F. Pareschi, G. Setti, and R. Rovatti, "Implementation and Testing of High-Speed CMOS True Random Number Generators Based in Chaotic Systems," *IEEE Trans. Circ. & Sys.*, vol. 57, no. 12, pp. 3124-3137, Dec. 2010.

[59] I. Kanter, Y. Aviad, I. Reidler, E. Cohen, and M. Rosenbluh, "An Optical Ultrafast Random Bit Generator," *Nature Photonics*, vol. 4, pp. 58-61, 2010.

[60] W. Wei, G. Xie, A. Dang, and H. Guo, "High-Speed and Bias-Free Optical Random Number Generator," *Photonics Technology Letters*, vol. 24, no. 6, pp. 437-439, June 2012.

[61] D. Stucki, S. Burri, E. Charbon, Chunnilall C., Meneghetti A., and F. Regazzoni, "Towards a High-Speed Quantum Random Number Generator," in *Proc. SPIE*, vol. 8899, Sep. 2013.

[62] Samuel Burri, Damien Stucki, Yuki Maruyama, Claudio Bruschini, Edoardo Charbon, and Francesco Regazzoni, "Jailbreak Imagers: Transforming a Single-Photon Image Sensor into a True Random Number Generator," in *Proceedings International Image Sensor Workshop (IISW)*, 2013.

[63] S. Burri, D. Stucki, Y. Maruyama, C. Bruschini, E. Charbon, and F. Regazzoni, "SPADs for Quantum Random Number Generators and beyond," in *ASP-DAC*, Feb. 2014, pp. 788-794.

[64] Samuel Burri, Harald Homulle, Claudio Bruschini, and Edoardo Charbon, "LinoSPAD: a time-resolved 256x1 CMOS SPAD line sensor system featuring 64 FPGA-based TDC channels running at up to 8.5 giga-events per second," in *Proc. SPIE*, vol. 9899, apr 2016.

[65] J. R. Lakowicz, *Principles of Fluorescence Spectroscopy*. New York: Springer, 2006.

[66] Ivan Michel Antolovic, Samuel Burri, Claudio Bruschini, Ron Hoebe, and Edoardo Charbon, "Analyzing blinking effects in super resolution localization microscopy with single-photon SPAD imagers," in *Proc. SPIE*, vol. 9714, Mar. 2016.

[67] A. Carimatto and E. Charbon, "Large Scale CMOS Single-Photon Detector Arrays for PET Applications," in *Front End Electronics*, May 2014.

[68] S. Mandai and E. Charbon, "A Multi-Channel Digital Silicon Photomultiplier Array for Nuclear Medical Imaging Systems based on PET-MRI," in *Proceedings International Image Sensor Workshop (IISW)*, June 2013.

[69] Francesco Panzeri, Antonino Ingargiola, Ron R. Lin, Niusha Sarkhosh, Angelo Gulinatti, Ivan Rech, Massimo Ghioni, Sergio Cova, Shimon Weiss, and Xavier Michalet, "Single-molecule FRET experiments with a red-enhanced custom technology SPAD," in *Proc. SPIE*, vol. 8590, Feb. 2013.

[70] Antonino Ingargiola, Francesco Panzeri, Niusha Sarkhosh, Angelo Gulinatti, Ivan Rech, Massimo Ghioni, Shimon Weiss, and Xavier Michalet, "8-spot smFRET analysis using two 8-pixel SPAD arrays," in *Proc. SPIE*, vol. 8590, Feb. 2013.

[71] J. Mata Pavia, M. Wolf, and E. Charbon, "3D Near-Infrared Imaging Based on a Single-Photon Avalanche Diode Array Sensor," in *Proc. SPIE*, vol. 8460, Oct. 2012.

[72] J. Mata Pavia, M. Wolf, and E. Charbon, "Single-Photon Avalanche Diode Imagers Applied to Near-Infrared Imaging," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 20, no. 6, Nov. 2014.

[73] J. Blacksberg, Y. Maruyama, M. Choukroun, E. Charbon, and G. R. Rossman, "Combined Raman and LIBS for Planetary Surface Exploration: Enhanced Science Return Enabled by Time-Resolved Laser Spectroscopy," in *Proceedings NASA International Workshop on Instrumentation for Planetary Missions*, Oct. 2012.

