LinoSPAD: A Compact Linear SPAD Camera System with 64 FPGA-Based TDC Modules for Versatile 50 ps Resolution Time-Resolved Imaging

Samuel Burri [ID], **Claudio Bruschini** * [ID] **and Edoardo Charbon**

Advanced Quantum Architecture Lab (AQUA), Microengineering Department, Ecole Polytechnique Fédérale de Lausanne (EPFL), 2002 Neuchâtel, Switzerland; samuel.burri@epfl.ch (S.B.); edoardo.charbon@epfl.ch (E.C.)

* Correspondence: claudio.bruschini@epfl.ch; Tel.: +41-21-695-4344

Abstract: The LinoSPAD camera system is a modular, compact and versatile time-resolved camera system, combining a linear 256 high fill factor pixel CMOS SPAD (single-photon avalanche diode) sensor with an FPGA (field-programmable gate array) and USB 3.0 transceiver board. This modularization permits the separate optimization or exchange of either the sensor front-end or the processing back-end, depending on the intended application, thus removing the traditional compromise between optimal SPAD technology on the one hand and time-stamping technology on the other hand. The FPGA firmware implements an array of 64 TDCs (time-to-digital converters) with histogram accumulators and a correction module to reduce non-linearities. Each TDC is capable of processing over 80 million photon detections per second and has an average timing resolution better than 50 ps. This article presents a complete and detailed characterization, covering all aspects of the system, from the SPAD array light sensitivity and noise to TDC linearity, from hardware/firmware/software co-design to signal processing, e.g., non-linearity correction, from power consumption to performance non-uniformity.

1. Introduction

Many time-resolved applications can benefit from a compact, versatile, and simple-to-use single-photon detector. They are beneficial when the timing characteristics of photons carrying important information, or when the overall number of photons is very low. While single-photon detectors, such as photomultiplying tubes (PMTs) [1] or multi-channel plates (MCPs) [2,3], have existed for decades, they are generally bulky, delicate, and limited in the number of pixels. CMOS SPADs (single-photon avalanche diodes) have existed since the early 2000s and they are on their way to replace PMTs and MCPs in many applications requiring high compactness, large pixels counts, high robustness and reliability [4,5].

A CMOS SPAD is an ordinary diode in a standard microelectronic circuit, which is reverse biased above its breakdown voltage. In the absence of charge carriers, the electric field is sustained without causing a large current flow. A single charge generated upon a photon absorption in the high electric field can trigger an avalanche through impact ionization. This avalanche of charges results in a macroscopic current, which is sensed and quenched by additional circuitry.

The timing response of single-photon detectors is one of the most important characteristics in many applications. A CMOS SPAD is capable, like a PMT, of generating an avalanche of electrons with a timing precision in the picosecond range from a single photon. The CMOS SPAD, however, can achieve this within an area of a few micrometers on a silicon chip and using a relatively low voltage

(a few tens of Volt). In addition, the output of a CMOS SPAD is compatible with the logic levels of electronic circuits, thus enabling the integration of many pixels, together with ancillary processing electronics, in a single chip.

SPADs have become increasingly useful in the imaging of fast, repetitive phenomena, such as fluorescence lifetime imaging microscopy (FLIM) [6] using pulsed illumination, or correlated photon detections with high time precision, such as in positron emission tomography (PET) [7] or fluorescence correlation spectroscopy (FCS) [8]. Any application requiring time-resolved statistics of possibly very few photons is indeed a possible target for SPAD imagers [9–11].

Linear arrays of CMOS SPADs started to appear after the introduction of SPADs in standard CMOS. Early examples are the works of Niclass, Sergio, and Charbon [12,13], which implement a 4 × 112 and 2 × 128 array with on-chip timing electronics. More recent developments include the 2 × 256 array of Krstajić et al. [14], the 2 × 128 array of Nissinen et al. [15], and the 8 × 1024 array by Maruyama et al. [16], targeting respectively time-resolved fluorescence and Raman spectroscopy. These examples include processing electronics in the sensor itself and produce streams of timestamps, histograms, and/or fluorescence lifetime information.

Recent advances in CMOS fabrication technology have made it possible to revisit the so-far mostly monolithic design of SPAD imagers. With shrinking feature sizes, more and more processing circuitry can be implemented in the same area, while the photosensitive diodes cannot be shrunk as much without sacrificing sensitivity and fill factor (ratio of photosensitive to overall pixel area). Established technologies have the advantage of an optimized fabrication with reduced defects in the silicon lattice, resulting in lower noise for SPADs, thus making it advisable to use well controlled and in comparison conservative fabrication processes for the sensor and the most advanced node possible for the processing. A possible step is 3D-stacking, which connects two chips at very high density, however this process is not yet widely available and does not offer the option of a quick (re-)design turnaround [17].

LinoSPAD combines a technologically conservative SPAD sensor front-end with an advanced processing back-end through a field-programmable gate array (FPGA), in a novel way. This modularization permits the separate optimization or exchange of either the sensor front-end or the processing back-end, depending on the intended application, thus removing the traditional compromise between optimal SPAD technology on the one hand and time-stamping technology on the other hand. The SPAD sensor chip implements a row of 256 pixels with integrated quenching in a 0.35 μm fabrication process. Through a carrier PCB, every pixel is connected to a Spartan 6 FPGA fabricated in a 45-nm process and providing enough processing resources to adapt the system to many applications [18]. However, even more advanced CMOS processes nodes will soon be available using the Artix-7 and other FPGA families.

The paper is organized as follows: Section 2 describes the camera system, covering the sensor and the FPGA architecture. Section 3 presents the detailed characterization of a single camera, while Section 4 reports variations across multiple sensors and FPGAs. Section 5 concludes the paper.

2. LinoSPAD: A Versatile SPAD Line Sensor

The LinoSPAD system is composed of the sensor, featuring a line of 256 SPAD pixels, and an FPGA acting as data processing and communication unit (DPCU) connected to a computer. Enabled by direct connections of SPAD pixels to FPGA inputs, the DPCU can implement processing functionality for a wide range of applications. This makes the LinoSPAD camera an ideal prototyping platform for SPAD-based systems.

2.1. Sensor Architecture

The SPAD line sensor, denominated LinoSPAD, is where photons are absorbed and converted into electronic digital pulses that are fed to the processing logic. The goal of LinoSPAD is to achieve high

fill factor, low noise, low jitter, and high photosensitivity, while maximizing its versatility by having only minimal circuitry for biasing and quenching photon-induced avalanches integrated on chip.

The fabrication of the sensor is done using the proven AMS 0.35 µm high-voltage process. A p+ to deep n-well diode is reverse biased to reach the high electric field needed to cause avalanches by impact ionization. The fill factor reaches 40% thanks to the optimized shape of the diode and the shared well for the cathodes. The complete on-chip pixel is composed of the SPAD diode connected in series with a quenching transistor and two inverters as shown in Figure 1b. We employ a passive quenching architecture with a single transistor acting as a non-linear resistance in series with the diode. The diode is biased at V_{OP}, which is V_{ex} above the breakdown voltage V_{bd}. The gate voltage V_Q of the quenching transistor influences the equivalent resistance and thus the current flowing through the SPAD. This current determines the SPAD dead time, i.e., the time that it takes for the SPAD to return to bias voltage V_{OP} after an avalanche. A typical value for the dead time is 100 ns. The inverters measure the voltage drop across the quenching transistor and decrease the SPAD output impedance to drive an output pad with a sufficiently fast response. The full sensor chip and the layout of two pixels are shown in Figure 1.

To accommodate the high number of required I/Os in a limited area, additional pads were placed inside a traditional pad ring of 192 elements. The sensor area measures 6.8 mm by 1.7 mm. The small size and conservative design led to very good fabrication yield, in that no defective pixels have been found yet.
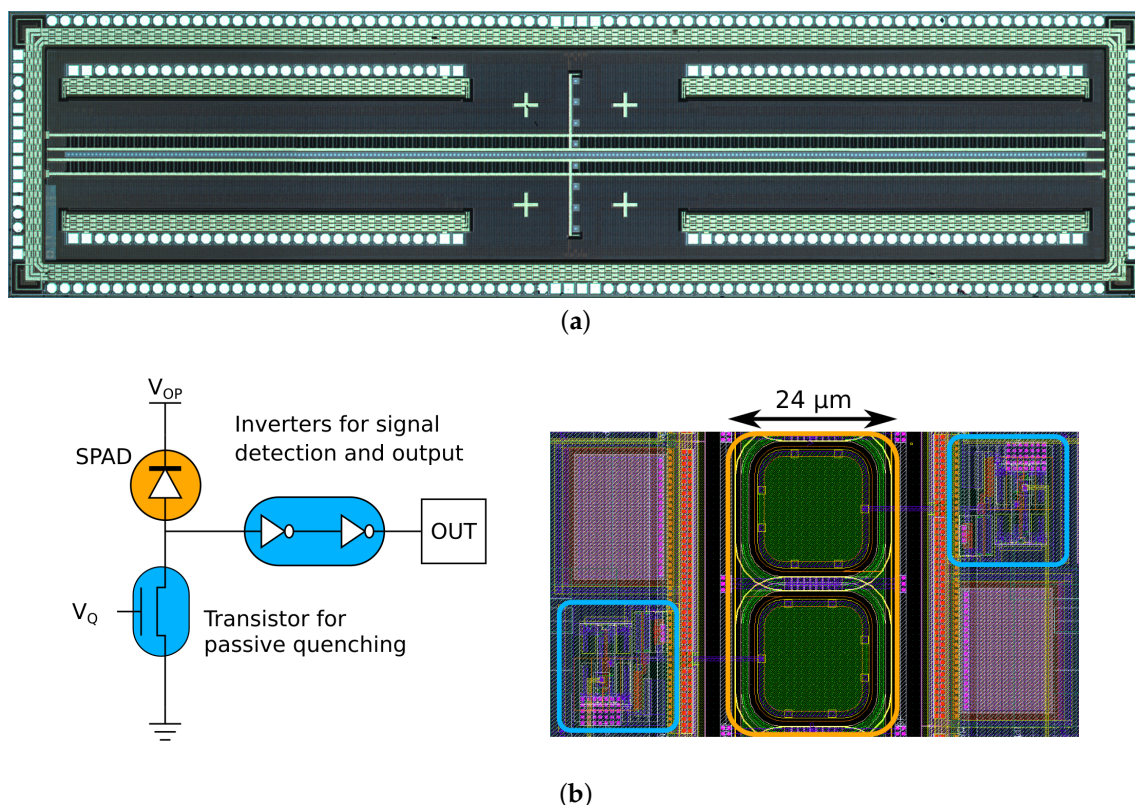

(**a**)

(**b**)

Figure 1. (**a**) Micrograph of the 6.8 mm by 1.7 mm LinoSPAD sensor. A total of 312 I/Os are laid out around the main line of pixels. Eight auxiliary pixels (with application specific pitch) and four alignment marks can be seen in the center; (**b**) Schematic of a single pixel and corresponding layout for a pair of pixels showing the alternating placement of pixel logic next to the densely packed diodes. The active area of the sensor corresponds to 40% of the highlighted SPAD (single-photon avalanche diode) area.

