A SCALABLE ALL DIGITAL 64 x 48 PIXEL FLASH LIDAR IMAGE SENSOR

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Hamid Partovi

August 2020

This dissertation is online at: http://purl.stanford.edu/hf634zg5572

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Mark Horowitz, Primary Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Amin Arbabian

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Gordon Wetzstein

Approved for the Stanford University Committee on Graduate Studies.

Stacey F. Bent, Vice Provost for Graduate Education

This signature page was generated electronically upon submission of this dissertation in electronic format. An original signed hard copy of the signature page is on file in University Archives.

Abstract

In recent decades, there has been an increasing interest in development of depth-sensing systems with a wide range of applications including autonomous driving cars, computer vision, motion-detection and object recognition. One class of such depth-sensors is known as Time-of-Flight (ToF) LiDaR, which estimates the distance of an object by illuminating it with a laser beam, and measuring the time of flight of the reflected light. There are two methods with which LiDaR systems generate a full 3D image of a scene. Most systems used today are Scanning LiDaRs. These devices illuminate a scene sequentially and mostly use mechanical motion to move the beam. The Flash architecture removes the need for beam motion by illuminating an entire scene at once.

This thesis explores the design space for a Flash LiDaR detector. Starting with the Poisson process, we derive the probability of photon detections as a function of time. The distribution, which includes the contributions of background and laser light sources, can best be described as a mixture exponential distribution that is used to build multi-exposure histograms under different conditions. Using the aforementioned procedure, we developed a simulator to analyze measurement uncertainty as a function of parameters such as photon flux, *SNR*, number of exposures, and object distance. An exponential weighting function which scales the histogram is then shown to improve recovery of distant objects. Finally, we provide an overview of the Flash LiDaR ICs and the issues associated with such designs comprising collisions, detection-rate, and timing accuracy. As photon detections are inherently asynchronous events, we use a classical circuit technique, self-resetting logic, to eliminate collisions and reliably increase the detection rates. We also use array-partitioning techniques to scale the design to large-scale arrays while maintaining high-performance and reliability. The Time-to-Digital Converter (TDC) circuits which timestamp photon arrivals use digitally-assisted free-running oscillators to decrease complexity, precluding the need to distribute high-frequency clocks or replica biasing. As a result, the design uses a single low-frequency external clock-source (<500MHz) which simplifies clock routing and reduces clock-distribution timing uncertainty.

Acknowledgements

First and foremost, I would like to thank my adviser, Professor Mark Horowitz for his mentorship, encouragement and for challenging me to dig deep. I am also grateful to him for his help in writing the thesis!

I would like to thank Pietro Caragiulo, Bojan Markovic and Byong Lim for assisting me in designing the Flash LiDaR IC.

I also want to take this opportunity to let my friends know how much I appreciate their company, advice and their help in difficult times: Mahroo Safai, Alfred Yeung, Luca Ravezzi, Ghavam Ghavami Shahidi, and Amy Fritz – thank you.

Finally, I am grateful to my brother, Mahmoud Partovi, my sister, Roshanak Partovi, and her family Sheena, Darya, and Mike O'Dowd for their caring support from afar. Last but not least, I am infinitely indebted to my Mum and Dad who instilled in me the desire to learn and to never stop learning. Without them I won't be here.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

In recent decades, there has been an increasing interest in design and development of depth-sensing systems with a wide range of applications including autonomous driving cars, computer vision, motion-detection and object recognition. Depth-sensing techniques can be divided in two categories: i) those that measure depth using multi-camera stereo-vision based on image disparity [1], and ii) those that use the **T**ime-**o**f-**F**light (*ToF*) data from either sound or electromagnetic waves, to measure distance based on the round-trip delay from emission to detection. While stereo vision technology only needs conventional cameras, their depth resolution decreases with distance, and depends on the physical distance between the cameras. Compared to stereo vision, ToF systems have the advantage that the size of the system is independent of the depth range and they provide a depth accuracy that slowly degrades with distance.

One class of ToF depth-sensors is known as LiDaR [2] - an acronym for **L**ight **I**maging, **D**etection and **R**anging which estimates the distance of an object by illuminating it with a light-source, for example a laser beam, and detecting the reflected light. In comparison to Sound Navigation and Ranging (Sonar)[1][3] which transmits and detects sound waves, and Radio Detection and Ranging (Radar)[4] which uses radio waves, LiDaR provides superior spatial accuracy, potentially in the millimeter range over a wide range of target materials, and distances from centimeters to kilometers. The capacity of LiDaR to successfully form an image of a scene is adversely affected by background illumination, poor object-reflectivity, and the presence of obscurants in the atmosphere. To ameliorate this shortcoming, range-detection systems sometime use LIDAR along with *secondary sensors* such as Radar, cameras or ultrasonic sensors.

There are two methods with which LiDaR systems generate a full 3D image of a scene [5]. Most systems used today are **Scanning** LiDaR. These devices illuminate a scene sequentially, block by block. Blocks can be as small as a single point (point scan) or can comprise row of pixels as in line scanning. These systems mostly use mechanical motion to move the beam, but some teams are working on methods to steer the beam electrically. The **Flash** architecture removes the need for beam motion by illuminating the entire scene of interest all at once. Chapter 2 gives more background about these LiDaR systems.

[1] Sonar has a short detection range and poor noise performance when transmission medium is air.

This thesis explores the design space of a Flash LiDaR detector. To accomplish this goal, we first analyze the accuracy of depth measurements based on the *ToF* technique subject to various environmental and system parameters. Next, we present a Flash LiDaR IC and propose circuit and architectural techniques that address the issues associated with such systems to enable the design of a high-performance high-reliability 3D image sensor.

1.1 Depth Measurement Uncertainty in ToF LiDaR

The uncertainty in depth measurement depends on a number of system and environmental factors, of which the power of the Laser source (signal) and the background illumination (noise) are two key parameters. Other factors include the object distance, reflectivity, and optical and electrical efficiency of the system.

Starting with the Poisson process to describe the statistical nature of light, Chapter 3 derives the probability of photon detections (P_{av}) as function of time. The distribution, which includes the contributions of both light sources, can best be described by a mixture exponential distribution. As each exposure can at most produce a single detection per pixel, a multi-exposure histogram must be constructed to recover object position. When convolved with a filter matched to the signal source the distance of an object can be estimated. Using the aforementioned procedure, the chapter analyzes measurement uncertainty as a function of parameters such as *SNR*, number of exposures, and object distance. *Time-gating* is then shown to improve depth estimation for *distant objects*. While time-gating precludes visibility of near objects, an improved recovery technique which includes time-domain exponential filtering of data has the ability of exposing a whole distant scene without compromising long-range recovery accuracy.

Finally, in Chapter 4, based on the multinomial distribution, we present a mathematical formulation for depth uncertainty of a recovered object, as well as conditions leading to *false-detections*. To gain insight into conditions affecting the uncertainty in depth measurement, a Gaussian approximation to the multinomial distribution is derived.

1.2 Circuit and Architectural Techniques for High Performance Flash LiDaR

In Chapter 5, we first provide an overview of the Flash LiDaR ICs and the issues associated with such designs comprising collisions, detection-rate, and timing accuracy. As photon detections are inherently asynchronous events, we use a classical circuit technique, self-resetting logic, to

eliminate collisions and reliably increase the detection rates. We also use array-partitioning techniques to scale to large-scale arrays while maintaining high-performance and reliability.

This design uses per-column Time-to-Digital Converter (TDC) circuits to timestamp photon arrivals. TDCs employ digitally-assisted free-running oscillators to decrease complexity, precluding the need to distribute high-frequency clocks or build precision clock-generators. As a result, the IC uses a single low-frequency external clock-source (<500MHz) which simplifies clock routing and reduces clock-distribution timing uncertainty.

Chapter 2. Background

In this chapter we provide an overview of LiDaR imaging. Many of these systems use Time Correlated Single Photon Counting, as well as the Single Photon Avalanche photodetector (SPAD), so we explain how both of these techniques work. The chapter concludes with an outline of the environmental and system parameters that affect the performance of LiDaR systems. Chapter 3 and Chapter 4 employ the insights gained in this chapter to address the LiDaR depth estimation accuracy in the presence of background noise, and to direct the design of a LiDaR image sensor described in Chapter 5.

2.1 Time-of-Flight LiDaR Imaging

Time-of-Flight LiDaR imaging estimates distance of an object, or depth of a scene, by measuring the time of arrival of object-reflected light with respect to a reference time. ToF LiDaR uses one of two techniques to measure distance [6, 7]. The first, which is called the **direct** method is based on the time difference between an emitted pulse, and its corresponding received pulse. In the second, the **indirect** method uses the phase difference between an outgoing sinusoidal light wave and its respective incoming wave to measure distance. While both direct and indirect ToF techniques provide high spatial accuracy, the former is better suited in applications, such as autonomous driving cars, where high-speed environment sensing is required [8].

Figure 2.1 illustrates a simplified block diagram of a direct ToF LiDaR system. As shown, a light pulse periodically illuminates a scene. When each pulse is launched a timer is started. This timer stops the instant that the reflected light reaches each pixel so the recorded delay corresponds to the round-trip time from the light-source to the object and from the object to the image sensor. Using this delay and the speed of light (C), the object distance can be calculated:

$$D_{Obj} = \frac{1}{2} C * ToF$$

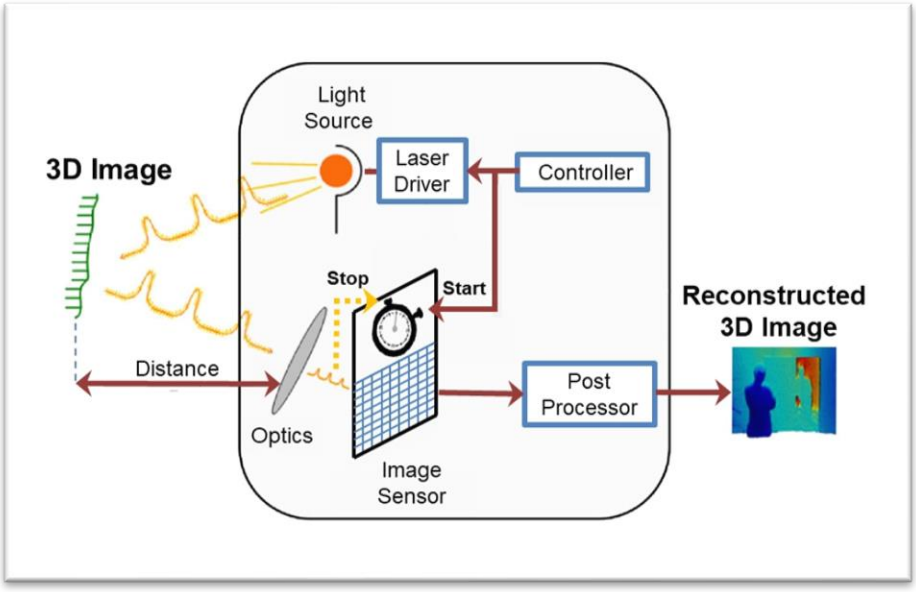

Figure 2.1: Simplified Block Diagram of a ToF LiDaR System

As seen in Figure 2.1, the LiDaR system contains many components [9]. A controller manages the system operation, and sets the laser power to maximize the signal to noise ratio while observing eye safety limits. In addition, the controller ensures that the laser driver and the image-sensor work synchronously. Light pulses reflected from a scene, are collected by a receiver lens and projected unto the image sensor. As we will see later, to maintain a good signal-to-noise ratio, the optical setup includes a narrowband optical filter to suppress the background illumination spectra to within a narrow band around the wavelength of the laser source. The image sensor, over which the scene of interest is projected, comprises an array of photo sensitive pixels, timers to record the arrival time of detected photons and storage to record the spatio-temporal information of detected photons for later processing. Finally, a post-processor receives the data from the sensor and reconstructs a 3D image of the scene with pixel locations providing the spatial 2D image, and ToF data supplying the 3rd dimension.

ToF image sensors generally operate at photon flux levels so it is unlikely that multiple photons impinge a pixel during the very short (100ns) exposure times. In fact, only a fraction of the pixels in the imager will detect a photon during each exposure. As a result, many direct ToF systems use **T**ime **C**orrelated **S**ingle **P**hoton **C**ounting (TCSPC) [10] that is predicated on the detection of a single photon per pixel per exposure whose arrival time is measured from the launch of the initiating laser pulse.

Consequently, to construct a 3D image of the scene, a large number of exposures are required. TCSPC operates on the collected ToF data from multiple exposures to recover a scene. Temporal readings for each pixel are sorted into timing bins to produce a histogram spanning the minimum and maximum travel-times corresponding to the shortest and the longest depths of interest. The histogram, along with an object recovery algorithm is used to estimate the depth of a scene. For a desired depth resolution, the number of exposures depends on system and environmental parameters such as object-distance, efficiency of optics, the laser optical power and background noise. Chapter 3 derives this dependence.

As stated earlier, there are two methods with which LiDaR systems generate a full 3D image of a scene. The **Flash** architecture illuminates an entire scene of interest all at once. On the other hand, **Scanning** LiDaR illuminates a scene sequentially, block by block. As can be expected, **Scanning** LiDaR almost always includes mechanical components such as rotating, or vibrating mirrors to capture parts of the image one at a time. It should be noted that since scanning a scene block by block produces additive delays in succession, objects with significant motion in comparison to the scanning speed undergo the undesirable artifact of *motion-blur*. In Section 3.13, we discuss the trade-offs between Flash and Scanning LiDaR systems.

LiDaR owes its spatial accuracy to the small Laser beam divergence, as well as a highly sensitive photo-detector that can detect the arrival of a single photon. This device, known as the Single Photon Avalanche Diode, or SPAD, has almost infinite photoelectric conversion gain with timing accuracies in the 10s of picoseconds range [11]. The next section describes its operation.

2.2 Single Photon Avalanche Photodetector

Given that during a normal acquisition of a LiDaR system, each pixel receives a small, generally zero or one photon, it is important to use a detector that can capture this photon, and precisely time its arrival. A **S**ingle **P**hoton **A**valanche **D**iode (SPAD), is an excellent detector for this type of application. It is a binary light detector, each time a SPAD detects a photon, the photon generates a very large current surge, which rapidly changes the voltage across the diode and is easily detected. The detector then becomes inactive for a period of time until the voltage recovers to the starting state. At that point, the SPAD is active again, ready to detect the next photon.

This digital behavior is accomplished by biasing a light sensitive P/N junction diode above its breakdown voltage; this is sometimes called Geiger Mode. At this excess bias, any carrier in the depletion region of the junction will be accelerated by the electrical field in that region so fast that when it scatters, it will generate additional carriers. These carriers are again accelerated causing

even more carriers to be generated. This positive feedback loop is what causes the device to "breakdown." At this bias point, the electric field across the junction is of such magnitude that a single photon that generates an electron-hole pair will trigger avalanche multiplication to create a large breakdown current surge across the diode. The next section describes the peripheral circuitry needed to prevent this breakdown current from destroying the diode. As they are responsive to single photons, SPADs are suitable in applications where the incoming photon flux is low. Notice that when biased past avalanche, **any** carrier in the depletion region will cause the diode to avalanche, including carriers randomly generated from thermal energy (thermal noise). Since these carriers also trigger the detector, the detector will occasionally fire when no light is present. This is analogous to dark current in a normal photodetector.

Although SPAD arrays, fabricated in exotic technologies such as InGaAs, have been utilized for decades [12], it is only recently that silicon-based SPADs were shown to provide acceptable performance levels for detection efficiency and dark noise [13, 14]. The viability of SPAD photosensors in standard CMOS processes, has allowed the integration of high-density SPAD arrays along with back-end electronics to implement a 3D image sensor on a single chip.

2.2.1 Operation

Figure 2.2 includes a simplified SPAD circuit, its IV characteristics, and operating waveforms. If a SPAD is impinged on by a photon, it may enter avalanche break-down, in which case, initially, a current of appreciable size flows through the diode, elevating its anode voltage V_{Anode}. In response, the output voltage of the circuit, V_{Out}, transitions to a logical "1".

V_{SPAD} is set to be some over voltage, V_{Ex}, above the diode's breakdown voltage, V_{BD}. V_{Ex} must be smallter than the voltage swing on V_{Anode}, so that once triggered the voltage across the diode reduces from its over-biased voltage, $V_{BD} + V_{Ex}$, to below V_{BD}, quenching its current and protecting the detection diode. The reduction of voltage across the SPAD renders it unresponsive to subsequent arrivals of photons for a period called dead time until the Anode voltage is discharged to ground by way of the Anode resistor, R_{RCH}. Once V_{Anode} falls below the trip-point of the Schmitt Trigger, the circuit output, V_{Out} returns to ground - its quiescent state.

Rearming the SPAD, in the simplest form, is accomplished passively by R_{RCH}, whose value must be large - typically, in the 100s of KΩs - to reduce the initial current surge during an avalanche to avoid *self-heating* and increased dark noise, and to minimize the after-pulsing probability[2] [15].

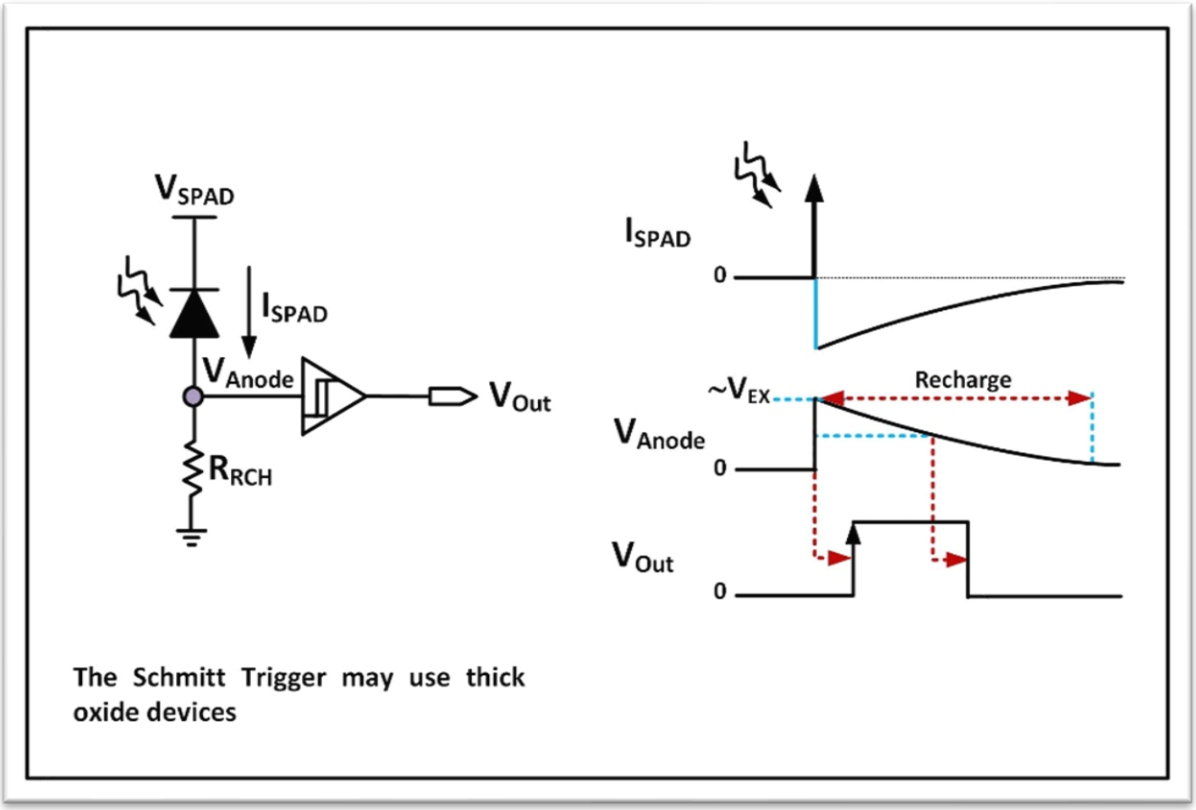

Figure 2.2: SPAD Circuit, its IV Characteristics, and Operating Waveforms

In order to shorten the dead-time and yet control after-pulsing, **A**ctive **Q**uench and **R**echarge (AQR*)* circuits [16, 17] can be used in place of the resistive element.

2.2.2 Active Quenching and Recharge

To provide this better control, an AQR circuit, upon detection of an avalanche, actively drives the voltage across the diode a known voltage below the breakdown voltage, substantially reducing the

[2] A small percentage of avalanche-generated carriers are trapped in deep levels within the junction and then released with significant delay after the primary impact. This can trigger the SPAD again. To avoid this phenomenon, called after-pulsing, an impinged SPAD cannot be immediately re-armed.

field. After a pre-determined hold-off time, this circuit relatively rapidly returns the voltage across the diode is returned to its armed, over-biased condition.

Figure 2.3 provides an example AQR circuit. In its quiescent state, V_{anode} is held at ground by the weak transistor, M_{Keep}, while M_{P1} and M_N are both off (V_{Qnch} = 1, V_{RCH} = 0). The current of M_{Keep} needs to be only large enough to deal with transistor leakage currents. This means V_{Out} = 0. On a photon initiated avalanche, V_{Anode} rapidly rises from the avalanche current forcing V_{Qnch} to transition low. This, in turn, de-activates the keeper transistor, M_{Keep}, and turns M_{P1} on. As a result, M_{P1} drives the Anode to V_{DD} which reduces the SPAD voltage, moving it further out of the avalanche regime. At the same time the circuit output switches to a logical "1" indicating detection of a photonic event. Note that during this time V_{Time} is still low, which enables M_{P1} to drive V_{Anode} high through M_{P2}. This starts the quench time of the detector.

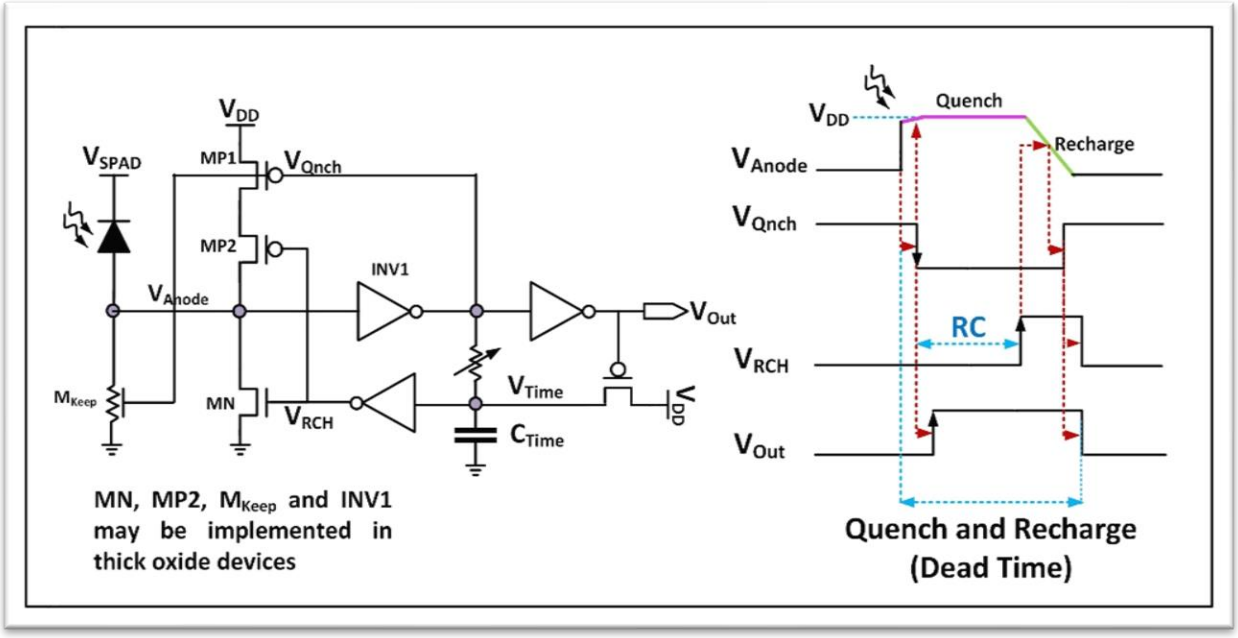

Figure 2.3: Active Quench and Recharge Circuit

This period ends when the output of a timer, shown as an adjustable *RC* delay[3] in the figure, V_{Time}, transitions. This RC delay is triggered when V_{Qnch} transitions. The *RC* delay controls the circuit dead time, and is adjusted to minimize the dead time while maintaining an acceptable low after-pulsing probability. After this delay, V_{Time} falls below the trip point of its succeeding inverter, V_{RCH}

[3] The *RC* delay circuit can be replaced with other delay elements such as a chain of variable starved inverters.

transitions high, discharging the Anode to ground and resetting the output to a logical “0”. At this point, the recharge process is complete; the SPAD returns to its quiescent state and is again responsive to new photonic events.

2.2.3 Key SPAD Metrics

A number of SPAD characteristics are critical when used in a LiDaR system. So far, we have talked about how the capture of a photon generates a digital output. Since we are extracting the arrival time of the photon, the most critical parameter of this circuit is the timing uncertainty of the SPAD. This is called SPAD timing jitter:

> **Timing Jitter**, that is the uncertainty in timing from photon arrival to the onset of avalanche, or the photo-electric conversion time, is primarily due to avalanche buildup statistics, the impingement position, and lateral propagation of the avalanche current. Timing jitter is dependent on technology, SPAD design, and the excess bias. Presently, in standard CMOS processes, timing jitter in the range of 10s of picoseconds have been reported [18].

The next parameter represents the effects of noise in this circuit, which, as mentioned earlier caused output pulses not triggered by light:

> **D**ark **C**ount **R**ate (*DCR*) is a measure of non-photon related avalanches and can be thought of as the internal noise associated with the SPAD. It is caused by trap-assisted thermal generation of carriers and tunneling in the depletion region [19]. The rate increases with the SPAD excess bias; it can be reduced by lowering the operating temperature.

The remaining parameters deal with how effective the SPAD is in converting photons to carriers that can cause an avalanche. The first is the probability that a photon striking the SPAD detector causes carrier generation in the avalanche region:

> **P**hoton **D**etection **P**robability (*PDP*) also known as Photon Detection Efficiency (PDE), is the likelihood that a photon impinging a SPAD will cause an avalanche. It is strongly dependent on technology, the photon energy, and the operating excess bias. In the near infrared range, the PDP of silicon SPADs is in the single percentage range, while it increases to as high as 30% in the visible spectrum [20].

The probability of generating an avalanche is even less than this, since the SPAD only covers a fraction of the pixel area:

Fill Factor (*FF*) is the ratio of the photosensitive area of an array of SPADs to its total area. Generally, it is determined by pixel-to-pixel separation required to suppress cross-talk. In some special cases, the in-pixel back-end electronics dictates the fill factor.

The remaining parameter is from optical noise – photons that are caught by the wrong detector:

Optical Cross-talk alludes to the spurious avalanche of a SPAD caused by an avalanche in a neighboring SPAD. When a SPAD is triggered into avalanche, it can generate secondary photon emissions by hot-carrier relaxation. These photons can be absorbed by a nearby SPAD and cause it to be triggered [21].

In summary, it can be seen that, in addition to process technology and SPAD engineering, the predominant parameters affecting the SPAD performance are the wavelength (energy) of the light source, and the excess bias voltage in the Geiger mode.

2.3 Environmental and LiDaR System Parameters

Since a LiDaR system time stamps each incoming photon (and some thermally generated carriers), the system performance strongly depends on the arrival rate of incoming photons, and whether those photons were generated by our laser illumination pulse, or from "stray" light sources that our system doesn't control. To compute the photons flux rate per pixel, the next section derives the relationship between scene illumination, object distance, and the number of photons collected by the receiver lens. This relationship makes clear that the ratio of the laser vs. background illumination intensity is a critical factor in LiDaR systems. Section 2.3.2 describes the factors, eye safety and solar illumination, that determine this ratio. After describing the detection algorithm in Section 2.3.5, the chapter concludes by outlining the sources of jitter in a LiDaR system.