[74] Y. Maruyama, J. Blacksberg, G. R. Rossman, and E. Charbon, "A time-resolved 128x128 SPAD camera for laser Raman spectroscopy," in *Proc. SPIE*, vol. 8374, May 2012.

[75] Y. Maruyama, J. Blacksberg, and E. Charbon, "A 1024x8, 700-ps Time-Gated SPAD Line Sensor for Planetary Surface Exploration With Laser Raman Spectroscopy and LIBS," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 1, pp. 179-189, Jan. 2014.

[76] Y. Maruyama and E. Charbon, "A Time-Gated 128x128 CMOS SPAD Array for on-Chip Fluorescence Detection," in *Proceedings International Image Sensor Workshop (IISW)*, June 2011.

[77] I. M. Antolovic, S. Burri, C. Bruschini, R. Hoebe, and E. Charbon, "Nonuniformity Analysis of a 65-kpixel CMOS SPAD Imager," *IEEE Transactions on Electron Devices*, vol. 63, no. 1, pp. 57-64, Jan. 2016.

[78] Xilinx, Spartan-6 FPGA Configurable Logic Block User Guide (UG384), 2010.

[79] Jinyuan Wu, "Several Key Issues on Implementing Delay Line Based TDCs Using FPGAs," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 3, pp. 1543-1548, June 2010.

[80] Weibin Pan, Guanghua Gong, and Jianmin Li, "A 20-ps Time-to-Digital Converter (TDC) Implemented in Field-Programmable Gate Array (FPGA) with Automatic Temperature Correction," *IEEE Trans. Nucl. Sci.*, vol. 61, no. 3, pp. 1468-1473, June 2014.

[81] Bui Van Hieu, Seunghyun Beak, Seunghwan Choi, Jongkook Seon, and Taikyeong Ted Jeong, "Thermometer-to-binary encoder with bubble error correction (BEC) circuit for Flash Analog-to-Digital Converter (FADC)," in *International Conference on Communications and Electronics*, aug 2010.

[82] Claudio Favi and Edoardo Charbon, "A 17ps Time-to-digital Converter Implemented in 65nm FPGA Technology," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, Feb. 2009, pp. 113-120.

[83] Xilinx, Command Line Tools User Guide (UG628), 2013.

[84] Frédéric Rivoallon, "Improving Performance in Spartan-6 FPGA Designs (WP311)," Xilinx, Tech. rep. 2011.

[85] Xilinx, Hierarchical Design Methodology Guide (UG748), 2013.

[86] H. Homulle, F. Regazzoni, and E. Charbon, "200MS/s ADC implemented in a FPGA employing TDCs," in *International Symposium on Field-Programmable Gate Arrays*, Feb. 2015.

[87] Claudio Favi, *Single-Photon Techniques for Standard CMOS Digital ICs*. Lausanne, Switzerland: EPFL, 2011, vol. 4954, PhD thesis.

[88] M. W. Fishburn and E. Charbon, "Statistical Limitations of TDC Density Tests," in *Proceedings IEEE Nuclear Science Symposium (NSS)*, Oct. 2012.

[89] Wilfried Uhring, Chantal-Virginie Zint, and Jeremy Bartringer, "A low-cost high-repetition-rate picosecond laser diode pulse generator," , vol. 5452, 2004, pp. 583-590.

[90] M. W. Fishburn, H. Menninga, and E. Charbon, "A 19.6ps, FPGA-Based TDC with Multiple Channels for Open Source Applications," *IEEE Transactions on Nuclear Science*, vol. 60, no. 3, pp. 2203-2208, June 2013.

[91] Sebastian Bourdeauducq, "Time to Digital Converter core for Spartan 6 FPGAs," *http://www.ohwr.org/projects/tdc-core/documents*, 2011 (accessed March 4, 2016).

[92] Jan Buchholz, *Evaluation of single photon avalanche diode arrays for imaging fluorescence correlation spectroscopy*. Heidelberg, Germany: DKFZ, 2014, PhD thesis.

[93] J. Mata Pavia, M. Wolf, and E. Charbon, "Measurement and modeling of microlenses fabricated on single-photon avalanche diode arrays for fill factor recovery," *Optics Express*, vol. 22, no. 4, pp. 4202-4213, Feb. 2014.