2.2. FPGA Interface Card

There is no readily available chip package supporting the pad layout of the LinoSPAD sensor. Therefore, we designed our own sensor PCB with a central pad for the chip and four rows of landing pads on the sides for the bond wires. This resulted in a PCB pin pitch of 160 μm, twice the pitch of the bonding pads on the chip, but still requiring PCB features of 50 μm. The fabrication of these PCBs poses interesting challenges for commercial PCB manufacturers. A close-up picture of the LinoSPAD daughterboard with visible bond wires is shown in Figure 2a. The wires are carefully protected by a resin while leaving the SPADs exposed to light.

The versatility of the LinoSPAD system depends largely on the FPGA motherboard, which connects the sensor to a host computer. We eventually decided to build our own flexible system when we could not find a readily available board satisfying our needs, i.e., provide: (a) enough available I/O to connect image sensors with wide parallel busses and a fast interface with a computer; (b) connectors for synchronization signals and bias voltages to simplify the setup of a camera; and (c) an expansion header to respond to future requests for a more powerful FPGA or increased bandwidth.
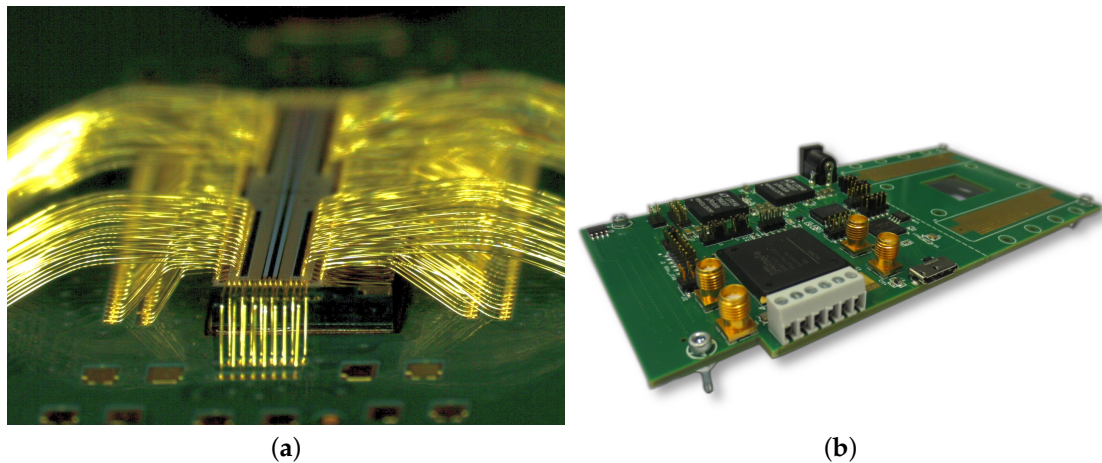

(**a**)　　　　　　　　　　　　　　　　　　　(**b**)

Figure 2. (**a**) Close-up of the LinoSPAD sensor glued to its PCB. Due to the high number of bond wires, the PCB layout was made following specifications from the bonding company. All pixels have at least a 90° aperture for incoming light (Image courtesy of Microdul AG, Zürich); (**b**) FPGA (field-programmable gate array) motherboard with two large 10 × 40 contact arrays around a hole foreseen for backside cooling. Two spring connector arrays connect the daughter PCB, which contains only few decoupling capacitors in addition to the sensor.

We decided to use a Xilinx® Spartan™ 6 FPGA, which offers a good trade-off between performance and cost. It is available with a high number of user I/O and sufficient logic elements to enable our foreseen applications. An FPGA with higher performance would offer faster logic and more memory, but at a prohibitive price point for one-off prototype system. The motherboard is shown in Figure 2b.

2.3. FPGA Architecture

The FPGA plays the most important role in the LinoSPAD camera system. It is where the signals of all pixels come together and have to be processed into the data stream sent to the computer. Each pixel in the sensor is generating electrical pulses upon the arrival of photons, carrying the timing information we want to digitize. The following subsections detail how this functionality is realized in the FPGA firmware.

2.3.1. Global Architecture

Single-photon detectors measure two main illumination characteristics, the number of photons and/or the time of their arrival. Counting is a trivial operation for digital circuits, whether the source signal is used directly as clock signal of the circuit or whether a sampling technique is employed. Both techniques have limits imposed by the maximum clock frequency and necessary waveform integrity. In contrast, measuring signal arrival with a precision better than the operating clock frequency is not trivial. As one of the distinguishing features of a SPAD sensor is a time-response in the picosecond range, we wanted to realize a circuit that exploits this precision as much as possible, within the limits imposed by an FPGA architecture.

We fixed the number of independent TDC modules to 64, as this is the largest number one can realistically pack in the given FPGA. Since we could not implement a TDC for each pixel to count photons in parallel, we added simple count registers in parallel. In addition, the TDC modules were optimized for nearly continuous operation such that data transmission does not further reduce the usable recording time for pixel data.

Figure 3 shows the FPGA firmware architecture implemented as a result of our considerations. Different modules are connected as slaves on a bus, linking them with a host computer. Data are exchanged through a bidirectional 32-bit FIFO running at 100 MHz. The main blocks of the firmware are the clock generation and synchronization circuitry, which is largely independent from the processing blocks consisting of synchronous sampled counters and the array of TDCs. The FPGA operates in response to requests received over the USB interface by performing data acquisition and processing, and sending results back to the computer. The following subsections detail the implementation of the main modules.
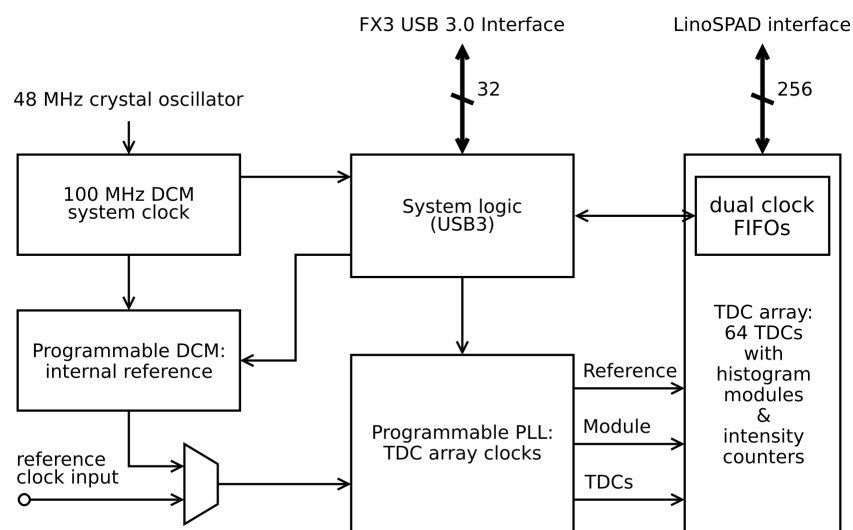

Figure 3. The LinoSPAD camera firmware is composed of two major subsystems with their state-machines controlled from a USB interface. The clock control part is responsible for generating the system clocks and synchronizing an illumination system. The time-to-digital converter (TDC) array, which contains the delay lines, histogram generation and post-processing, interfaces with the SPAD sensor and processes the pixel signals.

2.3.2. Clock Architecture

The capabilities and flexibility of the clock and synchronization in the LinoSPAD camera are dictated by the PLL (phase-locked loop) blocks available in the FPGA. The delay-line of the TDC needs a fixed clock frequency of 400 MHz to function properly and any timing information it generates is based on that clock. A PLL in the FPGA is programmed to generate the sampling clock from a lower

frequency reference that can optionally be sourced from an external clock generator or illumination system. It is also possible to use a clock derived from a crystal on the FPGA PCB to provide a reference output for external circuits.

From a base clock between 20 MHz and 100 MHz, the FPGA PLL generates a sampling clock running at 400 MHz and a slower clock for the processing and memory blocks in the TDC module. A fixed 100 MHz clock is used for the USB transceiver communication and the SPAD event counter module.

2.3.3. Event Counter Array

The event counter array consists of a synchronous event counter for every pixel. The counters work using the output line state of pixels sampled at 100 MHz: one counter increments every time the line is seen active, while another increments only for a 0-to-1 transition of the line. Together, the counters can be used to read the illumination on each pixel and to detect saturation conditions when the line stays active for extended periods. To detect such saturation conditions, it is necessary to observe the average pulse length obtained by dividing the active counter values by the transition counter values. The average pulse length for a detection depends on the excess bias voltage, and the quenching voltage, and is comprised between 40 ns and 400 ns.

The counters are implemented as 8-bit registers backed by 32-bit memory. The controlling state-machine switches trough the pixel counters and updates the memory locations with new values every 256 clock cycles. During the update, there is the option to reset the counters and to send their current values to the USB transceiver.