2.3.1 Received Optical Photons

We can analyze a LiDaR like a normal camera to determine the intensity of the light reaching each pixel. In LiDaR, two light sources illuminate the object: the laser and the background illumination, and in this analysis, we will be interested in keeping track of the photons received from each source. The main difference with a normal camera is that the exposure time is short enough that the Poisson statistics of the light is very important in our case, so in this work, *photon flux*, λ, is defined as the number of photons per units of time impinging a unit of sensor area, or a pixel. We will break this

total flux into λ_L, return photons from the laser that are our desired signal, and λ_N,[4] return photons from the background illumination that act as a source of noise in our system.

We will assume that the object that we are going to measure is Lambertian, so the illumination striking the object is scattered in all directions. Of all the light power that is scattered back from a point P_P, the lens will only receive:

$$P_{Rec} = \frac{A_{Lens}}{\pi D_{Obj}^2} P_P$$ [5]

where A_{Lens} is the area of the lens aperture, and D_{Obj} is the distance of the object of interest from the lens.

The power that is scattered back is proportional to the incident illumination times the reflectance of the object (at the wavelength of interest). This allows us to compute the total light power that will strike a pixel in the imager as:

$$P_{Rec} = \eta_{Sys} \rho \frac{A_{Lens}}{\pi D_{Obj}^2} P_T \quad (2.1)$$

which is a variant of the LiDaR equation [22, 23, 24]. In this equation, P_T is the total light power that is striking the object, in the field of view (*FoV*) of a pixel, ρ is object reflectivity, and η_{Sys} is system efficiency (which is usually close to 1).

As can be expected, the received power, P_{Rec}, is proportional to the illumination power and A_{Lens} (2.1), and inversely scales with the square of object distance. To convert this to photon flux rate, we need to divide the power by the energy of each photon, $E_{Ph} = hc/\lambda_{Ph}$. Finally, as was mentioned in the prior section, not all photons that strike the pixel will cause the SPAD to fire, so we scale the photon flux by the fill factor (*FF*) and photon detection probability (*PDP*) to get rate of SPAD events:

---

[4] We will add the thermally generated dark current into this photon rate. In the cases we explore the rate of thermal noise is small compared to background photon detection.

[5] If scattering is near Lambertian, light is reflected into π steradians and the lens only captures a solid angle of $\frac{A_{Lens}}{R^2}$ steradians.

$$\lambda = \eta_{Sys}\rho\frac{A_{Lens}}{\pi D_{Obj}^2}\frac{P_T}{E_{Ph}}FF * PDP \tag{2.2}$$

If there is no ambient light, all the light striking the object comes from the LiDaR's laser source. Assuming that the laser is focused to illuminate only the *FoV* of the entire imager, $P_T = \frac{P_L}{N_{Pix}}$, where P_L is the laser power, and N_{Pix} is the number of pixels in the imager. This gives the SPAD event rate of the signal we want to detect:

$$\lambda_L = \eta_{Sys}\rho\frac{A_{Lens}}{\pi D_{Obj}^2}\frac{P_L}{E_{Ph}*N_{Pix}}FF * PDP \tag{2.3}$$

2.3.2 Signal to Noise Ratio

Unfortunately, in most situations the object has some ambient illumination that adds additional photons to our detector, in addition to the thermal noise of the detector. The relationship between these incoming noise photons and the ambient illumination is nearly the same as Eq. (2.3), with one important difference: as the object moves farther away, the area in the *FoV* of a pixel increases quadratically with distance. This means if the ambient light has a constant irradiance (power per unit area), the incident power per pixel increases as D_{Obj}^2, which makes the noise photon rate independent of distance.

If we define I_{BG} as the power per unit area of the background illumination, and $FoV_{V|Pix}$, the angular vertical field of view for a pixel in radians, and $FoV_{L|Pix}$, the angular lateral *FoV*, then the background power received by a pixel is:

$$P_{BG} = \eta_{Sys}\frac{\rho}{\pi}A_{Lens}I_{BG}(FoV_{V|Pix}FoV_{L|Pix})$$

Letting $FoV_{Pix} = FoV_{V|Pix} = FoV_{L|Pix}$, the background flux density per pixel becomes:

$$\lambda_N = \eta_{Sys}\rho\frac{A_{Lens}I_{BG}FoV_{Pix}^2}{\pi E_{Ph}}FF * PDP \tag{2.4}$$

Now that we have both the desired photon flux, the signal, and the noise photon flux, we can combine Equations (2.3) and (2.4), and generate the signal to noise ratio:[6]

[6] In this formula, and for the rest of the thesis, we will assume that the dark current rate is much less than the arrival rate photons from background illumination.

$$SNR = \frac{\lambda_L}{\lambda_N} = \frac{P_L}{I_{BG} * D_{Obj}^2 * FoV_{Sensor}^2} \quad (2.5)$$

where $FoV_{Sensor}^2 = N_{Pix} FoV_{Pix}^2$ is the field of view of the entire sensor. This makes sense, since the SNR is just the power ratio between the laser illumination (P_L) and the power of the background illumination (on the area being imaged), $I_{BG} * D_{Obj}^2 * FoV_{Sensor}^2$. To maximize the SNR, we want to make the laser power as large as possible, and the background illuminance as small as possible.

Since background illumination generally has a wide wavelength spectrum, most LiDaR systems have a narrow bandpass filter in the optical path to block most of the energy of this light. Put differently, the received light is band-pass filtered to a few nanometers around the laser wavelength. This means that we are interested in the power of the ambient illumination only in this narrow band of wavelengths. More formally, I_{BG} is the filtered background irradiance, and is related to the spectral irradiance, $I_{\lambda|BG}$ @ $\lambda = \lambda_{Ph}$ by:

$$I_{BG} = I_{\lambda_{Ph}|BG} BW_{Opt}$$

where, BW_{Opt} is the bandwidth of the optical filter.

The achievable signal to noise ratio is limited by the maximum power of the illumination laser, which is constrained by eye safety concerns. LiDAR products must conform to the eye-safety restrictions as set by Standard 60825 of the International Electrotechnical Commission (IEC) [25]. These safety restrictions limit both the energy of each short laser pulse and the average power of the pulse train. These safety limits have a strong dependence on the wavelength of light that is used, with the power growing as the light moves to longer (more infrared) wavelengths. Unfortunately, the *PDP* of a silicon detector falls at longer wavelengths, going to zero at about one micron. As a result, balancing eye safety, and sensor photon detection efficiency, LiDaR systems with silicon detectors operate at wavelengths in the near-infrared (NIR) range.

2.3.3 Outdoor Example

To provide a concrete example, we will choose a laser wavelength of 905*nm*.[7] At this wavelength, using the Standard 60825 data, the average power of a laser that will be operating continuously is around $3mW/cm^2$ of lens opening, and the peak energy allowed for a pulse with a pulse width of 1-10ns is $2\mu J/cm^2$ of lens opening. This means that the peak power can be as large as $2KW/cm^2$ if one

[7] Wavelength of choice for LiDaR products for example, the Velodyne Puck.

could create a 1ns laser pulse. However if this pulse power is used, we would need to wait 0.7ms before starting the next pulse to stay below the average power limit of 3mW/cm^2.

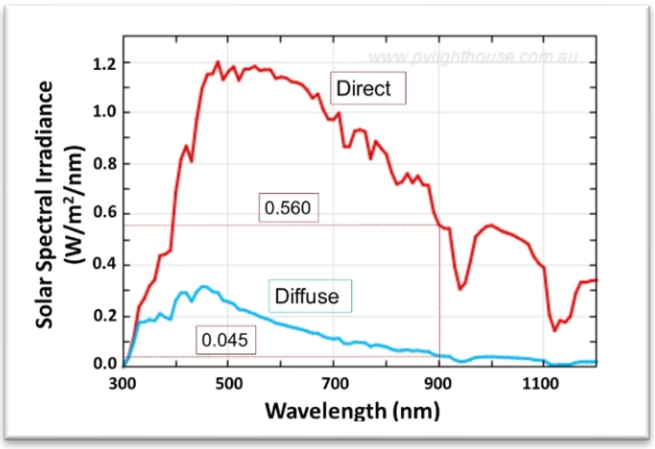

Figure 2.4: Direct and Diffuse Solar Spectral Irradiance[8]

To estimate the noise photon flux, we will assume a worst-case situation: that the LiDaR system operates outdoors, where the background illumination is generated from the Sun. Figure 2.4 includes the Spectral Irradiances for direct and diffuse sunlight: light that has been scattered by particles in the atmosphere such as clouds or reflections from ground.[9] For 905nm, the direct illumination is around 0.56 $W/(m^2\ nm)$, so the total power through a 4nm bandpass filter would be 2.24W/m^2 in direct sunlight. If we assume the lens of the detector has a the FoV of 22.5° x 22.5°, the FoV that the detector can see is around $\left(\frac{\pi}{8}\right)^2 D_{Obj}^2$, so the total ambient light energy would be 0.33$W/m^2 \times D_{Obj}^2$. For a SNR of 1, objects that are 20m away require a laser power of 131W.

If the sensor contains 64 x 64 pixels, the FoV of each pixel is, $FoV_{Pix} = 3mRad$. For this example, we will assume the system efficiency, $\eta_{Sys} = 1$, the system uses a 1cm^2 lens area, and that the object reflectivity is, $\rho = 0.2$. We will also use our estimated values for photon detection probability and fill factor for the detector we describe later in this thesis: PDP = 0.02, FF = 0.4. Plugging these values into Eq. (2.4) and using the energy of a 905nm photon of 2.2*10^{-19}J gives:

8 Courtesy of pvlighthouse.com

9 Direct and Diffuse reflections from a 70° tilted sun-facing surface.

$$\lambda_N = 0.2 \frac{10^{-4} m^2 * 2.24 \frac{W}{m^2} 0.003^2}{\pi\ 2.2\ 10^{-19}\ J} 0.4 * 0.02 = 4.6 Ph/\mu s$$

This data means that even in this "bright" environment, only a few photons will strike a pixel each microsecond.

If the scene is not in direct sun light and is diffusely illuminated, the background illumination drops by over 10x to I_{BG} = $0.18 W/m^2$ ($\lambda_N = 0.38 Ph/\mu s$) . To achieve a *SNR* = 1 at a distance of D_{Max} = $20m$, the laser power is now $11W$.

2.3.4 Pulse Shape, Pulse Period and Frame Rate

Peak power requirements, the pulse width, and eye safety constraints determine the maximum pulse rate of the laser. In these systems, shorter pulse times work better, but it is hard to rapidly turn on and off a 100W laser. Faster pulses are possible, but they use mode-locked lasers, which are bulkier and more expensive [26]. Some available drive circuits have minimum pulse widths of around 4ns [27], which is the value we use in this thesis. Compliance with eye safety limits dictates a pulse period of $\frac{P_L * 4 * 10^{-9}}{0.003}$ for this pulse width. For a P_L of $11W$, the period between pulses is $14.7\mu s$.

The overall frame rate for Flash LiDaR is set by the required time between pulses for eye safety, times the number of pulse exposures required to estimate distance with sufficient precision. If we define T_{LSL} as the shortest laser launch period in compliance with eye safety limits, and N_{Exp} as the number of exposures required to capture a scene at once with adequate resolution, we can define the frame rate of a Flash LiDaR system as:

$$Frame\ Rate_{Flash} = [N_{Exp} T_{LSL}]^{-1} \tag{2.6}$$

The number of exposures itself is determined by the desired standard deviation of depth error (σ_{Err}) while considering other factors such as the laser optical power; that is:

$$N_{Exp} = f(P_L, \sigma_{Err})$$

Scanning LiDaR, additionally includes a timing overhead (T_{Ov}) associated with the mechanical movement of mirrors from one parallel slice to the next, which is typically in the 100s of microseconds to the millisecond range [28]. There are other overhead latencies, common to both architectures, which correspond to data storage and post processing of the spatio-temporal data, whose contribution can be minimized with parallelism and are left out for simplicity.

We define the frame rate for the Scanning architecture as:

$$Frame\ Rate_{Scan} = [N_{Slice}(N_{Exp}T_{LSL}) + (N_{Slice} - 1)T_{Ov})]^{-1} \quad (2.7)$$

Where N_{Slice} is the number of slices comprising the scene that are sequentially exposed.

2.4 Detection Algorithm

There are a number of object detection algorithms that have been proposed for ToF sensors. They include Maximum Likelihood Estimation, Markov Chain algorithm [29] and Linear Cross Correlation [30]. The most widely used algorithm is cross correlation, which measures the response through a matched filter. While a matched filter is optimal for Gaussian noise, it is not optimal for Poisson noise. For these situations a log-matched filter [31] is optimal if there is no background illumination. In this work we assume a rectangular light pulse, so all the methods are the same – to sum up the photons that arrive during the time of the laser pulse.

The first two algorithms attempt to improve object detection in situations where a single pixel gets input from more than one surface position. Thus, they aim to detect *several* surface reflections in a pixel. Wallace, in "Detecting and Characterizing Returns in a Pulsed Ladar," [32] uses two steps, first to find initial estimates of the position and amplitude of the returns, and second to apply the Poisson-maximum likelihood estimation to find the peak parameters. Hernandez-Marin in "Multilayered 3D LiDAR Image Construction," [33] addresses LiDaR data processing using Bayesian approach that incorporates spatial constraints through a Markov Random Field.

Using cross correlation is the most commonly used detection method. This approach applies a "matched filter" [34] to the histogram data to maximize the signal to noise ratio in the detected signal. The matched filtering is formed by taking the cross correlation of the continuous laser pulse energy with the discrete-time, binned, histogram of the scene. The cross correlation is equivalent to convolving the measured response with the matched filter version of the input signal.[10] Ideally, the peak of the filtered histogram is now the time of flight delay for the light to reach the object and return. The next chapter will describe how we modified this filter to work better with data from SPAD detectors.

[10] Convolution is equivalent to cross correlation of two signals when the kernel (laser pulse) is symmetric, and the matched filter of a real input is just the input.

While the detection algorithm returns its best estimate of the position of the object, errors in that estimate arise from three primary sources:

1. Noise: Background photons will broaden the peak of the filtered histogram and will cause uncertainty in the measured results, especially at low *SNR* levels. Chapter 3 gives simulated results of this effect, and Chapter 4 derives the statistics of this uncertainty.

2. Photon Quantization: Given the expected number of captured signal photons in each timing bin is small, photon quantization (often called photon shot noise [35] can also limit the accuracy of the estimated time. This noise exists even when the background flux rate is very small. Thus the accuracy decreases when the total number of photons captured during each time bin is very small

3. Time Quantization: Compared to the other two noise sources, background illumination and shot noise, the ultimate accuracy will be limited by time quantization caused by the finite width of the time bins.

In the case of first noise source, if large, or in a situation where an object doesn't exist within the detection range, the detected peak will correspond to a *noise* peak, and is uncorrelated to object distance. To reduce such failures, the output of the filtered histogram must exceed a threshold. Chapter 3 will describe how we set this threshold.

In addition to the fundamental noise that is present in the detection algorithm, there are a number of other system noise sources that can affect accuracy. These errors arise from timing jitter in the components generating the laser pulse, jitter associated with detecting a photon (dominated by the SPAD jitter) and the jitter in the clock that is driving the SPAD imager. Typically, these can be driven to be small enough not to limit performance, but need to be considered during system design.

2.5 Summary

In this Chapter, we presented the principles governing LiDaR imaging and the Time Correlated Single Photon Counting (TCSPC) as a method to estimate distance of objects. These systems require a highly-sensitive, low-jitter, photon detector, which is usually accomplished using Single Photon Avalanche Diodes, or SPADs as the photodetector of choice. These devices need some supporting circuits that can use passive RC components, but adding active devices yields a design that is easier to control.

After introducing the basic system, we showed that the signal to noise ratio was simply the ratio of the irradiance caused by the pulsed laser illuminator, to intensity of the background irradiance,

which is why shorter laser pulses create better results. Even with high illumination levels, an average pixel will not detect a photon during each pulse; this means that creating a timing estimate with good fidelity requires accumulating the results from a large number of laser pulses. The next chapters look into this relationship in more depth.

Chapter 3. Statistics of Object Recovery in Presence of Noise

In this chapter, we will derive the probability distribution for *photon detections* caused by the light originating from the laser source and the background illumination *as a function of time*. The probability of a SPAD detection is a little different from the base Poisson statistics of light, since a SPAD only detects the first photon that impinges on it.[11] This effect causes the effective sensitivity to decay exponentially with time constant dependent on the background illumination. Using these statistics, we then create a simulator and use it to explore how the accuracy of object distance detection depends on system parameters such as laser and background flux levels, as well as exposure time. From this information we develop an improved detection algorithm.

3.1 Photon Statistics

As mentioned in Chapter 2, only a fraction of the photons incident on a SPAD pixel could start an avalanche. This fraction must strike a photo sensitive region (*FF*), and cause an avalanche (*PDP*). We lumped these factors into an effective arrival rate of photons that could cause an avalanche, and called that λ. The actual *number of arrivals* in a given time-interval, Δt, is *random* in nature with an *average number* given by $\lambda.\Delta t$. [12] The process governing the number of photon arrivals is *a* ***Poisson Distribution*** with the following Probability Mass Function (PMF):

$$P(Ph = k) = \frac{(\lambda.\Delta t)^k . e^{-\lambda.\Delta t}}{k!}$$

Having a mean and standard deviation given by $\mu = \lambda.\Delta t$ and $\sigma = \sqrt{\lambda.\Delta t}$. [36]

For LiDaR applications, the detector uses fine grained timing bins (usually measured in tens of picoseconds) to get good range accuracy. Thus, at reasonable light intensities $\lambda\,\Delta t$ for each time bin will be very small, and the expected probability that one photon arrives in this time slice is approximately $\lambda\,\Delta t$. Figure 3.1 includes the PMF and the Cumulative Distribution (CDF) Functions for the Poisson Process for 3 different mean photon arrivals.

---

[11] The time for the SPAD to recharge is generally longer than the acquisition time, so each SPAD will only fire once during each acquisition

[12] We will continue to use detectable photon rates, so the λ's take *FF* and *PDP* into account.

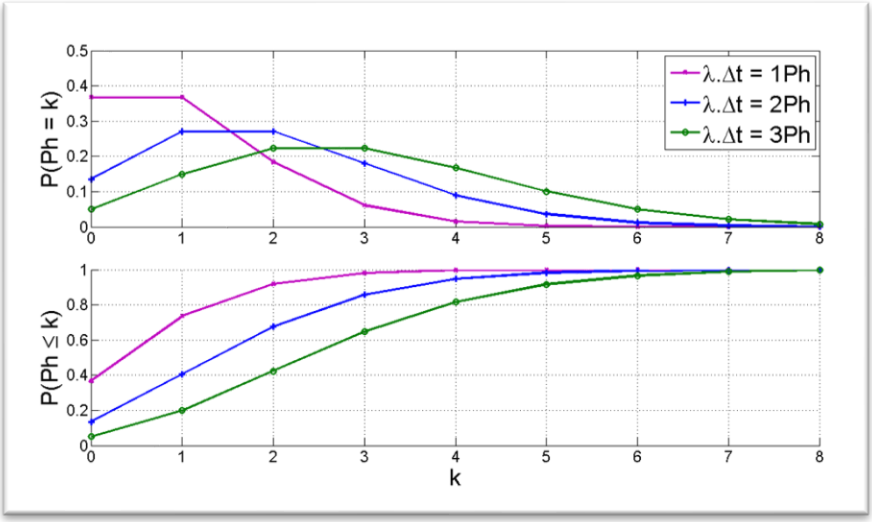

Figure 3.1: PMF (top) and CDF (bottom) Plots for the Poisson Process with different arrival probabilities in a time bin

The SPAD firing statistics are slightly different from the raw detectable photon rate. *The First Photon Postulate* (Appendix A) states that, once the first photon in a stream is detected, i.e. it triggers an avalanche, subsequent photon arrivals on the SPAD are ignored (since the SPAD is inactive while it recovers). Thus, for a detectable photon to be detected, it has to be the first photon that hits that SPAD during the observation time. This means that for a SPAD to fire during a time window two conditions are necessary: a detectable photon arrives in this time window, and there must have been no detectable photons during the entire time prior to this time window. For a uniform detectable photon arrival rate of λ, and a small time window of Δt, which is located t from the start of acquisition,

$$P(Avalanche) = \lambda\, \Delta t\, e^{-\lambda . t} \tag{3.1}$$

where $\lambda\, \Delta t$ is the probability that a detectable photon arrived in this time bin, and the exponential is just the probability that no photons arrived during the prior time.

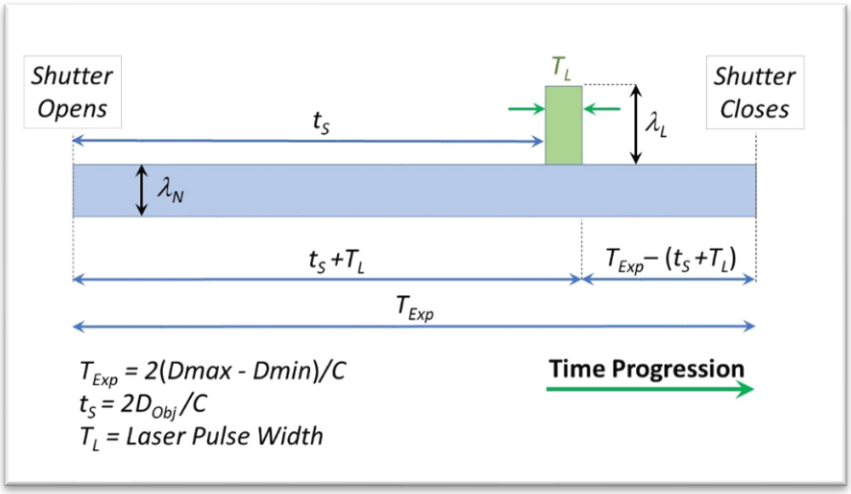

Figure 3.2: Flux of Photons Incident on a Pixel During an Exposure. This consists of the ambient background flux which is on during the whole exposure, and the reflected laser light. The total exposure time is set by the distance range that is of interest.

In our application the arrival rate of photons on a pixel is the sum of two arrival rates, one constant, and the other a short pulse as shown in Figure 3.2. The ambient illumination is constant during the detection window and gives rise to a detectable photon rate of λ_N. When the reflection of the laser pulse returns, the flux increases to $\lambda_S = (\lambda_N + \lambda_L)$. In the following, we will show that a ***Mixture Exponential Distribution*** best describes avalanche statistics in this case. Once the laser pulse is launched, and the image sensor is enabled, the pixel array is immediately exposed to background illumination for a time, t_S, ***before*** the return of the reflected laser pulse (T_L). The arrival of the laser pulse increases the photon flux from λ_N to λ_S for the duration of the laser pulse ***after*** which time the flux drops back to λ_N for the remaining exposure time. The following analysis is based on the "first-photon postulate", which states that avalanche ***may occur*** at times $T \geq t_{Av}$ ***if and only if it didn't occur prior to*** t_{Av}. The probability of avalanche in these three time-intervals is calculated as follows.

1. Avalanche before the return of the laser pulse (near-end), derived by integrating Eq. 3.1:

$$P(Avalanche, t \leqq t_S) = 1 - e^{-\lambda_N t}$$

2. Avalanche during the time of the pulse from the laser-source:

$$P(Avalanche, t_S < t \leqq t_S + T_L) = e^{-\lambda_N t_S}\left(1 - e^{-\lambda_S (t - t_S)}\right)$$

3. Avalanche ***after*** the end of the laser pulse (far-end):

$$P\left(Avalanche, t_S + T_L < t \leqq T_{Exp}\right) = e^{-(\lambda_N t_S + \lambda_S T_L)}\left(1 - e^{-\lambda_N (t - t_S - T_L)}\right)$$

Let us define the CDF of avalanche as a function of time: $F_{Av}(t) = P(Avalanche, T \leqq t)$:

$$F_{Av}(t) =$$

$$P(Avalanche, T \leqq t) \qquad for: t \leqq t_S$$

$$F_{Av}(t_S) + P(Avalanche, t_S < T \leqq t) \qquad for: t_S < t \leqq t_S + T_L$$

$$F_{Av}(t_S + T_L) + P(Avalanche, t_S + T_L < T \leqq t) \qquad for: t > t_S + T_L$$

Consequently, the CDF of *the Mixture Exponential Distribution* comprises:

$$\boldsymbol{F_{Av}(t)} = \tag{3.2}$$

$$\boldsymbol{1 - e^{-\lambda_N t}} \qquad \boldsymbol{t \leqq t_s}$$

$$\boldsymbol{1 - [e^{(\lambda_S - \lambda_N) t_s}] e^{-\lambda_S t}} \qquad \boldsymbol{t_S < t \leqq t_S + T_L}$$

$$\boldsymbol{1 - [e^{(\lambda_N - \lambda_S) T_L)}] e^{-\lambda_N t}} \qquad \boldsymbol{t > t_S + T_L}$$

$F_{Av}(t)$ in Equation (3.2) is indeed a valid CDF, since it is a non-negative, monotonic, non-decreasing function of time, and $F_{Av}(t) \to 1$, as $t \to \infty$. Finally, by differentiating the CDF, the PDF for the Mixture Exponential Distribution is derived:

$$\boldsymbol{f_{Av}(t)} = \tag{3.3}$$

$$\boldsymbol{\lambda_N e^{-\lambda_N t}} \qquad \boldsymbol{t \leqq t_S}$$

$$\boldsymbol{\lambda_S [e^{(\lambda_S - \lambda_N) t_S}] e^{-\lambda_S t}} \qquad \boldsymbol{t_S < t \leqq t_S + T_L}$$

$$\boldsymbol{\lambda_N [e^{(\lambda_N - \lambda_S) T_L)}] e^{-\lambda_N t}} \qquad \boldsymbol{t > t_S + T_L}$$

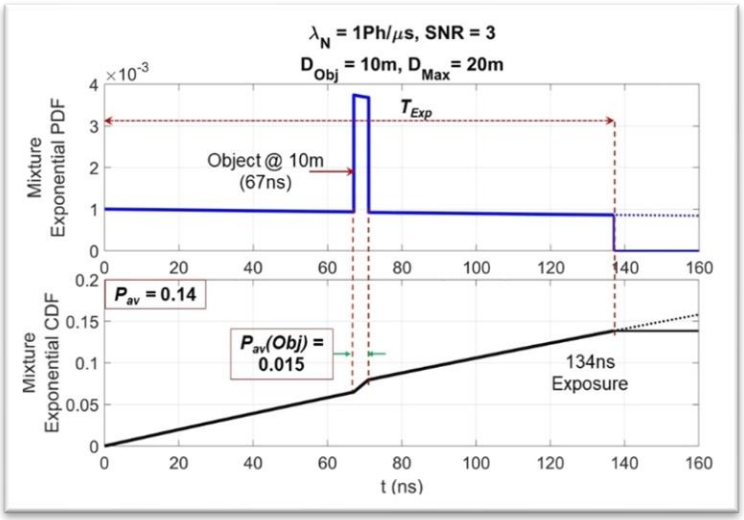

Figure 3.3: Mixture Exponential Distribution

Given the signal and background flux levels, the distribution can be used to determine the occurrence likelihood of an avalanche in a given time period. Figure 3.3 depicts the PDF and CDF of the distribution for a maximum coverage distance of 20m (T_{Exp} = 134ns) with an object placed 10m (t_S = 67ns) away from the sensor. In this figure we assume the background illumination generates an incident flux of λ_N = 1$Ph/\mu s$ to a pixel, and a 4ns wide (T_L = 4ns) laser-pulse incident unto the scene providing a signal flux of λ_L = 3$Ph/\mu s$ to each pixel. Given the flux levels and the exposure time, the figure shows a total probability of Avalanche of 0.14 for each exposure, and only 11% of these detections occur during the laser pulse.

If the probability of avalanche is small, the exponentials in Eq. (3.4) will be close to 1, so $\boldsymbol{f_{Av}(t)}$ can be approximated as:

$$f_{Av|App}(t) = \begin{cases} \lambda_N & t \leqq t_s \\ \lambda_S(1-\lambda_N t_S) \approx \lambda_S & t_S < t \leqq t_S + T_L \\ \lambda_N(1-\lambda_N t_S - \lambda_S T_L) \approx \lambda_N & t_S + T_L < t \leqq T_{Max} \\ 0 & t > T_{Max} \end{cases} \quad (3.4)$$

which is a standard Poisson process with a time dependent arrival rate. While as Figure 3.3 shows the avalanche probability is often small, the derivations in this thesis use the Mixture Exponential Distribution, unless otherwise stated.