[94] S. Donati, G. Martini, and M. Norgia, "Microconcentrators to recover fill-factor in image photodetectors with pixel on-board processing circuits," *Optics Express*, vol. 15, pp. 18066-18075, 2007.

[95] S. Donati, E. Randone, M. Fathi, J.-H. Lee, E. Charbon, and G. Martini, "Uniformity of Concentration Factor and Back Focal Length in Molded Polymer Microlens Arrays," in *Conference on Lasers and Electro-Optics (CLEO)*, 2010.

[96] Xilinx, Spartan-6 FPGA Data Sheet: DC and Switching Characteristics, 2015.

[97] H. Menninga, C. Favi, M. W. Fishburn, and E. Charbon, "A Multi-channel, 10ps Resolution, FPGA-Based TDC with 300MS/s Throughput for Open-Source PET Applications," in *Proceedings IEEE Nuclear Science Symposium (NSS)*, Oct. 2011, pp. 1515-1522.

[98] S. L. Lin, S. Krishnan, and S. Mourad, "A self-binning BIST structure for data communications transceivers," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 5, pp. 1399-1407, Oct. 2003.

[99] A. Raychowdhury, S. Ghosh, and K. Roy, "A novel on-chip delay measurement hardware for efficient speed-binning," in *11th IEEE International On-Line Testing Symposium*, 2005.

[100] M. Y. Berezin and S. Achilefu, "Fluorescence Lifetime Measurements and Biological Imaging," *Chemical Reviews*, vol. 110, pp. 2641-2684, 2010.

[101] ISS. (2016) Lifetime Data of Selected Fluorophores. [Online]. http://www.iss.com/resources/pdf/datatables/LifetimeDataFluorophores.pdf

[102] Anna Gerega, Norbert Zolek, Tomasz Soltysinski, Daniel Milej, Piotr Sawosz, Beata Toczylowska, and Adam Liebert, "Wavelength-resolved measurements of fluorescence lifetime of indocyanine green," *J. Biomed. Opt.*, vol. 16, no. 6, 2011.

[103] J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308-313, Jan. 1965.

[104] A. Migdall et al., Eds., *Single-Photon Generation and Detection*, 1st ed.: Elsevier, 2013.

[105] Christopher J. Chunnilall, Ivo Pietro Degiovanni, Stefan Kück, Ingmar Müller, and Alastair G. Sinclair, "Metrology of single-photon sources and detectors: a review," *Optical Engineering*, vol. 53, no. 8, July 2014.

[106] NIST, A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications, 2010, Pub 800-22 rev1a.

[107] F. Xu, B. Qi, X. Ma, H. Xu, H. Zheng, and H.-K. Lo, "Ultrafast Quantum Random Number Generation based on Quantum Phase Fluctuations," *Optics Express*, vol. 20(11), pp. 12366-12377, Nov 2012.

[108] M. Matsumoto, S. Yasuda, R. Ohta, K. Ikegami, T. Tanamoto, and S. Fujita, "1200um^2 Physical Random-Number Generators Based on SiN MOSFET for Secure Smart-Card Application," in *Proceedings of the IEEE International Solid-State Circuits Conference*, Feb. 2008, pp. 414-415.

[109] Jan W. Krieger, *Mapping diffusion properties in living cells.* Heidelberg, Germany: DKFZ, 2014, PhD thesis.

[110] Jan W. Krieger, Jan Buchholz, Samuel Burri, Claudio Bruschini, Edoardo Charbon, Christoph S. Garbe, and Jörg Langowski, "Imaging Fluorescence Correlation: Novel Results on New Image Sensors (SPAD Arrays) and a Comprehensive New Software Package (QUICKFIT 3.0)," in *Focus on Microscopy*, 2015.

[111] Jan W. Krieger, Anand P. Singh, Nirmalya Bag, Christoph S. Garbe, Timothy E. Saunders, Jörg Langowski, and Thorsten Wohland, "Imaging fluorescence (cross-) correlation spectroscopy in live cells and organisms," *Nature Protocols*, vol. 10, no. 12, pp. 1948-1974, Nov. 2015.

[112] Douglas Magde, Elliot L. Elson, and Watt W. Webb, "Fluorescence correlation spectroscopy. II. An experimental realization," *Biopolymers*, vol. 13, no. 1, pp. 29-61, Jan. 1974.