2.3.4. TDC Core

The TDC core implemented in LinoSPAD can process a rate of over 80 million events-per-second and of generating time-stamps with an average resolution (LSB) better than 25 ps. The firmware contains 64 TDCs, each shared sequentially among 4 pixels. The TDCs are connected to histogram accumulation logic described later. Figure 4 shows a schematic of the TDC array together with shared post-processing. The main parts of the TDC are the delay line and the encoder.
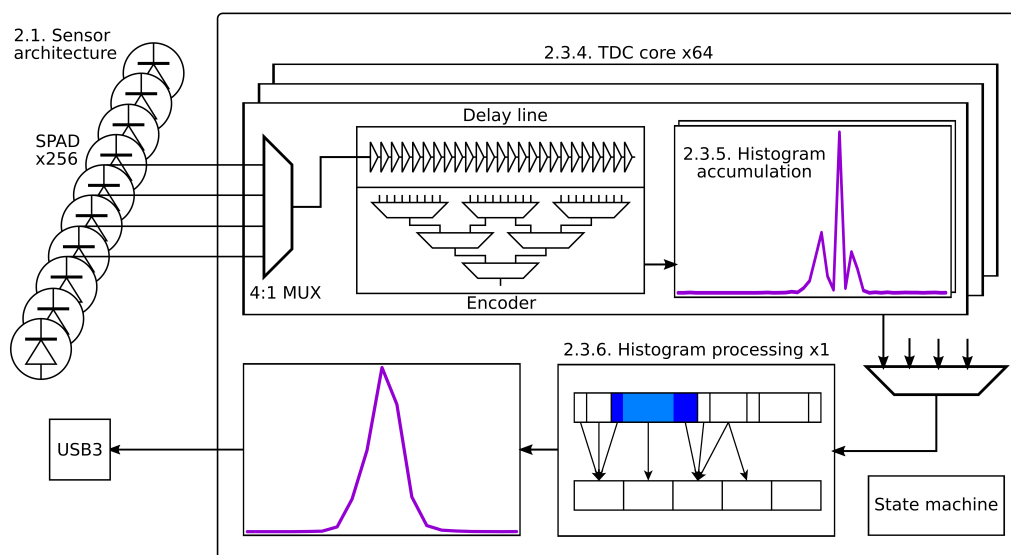

Figure 4. TDC array block detail showing the 64 delay lines with encoders and histogram accumulation engines. A state-machine is used to switch the multiplexer from one pixel to the next and sequence the readout of the accumulated histograms. The histogram engines share the post-processing, which is implemented in the path between the memory and the USB transceiver to process the histograms in real time while they are transferred to the computer. (Labels include section numbers where applicable).

Delay line: The TDC delay line uses the carry-chain structures of the FPGA logic blocks similar to previous designs [19–22]. These structures are usually employed to implement fast arithmetic circuits and provide the fastest connections inside the FPGA. Incidentally, they are the only routing structure exposed to the programmer of the FPGA. To exceed the timing precision of the sampling clock, a signal needs to be routed along a defined path where it reaches a variable number of registers depending on its time-of-arrival relative to the clock; upon freezing the associated memory elements one can evaluate the wanted time difference during the following clock period.

In Spartan 6 FPGAs the carry logic is exposed as primitive of 4 bit length, which contains dedicated connections to form longer chains. Each bit is connected to a flip–flop where a clock signal registers its state. Ideally, the clock signal reaches all registers at the same time and the register inputs are stable. Unfortunately, these conditions are not guaranteed and neither is the internal structure of the carry element, which does not correspond to a regular ripple carry chain. Taken together, these non-idealities cause a high non-linearity in the observed delays from one output state of the delay line to the next.

The TDC used in LinoSPAD takes the non-linearity of the delay line into account through its encoder and guarantees a monotonic increasing code with increasing time-of-arrival of the event, as well as no missing events. This is achieved by using a delay line with a length, which is always longer than the 2.5 ns sampling period, and an encoder that effectively implements a bit population count for valid samples. The Spartan 6 datasheet [23] specifies a delay of 80 ps for a 4-bit carry element, which results in a single bit flipping every 20 ps. Due to the non-idealities mentioned above, the observed delays are very non-uniform, ranging from almost 0 to 100 ps. However, the observed mean delay for one bit is below 20 ps, according to the datasheet values. Furthermore, it turns out that 35 carry elements, corresponding to 140 bits, are enough to resolve 2.5 ns in all operating conditions.

Encoder: We use an encoding module running at 400 MHz and optimized for the FPGA architecture to convert a sample of 140 bits into a binary code. A thermometer-to-binary conversion is employed, with two modifications: the first is an overlapping of the blocks to account for occasionally observed bubbles in the delay line, whereas a bubble occurs when a schematically later bit flips before an earlier one because of an optimized carry block implementation or clock skew [24,25]. The second is a decomposition in three blocks with two decision bits to better map the circuit into the 6-input lookup-tables available in the Spartan 6.

The encoder is fully pipelined to run at 400 MHz and combines a first stage of modified thermometer-to-binary conversion with a conventional population count to satisfy the conditions outlined above. For each clock cycle producing a 140-bit sample of the delay line, the encoder produces an 8-bit code representing the number of active bits, which relate the signal arrival time to the sampling clock.

2.3.5. Histogram Accumulation

The histogram accumulation logic receives its input from the delay line encoder running at 400 MHz. The raw code is extended using a slower coarse counter to obtain a 28-bit code that can resolve approximately 4.8 ms. The coarse counter counts in steps of 140 for every 2.5 ns sampling period and is reset synchronously to the reference clock of the system. This reference has a period, which is a multiple of 2.5 ns and is usually synchronized with the illumination system.

A histogram is accumulated in the FPGA memory to store a stream of timestamps. The possible histogram size is limited by the available memory capacity, e.g., 1024 bins with 16-bit resolution, which is sufficient to record full period histograms up to a length of 18.2 ns. With the possibility to discard two LSBs of a timestamp, effectively compressing the timing information by a factor of four, the maximum period for a histogram can be extended to 72.8 ns.

An alternative firmware mode uses the histogram accumulation memory to directly store up to 512 timestamps with 28-bit resolution for advanced applications. The post-processing part of the firmware is not available in this configuration. When histogram accumulation is completed, the memory is read out and reset, while the following accumulation uses a second memory.

2.3.6. Histogram Processing

Due to the inherent non-linear transfer characteristic of FPGA TDCs, the histograms produced in our system usually need to be corrected before they can be processed further. To make our system more useful and enable real-time applications, we implemented a statistical post-processing module to linearize in real time the recorded time-of-arrival histograms.

The first step of this approach consists in measuring the raw TDC characteristics. Using non-time-correlated illumination (or sensor noise), we collect histograms for each TDC without applying any correction. The TDC construction guarantees monotonic increasing codes and no missing events. We use this information to calculate the delay and offset of each code inside a sampling period of 2.5 ns.

The input histogram H_{in} has $k = 140$ bins for a 2.5 ns TDC period (τ_{TDC}). We assign the size $S_{\text{in},i}$ and position $P_{\text{in},i}$ for each code as calculated from the histogram counts $C_{\text{in},i}$:

$$S_{\text{in},i} = \tau_{\text{TDC}} \times \frac{C_{\text{in},i}}{\sum_{j=0}^{k-1} C_{\text{in},j}} \tag{1}$$

$$P_{\text{in},i} = \sum_{j=0}^{i-1} S_{\text{in},j} \tag{2}$$

for i from 0 to $k - 1$ raw input bins.

Afterwards, we calculate the weights needed to generate a uniform histogram of N bins with sizes $S_{\text{out},i} = S_{\text{out}} = \frac{\tau_{\text{TDC}}}{N}$ and positions $P_{\text{out},i} = i \times \frac{\tau_{\text{TDC}}}{N}$. The weights of the (sparse) $N \times k$ correction matrix M are given by

$$m_{i,j} = \max\left(0, \frac{\min\left(P_{\text{in},j} + S_{\text{in},j}, P_{\text{out},i} + S_{\text{out}}\right) - \max\left(P_{\text{in},j}, P_{\text{out},j}\right)}{S_{\text{in},i}}\right) \tag{3}$$

The calculated weights of M are given by the overlap between the input and output histogram bins corresponding to the position in the matrix. The processed histogram H_{out} is then given by the multiplication:

$$H_{\text{out}} = M \times H_{\text{in}} \tag{4}$$

Figure 5 illustrates the histogram processing implemented in LinoSPAD. A histogram from the recorded TDC codes is processed into a (usually shorter) histogram with uniform bin size. The matrix multiplication is implemented in real time by considering the input bins one after the other and calculating the corresponding output bins. The implementation is helped by the fact that the correction matrix is sparse due to the monotonic nature of the input histogram. The processing has been implemented such that the number of events in the output histogram corresponds to the number of events in the input histogram. The correction matrix is stored in a reduced representation, exploiting the fact that the columns add up to one. This allows an on-the–fly calculation of the remaining values. The arithmetic uses an 8-bit fixed-point representation for the matrix elements.
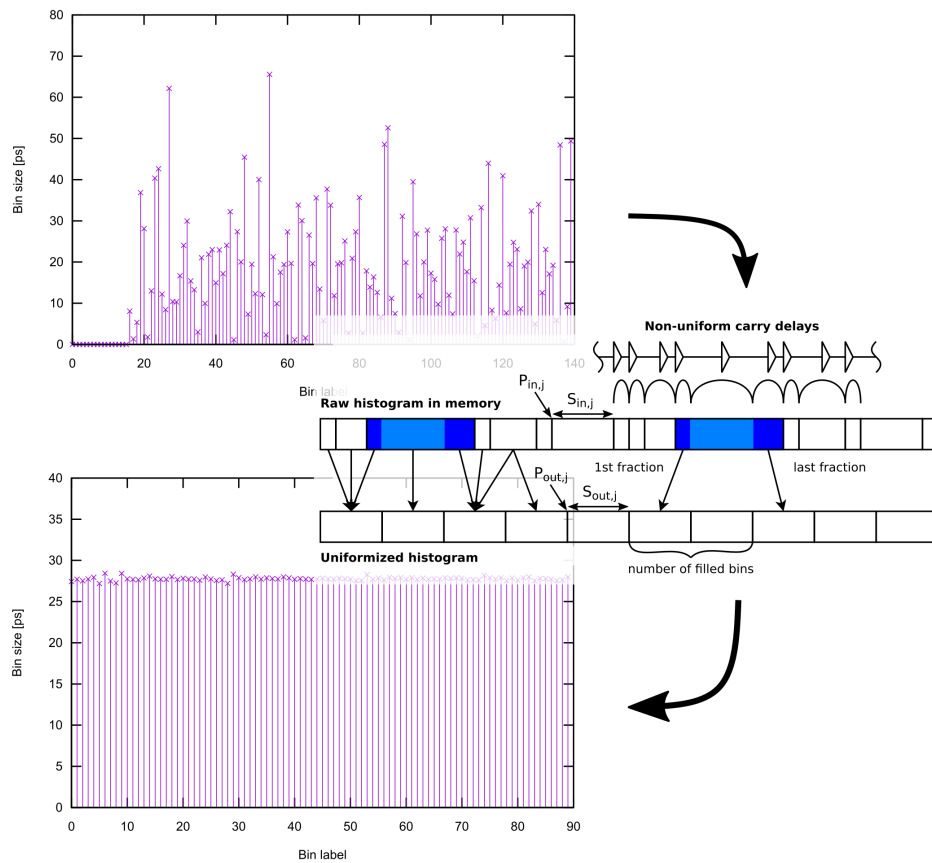
Figure 5. Two density measurements before and after histogram equalization with a schematic representation of the post-processing. The post-processing module is programmed with a compressed representation of the correction matrix for each TDC such that it is capable to correct non-linearities during readout in real-time. The most prominent non-linearity in the input histograms corresponds to unused bins from code 0 to the earliest samples. They occur because the delay line is longer than the sampling period, such that not all codes are used. Each TDC has a different number of unused codes.

2.4. FPGA Implementation

The implementation of a large number of TDC modules in an FPGA is challenging due to the large size of the circuit and the high clock rate needed to keep the circuit from becoming even larger. During design exploration for LinoSPAD, we needed to reduce the amount of logic running at 400 MHz and to isolate it as well as possible to minimize switching noise into adjacent TDCs. The number of routes connecting different clock domains should also be kept small, especially for fast frequencies, because uncertainties impose even tighter timing on the corresponding paths.

When exploring how to extend our single TDC firmware to a 64 TDC module implementation, we rapidly faced increasing real estate and compilation time limits. An effective approach has been to decouple those segments of the firmware running at 400 MHz from those running at slower speeds. Using design partitioning, we could fix the 400 MHz segments of the TDCs and allow the full design to synthesize and pass timing closure. More detail on the implementation is provided in [26].

3. LinoSPAD Characterization

In the characterization of LinoSPAD, unless stated otherwise, we used a quenching voltage of 1 V and an excess bias voltage adjusted to account for differences in breakdown voltage across multiple chips.

3.1. Breakdown Voltage

The measurement of the breakdown voltage of a SPAD sensor needs to be carried out before most other characterization measurements, given that most metrics depend on its precise value. The breakdown voltage of LinoSPAD is estimated for each pixel by using the excess noise method. The bias voltage is swept in increments of 5 mV and the dark count rate (DCR) is measured in each pixel. The sweep ends 200 mV above the point where the last pixel turns develops non-zero DCR. A measurement period of 5 s at each voltage step is used. The breakdown voltage is then extracted from a fit using a two-piece linear function, and subtracting 0.6 V to take into account the digitizing inverter threshold [27].

A typical distribution of pixel breakdown voltages on a sensor is shown in Figure 6a. A standard deviation below 100 mV is commonly observed. Variations between multiple chips are more important, as will be discussed later.
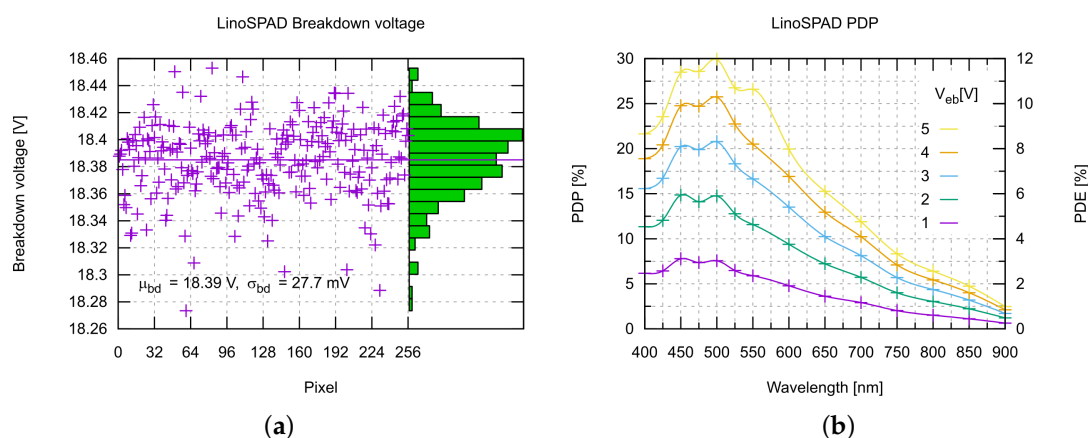

(a)

(b)

Figure 6. (**a**) Per-pixel breakdown voltage obtained using the excess noise method. In-chip standard deviation is typically below 100 mV; (**b**) Photon-detection probability and efficiency (PDP, PDE) for LinoSPAD in the wavelength range 400–900 nm measured for increasing excess bias using an integrating sphere and reference photodiode.

3.2. Photon Detection Probability (PDP) and Photo Response

The PDP of LinoSPAD was measured with two different approaches: the first was based on the use of an integrating sphere (Oriel/Newport part 77250) and monochromator (Oriel/Newport part 819D-SL-2) to measure the PDP over a wavelength range of 400 nm to 900 nm, through comparison with a reference photodiode (Hamamatsu part S1226-BQ). In this measurement only a few pixels with noise levels close to the median DCR value were taken into account and the DCR was subtracted to obtain the final result shown in Figure 6b. Taking into account a fill factor of 40%, the rightmost y-axis shows the photon detection efficiency (PDE).

The second method is based on the use of two LEDs with narrow spectra around 465 nm and 640 nm, respectively. The illumination was calibrated using a reference photodiode (Thorlabs S130C) to provide intensities from 1 to 10 μW/cm^2 across the area of the sensor. The high photon rates help reducing uncertainty in the reference calibration, but can lead to sensor saturation.

Two measurements with 465 nm illumination are shown in Figure 7. The first one sweeps the excess bias voltage from 0.5 V to 5 V at constant illumination intensity of 1 μW/cm^2, while the second one increases the illumination intensity up to 10 μW/cm^2 at a constant excess bias voltage of 2 V. The measured count rates are corrected for the individual pixel DCR values and illustrate the wide usable range of the LinoSPAD sensor.
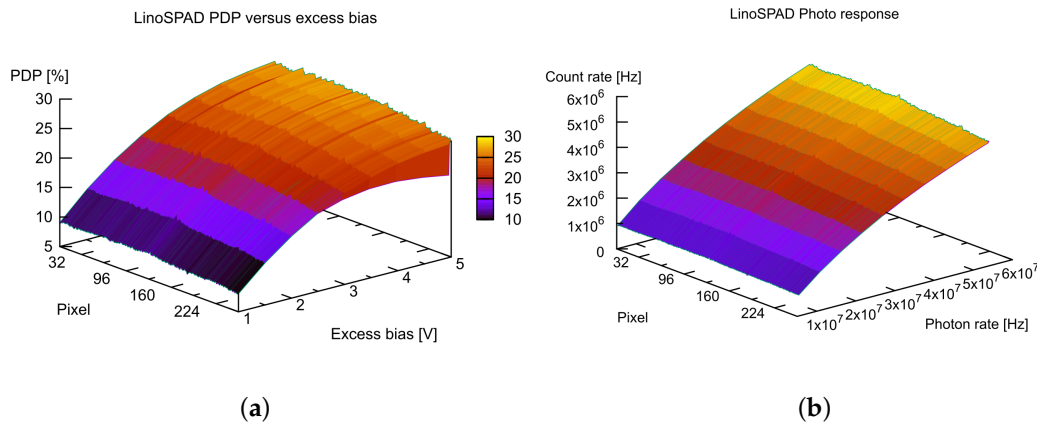
(**a**)   (**b**)

Figure 7. (**a**) PDP versus excess bias voltage; and (**b**) count rate versus photon rate (photo response). Both are recorded using a narrow spectrum LED at 465 nm. In (**a**), the illumination intensity is constant at 1 µW/cm^2 for excess bias voltages from 0.5 V to 5 V; while, in (**b**), the excess bias is constant at 2 V for illumination intensity levels between 1 µW/cm^2 to 10 µW/cm^2. Count compression occurs for the highest excess bias voltages and photon rates, where pixels reach saturation and the observed count rate decreases.