3.2 Methodology for Object Recovery

Since the probability of receiving a photon during an acquisition is small, to determine object position takes many acquisitions. To evaluate different object recovery approaches, we created a simulator of our SPAD imager. For each acquisition, the simulator randomly assigns the photon arrival times consistent with the user-set background flux rate and laser flux rate. The laser flux is controlled by setting the *SNR* of the resulting signal. The statistics are generated using the mixture exponential distribution that was derived in the prior section. The photons that arrive during the acquisition time are then stored in a histogram with the timing bin-size set by the resolution of the proposed imager.

To determine an object's distance, we first collect a number of acquisitions, (N_{Exp}), and then pass the histogram through a matched filter object detection algorithm, which provides the desired distance estimate. Each of these tasks is described in more detail in the following sub-sections.

3.2.1 Histogram Construction Algorithm

A simple way to generate sample points for any probability distribution is to select a random number uniformly between 0 and 1, and map that number through the inverse CDF of the distributions. This produces samples with the same probability as the underlying distribution [37].

More specifically, after we pick a number between 0 and 1, if that number is larger than the maximum value of the CDF calculated for the total exposure window (**a** in Figure 3.4), that sample

[13] $T_{Exp} \leqq T_{Max}$ and $\int_0^{T_{Max}} f_{Av|App}(t)dt = 1$

doesn't cause an avalanche and is discarded. Otherwise using the inverse map of the CDF, we compute its arrival time and increment the photon count in that bin. The arrival time determines whether the photon is a noise photon, (**c** in Figure 3.4) or a signal photon (b in Figure 3.4). This process is repeated N_{exp} times to create the final histogram, as shown schematically in Figure 3.5.

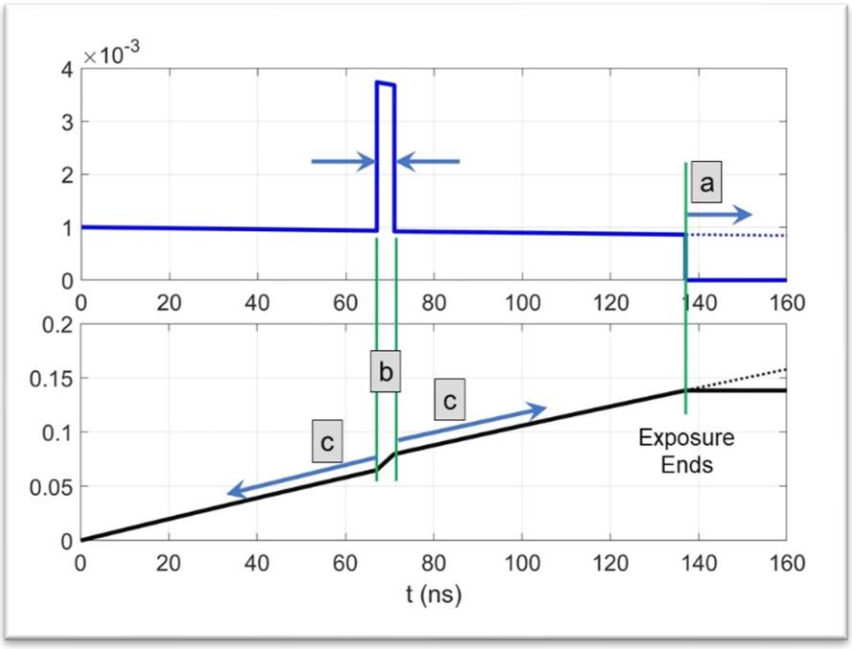

Figure 3.4: Construction Algorithm

Using this approach, we created a histogram for a single frame comprising 20,000 exposures, N_{Exp} = 20K, λ_N = 1$Ph/\mu s$, SNR = 1, and fine-bins of 100ps (1.5cm) width, which is illustrated in Figure 3.6. With 20K exposures, each 100ps bin will collect 2μs of light, so the expected number of photons in the noise bins is around 2 and is around 4 in the signal bins. The figure shows that given these parameters, noise and object bins occupy 0, 1, 2, 3…, 8 but no greater than 9 photons in this example. These results make sense, since from Poisson statistics with an average arrival rate of 4 photons in a signal time bin, the probability of getting 10 in a bin is 0.5%. This means that there is only a 20% chance that one of the 40 time-bins will have 10 photons in it. For the noise bins with an average arrival rate of 2 photons in a time bin, the probability of getting 10 photons is 4×10^{-5}. This means that the chance that one of the ~1300 noise bins has ten photons is around 5%.

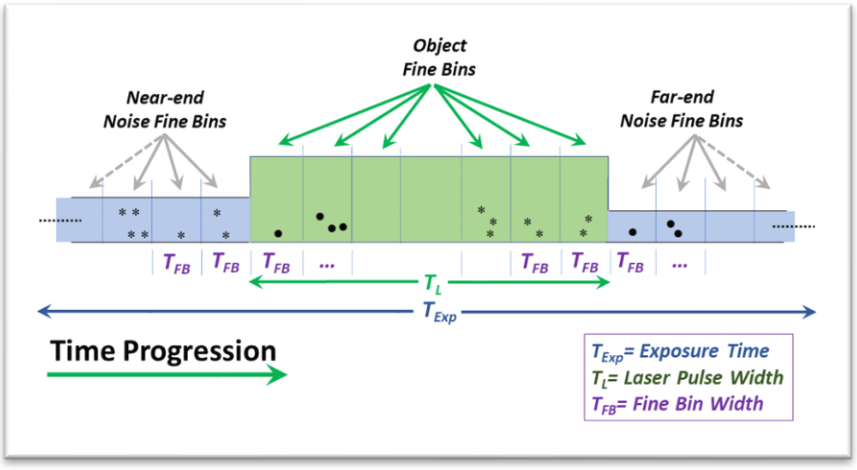

Figure 3.5: Discretization of the Exposure-time into Fine-timing Bins

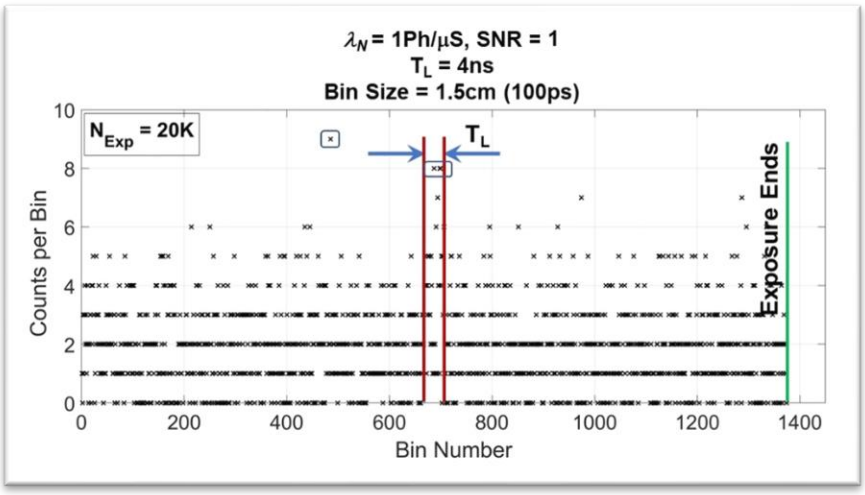

Figure 3.6: Constructed Histogram

3.2.2 Matched Filter

In order to recover an object buried in background noise, we employed the widely-used matched-filter followed by a ***peak-detection*** algorithm [38], where the histogram is convolved with the rectangular laser pulse, which in this thesis is a flat-top pulse of T_L= 4ns wide. This is accomplished by cross-correlating the histogram data with the laser pulse.

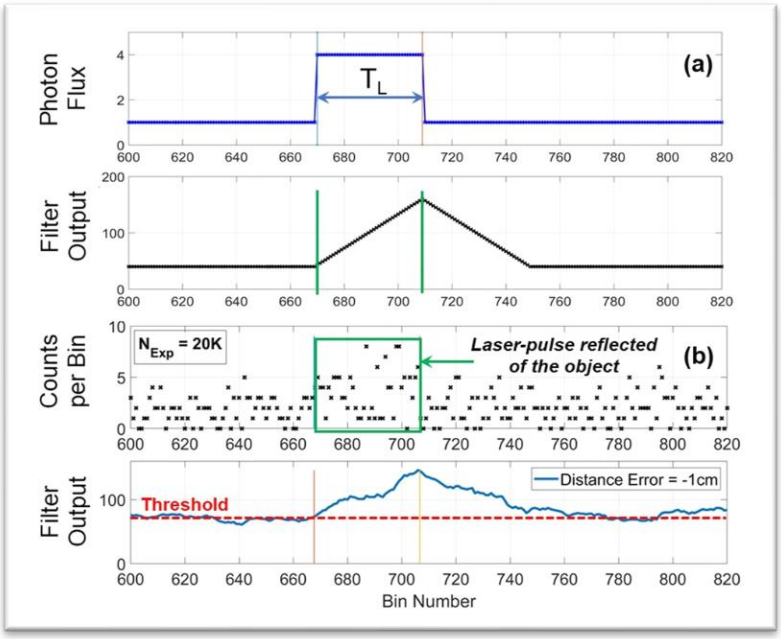

Figure 3.7: Matched-filtering with Ideal (a) and Histogrammed (b) Inputs

As a point of reference, Figure 3.7 (a) shows an ideal laser pulse, and the convolution with itself[14], with background illumination present throughout the exposure time. The filter output is an isosceles triangle whose base is raised by background noise. The vertex of the triangle; that is the maximum of the filter output, is coincident with the end of the laser pulse; $t_S + T_L$ as in Figure 3.2.[15] Therefore, subtracting T_L from the time at which the vertex occurs will produce t_S, and consequently the distance of the object. Figure 3.7 (b) shows an example of simulated photon counts for an object at 10*m* away from the sensor, and the output of the matched filter. In this example, the object is recoverable, i.e. the maximum filter output nearly coincides with the time of flight of the laser-pulse reflected from the object; the distance error is measured to be -1*cm.*

[14] For real, symmetric functions, convolution and cross-correlation provide equivalent results.

[15] Using convolution not cross-correlation can yield a time shift depending on where $t = 0$ is labeled for the pulse. In these figures, $t = 0$ was assumed to be the beginning of the pulse, which means the peak occurs at the end, and not at the pulse position. This shift could have been avoided by labeling the middle of the pulse as the point where $t = 0$.

In reality, the histogram, which comprises counts of events in bins associated with the object and noise photons, includes only *a sparse model of the reflected pulse*. Nonetheless, for any *SNR* and an adequate number of exposures, the distance of an object can be determined to within some error. The next section *estimates* the required number of exposures.

3.2.3 Number of Exposures and Object Recovery

The number of exposures per frame, N_{Exp}, the duration of each exposure, T_{Exp}, *SNR* (and λ_N), all affect the error in estimation of object distance. The relationship can be estimated using Poisson statistics; Chapter 4 gives a more complete derivation with the mixture exponential distribution.

If we assume that the laser pulse fully encompasses a pixel, then each output of the matched filter is the number of photons that were detected in a time period the length of the laser pulse (T_L). There are roughly, T_{Exp}/T_L independent coarse timing bins of this length during an exposure. Thus, we can use Poisson statistics to find the distributions of photons in a coarse "noise" bin and in the coarse "signal" bin. From these statistics we want to find two numbers: the ***threshold*** we should use to determine whether the number of photons in a region indicates that a true object has been detected; and given this threshold the probability that true objects *are* detected. These probabilities can be adjusted by changing N_{Exp}.

The desired false positive detection rate, *FPDR*, is the probability that received noise is detected as an object. This rate sets the threshold count used by the system. Given that there are T_{Exp}/T_L noise bins in each experiment, to achieve our overall error rate, the required probability for a single bin to exceed this threshold is $\frac{FPDR*T_L}{T_{Exp}}$. The threshold, N_{thres}, is just the number where $P(\lambda_{NoiseDetectionBin}, k \geq N_{thres}) \leq \frac{FPDR*T_L}{T_{Exp}}$. For the example in Figure 3.8 (N_{Exp} = 20*K*, λ_N = 1*Ph/μs*), the expected arrival into a noise coarse bin of 4*ns* is 80 photons, and the number of coarse bins is 34 (20*m*). Thus for a 1% *FPDR*, the probability per bin is 0.03%, yielding an $N_{thres} = 112$. For 10*K* exposures, the number of photons per bin drops to 40, yielding an $N_{thres} = 63$.

Given N_{thres}, we need to calculate the probability that true objects are not detected, which gives the false negative detection rate, *FNDR*, (percentage of true objects not detected) of the system. This is just $P(\lambda_{SigDetectionBin}, k \leq N_{thres})$. With 20*K* laser samples, the *FNDR* is extremely small (0.004%). For 10*K* samples, *FNDR* jumps to almost 3%. If this is not acceptable, one needs to accumulate more laser pulses. This situation is illustrated in Figure 3.8. In the 1st case, for the lower N_{Exp} = 10*K*, the peak of the matched-filter occurs in a noise bin, and not when the signal occurs. This peak is greater than the threshold, so a false object is reported. In the 2nd case, for N_{Exp} = 20*K*,

the matched-filter produces an output which ***succeeds*** in recovering the object position with a small estimated error of -1*cm*.

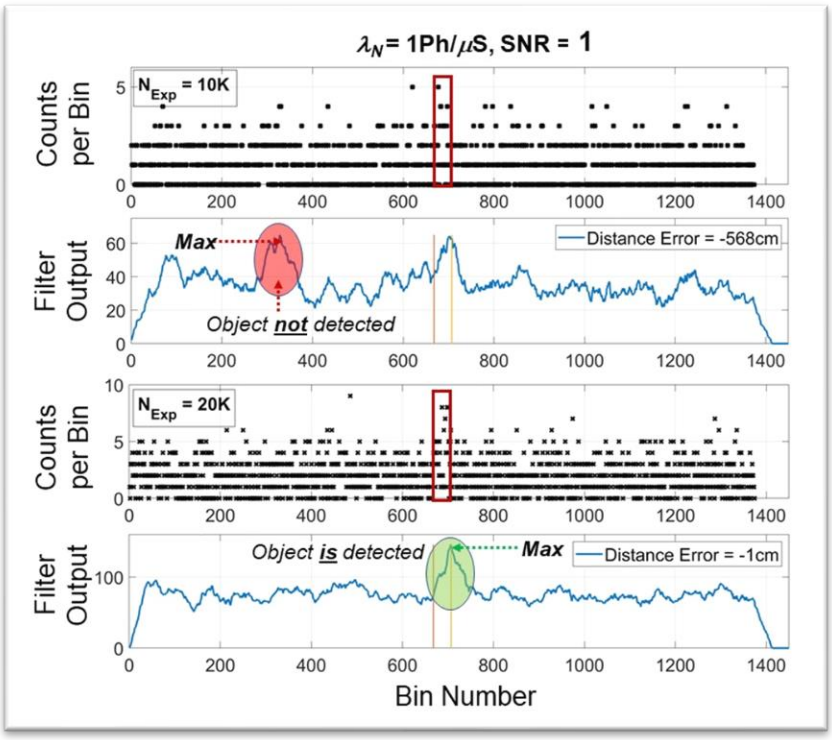

Figure 3.8: Histogram and Matched-filter Output – SNR = 1, N_{Exp} = 10K and 20K

3.3 Statistics of Depth Uncertainty

Section 2.4 introduced the three main sources of depth uncertainty in our system: background photon shot noise, shot noise of the signal photons, and time quantization. This section uses our simulator with the correct photon statistics to explore these effects, providing data and insight on how noise parameters affect measurement accuracy. Since photon arrival is stochastic, we create runs of around 1,000 measurements, with each measurement using N_{Exp} laser acquisitions. From each data run we extract estimates for mean and standard deviation of depth error.[16] Unless otherwise noted, these runs all use a background illumination flux, λ_N, of 1$Ph/\mu s$. We first look at

[16] Throughout this thesis, ***spatial error*** and ***depth error*** are interchangeably used.

the situations where uncertainty is dominated by the signal photons, and then explore situations with lower *SNR*.

3.3.1 Depth Error as a Function of *SNR and* the Number of Exposures

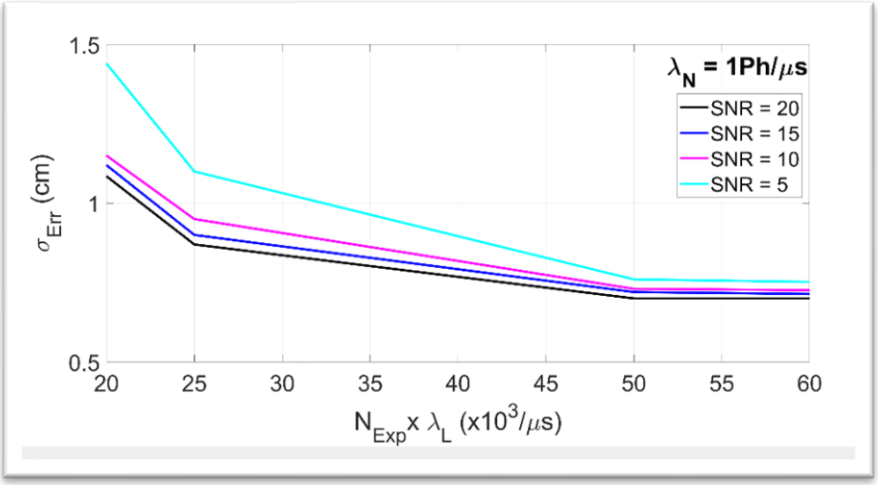

Figure 3.9: Depth Error at High *SNR*s as a Function of N_{EXP} * λ_L

When few background photons arrive, the uncertainty should be limited by the total number of signal photons collected which is clearly shown in Figure 3.9. This figure plots the depth uncertainty as a function of the aggregate laser photon arrival rate. In this plot, as the laser photon flux increases (higher SNR since λ_N is fixed) we decrease the number of exposures, to keep the aggregate laser photon arrival rate constant. At high *SNR*, the different *SNR*, and thus different number of exposures, curves have nearly the same depth error. This means that the depth uncertainty is set solely by the number of photons collected; the effect of the *SNR* and number of exposures only matter because they affect the total number of photons. At lower *SNR* ratios, the shot noise of the background illumination becomes more significant and causes the depth error to increase.[17]

[17] Interestingly the larger depth uncertainty in the *SNR* = 5 curve, comes from noise photons increasing the shot noise. Since 20% more photons (noise photons) are detected during the signal period, this increases the shot noise variance by 20 percent without increasing the signal. Thus, the resulting depth error is larger as shown in the plot.

The expected number of photons in each fine time bin is the aggregate photon rate times the width of the time bin. With our 100ps fine time bins, at $N_{EXP} * \lambda_L$ of $20K/\mu s$, the effective number of photons into each signal fine bin is 2 when the laser light is reflected, and the average number of photons collected during the laser pulse is 80. Given these numbers it is not surprising that at these low number of photons per bin, the depth uncertainty starts to increase. As can be seen in Figure 3.9, when the total number of collected photons is large (> 3 in each timing bin, 30*k*/μs), the uncertainty asymptotes to the quantization noise. In this figure, the distance was set to correspond to the boundary between two time bins, which gives a sigma of slightly larger than 0.75*cm*, which is the sigma for two 100*ps* time bins as will be shown in the next section.

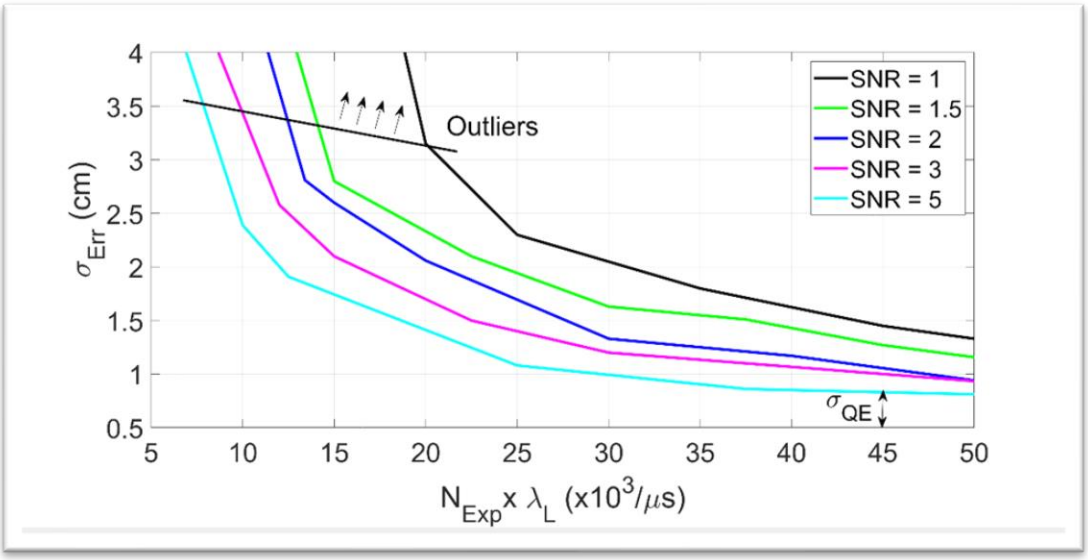

Figure 3.10: Distance Error vs. *SNR* and $N_{EXP} * \lambda_L$

Figure 3.10 explores the relationship between SNR and depth error. As the *SNR* decreases, the shot noise of the background photons adds to the noise of the laser photons and adds depth uncertainty. Since the variance of shot noise is proportional to the expected number of photons detected, at an *SNR* = 3, the variance will be 33% larger than at high *SNR*. At low *SNR*, the variation from the shot-noise of the background photons significantly increases the depth uncertainty. Low *SNR* signal require significantly larger number of signal photons to be received to achieve the same depth accuracy. To get the same distance accuracy of 1.5*cm* from an *SNR* = 1.5 signal requires around 2 times as many laser photons as a *SNR* = 3 signal. So, for a fixed background illumination, this will require 4 times the number of exposures (since each exposure provides half the signal photons). Chapter 4 analyzes this error dependence in more detail.

Also, a reduction in the number of exposures per frame causes a rapid deterioration in depth-error since this greatly decreases the number of signal photons detected. This both directly affects the measured distance accuracy, and also increases the probability of a false detection: either missing the object, or classifying noise as an object, presenting themselves as ***outliers.*** The outliers do not represent the error associated with a recovered object, but rather comprise failed detections, unrelated to object-position. Shown in Figure 3.10, the outlier breakaway line was constructed based on the method in Section 3.2.3 where for each *SNR*, the required aggregate photon arrival rate, $N_{EXP} * \lambda_L$, was calculated such that *FPDR* = *FNDR* = 0.1%.

With increasing number of exposures, depth error asymptotically approaches a lower limit dictated by quantization error. For the worst-case distances where the object is located between two bins, this leads to a standard deviation equal to ½ the size of a fine-bin, as shown in Figure 3.11. It should be clear that for an *efficient* recovery of a scene, the number of exposures must not exceed that needed to achieve the quantization-limited depth error.

$$\sigma_{QE} = \sqrt{\frac{(-\mu)^2}{2} + \frac{(\mu)^2}{2} - \left(\frac{\mu}{2} - \frac{\mu}{2}\right)^2} = \mu = \frac{Bin\ Size}{2}$$

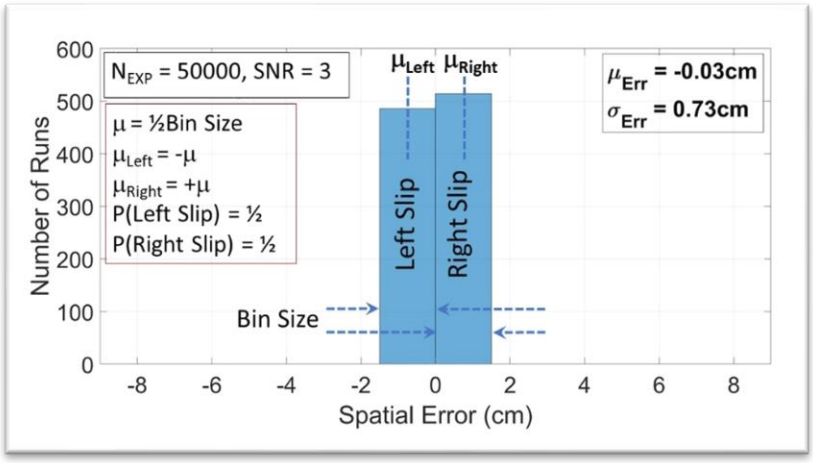

Figure 3.11: Quantization Error as a Limiting Case of Depth Error

3.3.2 Depth Error Histograms

The transition through different noise mechanisms is better visualized by plotting histograms of returned depth estimates. Figure 3.12 illustrates the histograms of depth error for *SNR* = 3 for an object at 20m away, built from runs where the exposures per frame is varied from 5*K* to 20*K*, which correspond to an average arrival of between 1.5 and 6 photons into each fine time bin. With 4.5 - 6 photons expected into each bin, and the bins aligned to the desired time, the distribution is set

almost completely in the two adjacent time bins, with few other returned values. At this *SNR*, even an average of 3 photons per bin (N_{EXP} = 10*K*) still leads to a slight increase in depth uncertainty, which increases rapidly as the number falls to 1.5.

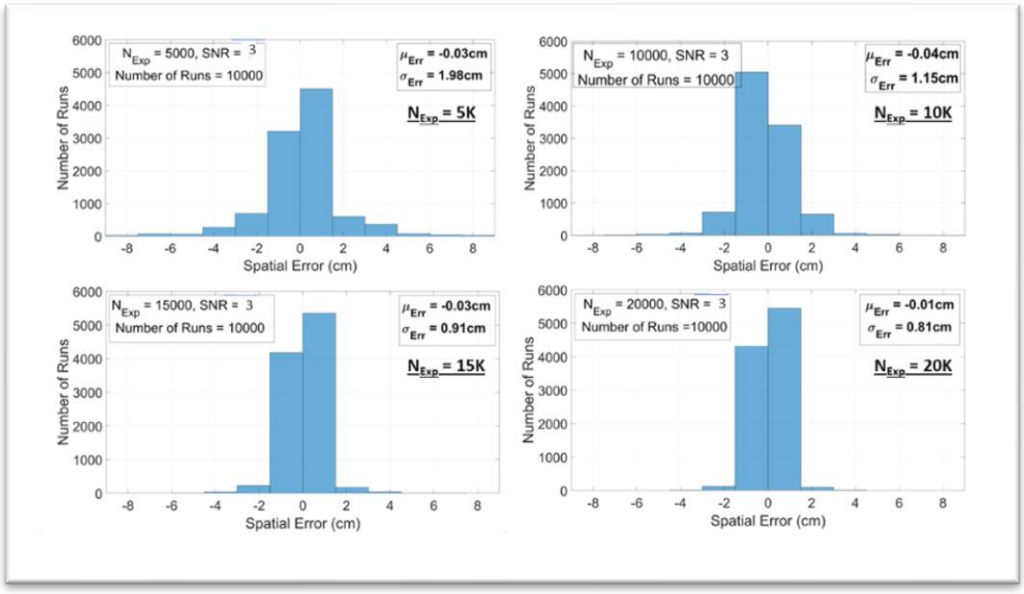

Figure 3.12: Depth Error vs. N_{Exp} for a Fixed *SNR* of 3

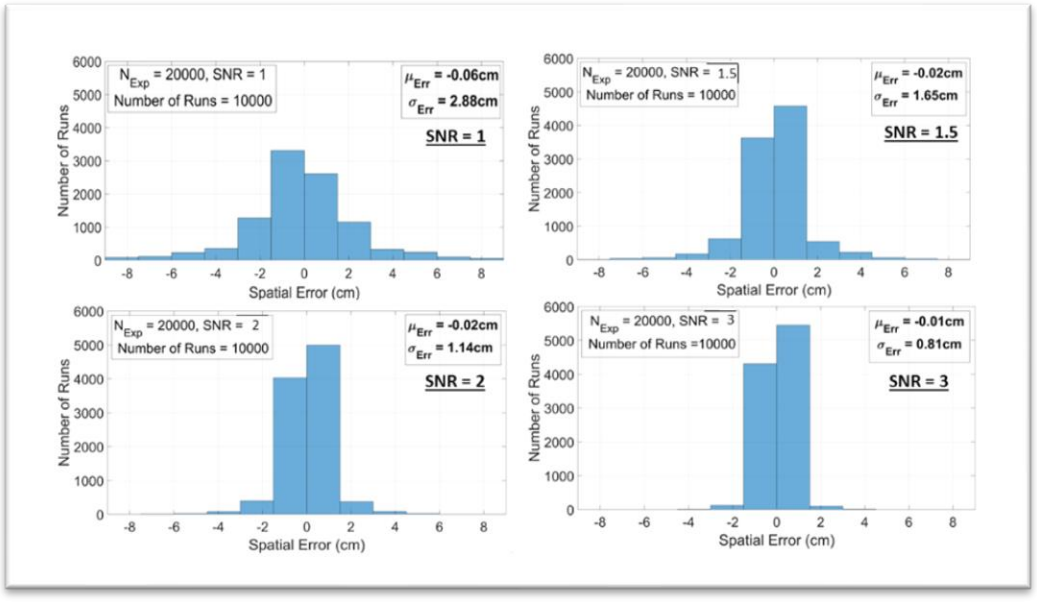

Figure 3.13: Depth Error vs. the *SNR* for a Fixed N_{Exp} of 20*K*

If we assume that the background flux and laser power are fixed, then as the objects get farther away, the *SNR* of the returning signals decreases. At a fixed N_{Exp}, as Figure 3.13 shows, this causes the measurement uncertainty to increase rapidly. This figure assumes a fixed λ_N of $1 Ph/\mu s$. As the *SNR* decreases the number of signal photons fall. Even with $20K$ samples, at *SNR* = 1 there are only on average 2 laser photons in each fine time bin during the laser pulse, in addition to the average of 2 photons in every bin from noise. This small number of signal photons, combined with the large number of noise photons, is why the uncertainty grows.