[113] Douglas Magde, Watt W. Webb, and Elliot L. Elson, "Fluorescence correlation spectroscopy. III. Uniform translation and laminar flow," *Biopolymers*, vol. 17, no. 2, pp. 361-376, Feb. 1978.

[114] Elliot L. Elson, "Fluorescence Correlation Spectroscopy: Past, Present, Future," *Biophysical Journal*, vol. 101, no. 12, pp. 2855-2870, Dec. 2011.

[115] Thorsten Wohland, Xianke Shi, Jagadish Sankaran, and Ernst H. K. Stelzer, "Single Plane Illumination Fluorescence Correlation Spectroscopy (SPIM-FCS) probes inhomogeneous three-dimensional environments," *Optics Express*, vol. 18, no. 10, pp. 10627-10641, May 2010.

[116] E. Betzig, "Proposed method for molecular optical imaging," *Optics Letters*, vol. 20, no. 3, pp. 237-239, Feb. 1995.

[117] Eric Betzig, George H. Patterson, Rachid Sougrat, O. Wolf Lindwasser, Scott Olenych, Juan S. Bonifacino, Michael W. Davidson, Jennifer Lippincott-Schwartz, and Harald F. Hess, "Imaging Intracellular Fluorescent Proteins at Nanometer Resolution," *Science*, vol. 313, no. 5793, pp. 1642-1645, 2006.

[118] R. M. Dickson, A. B. Cubitt, R. Y. Tsien, and W. E. Moerner, "On/off blinking and switching behaviour of single molecules of green fluorescent protein.," *Nature*, vol. 388, no. 6640, pp. 355-358, July 1997.

[119] S. W. Hell and M. Kroug, "Ground-state-depletion fluorscence microscopy: A concept for breaking the diffraction resolution limit," *Applied Physics B*, vol. 60, no. 5, pp. 495-497, 1995.

[120] Stefan W. Hell and Jan Wichmann, "Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy," *Optics Letters*, vol. 19, no. 11, pp. 780-782, June 1994.

[121] T. A. Klar, S. Jakobs, M. Dyba, A. Egner, and S. W. Hell, "Fluorescence microscopy with diffraction resolution barrier broken by stimulated emission.," *Proc Natl Acad Sci U S A*, vol. 97, no. 15, pp. 8206-8210, July 2000.

[122] W. E. Moerner and L. Kador, "Optical detection and spectroscopy of single molecules in a solid," *Physical Review Letters*, vol. 62, pp. 2535-2538, May 1989.

[123] I. M. Antolovic, S. Burri, C. Bruschini, R. Hoebe, and E. Charbon, "Super Resolution With SPAD Imagers," in *Focus on Microscopy*, Mar. 2015.

[124] M. Ovesny, P. Krizek, J. Borkovec, S. Zdenek, and G. M. Hagen, "ThunderSTORM: a comprehensive ImageJ plug-in for PALM and STORM data analysis and super-resolution imaging," *Bioinformatics*, vol. 30, no. 16, pp. 2389-2390, Apr. 2014.

[125] Robert P. J. Nieuwenhuizen, Keith A. Lidke, Mark Bates, Daniela Leyton Puig, David Grünwald, Sjoerd Stallinga, and Bernd Rieger, "Measuring image resolution in optical nanoscopy," *Nature Methods*, vol. 10, no. 6, pp. 557-562, Apr. 2013.

[126] Kenneth Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly Journal of Applied Mathmatics*, vol. II, no. 2, pp. 164-168, 1944.

[127] Donald W. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431-441, June 1963.

[128] Jinyuan Wu and Zonghan Shi, "The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay," in *Proceedings IEEE Nuclear Science Symposium (NSS)*, Oct. 2008.

List of Figures

List of Tables

List of publications

- I. M. Antolovic*, **S. Burri*,** R. Hoebe, Y. Maruyama, C. Bruschini, and E. Charbon, "*Photon-Counting Arrays for Time-Resolved Imaging*," Sensors, vol. 16, num. 7, 2016. (* equally contributing authors)

- I. M. Antolovic*, **S. Burri*,** C. Bruschini, R. Hoebe, and E. Charbon, "*Nonuniformity Analysis of a 65-kpixel CMOS SPAD Imager*," IEEE Transactions on Electron Devices, vol. 63, DOI: 10.1109/TED.2015.2458295, 2015. (* equally contributing authors)

- **S. Burri**, Y. Maruyama, X. Michalet, F. Regazzoni, C. Bruschini, and E. Charbon, "*Architecture and applications of a high resolution gated SPAD image sensor*," Optics Express (ISSN: 1094-4087), vol. 22, num. 14, p. 17573-17589, Washington, OSA, 2014.