3.3. Dark Count Rate (DCR)

DCR denotes the measured event rate in the dark. DCR measurements for LinoSPAD are shown in Figure 8. Two aspects were characterized: first, we measured the median noise level at temperatures ranging from −40 °C to 80 °C using a thermal test chamber; and, second, we measured the noise distribution for different excess bias voltages at room temperature.
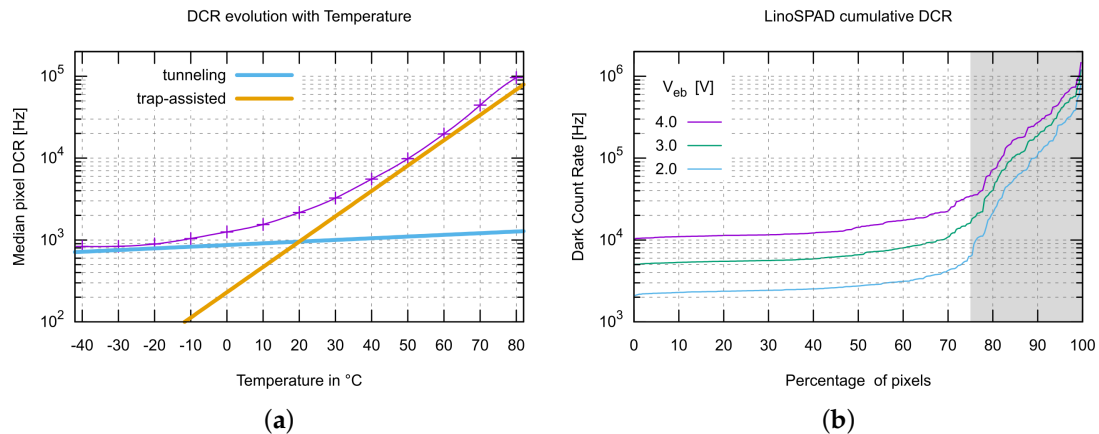

(**a**)   (**b**)

Figure 8. (**a**) Evolution of median dark count rate (DCR) when the sensor temperature varies from −40 °C to 80 °C. These values were measured using 2 V excess bias voltage; (**b**) The noise is recorded for excess bias voltages between 2 V and 4 V at room temperature. LinoSPAD typically has about 25% of pixels where the DCR exceeds the median value by an order of magnitude. However, no spatial pattern is present in the noise distribution.

The DCR versus temperature shows a well-known [28] characteristic for SPADs, whereby noise no longer decreases exponentially below a certain temperature due to the dominance of band-to-band tunneling over trap-assisted noise. For LinoSPAD the corresponding cut-off temperature is around 20 °C. Depending on environment temperature and activity, cooling can therefore be beneficial.

The spatial distribution of the noise and its dependence on the excess bias voltage are comparable to other SPAD sensors manufactured in the same technology [29]. LinoSPAD has a proportion of about 25% hot pixels, which have a noise level 10 times that of the median. A possible explanation for the higher proportion compared to previous sensors is the larger size of SPADs and square diode shape, which both increase the defect rate in silicon diodes.

Nevertheless, even the "hottest" pixels observed in LinoSPAD retain enough dynamic range to be used in practical applications, and no clustering of defects of any kind have been found.

3.4. Power Consumption

The power consumption measurement is the first characterization taking into account the combination of SPAD sensor and FPGA interface boards. The power was measured using a 1 V quenching voltage (V_Q) and 2 V SPAD excess bias voltage (V_{ex}). The sensor logic runs at 3.3 V, the FPGA board with voltage regulators is supplied with 5 V. We measured the system current flowing through the SPADs (V_{OP} voltage supply), the sensor (I/O) logic (V_{DD} voltage supply) and the FPGA system (V_{REG} voltage regulator supply) separately for different illumination conditions. Table 1 lists our results.

The FPGA has a power consumption of around 5 W regardless of chip activity, mainly because of the high clock rates and large design. The power increases only minimally with increasing switching activity due to photon detections. The sensor itself, however, features a power consumption, which is very closely related to the switching activity as the current is predominantly used to drive the output pads. From few milliwatts in the dark, the power rises to approximately 2 W for maximum activity, to then drops below the dark current consumption level when the sensor saturates and switching ceases completely. The current on the SPAD bias follows directly the illumination profile as photon incidence allows it to flow through the diodes.

Table 1. Current consumption for a LinoSPAD camera; the excess bias voltage was set to 2 V and the quenching voltage to 1 V. V_{OP} supplies the SPADs, V_{DD} the sensor I/O, and V_{REG} the power regulators on the FPGA PCB.

Condition/Supply	V_{OP} (2 V_{ex})	V_{DD} (3.3 V)	V_{REG} (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

3.5. TDC Response

The characterization of the TDC response consists in the measurement of the transfer curve relating arrival time relative to the sampling clock to the corresponding output value. This is similar to a transfer curve in an analog-to-digital converter that relates input voltage to output value, and thus uses the same metrics.

Differential non-linearity (DNL) measures the deviation of the actual step size from the ideal step size given by dividing the full range by the total number of steps. Integral non-linearity (INL) measures the deviation of the actual transfer curve from the ideal transfer curve. It corresponds to the integral of the DNL between the first and the current output code.

The TDC design in LinoSPAD ensures sound limits for both DNL and INL. The DNL is bounded by −1 with the guarantee that codes are monotonic, and the final value for the INL is zero with the guarantee that the full input range (clock period) is represented in the output codes.

To measure the TDC non-linearity we need to apply an input with a uniform distribution over the measurement range. In the case of LinoSPAD, this is achieved by using a non-correlated illumination source or dark noise. A histogram of TDC codes with uniform input distribution is also called density measurement. The histogram is interpreted as DNL after scaling the *y*-axis using the total number

of counts and the ideal bin size. A typical DNL of a TDC is shown in Figure 9a, where unused bins before code 0 have been removed.

Integrating the DNL in Figure 9a results in the typical INL shown in Figure 9b. The INL of uncorrected TDCs has a large negative excursion for low codes because of the unused bins. Here, it is already adjusted to exclude such unused bins, yet it still reaches almost −5 LSB. As explained, the final value is the ideal 0, as the full input range is present in the output.
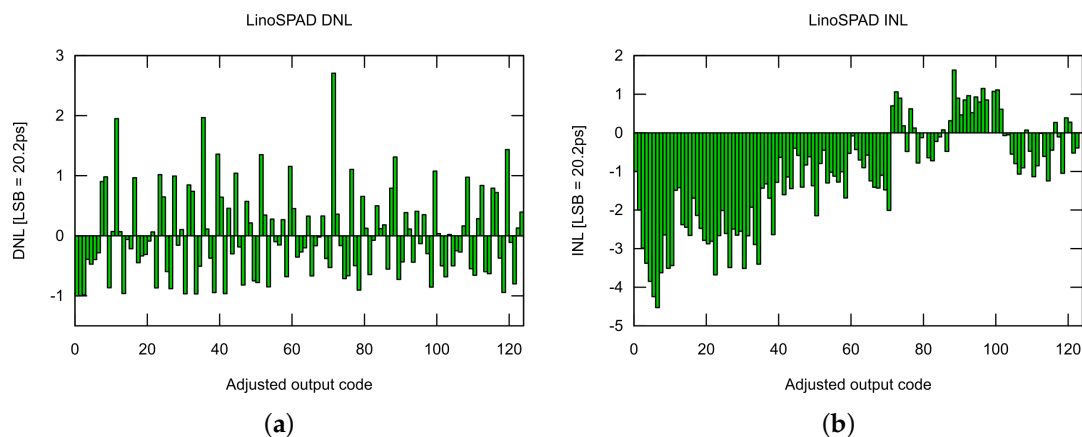

(**a**)   (**b**)

Figure 9. Uncorrected: differential non-linearity (DNL) (**a**); and integral non-linearity (INL) (**b**) characteristics of an FPGA TDC. Completely unused bins have been removed from the plots, resulting in a least-significant bit (LSB) of 20.2 ps. The first codes are rarely used, leading to an unusual negative excursion in the INL.

3.6. Temperature Effects in SPAD Sensors

Not only the sensor DCR is affected by temperature changes, but also the signal delays in the FPGA. Changes in temperature, caused by changes in switching activity, therefore induce a variation in the TDC characteristics [30]. Switching activity, in turn, is affected heavily by changes in the sensor illumination.

The FPGA on the LinoSPAD boards is usually equipped with an aluminum heatsink and a fan to lower the operating temperature and effectively dissipate approximately 5 W. Figure 10a shows the temperature evolution measured on the FPGA heatsink as the sensor illumination is increased. The heatsink temperature increases from 33 °C to 38 °C when moving from a combined 10 MHz event rate to 2 GHz of total counts.

Figure 10b shows the evaluation of the required delay line length to cover a period of 2.5 ns when the count rate, and therefore the temperature, increases. The average number of bits needed is almost constant at 124, but we can clearly observe delays becoming larger as the number of bits needed to cover 2.5 ns decreases with higher temperatures.

The data illustrate the importance of calibrating the LinoSPAD camera and the need to create a stable operating environment for precise measurements. The reported temperature measurements have been performed on the heatsink of the FPGA. Nevertheless, locally in the FPGA, temperature changes can occur much faster, quickly distorting measurements with fluctuating illumination conditions.
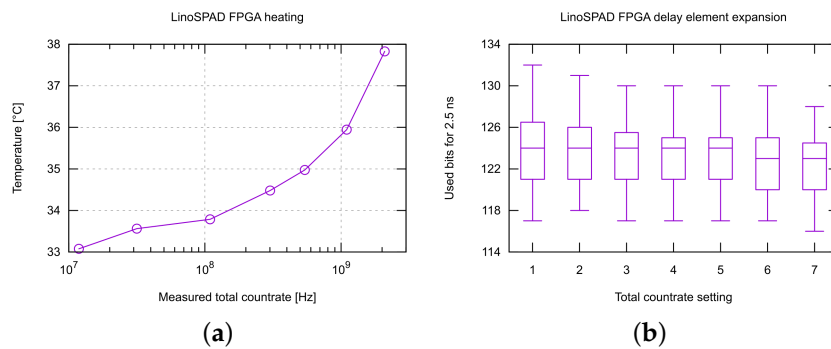
Figure 10. (**a**) Seven different illumination conditions for the sensor with their corresponding total count-rate and FPGA temperature; (**b**) Number of delay bits used under the same conditions for all the TDCs on the FPGA. We observe a decrease of used bits (longer delays) for higher temperatures.

3.7. Post-Processing

Histogram post-processing results are evaluated in the same way as for the TDCs, i.e., by measuring the resulting DNL and INL. The correction module is programmed with the transformation matrix defined in Section 2.3.6, calculated from many events gathered under stable illumination conditions. The data to post-process are afterwards gathered under the same conditions and analyzed.

The implemented matrix multiplication works as statistical correction using integer values and limited precision for the multiplication factors. As input histogram, we used a 12.5 ns period, spanning 700 bins, which are remapped to 450 bins, resulting in an ideal LSB of 27.7 ps. The FPGA module uses 8-bit matrix coefficients and a rounding that ensures that no counts are lost.