3.3.3 Depth Error vs. λ_N

Our analysis up to now has focused on how the depth uncertainty depends on the number of laser and background photons received. The results from this analysis don't depend on the actual photon flux rates (as long as N_{Exp} is adjusted to capture the same number of photons). However, as Figure 3.14 shows, because of the first photon principle, the flux rate <u>*does*</u> affect depth resolution. The figure plots depth uncertainty as the background flux, λ_N changes from 1 to 6 $Ph/\mu s$ for a fixed *SNR* of 3. Even for a short distance of $10m$ as the background flux rates increase, the depth error increases, since the effective sensitivity of the SPAD decreases, which reduces the effective photon flux that is captured by the SPAD.

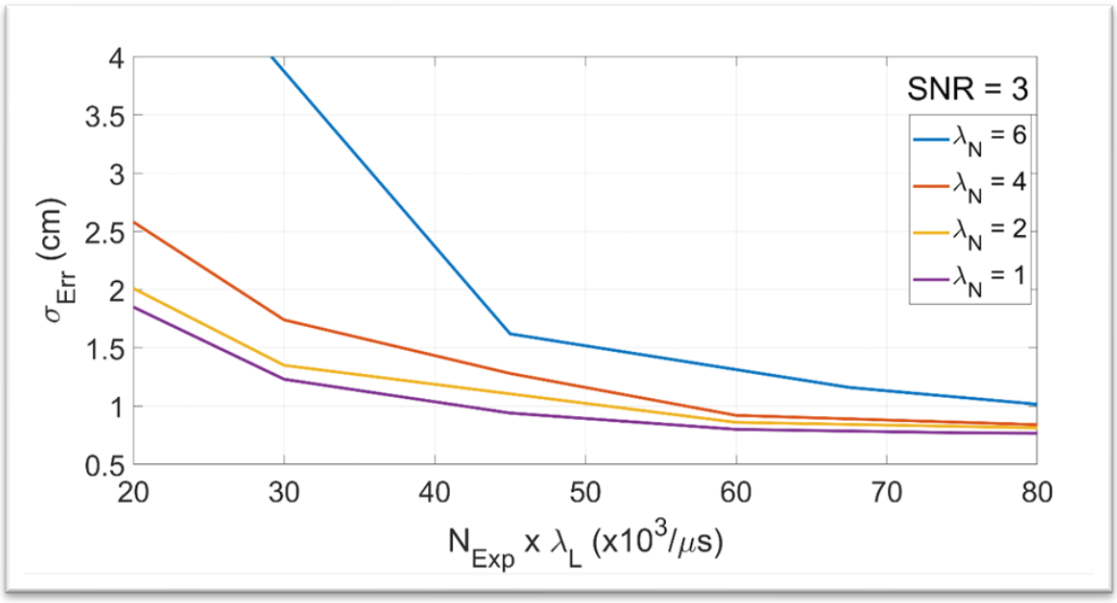

Figure 3.14: Depth Error vs. λ_N for a fixed *SNR* = 3 at a Distance of $10m$

This effect gets worse as the distance increases. Figure 3.15 provides the same plot, but now for an object that is 20m away. With double the distance, for the $133ns$ it takes for light to return, the

probability that a noise photon hasn't already hit the SPAD decreases (See Eq. (3.2)). For $\lambda_N = 6$ $Ph/\mu s$ it is $e^{-\lambda_N t_S} = e^{-6*0.133} = 0.45$, which means that less than half of the signal photons we should detect will be detected. This means that our *SNR* = 3 signal is really more like *SNR* = 1.3, and the actual flux of signal photons received is also only 45% of the listed value. For $\lambda_N = 4$ $Ph/\mu s$, the probability is $e^{-4*0.133} = 0.59$ which is still problematic. Both the lower effective *SNR* and flux rate cause the noise to start increasing more rapidly. In our data, these conditions led to a number of outliers, which is the reason errors go off scale.

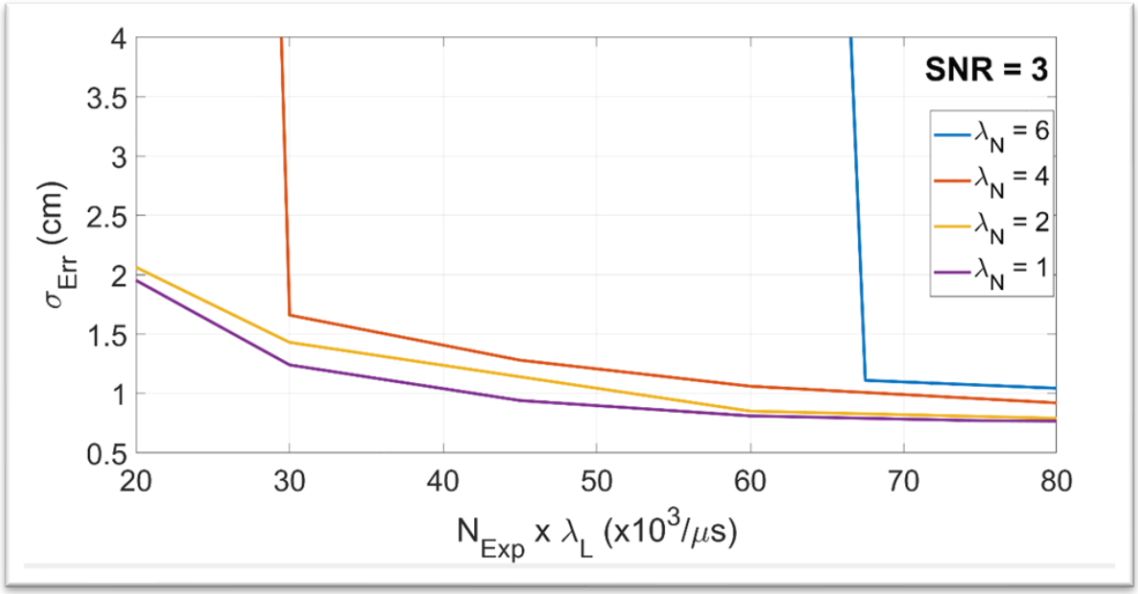

Figure 3.15: Depth Error vs. λ_N for a fixed *SNR* = 3 at a Distance of 20*m*

If the background flux is small enough that for the distances of interest the decrease in sensitivity is small, then the system described in the previous sections can be used. However, this data shows that if $\lambda_N * t_S$ is not very small, then additional methods will be needed to accurately measure distant objects.

3.4 Long-range Object-recovery

Since each SPAD only detects the first received photon, prior illumination desensitizes each pixel over time. Thus, the distance of an object from the sensor deteriorates the probability of detection in 2 ways:

1. Photon-flux falls off with square of object-distance as given by Equation (2.3).

2. The probability that an object causes Avalanche is affected by the length of time of exposure *before* the object is illuminated. It exponentially falls off with distance ($\tau = \lambda_N^{-1}$) as derived from Equation (3.2):

$$P(Avalanche, Object) = e^{-\lambda_N t_S}\left(1 - e^{-\lambda_S T_L}\right), \text{ where } t_S = 2D_{Obj} / \text{C}.$$

The next section provides two methods that improve the depth accuracy of distant objects.

We can better our estimates of distant objects by mitigating the effect of the prior photon arrival. The next section achieves this goal by gating the sensor, activating the SPADs only during certain regions of time. This has dual benefit: it decreases the number of noise bins, and improves the SPAD sensitivity during the reflected laser signal. The downside of gating is that multiple acquisitions are now needed to cover the entire range. Section 3.4.2 introduces a method that avoids the need for multiple runs by providing a simple filter which reverses the decreasing sensitivity, and yields a better single pass detection algorithm.

3.4.1 Time Gating

Time-gating is a technique used to improve the acquisition of distant objects by blocking a large portion of background illumination from triggering the sensor pixels. Another benefit of time-gating is that it minimizes the effect of obscurants, such as fog, rain, or smoke, in the atmosphere present between the object and the sensor. More specifically, time-gating delays the exposure of the image-sensor by some time, T_{TG}, after the launch of a laser pulse. With this method, only objects with round-trip delay of $t_S \geq t_{TG}$ can be imaged, and background illumination, as well as nearby images ($t < t_{TG}$) are blocked from interference in detecting the distant objects.

To demonstrate the noise-filtering property of time-delayed exposure, we consider two scenarios. We place an object 50m away from the sensor. The background and object-reflected flux-levels reaching the sensor are $\lambda_N = 1\ Ph/\mu s$ and $\lambda_L = 1\ Ph/\mu s$ respectively.

Figure 3.16 includes the two scenarios. In the 1st case, the exposure time begins with the launch of the laser pulse ($D_{Min} = 0$). Background illumination *prior to* the reflections from the object effectively degrades the reflected flux to $1.4 Ph/\mu s$, down by roughly 30%. In the 2nd case, exposure is delayed by $T_{TG} = 266ns$ ($D_{Min} = 40m$), filtering out background illumination, as well as objects

closer than 40*m*, from reaching the sensor. In this configuration, the effective flux decreases only to 1.9*Ph*/μs, a reduction of a mere 5%.

The *Ungated* Histogram of Figure 3.17 shows a wide range of measurement errors concentrated *near the beginning of exposure* with an exponential fall-off preceding the peak associated with the object-reflected pulse (yellow bar in the figure). In this case, around 20% of frames (each comprising 20,000 exposures) fail to detect the object correctly (either false positive or false negative detections).

In contrast, the *Time-gated* Histogram of Figure 3.18 includes the error-spread for the same object where the exposure is time-delayed by T_{TG} = 266ns (D_{Min} = 40*m*). In this case, there are no outliers present and depth error has a standard deviation of 2.88*cm*. This reduction in measurement failures is due increasing the effective signal photon arrival rate, and greatly reducing the number of "noise" time bins. Instead of covering from 0-333ns (50m round-trip time), we only need to measure for 67ns. This reduction in noise bins reduces the false positive error rate.

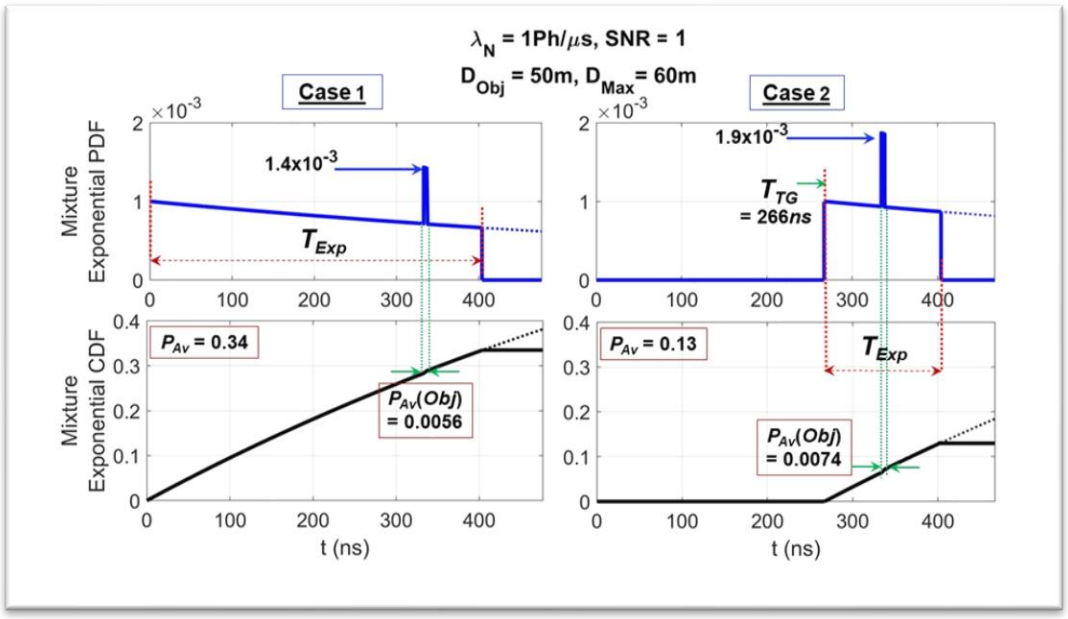

Figure 3.16: Time-gated Noise-filtering

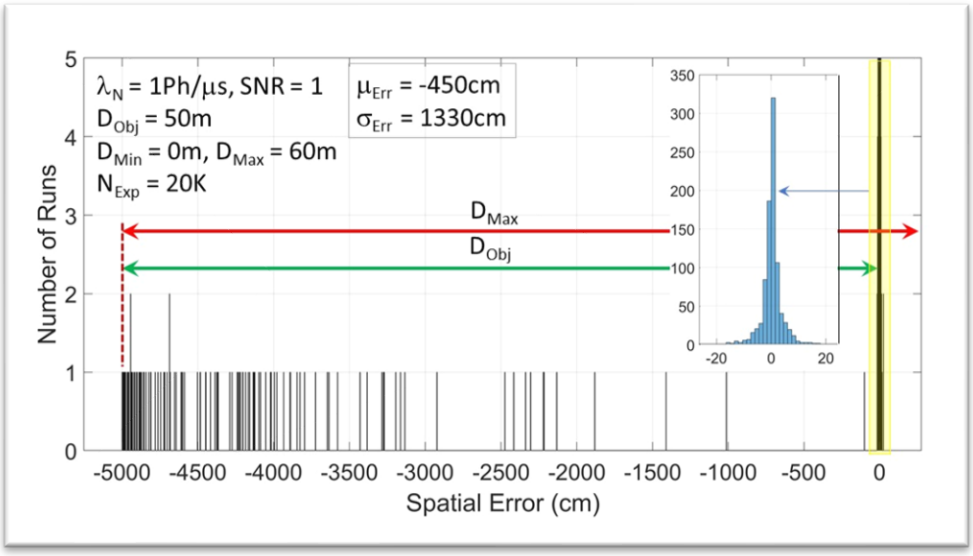

Figure 3.17: Ungated Histogram – D_{Min} = 0m, D_{Obj} = 50m

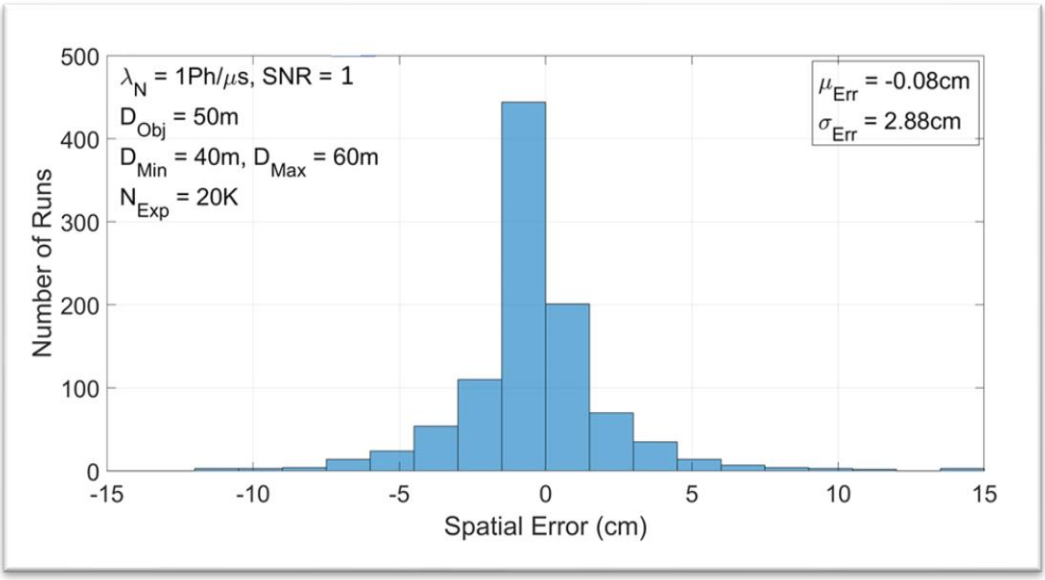

Figure 3.18: Gated Histogram – D_{Min} = 40m, D_{Obj} = 50m

While time-gating clearly works, it leaves the nearer distances unexposed. Therefore, to get an imprint of the whole scene, separate acquisitions for different distances of a scene are required. In our case, to cover the whole scene, additional exposures are necessary to recover objects up to **40*m***.

The *SNR* for these objects is no less than: $(50/40)^2 = 1.56$. If the remaining 40 meters of depth was exposed all at once, it would take $18K$ acquisitions to achieve a depth uncertainty of $1.78cm$.

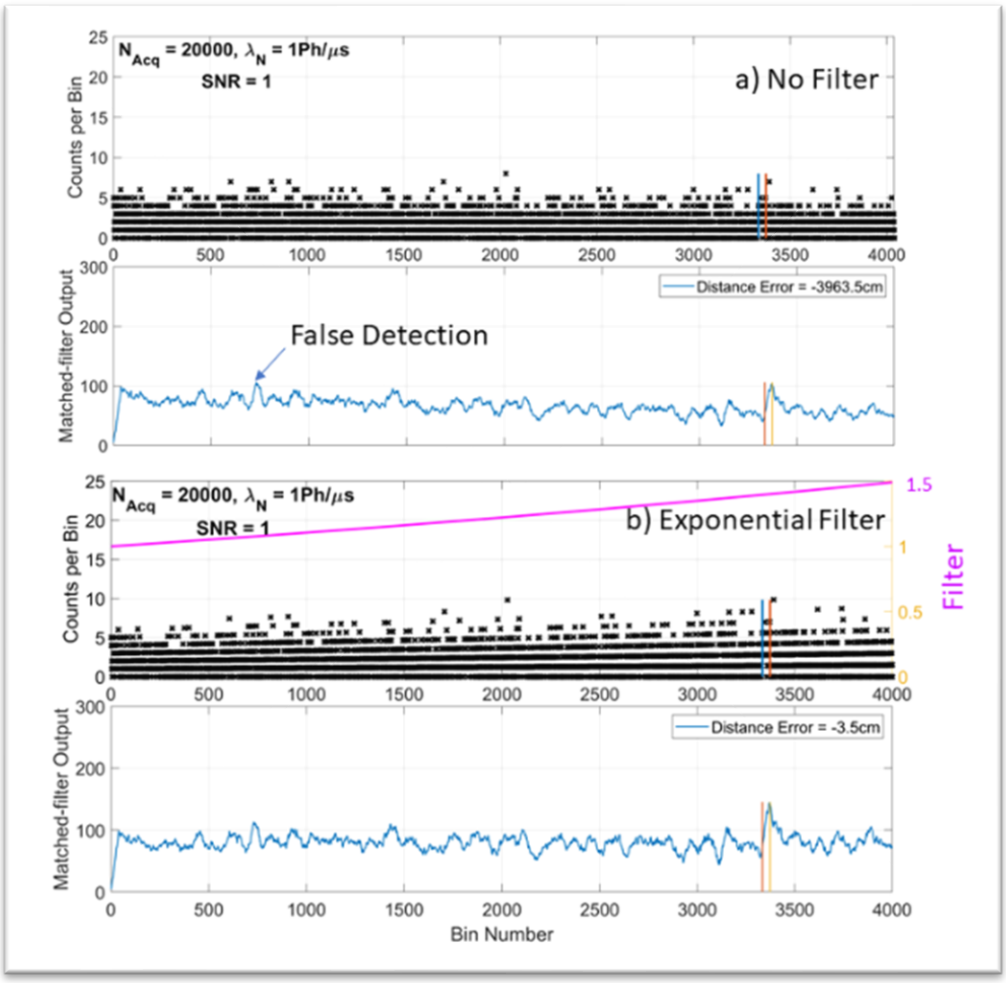

Figure 3.19: *Ungated* Long Range Object Recovery without and with the application of an Exponential Filter

3.4.2 An Improved Object Recovery Algorithm

To avoid the overhead of time gating, it is desirable if one could construct an *improved* object recovery algorithm which can expose a deep scene *all at once* and recover distant objects with good accuracy. Figure 3.17 provides a clue to the problem and how to fix it. In that figure it is clear that the outlier (false positive detections) are more prevalent at shorter times and their frequency decays with time. This makes sense given decreasing effective "sensitivity" of the SPAD with time. To correct for this effect, we can multiply the histogram by an increasing exponential weighting function, $e^{+\lambda_N t}$, which gives more weight to photons that arrive later, since they are less likely to

be detected. Of course this increased gain given to later photons also increases its shot-noise, so it is not as good as time gating, but does significantly improve detection accuracy.

Figure 3.19 show this effect. Part (a) of the figure shows the ***ungated*** histogram and the matched filter output for a 60*m* scene, where an early noise peak is larger than the signal from the 50*m* object. Multiplying this Histogram with our exponential prefilter allows that same object is detected correctly, as shown in Figure 3.19 (b). In comparison to (a), the Histogram is multiplied by the weighting function, where its effect can be seen in flattening of the matched filter output and the correct recovery of the object at 50*m* to within 5.5*cm*.

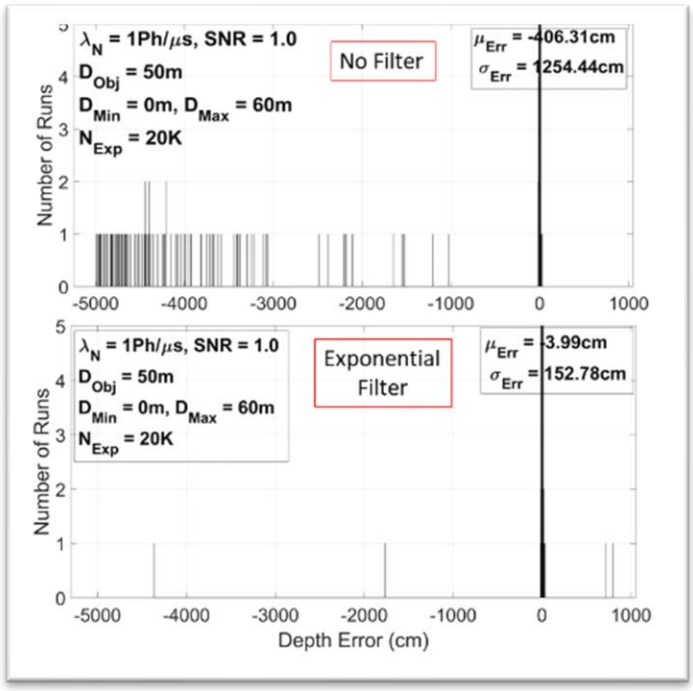

Figure 3.20: Depth Error Histograms without and with the Exponential Filter

To assess the statistical improvement in depth error, we produced depth-error histograms for 1000 runs for an object 50*m* away at an *SNR* = 1. The number of exposures, N_{Exp}, per run was 20*K*. Figure 3.20 illustrates the results. There are only 4 (false positive) outliers using the improved algorithm when compared to roughly 200 for the unfiltered histogram.

While this method requires the background photon rate, this is data is easily estimated from the histogram, by dividing the total number of photons received by the total time period. Since this total includes the laser photons, it will overestimate the rate, but this error will be large only when the *SNR* is very large; for these cases, the filtering operation will not affect the results. At high *SNR*, there are very few noise photons to worry about.

3.5 Maintaining High SNR – When is Scanning Better than Flash

The prior sections have pointed out a number of issues that arise when the *SNR* of the optical signal is small (< 3): the detector needs to receive many more laser photons to have the same distance accuracy, and even when one is willing to have more distance uncertainty, at low *SNR* more captured photons are needed to avoid outliers. Both problems get worse with distance. Since eye safety limits the average power that any laser can emit, when the required number of captured photons increases, the amount of time needed to do acquisitions must also increase. Given these issues, it seems clear that one wants to build LiDaR systems so their minimum *SNR* is above 3, to minimize the required acquisition times.

This desire for high *SNR* can be constrained by laser power: it might not be possible to find a laser strong enough to illuminate the entire field of view and achieve the desired *SNR*. In these cases, it makes sense to decrease the field of view of the laser illuminator, increasing the signal power in one region, and then scan the regions over the full field of view. Most commercial outdoor LiDaR systems take exactly this approach [39, 40] and scan a single pixel sensor and laser illuminator over the entire field of view.

Having an Aerial sensor provides an alternative approach. Here we leave the field of view of the sensor fixed, and simply scan the illumination over the sensor. To understand when partial illumination is beneficial we can look at the ***Frame Rate metric*** of Equations (2.6) and (2.7), which is a measure of how fast the LiDaR system can update its distance estimates:

$$Frame\ Rate_{Flash} = [N_{Exp|Flash} T_{LSL}]^{-1}$$

$$Frame\ Rate_{Scan} = [N_{Slice}(N_{Exp|Scan} T_{LSL}) + (N_{Slice} - 1)\, T_{Ov|Scan})]^{-1}$$

Where T_{LSL} is the time between laser pulses, and given eye safety constraints is proportional to the laser power, and $N_{Exp|Flash}$ is the number of exposures needed when the whole field of view is illuminated at once. Since we will assume the same laser power, T_{LSL} is the same for the scanning system, and it needs $N_{Exp|Scan}$ exposures. We also assume there is some time overhead, $T_{Ov|Scan}$ to move the beam to a new location. Since the scanning system is illuminating $1/N_{Slice}$ of the image, we will assume its *SNR* is N_{Slice} times higher than for the flash system.

If the flash system is running at high *SNR*, then $N_{Exp|Flash} = N_{Slice} * N_{Exp|Scan}$ since both systems need the same number of laser photons to be collected for each pixel, and the illumination in the scanning system is N_{Slice} times larger. Of course, this means that the first terms in both equations are the same and the scanning system is both slower and more complex, since it needs to

scan. The more interesting case occurs when the *SNR* of the flash system drops below 3, and the number of exposures start to rise. From Figure 3.10, relative to the *SNR* = 3 case, the required number of laser photons increases by 15% for *SNR* = 2, and doubles for *SNR* = 1. This additional required acquisition time will often be larger than the scanning overhead. This condition is formalized in the following:

$$Frame\ Rate_{Scan} \lesseqgtr Frame\ Rate_{Flash}$$

$$[N_{Slice}(N_{Exp|Scan}T_{LSL}) + (N_{Slice} - 1)T_{Ov|Scan})]^{-1} \lesseqgtr [N_{Exp|Flash}T_{LSL}]^{-1}$$

To illustrate, let's assume that there are 2 identical 48 x 64 image sensors. The first sensor operates in the scanning mode where the four 12x64 slices are illuminated sequentially (N_{Slice} = 4). The second sensor operates in the flash mode, where the entire array (48 x 64 pixels) is illuminated at once. Let's assume that for the flash sensor the *SNR* = 1, which makes *SNR* = 4 for the scanning case.