- T. Kluter, **S. Burri,** P. Brisk, E. Charbon, and P. Ienne, "*Virtual Ways: Low-Cost Coherence for Instruction Set Extensions with Architecturally Visible Storage*," ACM Transactions on Architecture and Code Optimization (ISSN: 1544-3566), vol. 11, num. 2, p. 3-28, New York, ACM, 2014.

- **S. Burri,** H. Homulle, C. Bruschini, and E. Charbon, "*LinoSPAD: a time-resolved 256 × 1 CMOS SPAD line sensor system featuring 64 FPGA-based TDC channels running at up to 8.5 giga-events per second*," Proc. SPIE, Optical Sensing and Detection IV, vol. 9899, 2016.

- I. M. Antolovic, **S. Burri,** C. Bruschini, R. Hoebe, and E. Charbon, *"Analyzing blinking effects in super resolution localization microscopy with single-photon SPAD imagers,"*., Proc. SPIE, Single Molecule Spectroscopy and Superresolution Imaging IX, vol. 9714, 2016.

- I. M. Antolovic, **S. Burri,** C. Bruschini, R. Hoebe, and E. Charbon, "*Super resolution with SPAD imagers*," Focus On Microscopy Conference, Göttingen, Germany, 2015.

- J. W. Krieger*, J. Buchholz*, **S. Burri**, C. Bruschini, E. Charbon, C.S. Garbe, and J. Langowski, "*Imaging fluorescence correlation: Novel results on new image sensors (SPAD arrays) and a comprehensive new software package (QuickFit 3.0)*," Focus On Microscopy Conference, Göttingen, Germany, 2015. (* equally contributing authors)

- **S. Burri,** F. Powolny, C. Bruschini, X. Michalet, F. Regazzoni, and E. Charbon, "*A 65k pixel, 150k frames-per-second camera with global gating and microlenses suitable for fluorescence lifetime imaging*," Proc. SPIE, Optical Sensing and Detection III, vol. 9141, 2014.

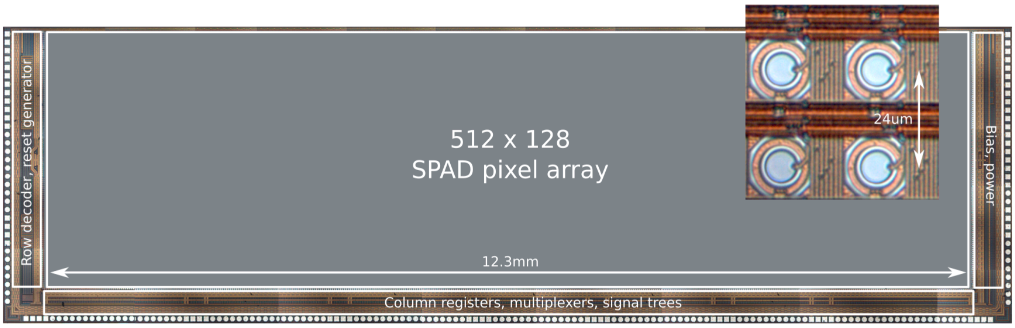
- J. Buchholz, J.W. Krieger, **S. Burri**, C. Bruschini, E. Charbon, U. Kebschull, and J. Langowski, "*Single photon avalanche diode arrays for single plane illumination fluorescence correlation spectroscopy*," Focus On Microscopy Conference, Sidney, Australia, 2014.

- **S. Burri**, D. Stucki, Y. Maruyama, C. Bruschini, and E. Charbon, and F. Regazzoni, "*SPADs for quantum random number generators and beyond*," ASP-DAC, p. 788-794, Singapore, 2014.