Figure 11 shows the result from evaluating 100 corrected histograms of about 100 k events each. For each bin in the output, we calculated the DNL and INL and present their span and mean values in Figure 11a,b, respectively. Looking at these results, we see that the FPGA correction can produce a TDC transfer curve with high linearity, yet this is largely dependent on the quality of the calibration measurements used to calculate the correction. Even small temperature changes in the FPGA, caused by changes in the event rate, will degrade the correction performance. As delay lines expand or shorten, the linearity of low codes degrades faster than that of high codes as a result of the TDC architecture. In realistic applications, however, the effect of short-term event rate changes is likely to be low-pass filtered (thermal inertia). Periodic recalibration should be envisaged for longer-term measurements, possibly coupled to a temperature control/stabilization system.
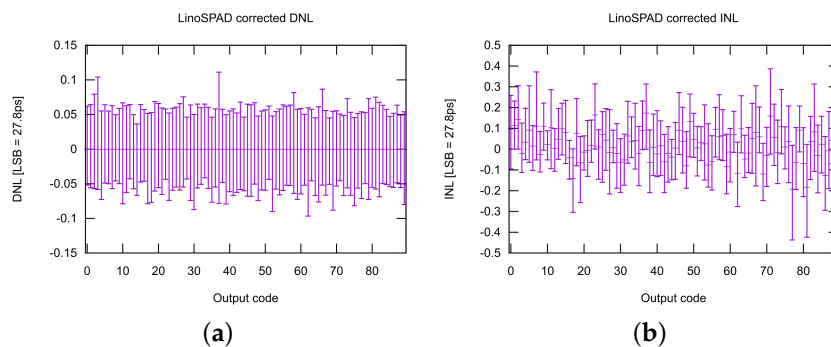

Figure 11. DNL (**a**); and INL (**b**) distributions across all output codes of a TDC when using the correction module after calibration. From 100 sample histograms, the plots show mean, minimum and maximum linearity values. The expected DNL variation from shot noise amounts to roughly 1/3 of the total measured variation.

3.8. Histograms

The verification of the correctness of histogram creation and management was achieved using time-correlated single-photon counting (TCSPC) techniques. The illumination is achieved using a 650 nm low power (5 mW) laser diode driven from a custom pulse generator using bipolar transistors [31]. Biasing the laser diode close to emission, we apply short current pulses to cause momentary population inversion and emit a laser pulse in the picosecond range. This illumination system is placed facing the chip without any optics and is synchronized with the FPGA.

We configure the FPGA firmware to generate a synchronization signal at 80 MHz to drive the illumination and measure arrival times relative to the 12.5 ns pulse period. Using calibration data obtained from a LED illumination of the same intensity, we program the correction module to remap the 700 bin raw histograms to 450 bin uniform histograms.

Figure 12 shows the resulting histograms for each pixel as transferred from the FPGA. A second processing module in the FPGA rotates the histograms one by one to align the peak positions at the center of the illumination period. From the flash illumination on the sensor, which suppresses the characteristics of the light pulse beyond the rising edge, we capture histograms with a mean FWHM below 100 ps. This result covers the full system including clock source, reference generation, illumination, detector and electronics, illustrating the level of precision, which is attainable when using FPGA PLL circuits for TDC clocking.
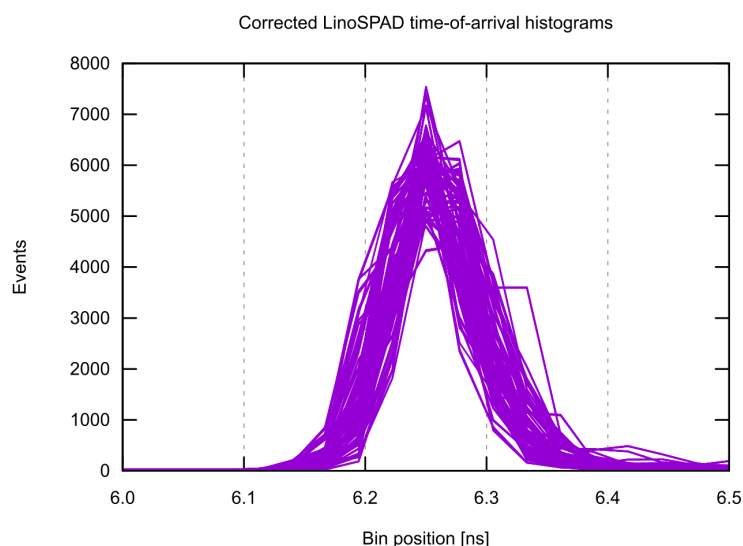

Figure 12. Calibrated FPGA response to a laser pulse shown for each pixel. The mean full-width at half-maximum (FWHM) of these histograms is below 100 ps.

3.9. Performance Summary and Comparison to other FPGA TDCs

The main characteristics of the LinoSPAD system are listed in Table 2. The unique feature of the LinoSPAD system is the combination of a minimal SPAD detector front-end with an FPGA processing back-end. To our knowledge, this has not been achieved previously to the extent presented in this paper. The FPGA characteristics apply to the current default firmware and can differ slightly for future updates or changes, but provide a solid baseline of what is to be expected from the system.

Table 3 adds a comparison with other FPGA based TDC systems and selected ASIC (application specific integrated circuit) TDCs. Generally, FPGA based TDCs have lower timing resolution and are affected by non-linearity due to FPGAs not being optimized to implement TDCs. On the other hand, they can have as many channels as fit in the logic array and a high number of I/Os to support

them. ASICs meanwhile have greater timing precision, but reduced number of channels due to I/O limitation and increasing integration complexity.

Table 2. LinoSPAD performance summary.

Parameter	Value
Chip size	6.8×1.68 mm^2
Technology	0.35 μm 4 M high-voltage
Number of pixels	256 (+8 perpendicular)
Pixel pitch	24 μm
Fill factor	40%
Dead time (V_{ex} = 3 V, V_Q = 1 V)	100 ns
Median DCR (V_{ex} = 3 V, 20 °C)	5–10 kHz
Spectral range (V_{ex} = 3 V, PDP > 5%)	400–775 nm
Light incidence	$\pm45°$ 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. LinoSPAD TDC comparison. The stated INL/DNL peak-to-peak ranges are uncorrected values. Technology key: S6, Spartan 6; V5, Virtex 5; V6, Virtex 6; C, CMOS ASIC; A, Unspecified ASIC.

		LinoSPAD [19]	[20]	[21]	[32]	[33]	[34]
Technology		S6	V5	V6	S6	C	C	A
Process	(nm)	45	65	40	45	350	130	-
# of TDCs		64	1	160	1	1	1	8
# of channels		256	1	160	1	1	1	8
LSB Resolution	(ps)	25	17	10	26	0.61	1	1
Event rate	(MHz)	133	300	300	125	0.8	10	12.5
Carry blocks	(TDC^{-1})	35	50	40	124	n/a	n/a	n/a
Clock frequency	(MHz)	400	300	600	125	200	80	n/a
DNL$_{pk-pk}$	(LSB)	4	4.6	3	n/a	0.5	0.8	0.04
INL$_{pk-pk}$	(LSB)	7	5.6	6	n/a	9	1.6	n/a

4. Extended Non-Linearity Characterization and Fabrication Variations

This section discusses the characterization of LinoSPAD with respect to dead time, afterpulsing and crosstalk, before considering variations across multiple TDCs in an FPGA and across multiple systems. We had the opportunity to characterize a small series of 10 systems using identical procedures.

4.1. Dead Time and Afterpulsing

LinoSPAD uses a passive avalanche quenching mechanism in the form of a variable resistor, implemented as channel of a transistor, which in turn determines the dead time of the SPAD. The dead time becomes shorter with a higher quenching transistor voltage, allowing a higher current to flow through the SPAD. At the same time, the fast quenching and recharge of the SPAD bias increases the probability that charges become trapped and result in afterpulses, which are false events correlated with an earlier event. Afterpulses are caused by the release of trapped charges after the SPAD has been recharged and result in skewed inter-arrival statistics for SPADs, particularly at short dead times.

Figure 13 presents the measurements performed to evaluate the dead time and afterpulsing probability in LinoSPAD. Figure 13a shows the dead time, i.e., the shortest inter-arrival time, and the afterpulsing probability when sweeping the quenching voltage from 0.6 V, where the quenching transistor is barely conductive, to 1.5 V, where the equivalent resistance is almost zero.

Figure 13b shows histograms of inter-arrival times for non-correlated illumination and increasing quenching voltage. The afterpulsing probability is computed from these histograms, knowing that uncorrelated illumination should result in exponentially distributed inter-arrival times due to the Poisson nature of arriving and detected photon statistics. Therefore, afterpulsing is visible in a histogram as extension above an exponential fit.

Crosstalk between pixels can be evaluated in the same way using non-correlated illumination. In the absence of correlated noise, the arrival time distribution between two pixels follows an exponential decay. Using this method, no significant crosstalk could be measured.
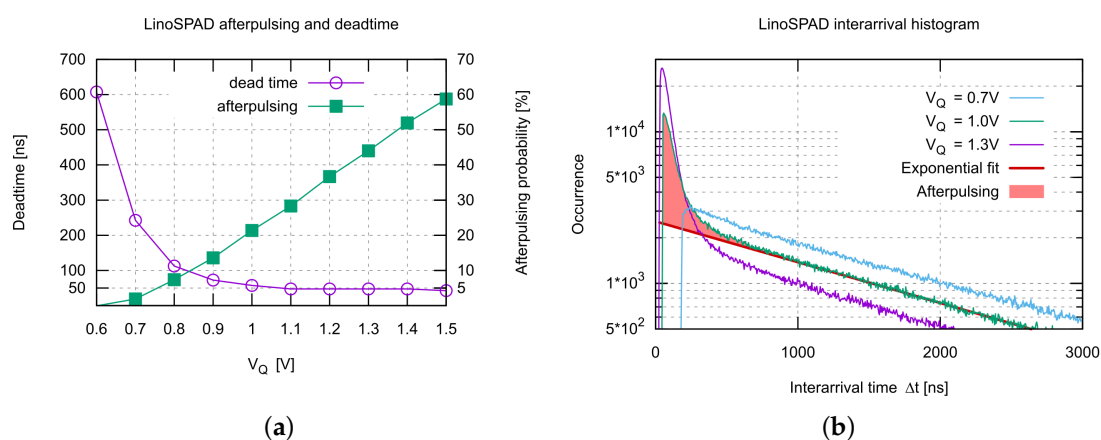

Figure 13. (**a**) Relationship between dead time and afterpulsing for the passively quenched LinoSPAD sensor; and (**b**) corresponding inter-arrival time histograms with clearly visible afterpulsing artifacts for higher quenching voltages, resulting in lower resistance and lower dead time.