From Figure 3.10, for a target accuracy of σ_{Err} = 1.5*cm*:[18]

$$N_{Exp|Scan} = 5.5k$$

$$N_{Exp|Flash} = 42k$$

And the above inequality is evaluated as:

$$(22k * T_{LSL} + 3T_{Ov|Scan})^{-1} \lesseqgtr (42k * T_{LSL})^{-1}$$

$$T_{Ov|Scan} \gtreqless 6.7k * T_{LSL}$$

This basically says that if the total scanning overhead is less than the total time used to form an image, the scanning system will be faster. Since the flash system has a low *SNR* overhead of 100%, this is exactly the expected result.

3.6 Summary

We started this chapter by covering the statistical nature of light which follows the Poisson distribution. Based on the SPAD first photon postulate, we arrived at the Exponential distribution

[18] This 8x increase in acquisitions can be broken down into a 4x increase to received number of laser photons (*SNR* = 4 vs *SNR* = 1) and roughly a factor of 2 needed to get the same depth uncertainty between an *SNR* 4 signal and an *SNR*1 signal.

which describes the probability of detected photons over time. We then derived a Mixture Exponential distribution to include the influence of background noise on photon detection. As each exposure causes at most a single detected photon, a large number of exposures are required to capture a scene. We constructed a simulator matching these photon statistics which creates histograms comprising multiple exposures defining a single frame. We then built multi-frame histograms of depth error, σ_{Err}, and analyzed the influence of background illumination, λ_N, *SNR* and the number of exposures, N_{Exp}, on the depth accuracy of a recovered object. These results showed that at high *SNR*s (higher than 3) we need around 3 photons in each fine time bin to maximize performance, and that number nearly doubles as the *SNR* drops to 1. For a fixed background illumination, this means at high *SNR*, the number of exposures is inversely proportional to *SNR*, but grows rapidly as *SNR* approaches 1. Even when tight distance accuracy isn't needed, the number of exposures can't be reduced too far to avoid problems with false detections. More distant objects make these issues worse, since SPADs only detect the first photon they encounter. This effect gets worse as λ_N increases.

Based on the insights gained from these analyses, we studied time-gating as a technique to improve the recovery of distant objects. While time-gating precludes visibility of near objects, an improved recovery technique which includes time-domain exponential filtering of data has the ability of exposing a whole (deep) scene without compromising long-range recovery accuracy.

Finally, we explored how illumination scanning can be used to reduce the extra acquisitions needed in low *SNR* situations. By only partially illuminating a scene, it is possible to increase λ_L, and move to a higher *SNR* operation point. Moving from $SNR = 1$ to $SNR > 3$ reduces total number of laser photons that the sensor needs to collect by 2 times, which, in an environment where the average laser power is constrained reduces the acquisition time by 2 times as well.

Chapter 4. Mathematical Derivation of Depth Uncertainty

Chapter 3 used Poisson statistical models to explain the more accurate simulations of the depth error for a LiDaR system. While this model could explain many of the simulation results, it didn't account for the underlying mixed-exponential nature of the photon arrivals. Thus, we needed to add corrections for distant objects, or observations that occurred with high background flux. This chapter develops a mathematical model for depth error based on Multinomial Distribution which includes all the effects of SPAD photon statistics. The model serves to ratify the observations made from simulations in Chapter 3, including *false-detections* statistics.

To gain insight into conditions affecting the uncertainty in depth measurement, we approximate the Multinomial model with an upper bound Gaussian Distribution. This formulation sheds light into the relation between depth uncertainty, *SNR* and N_{Exp}, as well as, laser and background flux rates and distance of the object.

4.1 Defining Bin Slips

As we showed in Chapter 3, there is, in general, some depth-error associated with object recovery. In this section we will express this error in terms of fine bin slips, which represent the number of fine timing bins the peak of the matched filter output is shifted from the correct position. The recovered object may have bin-slips to the left, to right or no slips at all.

$$P(LeftSlip) + P(RightSlip) + P(NoSlip) = 1$$

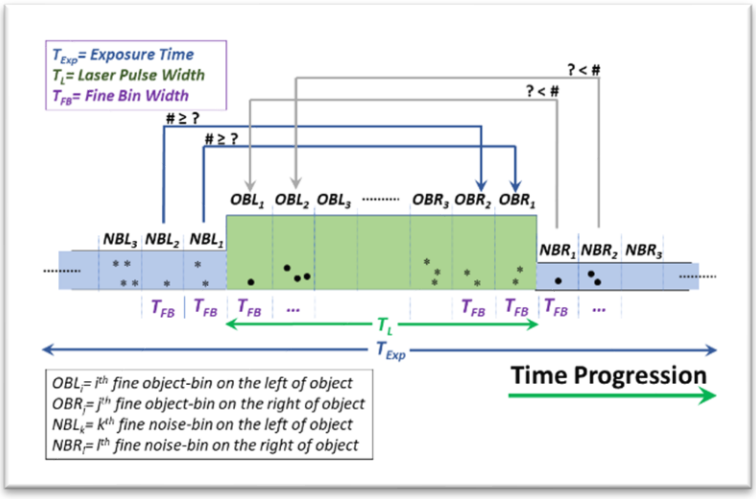

Figure 4.1: Differentiation of Fine Bins

Table 1: Fine-bin Nomenclature and Example

Bin Photon-counts

Fine Bins	Bin Number	Photon Count
Near-end Noise Bins (Left)	NBL_3	$\#NL_3 = 4$
	NBL_2	$\#NL_2 = 1$
	NBL_1	$\#NL_1 = 2$
Far-end Object Bins (Right)	OBR_3	$\#OR_3 = 3$
	OBR_2	$\#OR_2 = 2$
	OBR_1	$\#OR_1 = 2$
Far-end Noise Bins (Right)	NBR_1	$\#NR_1 = 1$
	NBR_2	$\#NR_2 = 2$
Near-end Object Bins (Left)	OBL_1	$\#OL_1 = 1$
	OBL_2	$\#OL_2 = 3$

Before we can define the statistics of a bin slip, we first define terminology that we will use in this chapter. Figure 4.1 is a more detailed variant of Figure 3.5 where fine bins are differentiated as in the following:

1. **O**bject **B**ins, fine time bins which capture the reflected laser light, are grouped by the proximity to the **L**eft, $\boldsymbol{OBL_i}$, and to the **R**ight, $\boldsymbol{OBR_j}$, of the reflected pulse
2. **N**oise **B**ins to the **L**eft (near-end) of the object are marked as $\boldsymbol{NBL_k}$, where $\boldsymbol{k}$ has a descending order moving towards the object

3. **N**oise **B**ins to the **R**ight (far-end) of the object are marked as ***NBRl***, where ***l*** has an ascending order moving away from the object.

The ***Depth error*** associated with a detected object, ***measured in the number of bin-slips***, is predicated on the relative count of photons in noise and the *corresponding* object bins:

- the **near**-end ***noise*** bins in comparison to the **far**-end ***object*** bins, and
- the **far**-end ***noise*** bins versus the **near**-end ***object*** bins.

Let the number of photons in a *left noise bin*, NBL_m, and in a *right noise bin*, NBR_n, be $\#NL_m$ and $\#NR_n$, respectively; further, let the number of photons in *left and right object bins*, OBL_p and OBR_q, be $\#OL_p$. and $\#OR_q$. Table 1 summarizes the nomenclature, as well as the per-bin photon-counts for the example of Figure 4.1.

4.1.1 Near-end Bin-slips

Since we are using the peak of the matched filter to indicate the location of the pulse, we can determine the probability of a shift to the left, by looking at when the photon count in *NBL*s close to the pulse are larger than the *OBR*s at the end of the pulse. For the pulse peak to appear *k* bins to the left of its actual position, two conditions must be met. First, the value of the filter must be higher at position *k* than at *k* - 1, *k* - 2 … 0, otherwise it would not be the peak. Similarly, its value must be higher than the value at *k* + 1, *k* + 2, … for the same reason. Since the matched filter is just integration, we can compute these probabilities by looking at the difference of photon count between signal and noise bins.

More formally, let ΔL_i be the photon count in a **left** noise-bin (*NBL*) *less* the count in the associated **right** object-bin (*OBR*):

$$\Delta L_i = \#NL_i - \#OR_i$$

To meet the first condition, we need to make sure the filter value at *k*, is higher than *k* - 1, $\Delta L_k > 0$, and higher than *k* - 2, $(\Delta L_k + \Delta L_{k-1} > 0)$, etc:

$$P\big(Filter(k) > Filter(j < k)\big) =$$
$$P\big[(\Delta L_k > 0) \wedge (\Delta L_k + \Delta L_{k-1} > 0) \wedge \ldots \wedge \big(\textstyle\sum_{i=2}^{k} \Delta L_i > 0\big) \wedge \big(\textstyle\sum_{i=1}^{k} \Delta L_i > 0\big)\big] \qquad (4.1)$$

For example, for the first 3 bin-slips to the left the following relations must hold:

1. if $\Delta L_1 > 0$; there will be ***at least 1*** bin-slip to the left

2. if **($\Delta L_2 > 0$** *and* **$\Delta L_2 + \Delta L_1 > 0$)** there will be ***at least 2*** bin-slips to the left

3. if **($\Delta L_3 > 0$** *and* **$\Delta L_3 + \Delta L_2 > 0$** *and* **$\Delta L_3 + \Delta L_2 + \Delta L_1 > 0$)** there will be ***at least 3*** bin-slips to the left.

Given this probability, we can compute the probability that there are ***exactly k*** bin-slips to the left by ensuring that the bins farther to the left don't have a higher peak:

$$P(Slips\ to\ the\ Left = k)$$
$$= P\Big(\big(Filter(k) > Filter(j < k)\big) \wedge \big((Filter(k+1) < Filter(k)) \vee ((Filter(k+2) < Filter(k)) \vee \dots)\big)\Big)$$
$$= P\Big(\big(Filter(k) > Filter(j < k)\big) \wedge \big((\Delta L_{k+1} \leq 0) \vee (\Delta L_{k+1} + \Delta L_{k+2}) \leq 0) \vee \dots\big)\Big)$$

For all cases that we care about, all the terms in the probability that the peak isn't farther than k, are going to be close to 1, and can be accurately approximated by its first term, $P(\Delta L_{k+1} \leq 0)$.[19] The resulting formula is:

$$\boxed{P(Slips\ to\ the\ Left = k) = P\Big(\big(Filter(k) > Filter(j < k)\big) \wedge (\Delta L_{k+1} \leq 0)\Big)} \quad (4.2)$$

If our experiment is done with a larger number of trials, so the distribution of photons in each bin are essentially independent, we can factor Eq. (4.) into two probabilities:

$$P\big(Filter(k) > Filter(j < k)\big) \text{ and } P(\Delta L_{k+1} \leq 0)$$

Notice that the probability, $P(\Delta L_{k+1} \leq 0)$, is just $1 - P(\Delta L_{k+1} > 0)$. Also, if the bins are independent they have the same statistics, $P(\Delta L_{k+1} > 0) = P(\Delta L_1 > 0)$. This means that the last term is always:

[19] To see this consider that the probability of any ΔL_k being positive is small, since the expected arrival rate in object bins is higher than noise bins. The first term has already removed all the cases where ΔL_{k+1} is positive, so the second term only removes cases where ΔL_{k+1} is negative, and $(\Delta L_{k+1} + \Delta L_{k+2})$ is still positive. This means that ΔL_{k+2} must be larger than a positive value $(-\Delta L_{k+1})$ rather than zero, which is an even smaller correction.

$$P(\Delta L_{k+1} \leq 0) = 1 - P(Filter(1) > Filter(0)) \quad (4.3)$$

4.1.2 Far-end Bin-slips

In the same manner as the near-end bin-slips, an equation for the probability of slips to the right can be formulated. Defining the photon count in a noise-bin to the **right** *less* the count in the associated object-bin to the **left** to be:

$\Delta R_i = \#NR_i - \#OL_i$

Then:

$$\boxed{P(Slips\ to\ the\ Right = k) = P[\big(Filter(k) > Filter(j < k)\big) \wedge (\Delta R_{k+1} \leq 0)]} \quad (4.4)$$

4.1.3 Relation between the Near-end and Far-end Bin-slips

The ***single-trial probabilities*** for a photon landing in a noise or an object bin can be calculated by using flux-levels reaching the sensor as in Equation (3.4). Given the width of a fine-bin, $\boldsymbol{T_{FB}}$:

$$p_{NL} = \lambda_N(1 - \lambda_N t_S).T_{FB} \qquad (near - end)$$

$$p_{Ob} = \lambda_S(1 - \lambda_N t_S)T_{FB} \qquad (object)$$

$$p_{NR} = \lambda_N(1 - \lambda_N t_S - \lambda_S T_L)T_{FB} \qquad (far - end)$$

where the noise rates are computed for the noise bins close to the laser pulse, and we assume that the laser pulse duration is short enough that we don't need to compute the single-trial probabilities for two signal bins, one for the left side of the pulse and a different one on the right side of the pulse. This assumption also means that, the near-end and far-end single-trial noise probabilities are nearly the same:

$$p_{NR} \cong p_{NL} = \lambda_N(1 - \lambda_N t_S).T_{FB} \quad (4.5)$$

With this approximation, we can write:

$$\boxed{P(Slips\ to\ the\ Left = k) \cong P(Slips\ to\ the\ Right = k)}$$

4.2 Computing Bin Slip Probabilities

Given the *SNR*, background flux, and number of exposures, the above equations enable us to compute the bin slip probabilities, and estimate the expected depth uncertainty for each measurement condition. The next section reviews the properties of a multinomial distribution. Then

we compute the probability of a single bin slip, before extending it to multiple bin slips. Since this exact formulation is complex, we next look at using a Gaussian model to estimate these numbers.

4.2.1 The Multinomial Distribution

Bin-slip probabilities are best modeled by the Multinomial Distribution, which is the generalization of the Binomial Distribution, the latter being a discrete distribution used to calculate the probability of the number of occurrences, x_1, x_2, of two random variables, X_1, X_2, with single-trial probabilities of p_1 and p_2, for n independent trials:

$$P(X_1 = x_1, X_2 = x_2) = \frac{n!}{x_1!\,x_2!} {p_1}^{x_1} {p_2}^{x_2}$$

Where:

$$p_1 + p_2 = 1, \text{ and}$$

$$x_1 + x_2 = n$$

The mean and standard deviation for X_1 and X_2 are given as:

$$\mu_1 = n.p_1$$

$$\mu_2 = n.p_2$$

$$\sigma_1 = \sigma_2 = \sqrt{n.p_1.p_2}$$

When the number of trials is large, and p_1 is small; that is, the expected value of X_1 is modest, bin statistics become independent of each other, and the Binomial Distribution can by approximated by the Poisson Distribution, where the expected *photon count*, λt, is just $\mu_1 = n.p_1$.

In the more general case of ***Multinomial Distribution*** [41] the probability mass function has the form:

$$P(X_1 = x_1, X_2 = x_2, X_3 = x_3, \ldots, X_k = x_k) = \frac{n!}{x_1!\,x_2!\,x_3!\ldots x_k!} p_1^{x_1} p_2^{x_2} p_3^{x_3} \ldots p_k^{x_k}$$

Where:

$$\sum_{i=1}^{k} p_i = 1$$

$$\sum_{i=1}^{k} x_i = n$$

The mean and standard deviation for X_i is given by:

$$\mu_i = n.p_i \tag{4.6}$$

$$\sigma_i = \sqrt{n.p_i(1-p_i)} \tag{4.7}$$

The random variables described by the Multinomial Distribution are correlated with the covariance of:

$$Cov(X_i, X_j) = -n.p_i.p_j$$

Again, in the limit of a large number of trials, if all but one of the p_i are small, this distribution can be approximated by a Poisson distribution. Notice that the covariance scales as $-n.p_i.p_j$, which means in this approximation (n goes to infinity, and $n.p_i \longrightarrow \lambda t$), the covariance goes to zero, and the processes become independent. The joint distribution is just the product of the Poisson distribution for each variable.

4.2.2 Calculating the Probability for a Single Bin-slip

Using Equation (4.5) we assign $\boldsymbol{p_N}$ to single-trial probabilities of the near, p_{NL}, and far-end noise bin, p_{NR}:

$p_N = p_{NL} \cong p_{NR}$

In addition, we represent the number of photons in (near and far) noise and (near and far) object bins by $\boldsymbol{\#N_i}$ and $\boldsymbol{\#O_i}$:

$$\#N_i = \#NL_i = \#NR_i$$

$$\#O_i = \#OL_i = \#OR_i$$

$$\Delta_i = \Delta L_i = \Delta R_i = \#N_i - \#O_i$$

Based on these generalized variables, p_N, $\#N_i$, $\#O_i$, Δ_i, the probability of a single bin-slip to the left or the right can be calculated. In this case, the multinomial distribution comprises 3 variables of $\#N_1$, $\#O_1$ and ***#None***, where *None* is the event that a photon lands in neither the noise bin nor in the object bin of interest. Thus:

$$\#None = N_{Exp} - (\#N_1 + \#O_1)$$

With the following single-trial probability:

$$p_{None} = 1 - (p_N + p_{OB})$$

Therefore, the probability that one bin-slip to the left or right yields a filter output that is higher than the correct position is the summation of probabilities for all possible values of $\#N_1$ and $\#O_1$, where the count in the noise bin is higher than the signal bin ($\#O_1 < \#N_1$):

$$P(Filter(1) > Filter(0)) = \sum_{\#N_1=1}^{\#n} \sum_{\#O_1=0}^{\#N_1-1} \frac{N_{Exp}!}{\#N_1!\,\#O_1!\,\#None!} p_N^{\#N_1} p_{Ob}^{\#O_1} p_{None}^{\#None}$$

where we set the upper limit on the number of photons in a noise bin, $\#n$, to be $\mu_N + 6\sigma_N$. To extend this to find the probability that there is ***exactly 1*** bin-slip to the left or the right, we need to multiply by the probability that $\Delta_2 \leq 0$. Again assume a large number of trials so we can use Eq. 4.3 we get:

$$\begin{aligned} &P(Slips\ to\ the\ Left = 1) = P(Slips\ to\ the\ Right = 1) \\ &= P\big(Filter(1) > Filter(0)\big) * \big[1 - P\big(Filter(1) > Filter(0)\big)\big] \\ &= P\big(Filter(1) > Filter(0)\big) - P\big(Filter(1) > Filter(0)\big)^2 \end{aligned}$$

4.2.3 General Expression for the Near and Far-end Bin-Slip Probabilities

The multinomial expression for any number of bin-slips is derived as follows.

$$\begin{aligned} P(Slips\ to\ the\ Left \geq k) &= P(Slips\ to\ the\ Right \geq k) \\ &= P[(\#N_k > \#O_k) \wedge (\#N_k + \#N_{k-1} > \#O_k + \#O_{k-1}) \\ &\wedge (\#N_k + \#N_{k-1} + \#N_{k-2} > \#O_k + \#O_{k-1} + \#O_{k-2}) \dots\dots] \end{aligned}$$

(4.8)

Let:

$\#Sum_n = (\#N_1 + \#N_2 + \cdots + \#N_k + \#N_{k+1})$

$\#Sum_o = (\#O_1 + \#O_2 + \cdots + \#O_k + \#O_{k+1})$

$\#None = (N_{Exp} - Sum_o - Sum_n)$

The probability that *Filter*(k) > *Filter*($j < k$) is

$$P\big(Filter(k) > Filter(j < k)\big) =$$

$$\sum_{\#N_k=0}^{\#n} \sum_{\#O_k=0}^{\#ok} \sum_{\#N_{(k-1)}=0}^{\#n} \sum_{\#O_{(k-1)}=0}^{\#o(k-1)} \dots\dots \sum_{\#N_1=0}^{\#n} \sum_{\#O_1=0}^{\#o1} \left\{ \frac{\boldsymbol{N_{Exp}!}}{(\#N_1! \dots \#N_{k+1}!)(\#O_1! \dots \#O_{k+1}!)None!} p_N^{\#Sum_n} p_{Ob}^{\#Sum_o} p_{None}^{\#None} \right\}$$

(4.9)

The summation upper limits are set to ensure that the inequalities in Eq. (4.8) hold.[20] To get the bin-slip probability, we take the result from Eq. (4.9) and multiply it by 1 – *P*(*Filter*(1) > *Filter*(0)), leading to the following expression for the probability for any number of bin-slips:

$$P(Slips\ to\ the\ Left = k) = P(Slips\ to\ the\ Right = k)$$
$$= \left(1 - \sum_{\#N_1=0}^{\#\boldsymbol{n}} \sum_{\#O_1=0}^{\#N_1-1} \frac{N_{Exp}!}{\#N_1!\,\#O_1!\,\#No_1!} p_N^{\#N_1} p_{Ob}^{\#O_1} p_{No_1}^{\#No_1}\right)$$

$$\left(\sum_{\#N_k=0}^{\#n}\sum_{\#O_k=0}^{\#ok}\sum_{\#N_{(k-1)}=0}^{\#n}\sum_{\#O_{(k-1)}=0}^{\#o(k-1)} \cdots \cdots \sum_{\#N_1=0}^{\#n}\sum_{\#O_1=0}^{\#o1} \left\{\frac{\boldsymbol{N_{Exp}}!}{(\#N_1! \ldots \#N_{k+1}!)(\#O_1! \ldots \#O_{k+1}!)None!} p_N^{\#Sum_n} p_{Ob}^{\#Sum_o} p_{None}^{\#None}\right\}\right)$$

(4.10)

Where $\#No_1$ and p_{No_1} represent the number and probability of *None* events for the <u>*one bin slip case*</u>. Equation (4.10) was used to calculate slip probabilities to construct histograms based on *SNR* and σ_{Err}. Figure 4.2 illustrates how the <u>*mathematical*</u> formulation of depth error compares to the <u>*simulated*</u> counterpart for *SNR* = 3, and two exposures with N_{Exp} = 5*K* and 10*K*. As can be seen, the theoretical derivation produces results that agree with simulations to within 5%.

Figure 4.3 plots the depth error of our simulated results (Figure 3.10) versus this theoretical model. Note that at lower signal flux rates, the theoretical curves increase not as rapidly as the simulated plots indicating the increasing presence of outliers.

20

$\#n = \mu_{NL} + 6\sigma_{NL}$

$\#ok = \#N_k - 1$

$\#o(k-1) = \{\#N_k + \#N_{(k-1)}\} - \#O_k - 1$

$\#o(k-2) = \{\#N_k + \#N_{(k-1)} + \#N_{(k-2)}\} - \{\#O_k + \#O_{(k-1)}\} - 1$

.

.

.

$\#o1 = \{\#N_k + \#N_{(k-1)} + \#N_{(k-2)} + \ldots + \#N_1\} - \{\#O_k + \#O_{(k-1)} + \#O_{(k-2)} + \cdots + \#O_2\} - 1$

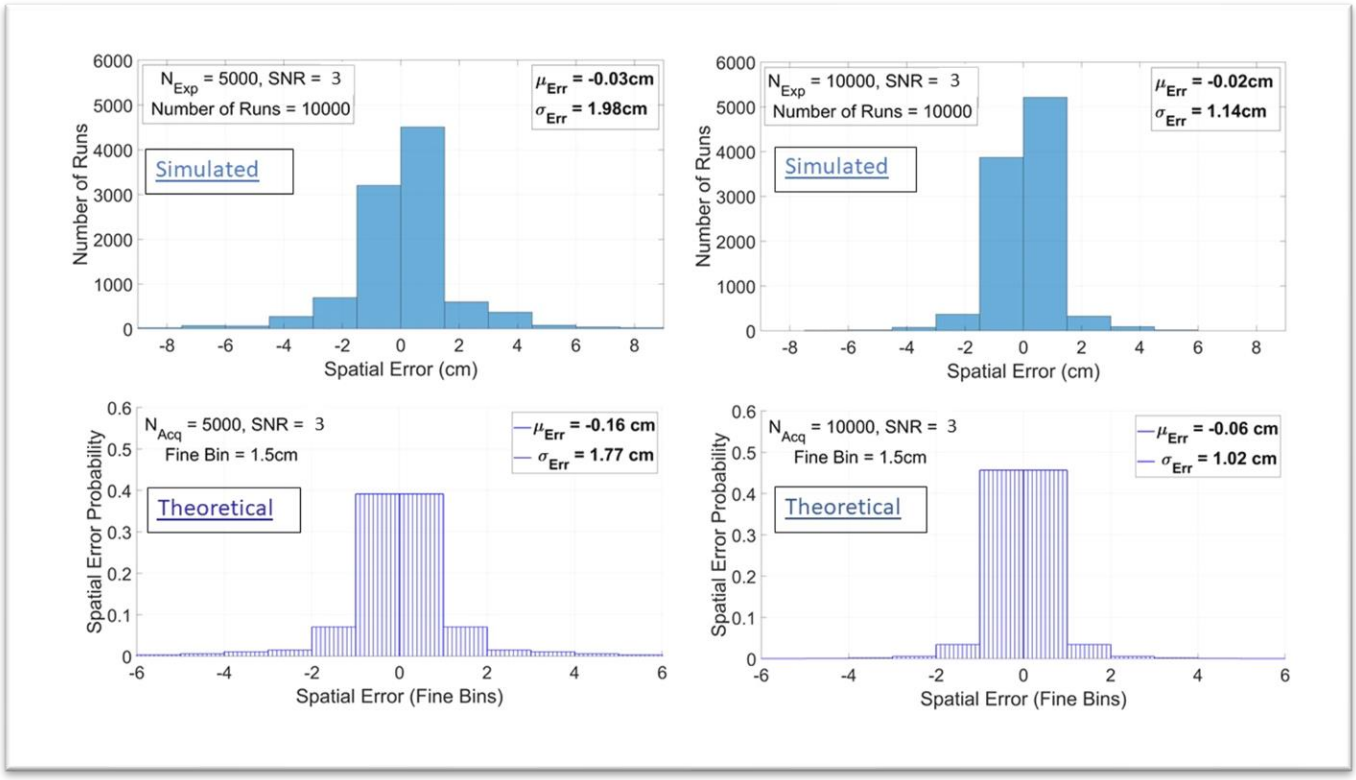

Figure 4.2: Simulated and Theoretical Histograms for Depth Error

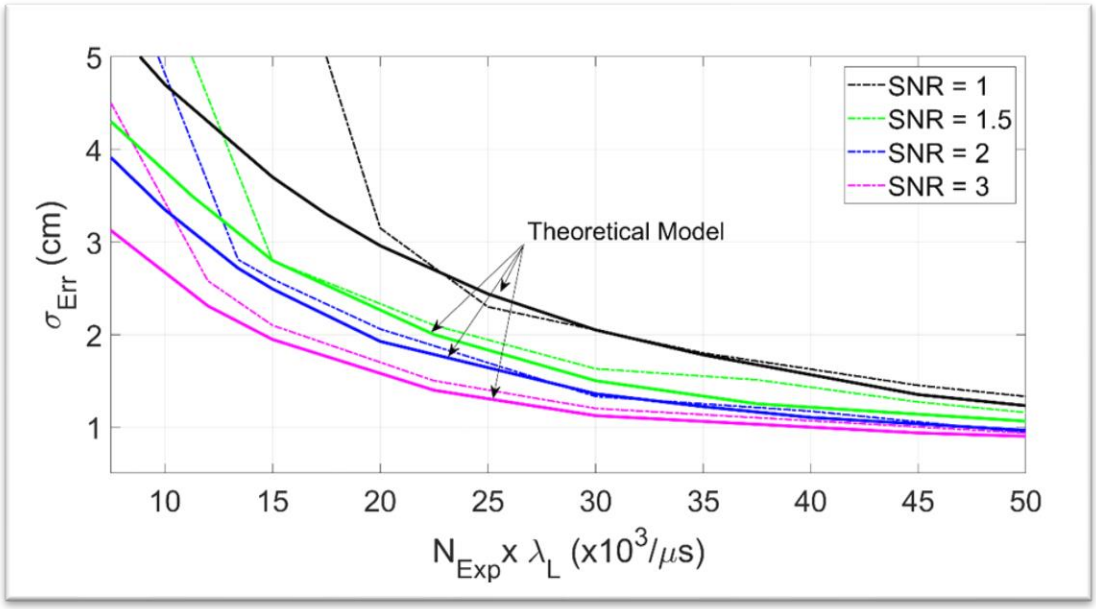

Figure 4.3: Simulated and Theoretical Curves of Depth Error for Various *SNR*s

4.2.4 The Gaussian Approximation for Depth Error

Having derived the bin-slip probability, we now look to see if we can approximate this expression to help derive algebraic expressions for distance errors. We do this by first creating an upper bound for the bin-slip probabilities, and then fit a Gaussian Distribution to the resulting statistics.