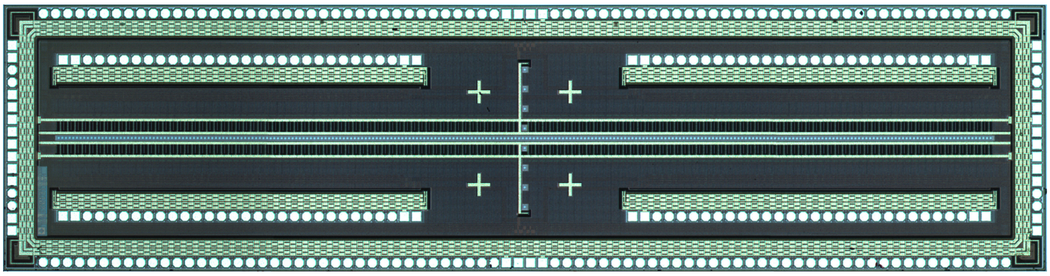
- V. Krishnaswami, **S. Burri,** F. Regazzoni, C. Bruschini, C. J. F. van Noorden, E. Charbon, and R. Hoebe, "*SPAD array camera for localization based super resolution microscopy*, " Focus On Microscopy Conference, Maastricht, the Netherlands, 2013.

- **S. Burri,** D. Stucki, Y. Maruyama, C. Bruschini, E. Charbon, and F. Regazzoni, "*Jailbreak Imagers: Transforming a Single-Photon Image Sensor into a True Random Number Generator*," International Image Sensor Workshop, Utah, USA, 2013.
- F. Powolny, **S. Burri,** C. Bruschini, X. Michalet, F. Regazzoni, and E. Charbon, "*Comparison of Two Cameras based on Single Photon Avalanche Diodes (SPADs) for Fluorescence Lifetime Imaging Application with Picosecond Resolution*," International Image Sensor Workshop, Utah, USA, 2013.
- D. Stucki, **S. Burri,** E. Charbon, C. Chunnilall, A. Meneghetti, and F. Regazzoni, "*Towards a High-Speed Quantum Random Number Generator*," SPIE Conference on Defense and Security, Dresden, Germany, 2013.
- **S. Burri** and E. Charbon, "*SPAD Image Sensors: From Architectures to Applications*," Imaging Systems and Applications 2012, Monterey, California, USA, 2012.
- T. Kluter, **S. Burri,** P. Brisk, E. Charbon, and P. Ienne, "Virtual Ways: Efficient Coherence for Architecturally Visible Storage in Automatic Instruction Set Extensions," HiPEAC, Pisa, Italy, 2010.

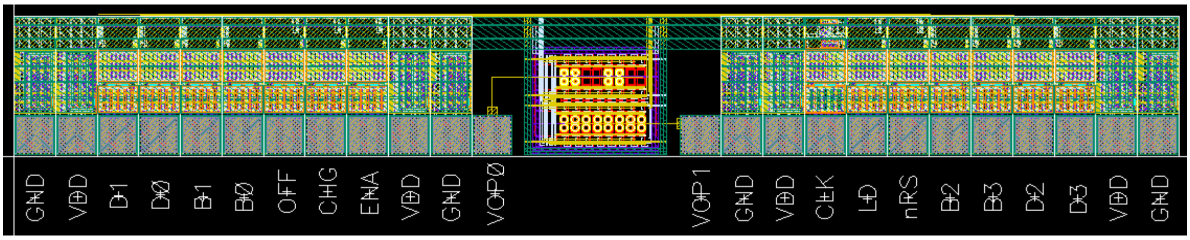
Chip gallery


512 x 128
SPAD pixel array
Row decoder, reset generator
Bias, power
24um
12.3mm
Column registers, multiplexers, signal trees