4.2. TDC-to-TDC Variation

The LinoSPAD firmware features 64 TDC blocks. The FPGA datasheet specifies a limit on the total delay of a carry block, but nothing is specified about uniformity. Other important timing values to create TDCs, such as clock skew and path delays in the logic blocks, are only provided implicitly in the timing verification tool of the synthesis toolchain.

We therefore decided to measure the variation between TDCs in a single FPGA to have an experimental verification of the actual LSB values. The measurements we carried out are similar or identical to the ones reported in the previous section, but now analyzed to compare the performance of different TDCs.

Figure 14a presents the evaluation of bin sizes for the delay line in each TDC. We observe that about 75% of the bins feature a size below 30 ps, but among the remaining 25%, there is almost in every TDC an outlier bin with a size around 70–80 ps. Our correction assumes that events that fall within a single bin are uniformly distributed, which might no longer hold for very narrow pulses in comparatively wide bins. Another observation is that the delay lines are largely uniform in performance across the whole FPGA. There are no locations that are significantly faster or slower than others, and no obvious clock region boundaries are visible.
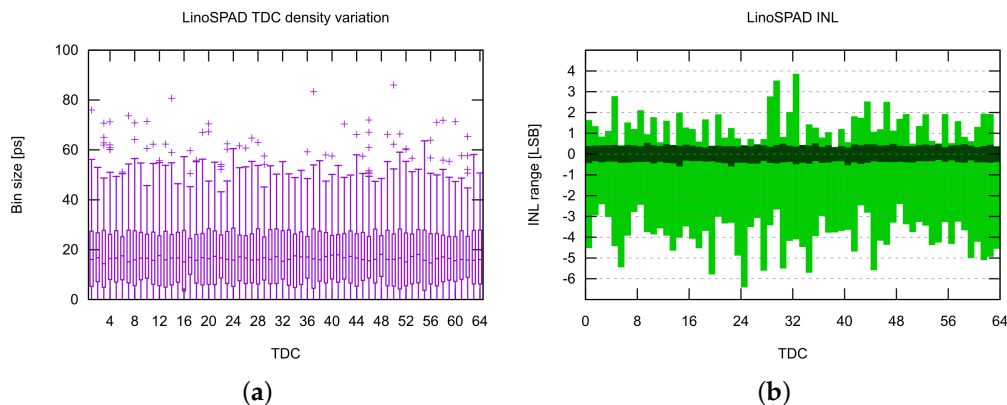
Figure 14. (**a**) TDC-to-TDC density statistics for all TDCs in a given FPGA. Generally, the mean bin size is well below 20 ps, but almost each TDC has some outliers with delays up to 80 ps. (**b**) INL correction efficiency across the 64 TDCs in a camera, showing its effectiveness regardless of the input distributions.

Figure 14b shows the performance of the correction module for every TDC. The light span shows the uncorrected INL values for each TDC, similar to Figure 9b for a single one, and the dark span shows the corrected values.

4.3. Sensor-to-Sensor Variation

From a set of ten cameras of a single fabrication run, we measured the same characteristics as discussed above to evaluate the variation to be expected from between cameras made to the same specifications.

Figure 15 shows the variation in breakdown voltage (Figure 15a) and DCR (Figure 15b) for the 10 aforementioned systems. The DCR was measured with uniform 2 V excess bias. The sensors have been fabricated in general-purpose multi-project wafer fabrication runs. SPADs use non-guaranteed properties of standard CMOS processes and always violate various design rules. This introduces potentially uncontrolled variations between chips, especially if they are fabricated in separate runs, and/or if the process has been optimized in-between. These variations are obviously taken care of once a design is transferred to an industrial production flow.
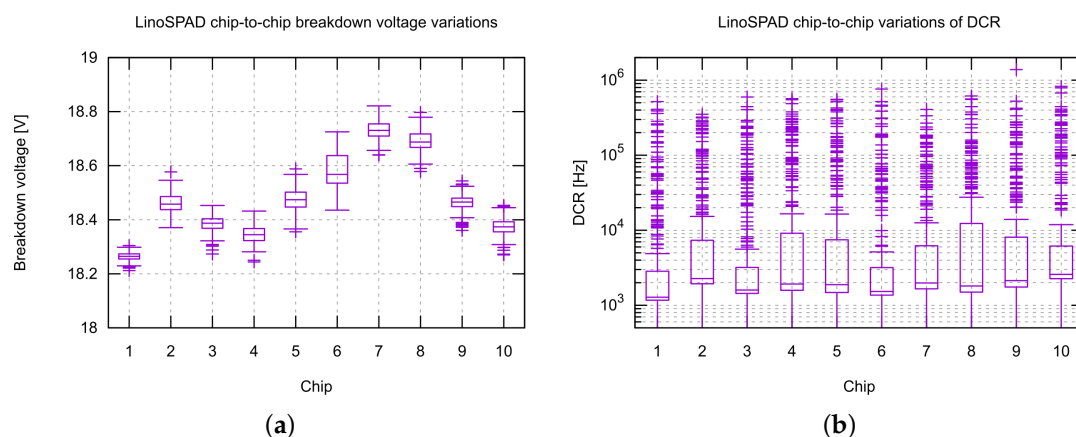

Figure 15. Chip-to-chip variations of: breakdown voltage (**a**); and DCR (**b**). The breakdown voltage has a typical deviation below 100 mV, but chip-to-chip variations can exceed 500 mV. Individual characterization remains therefore important. The difference in DCR is not as pronounced, yet there seem to be chips with significantly better than average performance.

Even for the small sample size from the same run that we analyzed, we observe a variation in breakdown voltage of around 500 mV, which is significantly larger than the variation between the pixels on a single chip (which is typically around 50 mV). Individual measurements and bias corrections are therefore advisable for optimal operation.

From the comparison of the noise, it is difficult to derive a general conclusion. Perhaps the best observation to make is that there seem to be chips with more uniform noise, expressed in a significantly reduced span for 75% of the pixels.

We also evaluated the PDP across the 10 sensors using LEDs with 465 nm and 640 nm wavelength. We did the measurements as reported in Figure 16 with adjusted excess bias for the sensor mean breakdown and subtracted dark counts on a per-pixel basis. As illustrated by our results, the sensitivity shows a good uniformity across all sensors, which suggests a well-controlled fabrication environment, despite the variation in breakdown voltage and DCR.
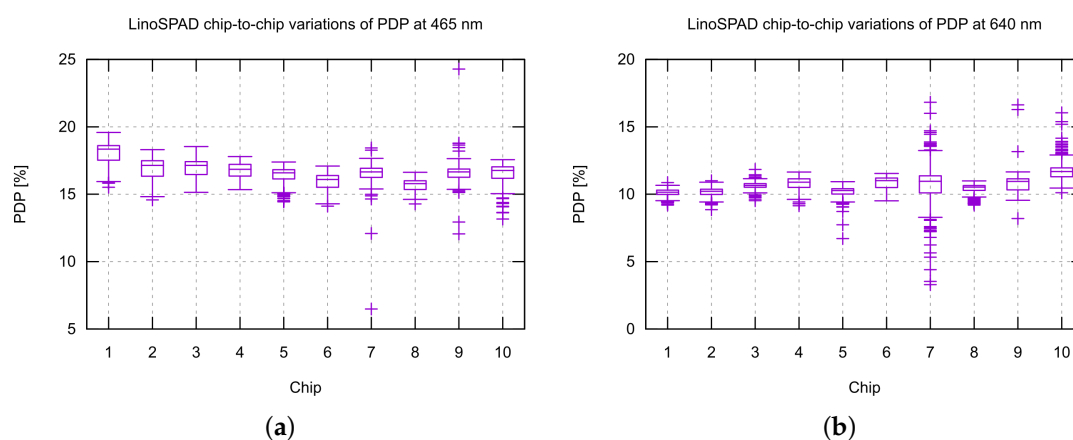

Figure 16. Chip-to-chip PDP variations at: 465 nm (**a**); and 640 nm (**b**) for uniform excess bias of 2 V. The variations are quite small, bearing evidence of a mature, well-controlled fabrication process.

4.4. FPGA-to-FPGA

Because a delay line in an FPGA makes use of circuit properties that are not guaranteed in the datasheet, we carried out measurements to gauge the performance variations between multiple FPGAs. The comparisons, which were based on the TDC characterization measurements outlined in Section 3.5, did not reveal any peculiar behavior. As we are using the fastest available speed grade of the Spartan 6, delay variations are indeed expected to be minimal. Nevertheless, comparing the same TDC across ten FPGAs revealed placements that appeared consistently slower than the rest. However, the differences were not large enough to warrant a design change, which would have resulted in the need for a much more complicated placement approach employing manual optimization of TDC locations.

5. Conclusions

We presented the design and characterization of the LinoSPAD sensor system. The compact size and integration, together with the demonstrated performance results, make it a useful tool for a wide number of time-correlated imaging applications. The integrated TDC modules with a resolution better than 50 ps and flexible synchronization options allow for the rapid integration of the sensor into many existing systems to replace older, bulkier and less capable time-resolved cameras. Future work on the firmware is planned to allow more flexible sharing of the TDCs among multiple pixels and more efficient memory usage for longer histogram duration. A post-processing scheme based on single-shot dithering is also planned. Finally, we are looking into upgrading the linear front-end and/or FPGA motherboard with next generation sensors and/or FPGAs, making full use of the system's modular construction.