To form the upper bound, we first approximate $\boldsymbol{P}(\boldsymbol{Slips\ to\ the\ Left} = \boldsymbol{k})$ to be $P(Filter(k))$ and higher than all previous filter outputs, ignoring the times when the shift is even larger, since that happens a small percentage of the time. We further simplify the equation by realizing that since the values of terms in (4.1) decrease in successive order; that is:

$$[P(\Delta L_k > 0)] > [P(\Delta L_k + \Delta L_{k-1} > 0)] > [P(\Delta L_k + \Delta L_{k-1} + \Delta L_{k-2} > 0)] > \cdots > [P(\sum_{i=1}^{k} \Delta L_i > 0)]$$

a simple ***upperbound*** for probability, P_{UB}, of <u>*bin-slips to the left*</u> can be obtained by just computing whether $P(Filter(k) > 0)$. This gives:

$$P_{UB}(Slips\ L = k) = P\left(\sum_{i=1}^{k} \Delta L_i > 0\right); \qquad P_{UB}(Slips\ R = k) = P\left(\sum_{i=1}^{k} \Delta R_i > 0\right) \tag{4.11}$$

Notice that summations $\sum_{i=1}^{k} \Delta L_i$ and $\sum_{i=1}^{k} \Delta R_i$ can be thought of as comparing the photon counts in <u>*bins k times as wide as the unit fine-bin*</u>. This means that random variables $(\sum_{i=1}^{k} \#\boldsymbol{N_i})$ and $(\sum_{i=1}^{k} \#\boldsymbol{O_i})$ are multinomially distributed[21], each having the following single trial probabilities:

$$p_N(k) = \boldsymbol{k}.\lambda_N(1 - \lambda_N t_S)T_{FB}$$

$$p_{Ob}(k) = \boldsymbol{k}.\lambda_S(1 - \lambda_N t_S)T_{FB}$$

With mean and standard deviations derived based on (4.6) and (4.7):

$$\mu_N(k) = N_{Exp}.p_N(k)$$

$$\sigma_N(k) \cong \sqrt{N_{Exp}.p_N(k)}$$

$$\mu_{Ob}(k) = N_{Exp}.p_{Ob}(k)$$

$$\sigma_{Ob}(k) \cong \sqrt{N_{Exp}.p_{Ob}(k)}$$

[21] On condition that $p_i, p_j \ll 1$, the covariance between the random variable, X_i and X_j is much smaller than their variance, i.e., $Cov(X_i, X_j) \ll \sigma_{i,j}^2$, and thus, they may be assumed independent for analytical simplicity.

The slip probability depends on the difference between variables $(\sum_{i=1}^{k} \#\boldsymbol{N_i})$ and $(\sum_{i=1}^{k} \#\boldsymbol{O_i})$. Since the difference of Poisson distributions don't have nice formula, to make computing this simpler, we will assume that they are Normally distributed, which allows the mean and variance to be added and subtracted. We create a new random variable $\Delta\boldsymbol{k}$:

$$\Delta\boldsymbol{k} = (\sum_{i=1}^{k} \#N_i) - (\sum_{i=1}^{k} \#O_i)$$

With mean and standard deviation calculated as:

$$\mu_{\Delta k} = \mu_N(k) - \mu_{Ob}(k) \cong k \cdot T_{FB} N_{Exp}(1 - \lambda_N t_S)(\lambda_N - \lambda_S) = -\#L_p \cdot k$$

$$\sigma_{\Delta k}{}^2 = \sigma_N(k)^2 + \sigma_{Ob}(k)^2 \cong k.T_{FB} N_{Exp}(1 - \lambda_N t_S)(\lambda_N + \lambda_S) = (\#L_p + 2 \cdot \#N_p) \cdot k$$

where $\#L_p$ and $\#N_p$ are the expected number of laser and noise photons in a time bin aggregated over the run. Thus, $\mu_{\Delta k}$ is the negative of the expected value of laser photons in a bin *k* times as large as T_{FB} and $\sigma_{\Delta k}{}^2$ is the variance in the differences between the signal (noise + laser) and noise bins, so it is the expected value of twice the noise plus the laser photons in a bin *k* times as large as T_{FB}.

The Gaussian *PDF* for $\boldsymbol{\Delta k}$ has the form:

$$f_{\boldsymbol{\Delta k}}(\delta k, k) = \frac{1}{\sqrt{2\pi\sigma_{\Delta k}^2}} e^{-\frac{(\delta k - \mu_{\Delta k})^2}{2\sigma_{\Delta k}^2}}$$

Therefore, using (4.11):

$$P_{UB}(Slips\ L = k) = P_{UB}(Slips\ R = k) = P(\boldsymbol{\Delta k} > 0) = \mathbf{1} - F_{\boldsymbol{\Delta k}}(\mathbf{0}, k) = 1 - \int_{-\infty}^{\mathbf{0}} f_{\Delta k}(\delta k, k) d(\delta k)$$

Finally, the probability of bin-slips in terms of the ***Standard*** Gaussian Distribution is given by:

$$P_{UB}(Slips\ to\ the\ Left = k) = P_{UB}(Slips\ to\ the\ Right = k) = 1 - Z\left(\Delta k' \leq -\frac{\mu_{\Delta k}}{\sigma_{\Delta k}}\right) =$$

$$= Z\left(\Delta k' > -\frac{\mu_{\Delta k}}{\sigma_{\Delta k}}\right) \qquad (4.12)$$

Where:

$$-\frac{\mu_{\Delta k}}{\sigma_{\Delta k}} = \sqrt{\frac{k.T_{FB} N_{Exp} * (1 - \lambda_N t_S)}{\lambda_N + \lambda_S}} (\lambda_S - \lambda_N) = \sqrt{SNR \frac{k.\boldsymbol{\lambda_L} T_{FB} N_{Exp} * (1 - \lambda_N t_S)}{SNR + 2}} = \sqrt{\frac{SNR}{SNR+2} \cdot \#L_p \cdot k}$$

(4.13)

Equation (4.13) confirms our insight into how various parameters affect depth-error:

- The key parameters are *SNR* and the expected number of laser photons in each timing bin
- As long as the *SNR* is above 3-4, further increases in *SNR* has small effect
    1. Moving from *SNR*=3, to *SNR*=1, increases $\#L_p$ by 1.8x
    2. Moving from *SNR*=5, to *SNR*=3, increases $\#L_p$ by 1.2x
- The larger the number of bin slips (*k*), the lower is its probability
- Since $\#L_p$ depends on the noise level ($N_{Exp} \cdot T_{FB}(1 - \lambda_N t_S) \cdot \lambda_L$) the value of the noise level matters
- The farther the object (t_S), the larger the noise shielding and the higher is the slip probability
- For a given error-spread, σ_{Err}, the quantity $\frac{\mu_{\Delta k}}{\sigma_{\Delta k}}$ is constant. If we also assume that the noise flux, λ_N, is constant, since $\lambda_L = SNR \cdot \lambda_N$, then:

    1. For small values of *SNR*, $N_{Exp} \propto {}^{1}/_{SNR^2}$, to shrink $\#L_p$ by *SNR*.
    2. For larger values of *SNR*, $N_{Exp} \propto {}^{1}/_{SNR}$, to keep $\#L_p$ constant.

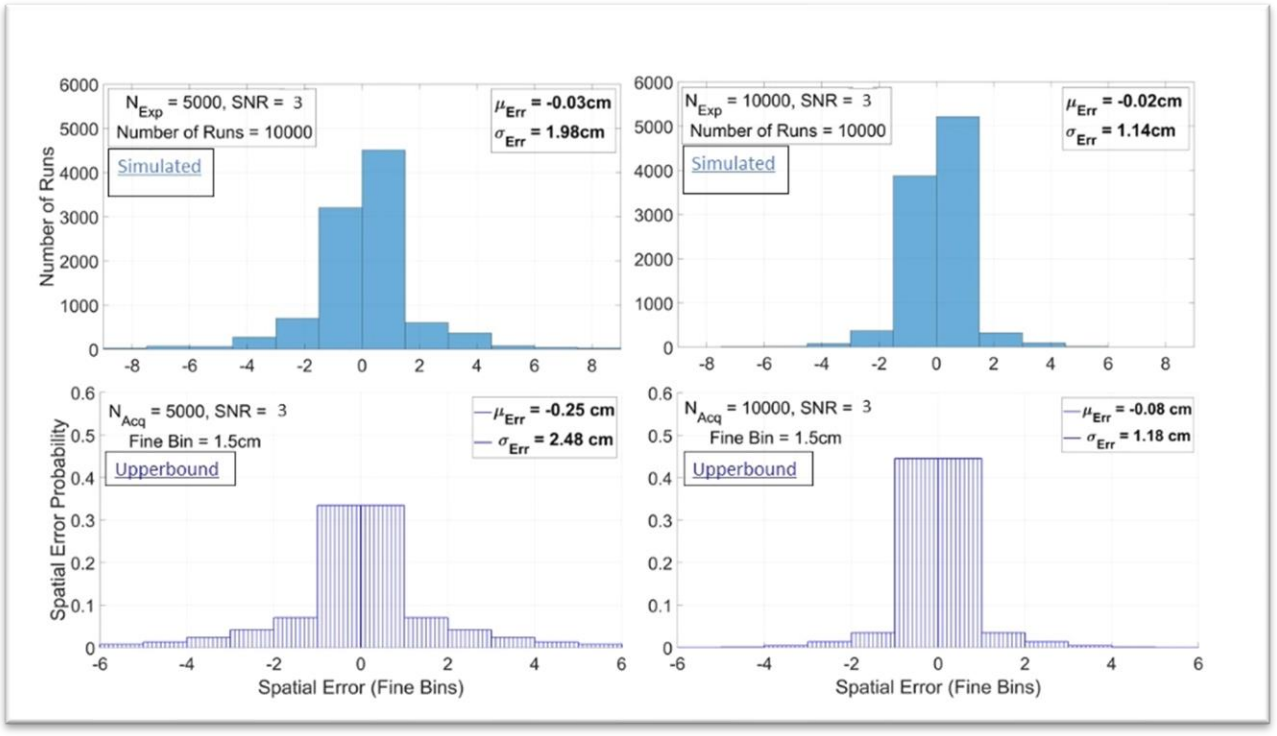

Figure 4.4: Simulated vs. Upper bound Histograms for Depth Error

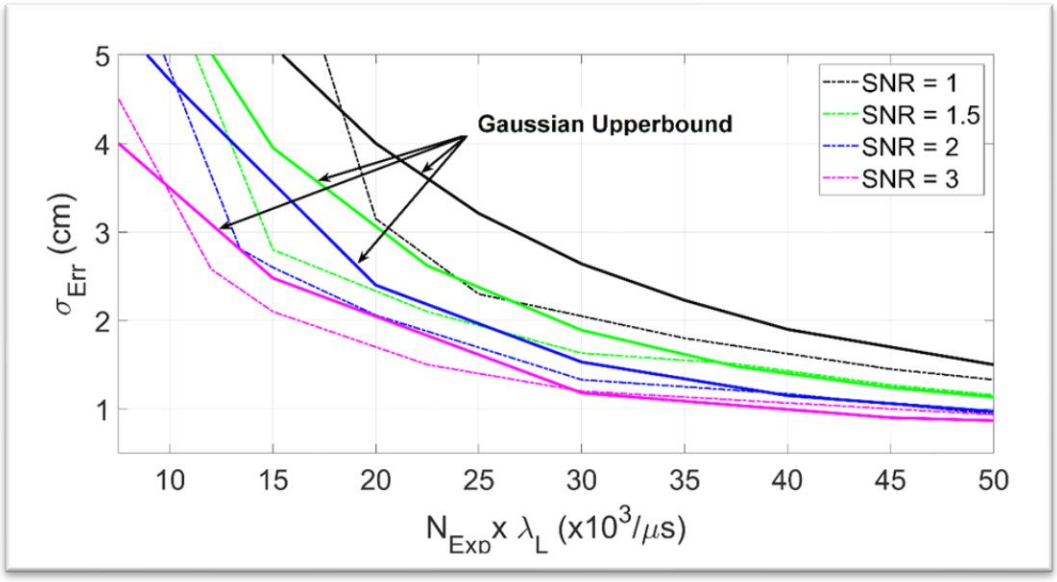

Figure 4.5: Upper bound Approximation to Depth Error

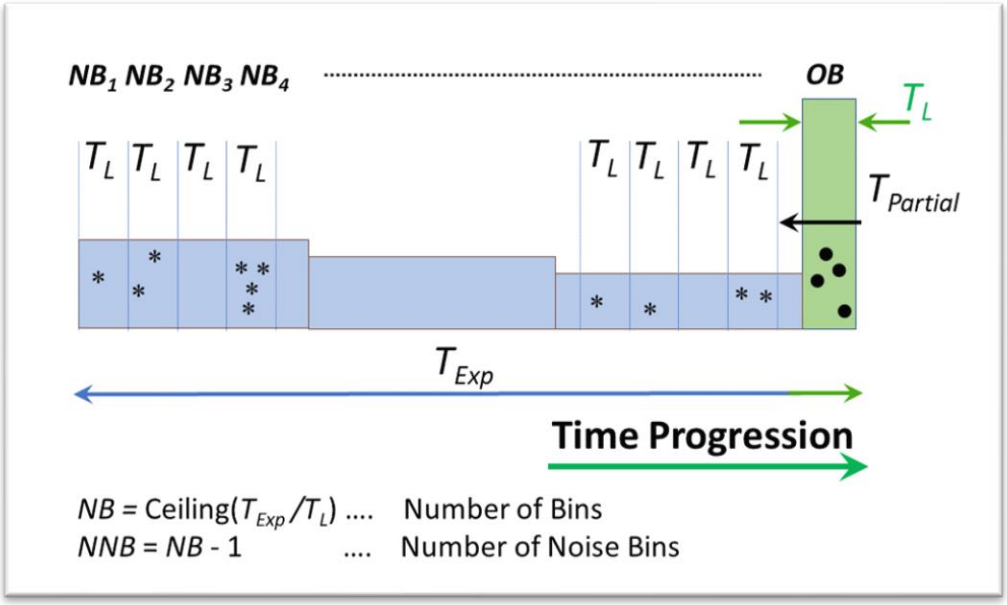

Figure 4.6: Break-up of Exposure Time into T_L-sized Bins

Equation (4.12) is used to calculate the upper-bound estimate of error-spread, the results for which are plotted in Figure 4.4 for $SNR = 3$, and two exposures with $N_{Exp} = 5K$ and $10K$. The error spread, σ_{Err}, is predicted to be within 25%. Figure 4.5 plots the distance error of our simulated results (Figure 3.10) versus this upperbound approximation. Note that at lower signal flux rates, the

upperbound curves fall below the simulated plots indicating that the simulated errors were corrupted by the presence of outliers.

4.3 False Detection Statistics

In Chapter 3, using a Poisson model we created a threshold to allow us to set error rate for false positive detections which in turn determined the rate for false negative detections. Using the accurate bin slip statistics, derived previously, this section looks at the minimum possible error rate as a function of *SNR* and total number of laser photons collected. To generate this minimum, rather than setting a fixed threshold, we will compute the probability that a collection of noise bins exceeds the value of bins which contain the returning laser pulse creating an uncorrectable "outlier" event. The outliers do not represent the error associated with a recovered object, but rather comprise failed detections, unrelated to object-position. What is important is to determine the minimum number of captured laser photons, or equivalently exposures for which the probability of false detection is below a threshold, for example 1%.

Let's consider Figure 4.6 where the exposure time (T_{Exp}) is divided into coarse bins, T_{CB}, equal in size to the laser pulse, T_L, $T_{CB} = T_L$. Thus:

- Total number of coarse bins: $NB = \text{Ceil}(T_{Exp}/T_L)$
- Number of ***noise*** Coarse: Bins: $NNB = NB - 1$
- ***Object*** Coarse Bin: ***OB***

We assign $\#NCB_i$ to the number of photons in the ***Noise*** Coarse Bin ***i***, and $\#OCB$ to the number of photons in the ***Object*** Coarse Bin. A false detection occurs if the photon-count in the Object Bin is *less than the count* <u>*in any*</u> of the Noise Coarse-Bins, $\#NCB_i$; that is:

$$P(False\ Detection) = P[(\#OCB < \#NCB_1) \vee (\#OCB < \#NCB_2) \vee (\#OCB < \#NCB_3) \ldots]$$

Or equivalently:

$$P(False\ Detection) = 1 - P[(\#OCB \geq \#NCB_1) \wedge (\#OCB \geq \#NCB_2) \wedge (\#OCB \geq \#NCB_3) \ldots]$$

$$P(False\ Detection) = 1 - P[\#OCB \geq Max(\#NCB_i)] \qquad \text{for } i = [1{:}\,NNB]$$

(4.14)

The number of photons in noise coarse-bins, *#NCB$_I$*, are identically distributed, and if assumed to be independent,[22] a simple formula for the distribution of ***Max(#NCB$_I$)*** can be derived. Assigning ***#NCB*** to all ***#NCB$_I$***, we have:

$$P(Max \leq n) = [P(\#NCB \leq n)]^{NNB}$$

Therefore (4.14)) is rewritten as:

$$\boxed{P(False\ Detection) = 1 - [P(\#NCB \leq \#OCB)]^{NNB}}$$

Random variables *#NCB* and *#OCB* are jointly multinomially-distributed:

$$P(\#NCB = \#n, \#OCB = \#o) = \frac{N_{Exp}!}{\#n!\,\#o!\,\#none!} p_N^{\#n} p_{Ob}^{\#o} p_{none}^{\#none}$$

where:

$$\#none = N_{Exp} - (\#n + \#o)$$

$$p_{none} = 1 - (p_{NCB} + p_{OCB})$$

And given Equation (3.4), the *single trial probabilities* of a photon landing in a *noise coarse-bin*, as well as that for the *object bin* are calculated:

$$p_{NCB} = \lambda_N(1 - \lambda_N t_S)T_L$$

$$p_{OCB} = \lambda_S(1 - \lambda_N t_S)T_L$$

Consequently, the probability that the number of photons in a noise-bin is less than the number in the object bin, as well as, the general formulation for false-detections can be expressed as:

$$\boldsymbol{F_{\#NCB}}(\#OCB) = \sum_{\#o=0}^{\boldsymbol{N_{Exp}}} \sum_{\#n=0}^{\#o} \frac{N_{Exp}!}{\#n!\,\#o!\,\#none!} p_{NCB}^{\#n} p_{OCB}^{\#o} p_{none}^{\#none}$$

$$\boxed{P(False\ Detection) = 1 - [\sum_{\#o=0}^{N_{Max}} \sum_{\#n=0}^{\#o} \frac{N_{Exp}!}{\#n!\#o!\#none!} p_{NCB}^{\#n} p_{OCB}^{\#o} p_{none}^{\#none}]^{NNB}} \quad (4.15)$$[23]

---

[22] $Cov(\#NCB_i, \#NCB_j) \ll \sigma^2_{\#NCB_{i,j}}$

[23] It should be noted that in the final step, N_{Exp} has been replaced by N_{Max} for computational efficiency:

$N_{Exp} \rightarrow N_{Max} = \mu_N + 6\sigma_N$

Figure 4.7 illustrates the probability of false detections as a function of the number of exposures for the familiar *SNR* values, and $\lambda_N = 1/\mu s$. Figure is based on the maximum distance of interest at 20m (T_{Exp} = 133ns) and 33 noise bins (T_L = 4ns). As false detections are to be rare, the relevant probabilities must be much smaller than 1. Using the data from Figure 4.7, Table 2 tabulates the number of exposures yielding error probabilities of 0.001, 0.01 and 0.1. Table also includes the depth uncertainty corresponding to these probabilities for *SNR* = 1 (For proper comparison, the influence of outliers on depth error is removed). Determination of N_{Exp} is based on the acceptable number of false detections *and/or* the allowable depth-error. For example, error for ***distant*** objects can be allowed to be larger when compared to those closer to the sensor, in which case, N_{Exp} is determined by the probability of ***false frames***. On the other hand, ***near*** objects necessitate smaller distance error and thus N_{Exp} is calculated based on the required ***error tolerance***.

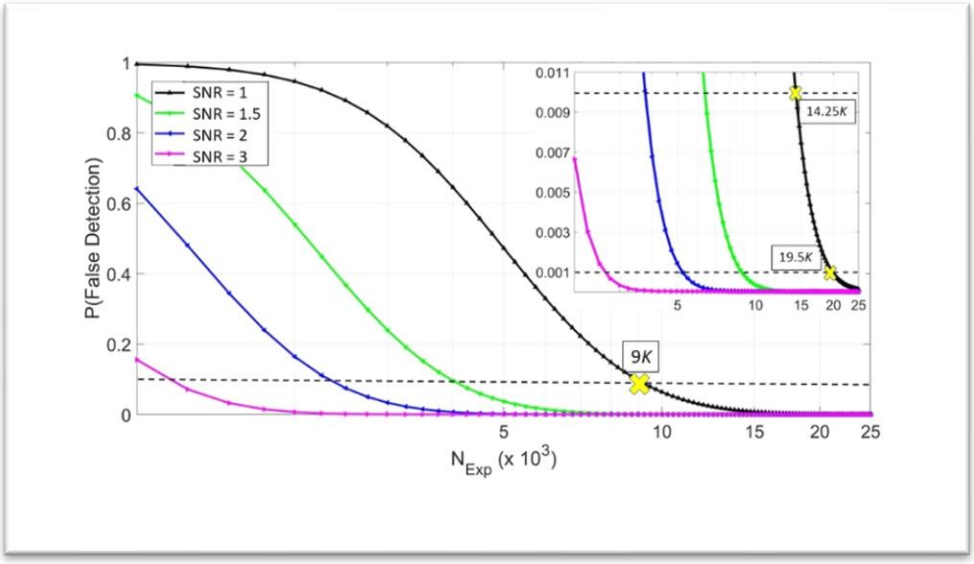

Figure 4.7: False-detection Probability vs. N_{Exp} for D_{Max} = 20m

P(False Detection)	***SNR***	***N_{Exp} (x10^3)***	***# L_p***	***σ_{Err} (cm)***
0.001	1	19.5	1.95	2.9
	1.5	8.75	1.3	
	2	5.25	1.1	

	3	2.75	.8	
0.01	1	14.25	1.4	3.9
	1.5	6.35	0.95	
	2	3.75	.75	
	3	1.90	.57	
0.1	1	9.0	0.9	5.4
	1.5	4.0	0.6	
	2	2.25	0.45	
	3	1.10	0.33	

Table 2: N_{Exp} and SNR for False-detection Probabilities of 0.1, 0.01 and 0.001

4.4 Summary

In this chapter presents, based on the multinomial distribution, a mathematical formulation for depth uncertainty of a recovered object, as well as conditions leading to *false-detections*. The formulation agrees with the simulated results to within 5%. To gain insight into conditions affecting the uncertainty in depth measurement, an upper bound Gaussian approximation to the multinomial distribution was derived. The upper bound formulation sheds light on the relation between *SNR* and N_{Exp} for a fixed error-spread, σ_{Err}, and can explain the observed dependency between depth uncertainty, SNR, and the expected number of photons in each timing bin. The accuracy of this estimation is to within 25% when compared to the simulated results.

Chapter 5. Image Sensor

In this chapter we provide a detailed description of a scalable, all-digital, column-parallel, 64x48 pixel ToF Flash image sensor. The image-sensor is implemented in TSMC's 130nm HV, 1.5V technology with 6 metal layers. The nominal FO4 delay for this process is 60ps. The SPAD pitch is 50µm x 50µm. The sensor is organized in two independent 32 x 48-pixel arrays; the block diagram is shown in Figure 5.1.

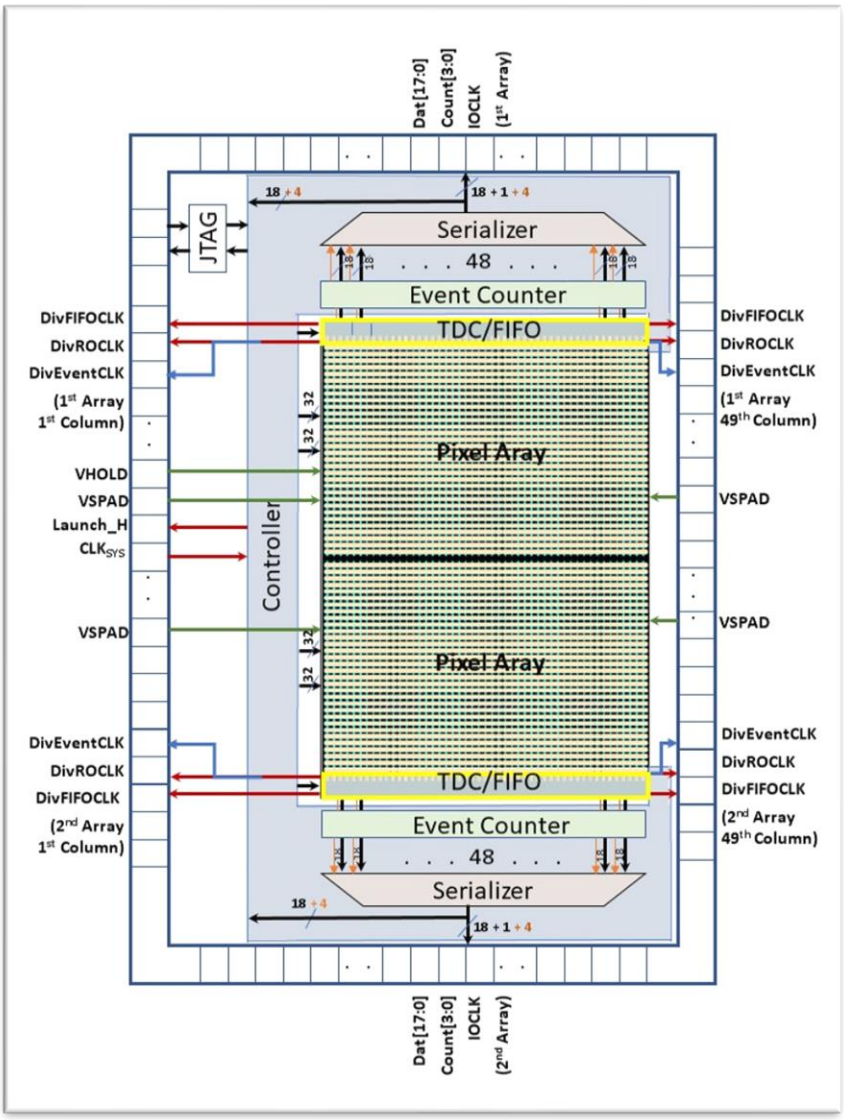

Figure 5.1: The Image Sensor Block Diagram

The design focuses on addressing some of the issues associated with designing a 3-D (A 2-D array of pixels, each pixel measuring the distance to the object) Flash ToF sensor, particularly how to get good pixel fill-factor, high column repetition rate and timing accuracy in a scalable design. In contrast to ToF image sensors where the TDC is incorporated within the pixel [42] which allows for a fully parallel operation, in order to improve fill-factor, this design uses a column parallel architecture [43, 44, 45, 46], where pixels on a column share common timing and address busses which propagate the spatiotemporal data of photonic events downstream to backend electronics on a per column basis.

The design of this imager is unique in many ways. First its internal operation is fully asynchronous, using a type of self-resetting or post-charge logic [47, 48] to maximize performance and minimize circuit overhead. The imager uses a single external clock source, around 500MHz, to control the IO data-transfer to a synchronous receiver. Unlike any other image sensor ICs, there are no analog components, nor any high-frequency global signals present in this design.