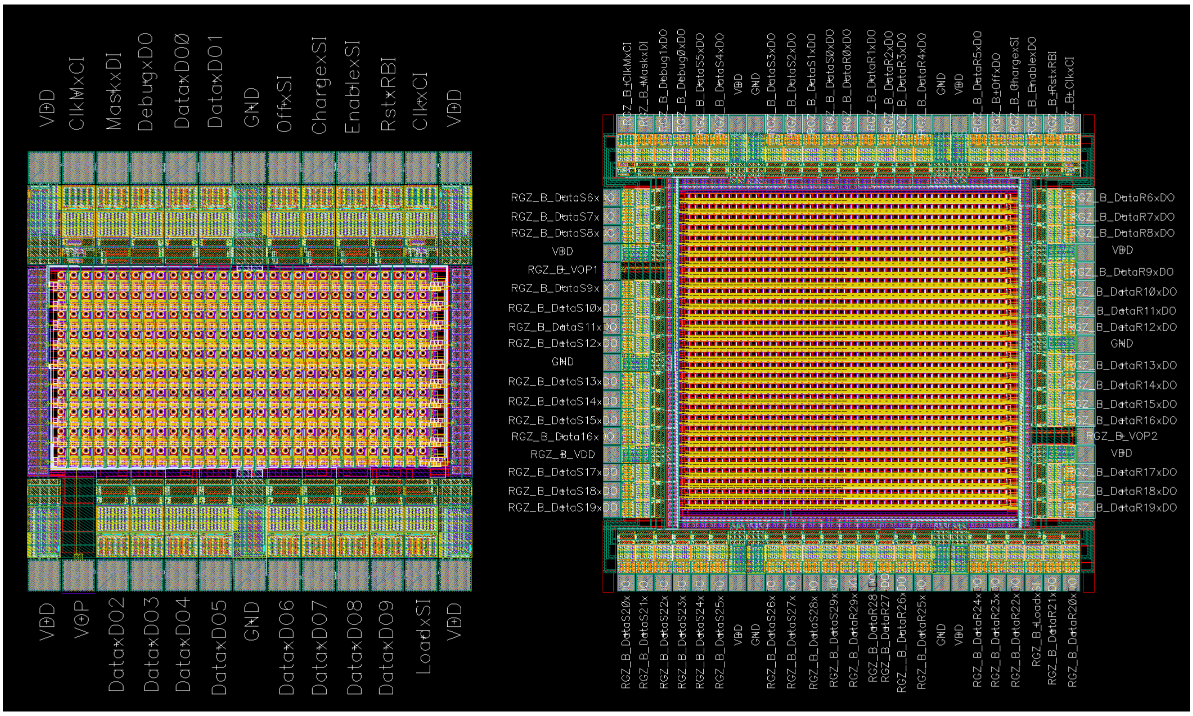
Die micrograph of the SwissSPAD sensor with the SPAD-array occupying most of the area and control and readout logic on three sides, such that two sensors can be abutted with a gap of a few pixels. The inset is a detail of the SPAD cells with a pitch of 24 µm.



Micrograph of the LinoSPAD sensor. Vertically in the center is an array of eight auxiliary pixels and four alignment crosses surround the center. Additional to the ring of 192 bonding pads there are 120 bonding pads in four blocks inside the ring.



Layout of a small test matrix for random number generation. The array is optimized for continuous operation with a shift-register style readout.



Small and large SPAD matrices for random number generation, modelled after the previous test design.

Acknowledgements

This thesis started, when I was working on my master thesis in the Processor Architecture Laboratory (LAP) at EPFL, and was approached by Professor Edoardo Charbon, looking for possible Ph.D. candidates to join his group working on CMOS SPADs. After obtaining my MSc, I joined the Advanced Quantum Architecture Group (AQUA) in the summer of 2010. In the six years leading to my Ph.D. I met many people who supported me with their advice, provided valuable feedback on my work, helped to obtain measurement data with my cameras or simply accompanied me on (part of) my way. I would like to thank every one of you and hope you understand that I cannot possibly mention everyone personally in the following paragraphs.

First, I would like to thank Edoardo for offering me the chance to undertake the journey and supporting me all the way. Though only part-time at EPFL during my work, he followed my developments closely and asked the difficult questions to guide my work when needed. As promised, he would always find the necessary resources to renew my work contract. Here I have to thank Claudio Bruschini as well for managing things at EPFL when Edoardo could not be present. Claudio tirelessly supported the last members of Edoardo’s group in Lausanne and provided me feedback on my work on countless occasions, especially in the writing of this thesis.

I had the chance to travel a few times with my cameras during this thesis and on these occasions met the members of my jury. The first visit was to the University of California Los Angeles (UCLA), where I met Dr. Xavier Michalet, who is working in the biochemistry group of Shimon Weiss. Together we evaluated an early SwissSPAD sensor in fluorescence lifetime applications. Xavier analyzed the details of the camera as good as I could provide them and did all the data analysis for the measurements performed at UCLA. In retrospect, my visit was too soon, as the SwissSPAD without microlenses did not have sufficient sensitivity for the applications Xavier had in mind. Still, I am deeply indebted to Xavier for all the feedback he provided in relation to my visit and last but not least for his thorough proofreading of the thesis draft.