Author Contributions: S.B. designed the hardware and firmware of the camera and carried out the characterization; C.B. coordinated and co-directed the work, provided feedback on every part and coordinated with external users; and E.C. proposed the LinoSPAD sensor design and co-architected the original delay line based TDC, proposed the combination of the sensor with the FPGA, co-designed the pixel, and co-directed the work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hamamatsu Photonics. *Photomultiplier Tubes: Basics and Applications*, 3rd ed.; Hamamatsu Photonics K. K., Electron Tube Division, TOTH9001E03a: Hamamatsu, Japan, 2007; Available online: https://www.hamamatsu.com/resources/pdf/etd/PMT_handbook_v3aE.pdf (accessed on 6 October 2017).
2. Lampton, M. The Microchannel Image Intensifier. *Sci. Am.* **1981**, *245*, 62–71. [CrossRef]
3. Michalet, X.; Siegmund, O.H.W.; Vallerga, J.V.; Jelinsky, P.; Millaud, J.E.; Weiss, S. Photon-Counting H33D Detector for Biological Fluorescence Imaging. *Nuclear Instrum. Methods Phys. Res. Sect. A Accel. Spectrom. Detect. Assoc. Equip.* **2006**, *567*, 133. [CrossRef]
4. Rochas, A.; Gösch, M.; Serov, A.; Besse, P.A.; Popovic, R.S.; Lasser, T.; Rigler, R. First Fully Integrated 2-D Array of Single-Photon Detectors in Standard CMOS Technology. *IEEE Photon. Technol. Lett.* **2003**, *15*, 963–965. [CrossRef]
5. Charbon, E. Single-photon imaging in complementary metal oxide semiconductor processes. *Phil. Trans. R. Soc. A* **2014**, *372*, 20130100. [CrossRef] [PubMed]
6. Gersbach, M.; Trimananda, R.; Maruyama, Y.; Fishburn, M.; Stoppa, D.; Richardson, J.; Walker, R.; Henderson, R.K.; Charbon, E. High frame-rate TCSPC-FLIM using a novel SPAD-based image sensor. *Proc. SPIE* **2010**, *7780*. [CrossRef]
7. Carimatto, A.; Charbon, E. Large Scale CMOS Single-Photon Detector Arrays for PET Applications. In Proceedings of the Front End Electronics, Lemont, IL, USA, 20–23 May 2014.
8. Krieger, J.W.; Buchholz, J.; Burri, S.; Bruschini, C.; Charbon, E.; Garbe, C.S.; Langowski, J. Imaging Fluorescence Correlation: Novel Results on New Image Sensors (SPAD Arrays) and a Comprehensive New Software Package (QUICKFIT 3.0). In Proceedings of the Focus on Microscopy, Göttingen, Germany, 29 March–1 April 2015.
9. Mosconi, D.; Stoppa, D.; Pancheri, L.; Gonzo, L.; Simoni, A. CMOS Single-Photon Avalanche Diode Array for Time-Resolved Fluorescence Detection. In Proceedings of the 32nd European Solid-State Circuits Conference, 2006 (ESSCIRC 2006), Montreux, Switzerland, 19–21 September 2006.
10. Panzeri, F.; Ingargiola, A.; Lin, R.R.; Sarkhosh, N.; Gulinatti, A.; Rech, I.; Ghioni, M.; Cova, S.; Weiss, S.; Michalet, X. Single-molecule FRET experiments with a red-enhanced custom technology SPAD. *Proc. SPIE* **2013**, *8590*. [CrossRef]
11. Perenzoni, D.; Gasparini, L.; Massari, N.; Stoppa, D. Depth-range extension with folding technique for SPAD-based TOF LIDAR systems. In Proceedings of the 2014 IEEE Sensors, Valencia, Spain, 2–5 November 2014.
12. Niclass, C.; Sergio, M.; Charbon, E. A Single Photon Avalanche Diode Array Fabricated in 0.35 μm CMOS and based on an Event-Driven Readout for TCSPC Experiments. *Proc. SPIE* **2006**, *6372*. [CrossRef]
13. Sergio, M.; Niclass, C.; Charbon, E. A 128 × 2 CMOS Single-Photon Streak Camera with Timing-Preserving Latchless Pipeline Readout. In Proceedings of the 2007 IEEE International Solid-State Circuits Conference (ISSCC 2007), San Francisco, CA, USA, 11–15 February 2007.
14. Krstajić, N.; Levitt, J.; Poland, S.; Ameer-Beg, S.; Henderson, R. 256 × 2 SPAD line sensor for time resolved fluorescence spectroscopy. *Opt. Express* **2015**, *23*, 5653–5669. [CrossRef] [PubMed]
15. Nissinen, I.; Nissinen, J.; Keränen, P.; Länsman, A.-K.; Holma, J.; Kostamovaara, J. A 2 × (4) × 128 Multitime-Gated SPAD Line Detector for Pulsed Raman Spectroscopy. *IEEE Sens.* **2015**, *15*, 1358–1365. [CrossRef]

16. Maruyama, Y.; Blacksberg, J.; Charbon, E. A 1024 × 8 700-ps Time-Gated SPAD Line Sensor for Planetary Surface Exploration with Laser Raman Spectroscopy and LIBS. *IEEE J. Solid State Circ.* **2014**, *49*, 179–189. [CrossRef]
17. Charbon, E.; Scandini, M.; Mata Pavia, J.; Wolf, M. A dual backside-illuminated 800-cell multi-channel digital SiPM with 100 TDCs in 130 nm 3D IC technology. In Proceedings of the 2014 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), Seattle, WA, USA, 8–15 November 2014.
18. Burri, S.; Homulle, H.; Bruschini, C.; Charbon, E. LinoSPAD: A time-resolved 256x 1 CMOS SPAD line sensor system featuring 64 FPGA-based TDC channels running at up to 8.5 giga-events per second. *Proc. SPIE* **2016**, *9899*. [CrossRef]
19. Favi, C.; Charbon, E. A 17ps Time-to-Digital Converter Implemented in 65 nm FPGA Technology. In Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays, Monterey, CA, USA, 22–24 February 2009.
20. Fishburn, M.W.; Menninga, H.; Charbon, E. A 19.6 ps, FPGA-Based TDC with Multiple Channels for Open Source Applications. *IEEE Trans. Nuclear Sci.* **2013**, *60*, 2203–2208.
21. Bourdeauducq, S. Time to Digital Converter core for Spartan 6 FPGAs. Available online: http://www.ohwr.org/projects/tdc-core/documents (accessed on 6 October 2017).
22. Dadouche, F.; Turko, T.; Uhring, W.; Malass, I.; Dumas, N.; le Normand, J.-P. New Design-methodology of High-performance TDC on a Low Cost FPGA Targets. *Sens. Transducers* **2015**, *193*, 123–134.
23. Xilinx. Spartan-6 FPGA Data Sheet: DC and Switching Characteristics (DS162). Available online: https://www.xilinx.com/support/documentation/data_sheets/ds162.pdf (accessed on 6 October 2017).
24. Wu, J. Several Key Issues on Implementing Delay Line Based TDCs Using FPGAs. *IEEE Trans. Nuclear Sci.* **2010**, *57*, 1543–1548. [CrossRef]
25. Van Hieu, B.; Beak, S.; Choi, S.; Seon, J.; Jeong, T.T. Thermometer-to-binary encoder with bubble error correction (BEC) circuit for Flash Analog-to-Digital Converter (FADC). In Proceedings of the 2010 Third International Conference on Communications and Electronics (ICCE), Nha Trang, Vietnam, 11–13 August 2010.
26. Burri, S. Challenges and Solutions to Next-Generation Single-Photon Imagers. Ph.D. Thesis, Swiss Federal Institute of Technology, Lausanne, Switzerland, 2016.
27. Fishburn, M.W. Fundamentals of CMOS Single-Photon Avalanche Diodes. Ph.D. Thesis, TU Delft, The Netherlands, 2012.
28. Charbon, E.; Fishburn, M.W. Monolithic Single-Photon Avalanche Diodes: SPADs. In *Single-Photon Imaging*, 1st ed.; Seitz, P., Theuwissen, A.J.P., Eds.; Springer: Heidelberg, Germany, 2011; pp. 123–157, ISBN 978-3-642-18442-0.
29. Burri, S.; Maruyama, Y.; Michalet, X.; Regazzoni, F.; Bruschini, C.; Charbon, E. Architecture and applications of a high resolution gated SPAD image sensor. *Opt. Express* **2014**, *22*, 17573–17589. [CrossRef] [PubMed]
30. Pan, W.; Gong, G.; Li, J. A 20-ps Time-to-Digital Converter (TDC) Implemented in Field-Programmable Gate Array (FPGA) with Automatic Temperature Correction. *IEEE Trans. Nuclear Sci.* **2014**, *61*, 1468–1473. [CrossRef]
31. Uhring, W.; Zint, C.-V.; Bartringer, J. A low-cost high-repetition-rate picosecond laser diode pulse generator. *Proc. SPIE* **2004**, *5452*. [CrossRef]
32. Keränen, P.; Kostamovaara, J. A Wide Range, 4.2 ps(rms) Precision CMOS TDC With Cyclic Interpolators Based on Switched-Frequency Ring Oscillators. *IEEE Trans. Circ. Syst.* **2015**, *62*, 2795–2805.
33. Wang, H.; Dai, F.F. A 14-Bit, 1-ps Resolution, Two-Step Ring and 2D Vernier TDC in 130 nm CMOS Technology. In Proceedings of the 43rd IEEE European Solid State Circuits Conference (ESSCIRC 2017), Leuven, Belgium, 11–14 September 2017.
34. PicoQuant. HydraHarp 400—Multichannel Picosecond Event Timer and TCSPC Module with USB Interface. Product Brief. Available online: https://www.picoquant.com/images/uploads/downloads/hydraharp400.pdf (accessed on 21 November 2017).