The design contains 4 main circuit structures, plus the controller and IO and each will be described in more detail in this chapter. The pixel circuits contain a SPAD and the active quench circuitry that we use to optimize its performance. It also contains the circuitry required to drive a pixel column. The pixel column aggregates information from a number of pixels, and delivers that information to the TDC, which needs to handle the fact that the photon arrivals are asynchronous, and might cause circuits to become metastable. Each column can unambiguously detect photonic events within a range of 20m with a sub-centimeter resolution. The column dead-time, or repetition period; that is, the minimum time between photonic events on a column that are distinguishable, and thus produce valid spatiotemporal information, is nominally 1ns effecting a column detection rate of approximately 1GHz.

In the case that events violate this minimum time, the information for the first event is preserved, while that for the second event is discarded. In contrast, the design presented in [49] uses collision detection circuitry to identify and discard the corrupted data without the ability of preserving any useful information from temporally proximate events.

The TDC timestamps the photon's arrival on the column and employs an un-calibrated free-running ring-oscillator whose frequencies are measured by timestamping multiple assertions of the system clock. Since the TDC timestamps the arrival time on the column, the system is designed to calibrate out the delay between the SPAD firing and the information arriving at the bottom of the column. These timestamped events are then passed to an asynchronous FIFO which stores the spatiotemporal data for post processing. The writes to the FIFO are event driven; that is a FIFO is written only when an event is detected on its column.

The design also includes some circuitry for I/O and testing, comprising a high-speed bus to communicate the image data to the host processor, and a command and control interface to configure the chip.

5.1 Self-resetting Logic and Signal Naming Convention

This section briefly reviews the operation of self-resetting logic, as the design of the image sensor heavily uses this logic family. All self-resetting signals start from a reset value. When triggered, they either assert, and momentarily change their value to their assertion value, or retain their reset value. If they assert, after a short time period the gate will reset itself to its reset value which is why they are called self-resetting circuits.

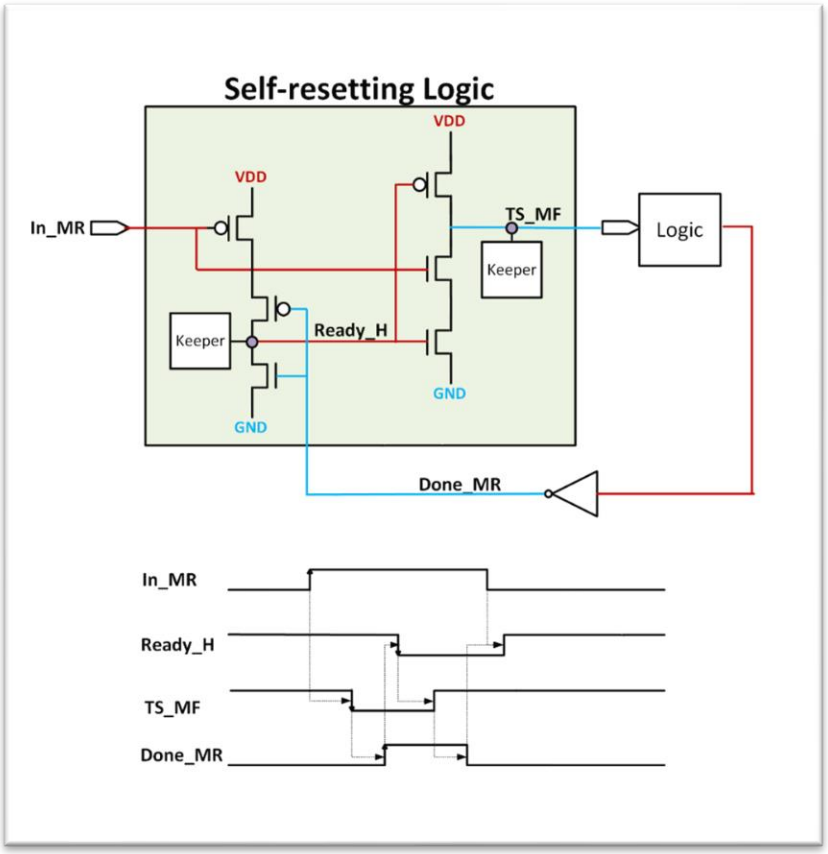

Figure 5.2: Example Diagram of Self-resetting Logic

Since all signals are pulses, we use a naming convention that indicates the reset and active values. Signals with names appended by _MR (or _MF) are monotonically rising (or falling) during their active period where the active edge is rising (or falling) respectively. This defines the reset value of the signal; 0 for _MR and 1 for _MF, but not their logical sense, which is denoted by _T, matches the value, and _C matches the complement of the value. Thus monotonic differential signal-pairs (True and Complement) are assigned names appended with _TMR/_CMR or _TMF/_CMF.

Figure 5.2 shows an example self-resetting logic circuit. All self-resetting gates are monostable circuits: once triggered, their output is reset after some delay. To ensure that the pulse is not too short, often, like illustrated in this figure, the reset is initiated by a signal that is generated by the evaluation of a downstream logic gate. In its quiescent state, monotonically rising signals, In_MR and Done_MR are both low, rendering the Ready_H node high. On the rising edge of In_MR, monotonically falling signal, TS_MF immediately discharges, and evaluates the subsequent logic block. As a result, Done_MR rises and grounds Ready_H which resets TS_MF. After some delay, Done_MR also resets to ground ending the self-resetting process. The circuit is ready to be triggered again only if In_MR has returned to ground charging Ready_H to VDD. From this vantage point this circuit is edge-triggered.

5.2 The Pixel

The pixel comprises a SPAD, the quench and recharge (AQR) and event-detection circuits. The goal of this circuit is to briefly drive the column with its row address each time the SPAD fires. This information should be on the column wires only long enough for the next stage to sense its value. Once this occurs the circuit self-resets and stops driving the bus.

Figure 5.3 illustrates the pixel circuit used in this design focusing on the event detection circuitry. The quench and recharge circuitry was shown in Figure 2.3 in Chapter 2. The output of this circuit then feeds into a mux that drives the detection circuit in the light green which also is in each pixel cell; the five NAND gates together drive Event_MR; they are distributed in the column circuity, as explained in the next section, and are shown here to understand the pixel's operation. The detection circuit receives the quench and recharge output of the activated i^{th} pixel**,** $SPAD_i$_MR.[24] In its reset-state, $SPAD_i$_MR, and Event_MR are both low, forcing the state-node, $State_i$, high. The five pairs of differential address lines, A[0-4]TMF and A[0-4]CMF have been precharged high, and are held at VDD by weak keeper devices since under these conditions, PU = 1, PD_i = 0, and the address-lines are not driven by the pixel circuits.

[24] SPAD_MR signals may be bypassed to electronically control the pixel array for calibration and test purposes by way of the Cal (calibrate) signal. This is described in more detail in Section 5.4.

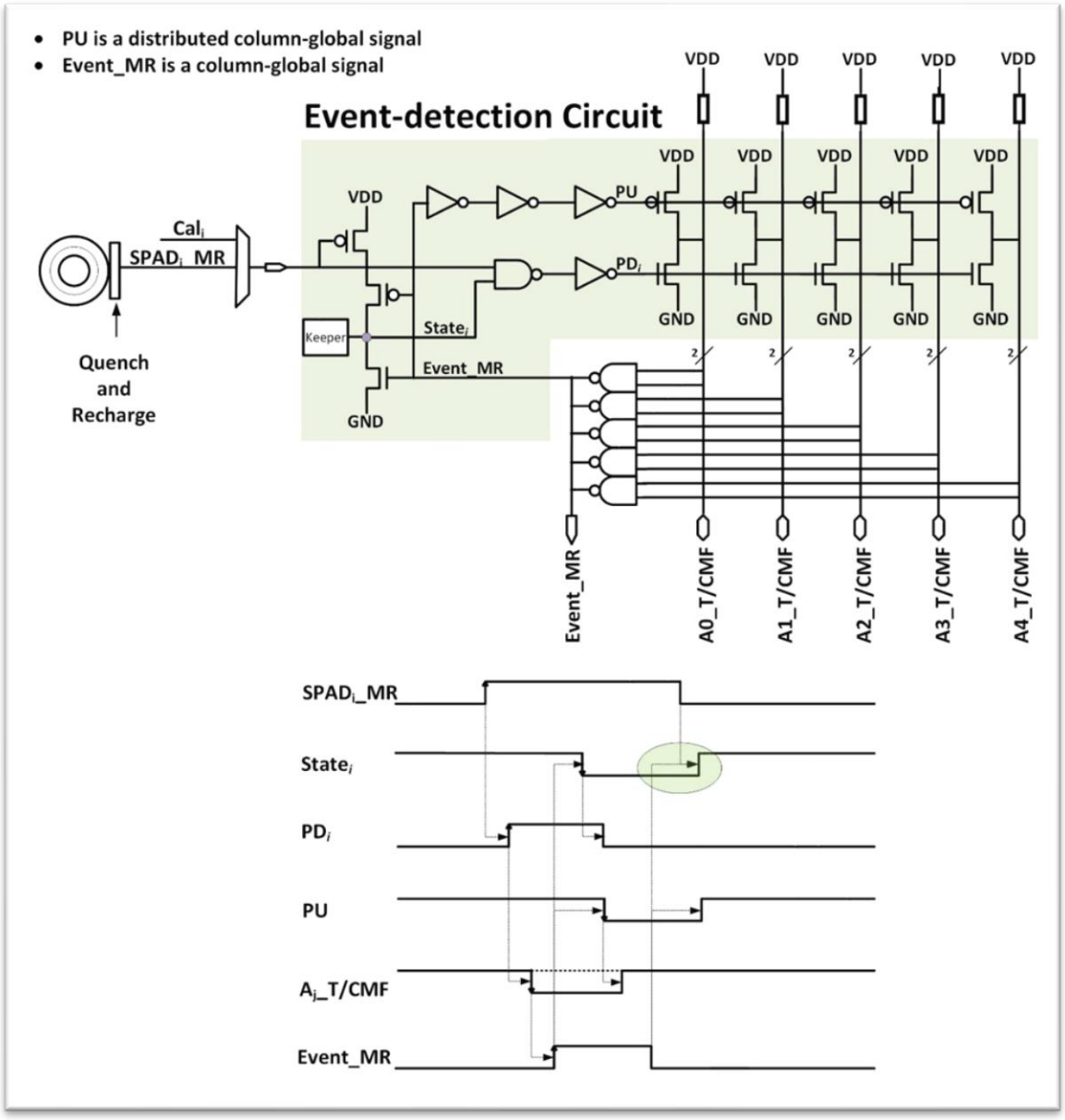

Figure 5.3: Pixel Event-detection Circuit

The assertion of $SPAD_i$_MR, initiates the circuit operation. With $State_i$ = 1, PD_i immediately transitions high, driving the row address of this pixel onto the differential address lines, causing one line of each pair to fall. Gates looking at these lines (the NAND gates in Figure 5.3) detect this change and assert a monotonically-rising signal, Event_MR. This signal then starts the reset of this pixel driver by immediately grounding $State_i$, which turns off the pull down driver, and asserts PU which starts the precharge of the differential address lines.

Note that Event_MR drives the state nodes of all other pixels on the column low, disconnecting those pixels from the column; i.e., $PD_i \rightarrow 0$, i = [31:0] and asserting PU in all the cells. Thus after the driven address has been received, PU transitions low in all the cells, post-charging the address-lines to VDD. Since all cells contribute to the post-charge, the size of the PMOS devices in each cell should be small. Once the columns are pulled to VDD, Event_MR transitions low, and PU de-asserts ($PU \rightarrow 1$) ending the post-charge operation, and again rendering the column responsive to new events.

An important point is to be made here. At the end of the aforementioned process, when Event_MR $\rightarrow 0$, the state-nodes are weakly held at ground. To activate the event-detection circuit of a pixel, its associated $SPAD_i$_MR must be at ground first so as to set the state-node to 1 (green oval in the figure). This arrangement ensures that the detection circuit is only triggered by a $0 \rightarrow 1$ transition of the $SPAD_i$_MR signal. Even if another SPAD on the column fired during this period and is still asserted when Event_MR falls, that photon event would be dropped, rather than reported with erroneous timing. This feature preserves the precise timing of reported events.

The event-detection circuit described here is monostable and belongs to the aforementioned digital circuits known as self-resetting logic. The circuit does not reset to its quiescent state until the succeeding stage has successfully transitioned. In this case, the event-detection circuit remains in its active state until the corresponding address lines have successfully discharged. This is contrast to the self-timed monostables used in a number of image sensors [50, 51] where the duration of the pulses is not directly related the detection circuit and the down-stream circuitry and thus requires delay-padding to avoid reliability problems.

Figure 5.4 shows the layout of the Pixel cell including the SPAD, the AQR and the Event-detection circuit. The SPAD occupies the largest percent of the pixel area and is circular to avoid any field enhancement at corners. While the area of the SPAD is large, occupying roughly 30% of the cell area (50μ x 50μ), the fill factor of the resulting cell is still a modest 12.5%. The main issue is the need for large spacing between the high voltage photodiode and the supporting circuitry, as well as spacing restriction to minimize cross-talk between neighboring pixels. The differential address (in green) lines run vertically on the right-hand side of the SPAD then jog further to the right to run over the detection circuit, where the NMOS and PMOS address drivers are located. After passing over these drivers, they jog back to the left, to run between the next row of pixel cells.

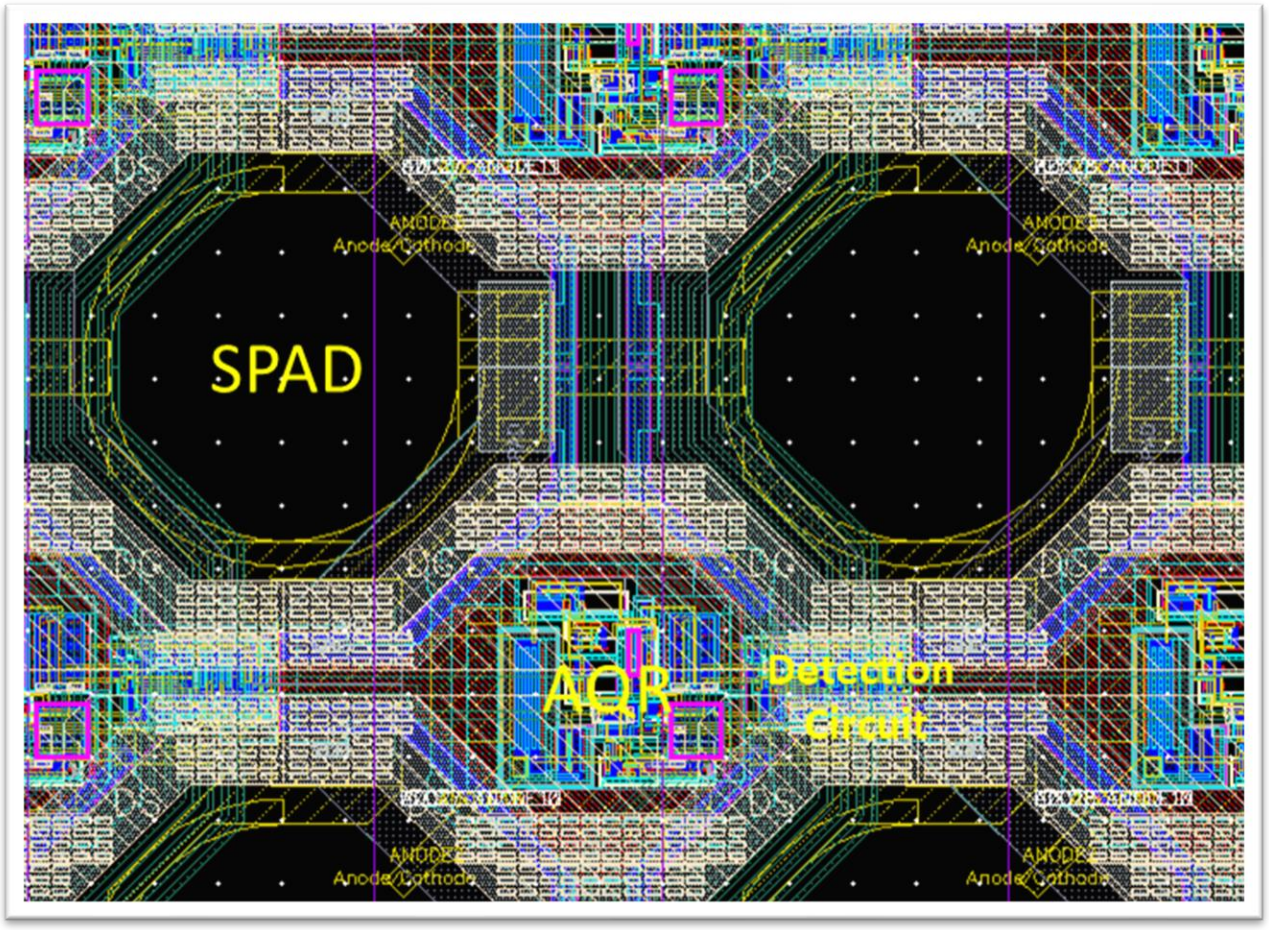

Figure 5.4: Pixel Layout

5.3 The Pixel Column

Figure 5.5 illustrates the pixel column in conjunction with its interface to the TDC and address latches. Pixels in each column share 5 pairs of monotonically falling differential address lines, A_i_CMF/A_i_TMF, i = [4:0]. These address lines encode the spatial information of an activated pixel, where they are transmitted to 5 self-strobed latches, described in Section 5.3.1, which statically store the corresponding spatial address, Add[4:0], of the event until a new distinguishable event is detected.

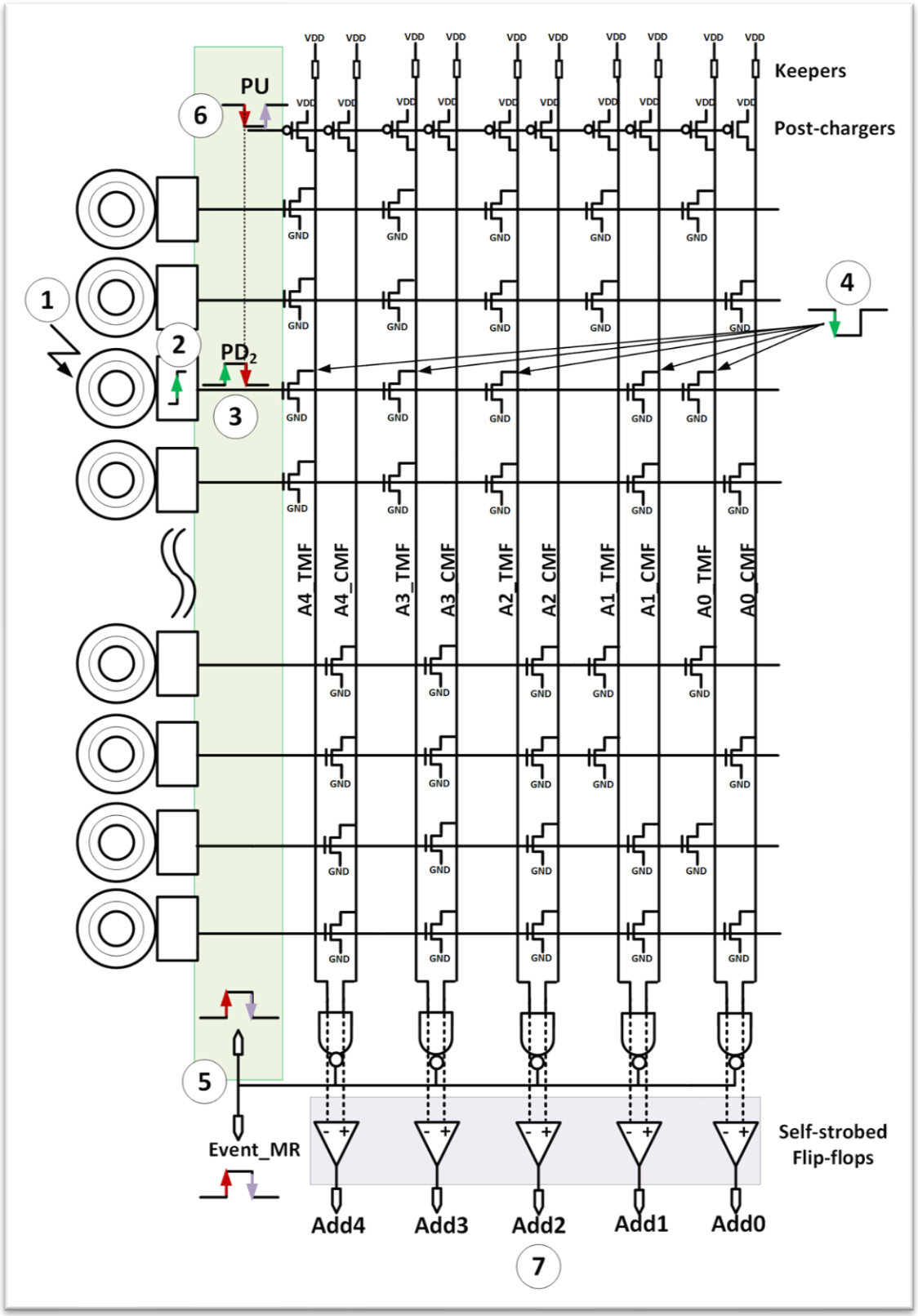

Figure 5.5: Simplified Schematic of the Pixel-column

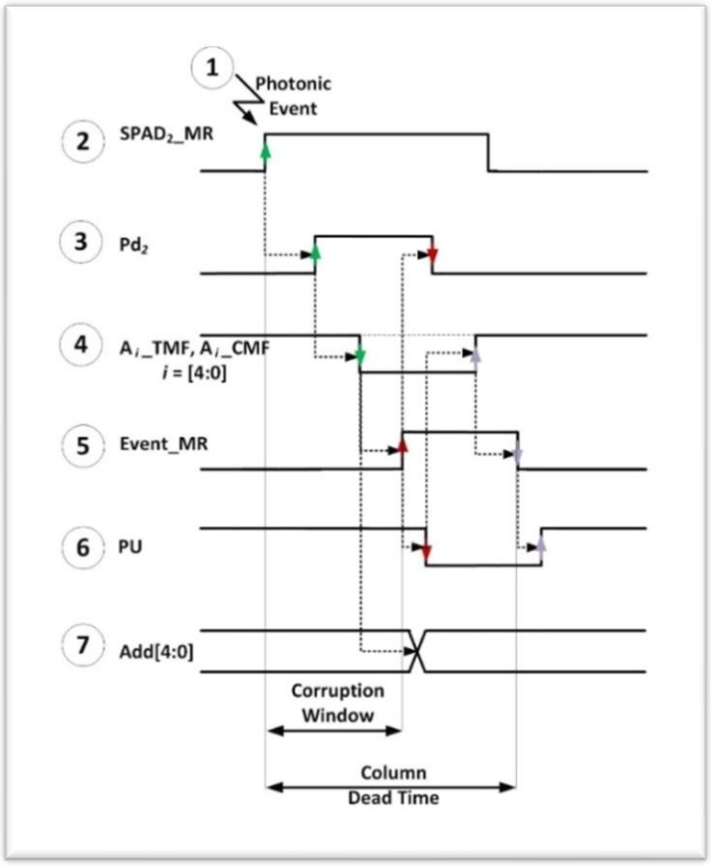

Figure 5.6: Timing Diagram for Column Response to a Photonic Event. The numbers correspond to the labeled nodes shown in Figure 5.5.

As discussed before, address lines also generate a monotonically rising signal, Event_MR which imbeds the timing information of the event. As this signal is an input to all the pixels, as well as to the TDC; it has a large fanout. To generate this signal with the smallest possible delay we short together the outputs of the 5 NAND gates that detect the evaluation of each address line. Since all bits of the address are driven at the same time, this increases our drive strength by 5x with little penalty. It does require that the timing of the address lines need to match well to prevent shorting gate outputs, but those delays match by layout symmetry.

The reset sequence, described in the previous section, includes disabling the impinged pixel, as well as blocking the access of other pixels to the column address lines (Event_MR $\rightarrow$ 1, $State_i$ $\rightarrow$ 0) to enable post-charging the address lines. On completion of column post-charge, Event_MR falls. A simplified timing diagram for the self-resetting operation of a pixel column from reception of a photonic event to de-assertion of Event_MR is included in Figure 5.6.

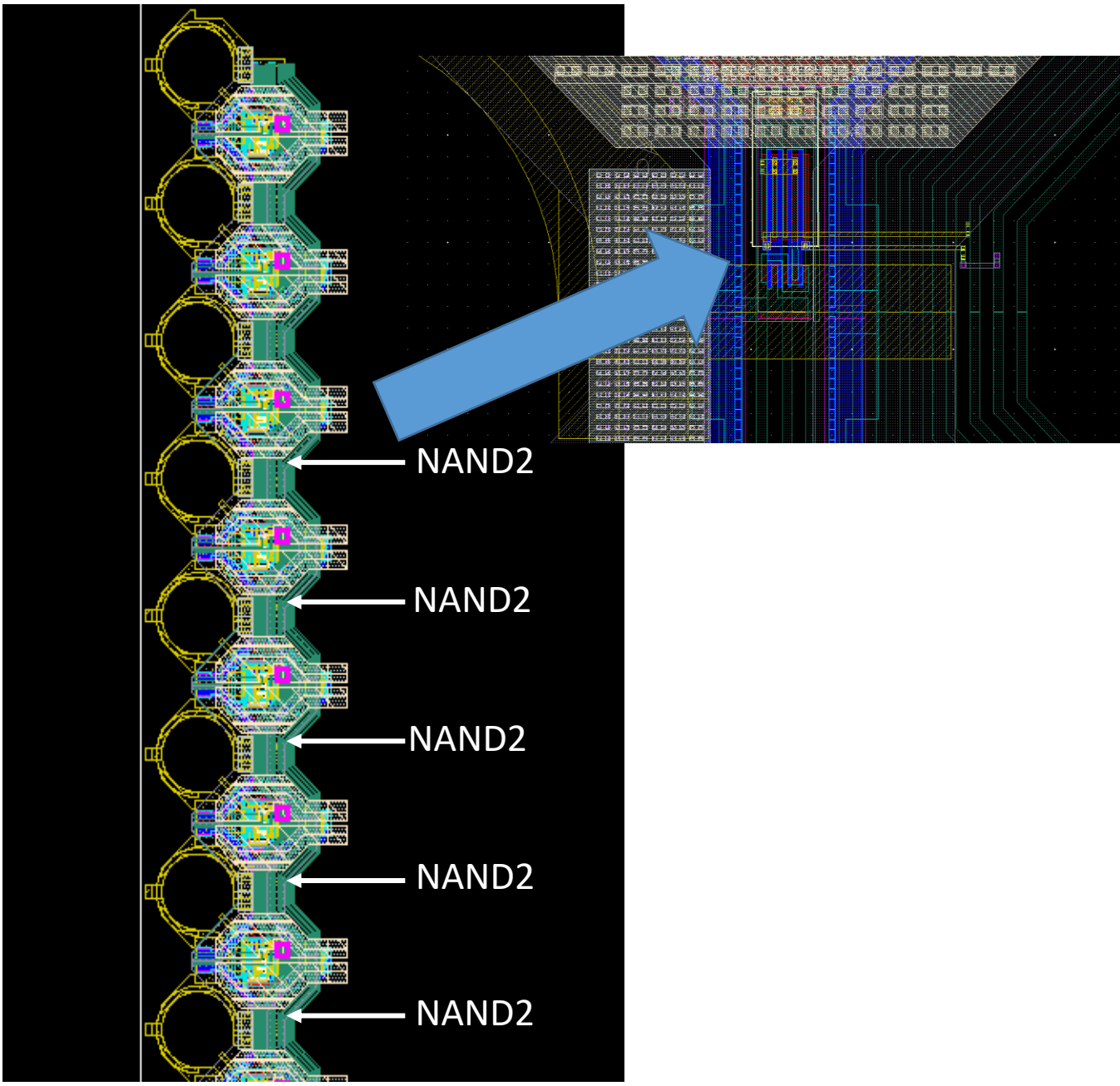

Figure 5.7: Layout of 8 Pixels in Column with Blow-up Showing NAND gate

In addition to distributing the precharge circuits in each pixel, the NAND gates that generate Event_MR are also distributed, and each column of 8 pixels contains a set of the 5 NAND gates, as is shown in Figure 5.7. There is room for a NAND gate under each pixel's AQR circuit. Distributing these NAND gates both saves area, and minimizes the added delay caused by the RC of the wires.