After the development of microlenses on SwissSPAD my next visit lead me to the German Cancer Research Institute (DKFZ) where I met Prof. Jörg Langowski and his two Ph.D. students Jan Buchholz and Jan Krieger Together we setup SwissSPAD in their SPIM-FCS system and performed initial characterization. Jan K. constructed the SPIM and wrote data analysis software while Jan B. implemented his own firmware to operate SwissSPAD and offload sensor data in real time to another FPGA for correlation analysis.

The last visit on this tour brought a LinoSPAD camera to the Paul Scherrer Institute, where I met Dr. Rasmus Ischebeck working on the new Swiss Free Electron Laser (SwissFEL). While at PSI, I met Cigdem Ozkan Loch and Manuel Knecht, whom I introduced to LinoSPAD. I hope the system will prove useful in the analysis of ultrafast phenomena at PSI.

Before I could bring LinoSPAD to PSI, the firmware needed to be developed and the sensor characterized. To help me on this task Harald Homulle made an internship at EPFL to work with François Powolny on FLIM

data analysis and had the goal to compare LinoSPAD and with the Fluocam. Unfortunately, the first version of LinoSPAD did not work at this time. Nevertheless, Harald could spark firmware development and, together with François, advance the state of FLIM on SwissSPAD through his work.

With François, I also have to mention Hyungjune Yoon, post-doc at AQUA at the same time as François and Christian Mester doing his post-doc before. Thank you for the time you shared with me and for your support in my projects.

Of my fellow Ph.D. students in Zürich, I am indebted towards Juan Mata Pavia for his microlens simulator and discussions related to the fabrication and performance of the lenses.

From Edoardo's group in Delft I want to mention especially Ivan Michel Antolovic, who took over projects with SwissSPAD and introduced the sensor to more people. Michel was responsible for many non-uniformity measurements and successfully performed the first super resolution microscopy with a SPAD sensor. I also want to thank Shingo Mandai for his support in my tapeouts and everyone else for the presentations and discussions in the Skype meetings. It was always a pleasure to listen in.

Francesco Regazzoni is difficult to fit in these acknowledgements due to the very orthogonal nature of the work I did in collaboration with him. Together we evaluated SwissSPAD for random number generation and developed new chips to this purpose. Thank you Francesco for the publications on random number generations and all our interesting discussions on cryptography and computer security.

I also want to thank all the people making my stay at EPFL very enjoyable. First among them Alain Berthoud, who shared my office when I started, and Mahdi, who was on the same way as me pursuing his Ph.D. on germanium-on-silicon SPADs.

A special thank you to the people animating the LAP, especially Chantal Schneeberger, who helped me with many administrative tasks, despite not being responsible for our group. Thank you André Guignard, for your mechanical expertise and the crossword puzzles solved together after lunch. Thank you René Beuchat, for teaching me embedded systems and FPGAs and for supervising my master thesis. Thank you Theo Kluter and Paolo Ienne, for the semester project leading to my first publication at HiPEAC 2010, before my Ph.D. even started.

I am thankful also for AQUA's former secretary Joëlle Radigois and the group's current secretary Brigitte Khan. They were always quick to help on administrative matters, especially regarding the travelling I did.

I am very grateful for my parents, brothers and sister who were always there for me from the start of this thesis and for my wife and her family, who joined me on the journey after the first year. Without you, I would not be where I am now.

About the author

Samuel Burri was born in 1985 in Bienne, Switzerland. He grew up near Bern, where he pursued his education to the university-entrance diploma. In 2005, he began his studies in electrical and electronical engineering at École Politechnique Fédérale de Lausanne (EPFL). After a year at École Polytechnique de Montréal in Canada, he obtained his BSc in 2008 and his MSc in 2010 from EPFL. While working on his master's thesis in embedded systems, he was recruited by Professor Edoardo Charbon to pursue a Ph.D. on the topic of Single-Photon Avalanche Diodes. Samuel Burri is currently working in collaboration with the Quantum Architecture Laboratory of Professor Charbon to bring the Single-Photon cameras and associated techniques developed during his thesis to a wider range of academic and industrial customers.

Samuel Burri married in 2014 and currently lives in Potsdam, Germany. His interests include rapid manufacturing techniques that involve putting to use his 3D printers and laser cutter and software-defined radio.