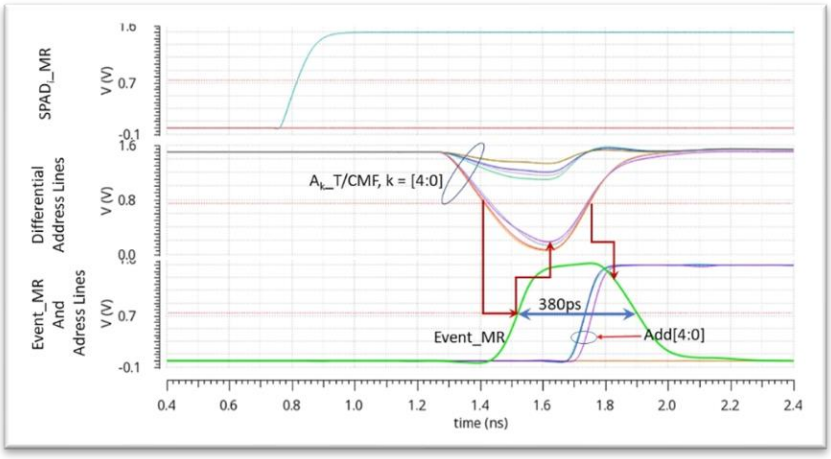

Figure 5.8: Column Response to a Single Photonic Event

Figure 5.8 shows a simulation of the key column waveforms for a single photonic event, starting with the initiating SPAD_MR event. It takes around 500ps before the pixel begins to drive the address lines. About 100ps later one of each of the address line pairs has fallen to VDD/2. In response to the monotonic address lines falling, Event_MR is asserted 100ps later. At the same time, self-strobed latches convert the monotonic signals to their static form, Add [4:0]. These signals remain stable until a new event impinges the column. Assertion of Event_MR immediately begins the recharge of the address lines to their quiescent state. In turn, Event_MR de-asserts, ending the sequence, at which point the column is ready to process a new event. The energy expended per column on detection and transfer of event data is 31pJ.

While the column operation is not complex, the asynchronous nature of photon arrivals complicates the requirements and analysis of this circuit. One needs to analyze what happens when two photons arrive at different pixels on the same column at similar times. There are two times that we would like to extract. The first is the minimum time between two photon arrivals that ensures that both pixels are cleanly detected. If photons hit the column with this spacing or larger, both photons are detected with no errors. The second time is the minimum separation between photons that allows the first photon to be detected without corruption. Since this spacing is less than the first timing, it means that the second photon will not be detected.

Reception of a photonic event by the i^{th} pixel produces the AQR signal, $SPAD_i$_MR, whose rising edge activates an event-detection circuit. The delay from this edge to the assertion, subsequent de-

assertion of Event_MR, and recharging of the node State to VDD defines the column dead time[25], where new photonic events impinged on the corresponding column are undetectable, and sets the minimum separation between photon detections that can be cleanly detected.

The post-extracted column dead-time in nominal PVT is measured to be ~980ps. That is, events with spacings greater than the dead-time are distinguishable and their spatiotemporal information can be correctly measured. As seen in Figure 5.9, two events at 1ns (> 980ps) apart separately discharge their respective address lines, producing their corresponding Event_MR and the latched pixel address.

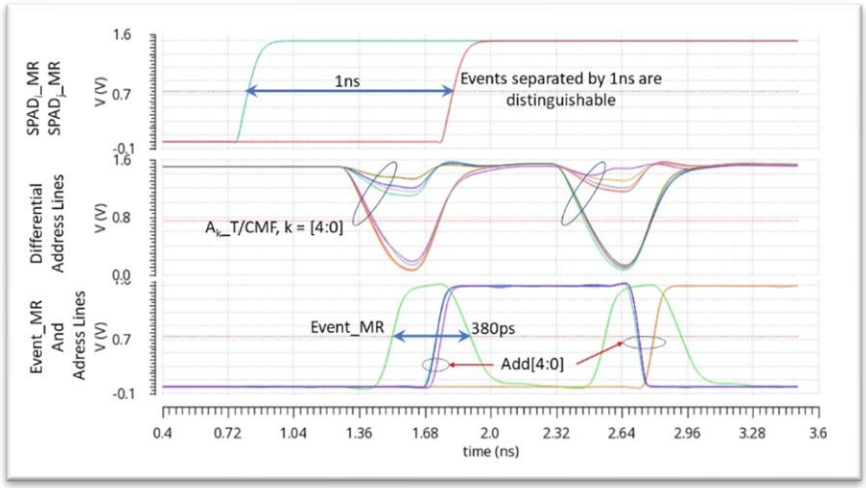

Figure 5.9: Distinguishable Events – Both Events Are Detectable

On the other hand, events with spacing less than the dead-time are not distinguishable. In this case, the spatiotemporal information of the earlier event is correctly captured and the information for the 2nd event is incorrect and discarded. Figure 5.10 shows such a scenario where the event spacing is 960ps (<980ps) violating the dead time. With this condition, the column's response to the 1st event is identical to that in the Figure 5.9, however, as the dead-time is violated, the second event is prevented from generating an Event_MR and a corresponding pixel address.

[25] The actual dead time is smaller by the delay through the input multiplexer that SPAD_MR flows through. What is essential is that the signal doesn't enter the Event-detection circuit, until the State node has been restored, but since there is delay through the multiplexer, the actual photon event can happen slightly before the node State has been driven high.

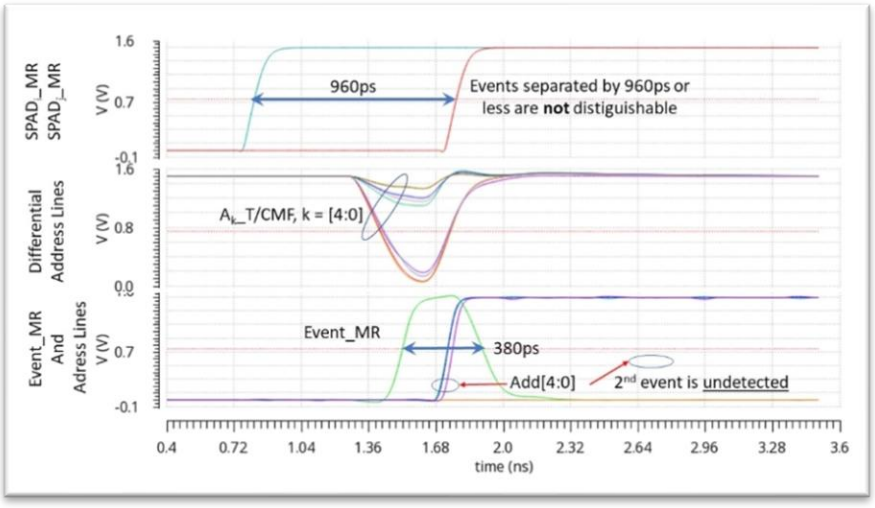

Figure 5.10: Indistinguishable Events – Only First Event is Detectable

It should be noted that the arrival of a second photonic event which starts driving the address lines prior to the assertion of Event_MR will corrupt the state of monotonic address lines, since it will start driving another address onto these lines. As the lines are precharged high, this can cause both bitlines in a pair to fall. A novel latching construct, the self-strobed latch, described in the next section, preserves the address of the 1st event, despite the corruption of the monotonic lines.

5.3.1 Self-strobed Latch

A cross-coupled NAND-pair circuit [52] can always be used to convert the differential monotonically falling signals of a dynamic structure to a static output signal, so long as the complementary nature of the inputs in the evaluation phase is guaranteed (Figure 5.11 a, transitions 1 and 2). And, in the case of column address line pairs:

Precharge phase: A_i_CMF = A_i_TMF = 1;

Evaluation phase: A_i_CMF = ~A_i_TMF.

However, if the complementary nature of the inputs in evaluation phase is not preserved, the conversion to a static signal will produce an erroneous output (Figure 5.11.a, transition 3).

As was mentioned in the prior section, this condition can arise in our design because of the random arrival times of photons on columns of the pixel-arrays. If events are far apart in time, then the differential address-lines, A_i_CMF/A_i_TMF, i = [4:0], are truly complementary. On the other-hand, if two events impinge a column within a small time-window (from one pixel starting to drive the

address lines, to Event_MR → 1 and disabling the other pixels), both pixels can drive the lines and it is possible that both lines in a pair discharge; the differential address-pairs are no longer complementary.

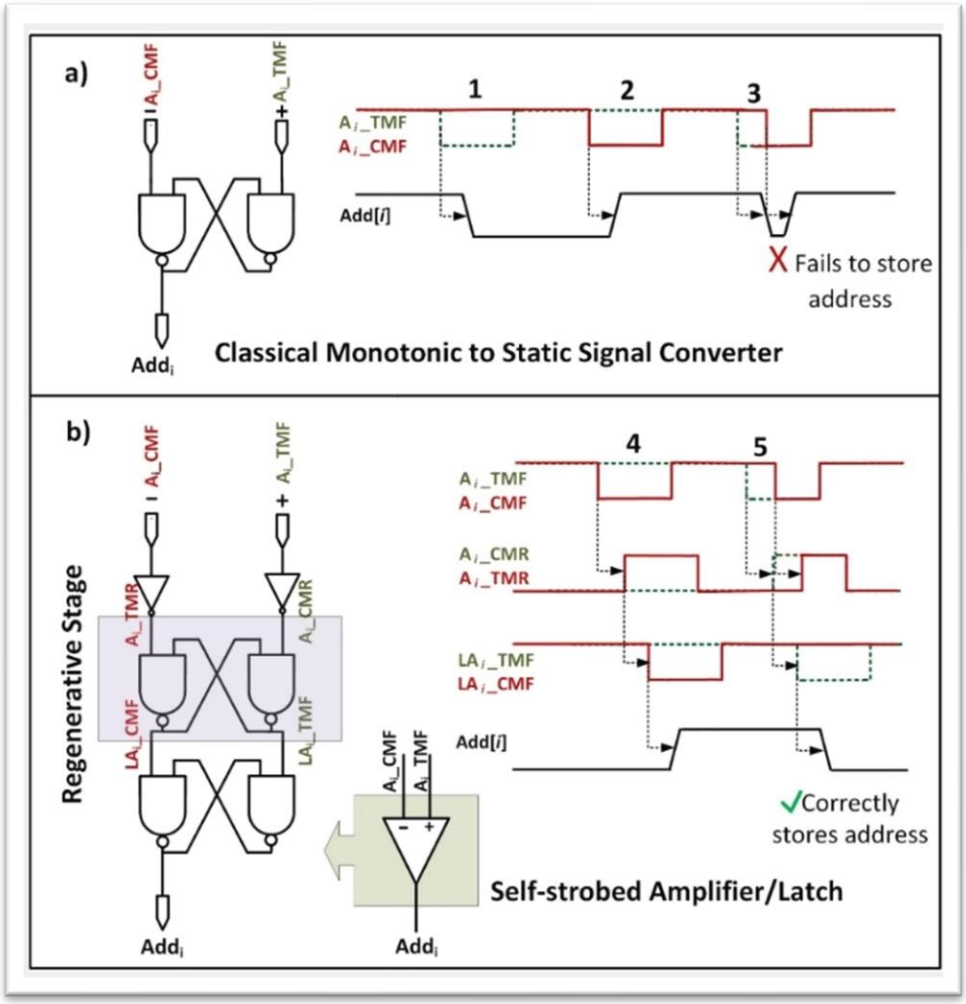

Figure 5.11: The Self-strobed Latch

To address this problem, the self-strobed latch circuit shown in Figure 5.11.b, includes a second regenerative stage. When the address bit-lines are in pre-charge, its inputs (A_i_TMR, A_i_CMR) are low, rendering the outputs, LA_i_TMF, A_i_CMF, high. This added latch captures which address line falls first, if they both transition low. Transition 4 in Figure 5.11.b shows the generic single event waveforms, while transition 5 comprises two closely spaced events. In this example, A_i_TMF falls first, forcing A_i_CMR to rise, grounding LA_i_TMF while LA_i_CMF continues to be high. At this point, the delayed discharge of A_i_CMF does not affect the state of the latch. Excepting the

condition where events occur at the same time[26], the self-strobed latch described here, correctly resolves the address of the first of closely-spaced events on a column, rejects new events within the dead time, and prevents the corruption of the data.

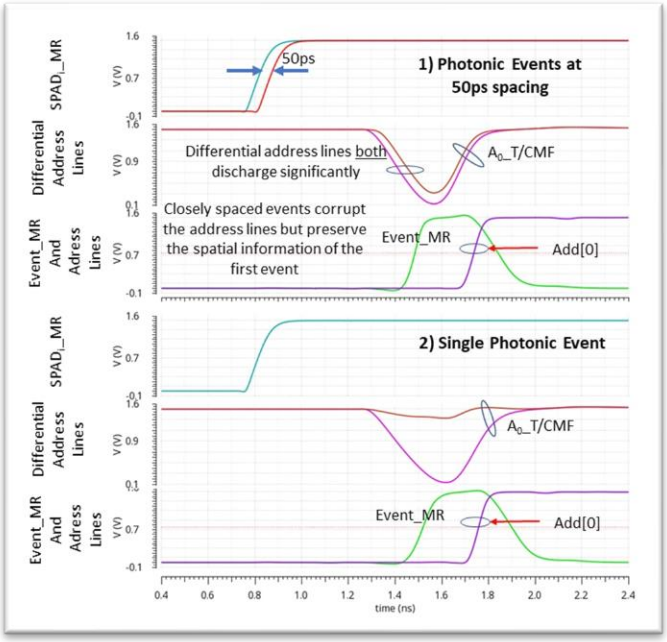

Figure 5.12: Indistinguishable Events with Address Corruption – First Event is Detectable

Figure 5.12 (1) shows such a simulation of this condition, where in response to two closely-spaced events, both complementary address lines A_0_CMF and A_0_TMF fall with a time difference of approximately 50ps. In response, the latched address, Add[0] asserts. To determine the validity of this outcome, we compare it to the latched address of a single event. As seen from Figure 5.12 (2), in this case, the monotonic nature of address line is preserved, producing a latched address, Add[0], which also asserts. This result proves that the self-strobed latch succeeds in correctly determining the correct address of the first event, even though the corresponding monotonic address are corrupted.

[26] Because of device mismatch, there is a low probability that events a few pico-seconds apart may produce erroneous spatiotemporal data.

5.3.2 Column Metastability Signature

As one can expect, due to the random nature of photon arrival times, and the monostable nature of the circuits used, the image sensor can experience metastability. Though infrequent, it is important to investigate how metastability affects the operation of the sensor and the system. In the case of sensor columns, two timing conditions can lead to an indeterminant sensor state. The first is related to the closely spaced photon arrival that we just discussed. If two photons arrive at exactly the same time, at least the differential lines of one bit of the address will fall at the same time. This situation can cause our self-strobed latch to become meta-stable. Since this latch output is only stored in a FIFO and doesn't affect timing or control, it can't cause any problems for the overall system.

The more troubling case comes when the node State in Figure 5.3 goes meta-stable. This situation occurs when a photon arrives after Event_MR falls and before the value of the node State reaches VDD. The keeper on this node will eventually drive the node to VDD or VSS, but it can stay in a meta-stable state for a while. In this state, PD, the signal which drives the NMOS pulldown transistors can be partially on. If they cause the address lines to fall, it will result in an event with bad timing. If the sensor column address lines partially discharge but fail to assert Event_MR, no data is produced. In this case, address lines recharge to their quiescent state, either by the column keepers, or by the arrival of another event, and the sensor recovers. Thus neither of these cause more than a single photon event to be corrupted.

5.4 The TDC

Because ToF image sensors employ a large number of TDCs, nearly all TDC designs use replica-biasing to lock a replica ring oscillator, RO, frequency to a precision system clock [53, 54, 55, 56]. A voltage or current reference is then routed to all the TDCs to set their frequencies. Due to long routs and device mismatch, it is difficult to precisely maintain the frequency of the TDCs.

To avoid this complexity and source of errors, this design doesn't lock the RO frequency to a reference source. Instead it is measured using an electronic calibration circuit. The input to the Event-detection Circuit in each pixel is driven by a mux, which selects either the SPAD output, or a calibration input. This calibration input is used both to timestamp the start of each acquisition cycle, and to measure the frequency of each TDC ring. When the laser pulse is emitted, the calibration signal is used to trigger one row to fire. This digitizes the count of each ring oscillator at the start of the measurement cycle, and is stored in the output FIFO. Reflected photons are then

timestamped with their ring oscillator count, and stored. At the end of the acquisition, another calibrate signal is sent and stored. Since the time between the start and end calibration pulses are known and timestamped by each of the 48 TDCs, the time difference between the 2 timestamped data determines the frequency of each RO. This information is then used to convert the ring oscillator counts into delay from the start of the laser pulse.

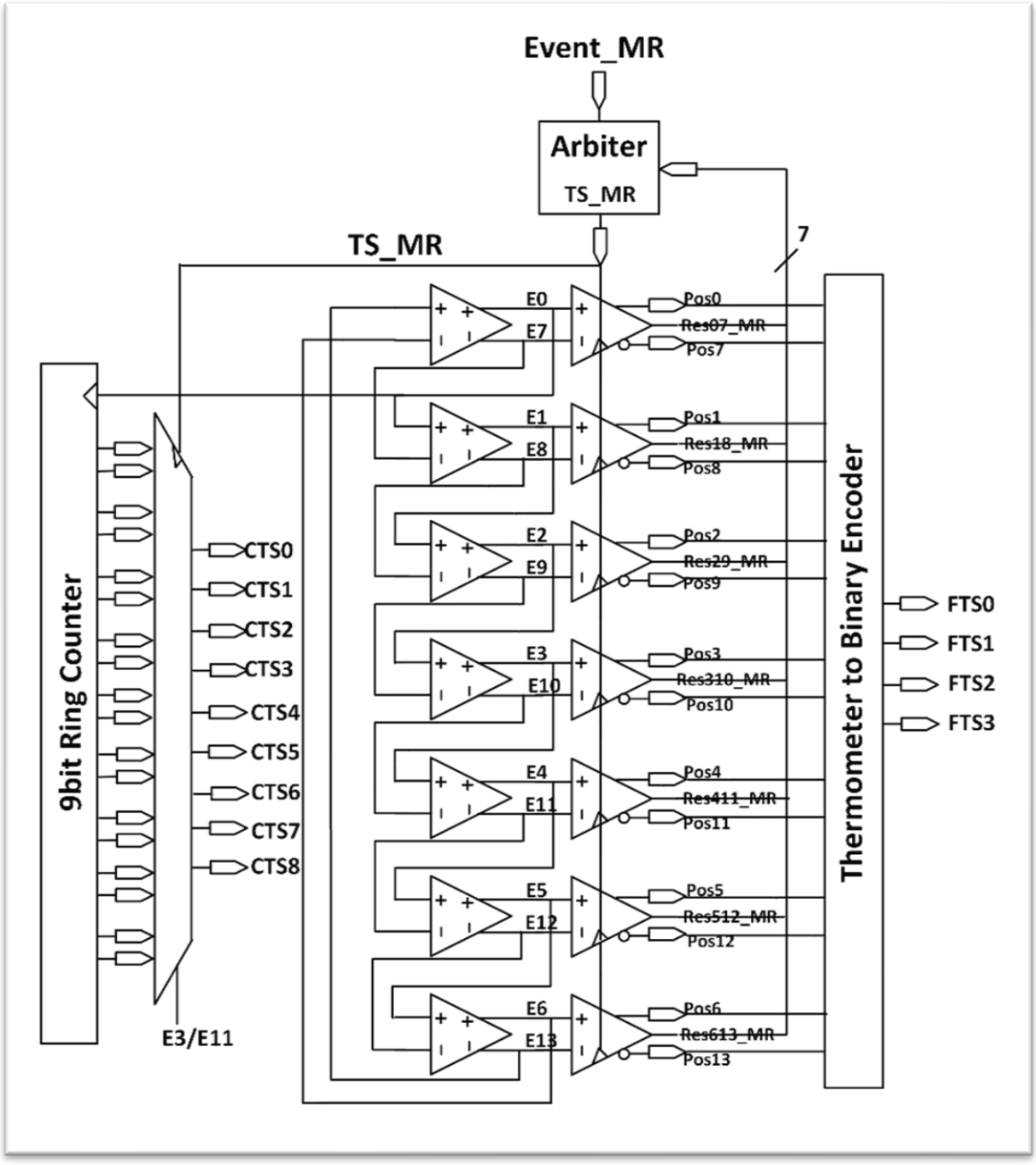

Figure 5.13: Simplified TDC Block Diagram

The block diagram of the TDC is shown in Figure 5.13. It includes an arbiter circuit which manages the interface to the pixel-column. The arbiter is needed in case the cycle time of the TDC is longer than the column. It drops a new event that happens before the TDC can timestamp it, and, on condition that TDC is ready to receive a new event, generates a buffered version of Event_MR, TS_MR, which is transmitted to the TDC and timestamped (Figure 5.14).

The TDC core comprises a free-running, un-gated, differential RO, a ring-counter, and flops that sample both the ring (the fine samplers) and the counter when an event occurs. To achieve better INL and reduce hardware costs, the number of the ring-oscillator stages is made as small as possible, while maintaining good operating margins for the ring-counter. The RO comprises 7 differential stages with a 720ps period (post-extracted nominal) and generates 14 evenly spaced edges, E0-E13, with a resolution of approximately 50ps. The energy consumption of the TDC is 11pJ per operation.

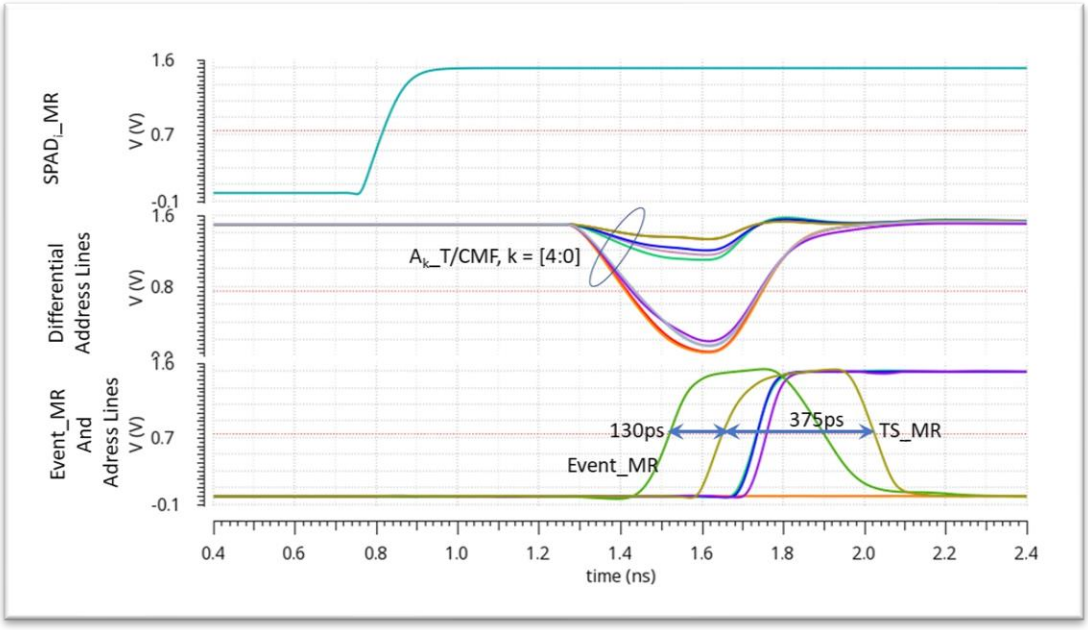

Figure 5.14: TS_MR Signal – the TDC Domain Event_MR

A dynamic synchronous 9-bit ring-counter with outputs, Cnt0 - Cnt8, is clocked by E0 and continuously counts the number of trips around the RO. Counter outputs, are re-latched on the falling edge of E0, to produce the phase-delayed signals, Cnt0_F – Cnt8_F (

Figure 5.15). The counter can unambiguously measure event-timings up to at least 128ns.

The rising edge of TS_MR indicates the arrival of a new event to timestamp, and causes:

- Seven differential samplers, which receive complementary RO signals of E0/E7, E1/E8, E2/E9, E3/E10, E4/E11, E5/E12, E6/E13 to be clocked by TS_MR and produce 14 fine position outputs, Pos0 – Pos13, of 7 zeros and 7 ones.
- TS_MR also captures the ring-counter (and its phase-delayed) outputs, to produce flopped outputs Cnt0TS – Cnt8TS, and Cnt0TS_F – Cnt8TS_F.

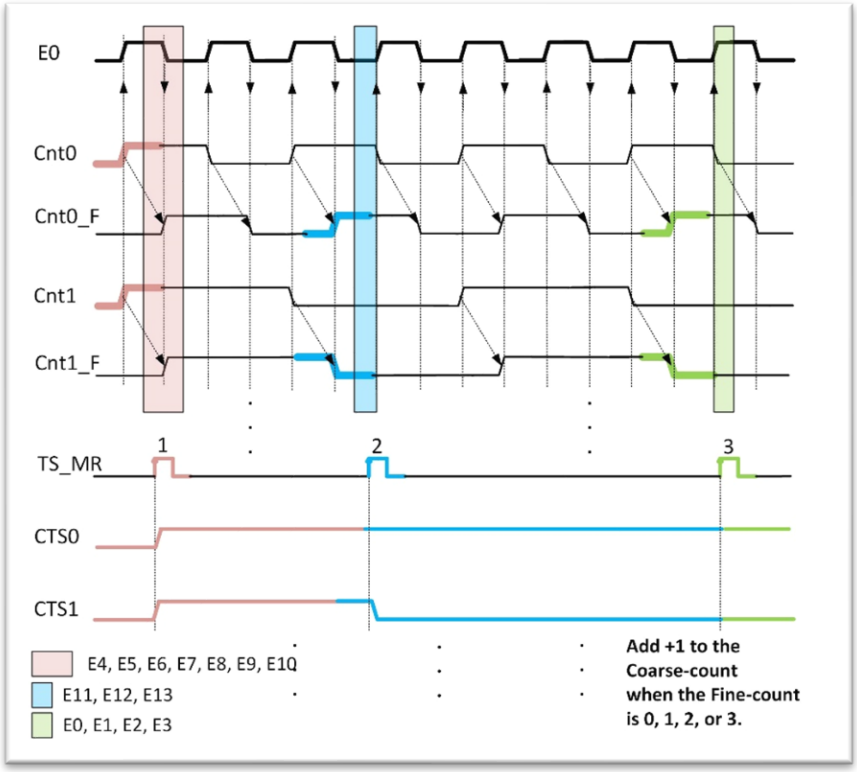

Figure 5.15: Coarse-count Synchronization to TS_MR Clock Domain

The unique 1 → 0 transition of two consecutive sampler outputs, indicates the temporal position of TS_MR within a single oscillator-cycle. A thermometer to binary converter maps Pos0 – Pos13 to (**F**ine-**T**ime **S**tamp signals, FTS0 - FTS3 to be stored in the FIFO.

Given that the counter outputs are sampled (by TS_MR) asynchronously with respect to the counter clock (E0), a problem can occur if TS_MR occurs around the E0 transition, since it can sample the counter outputs as they are changing giving an erroneous output. To avoid this situation we sample both ring-counter outputs, Cnt0TS - Cnt8TS, and their phase-delayed variant, Cnt0TS_F – Cnt8TS_F and, given the position of 1 → 0 transition, mux out the stable value of the counter, generating **C**oarse-**T**ime **S**tamp signals, CTS0 - CTS8, to be written in the FIFO, along with the associated fine-time stamps, FTS0 - FTS3.

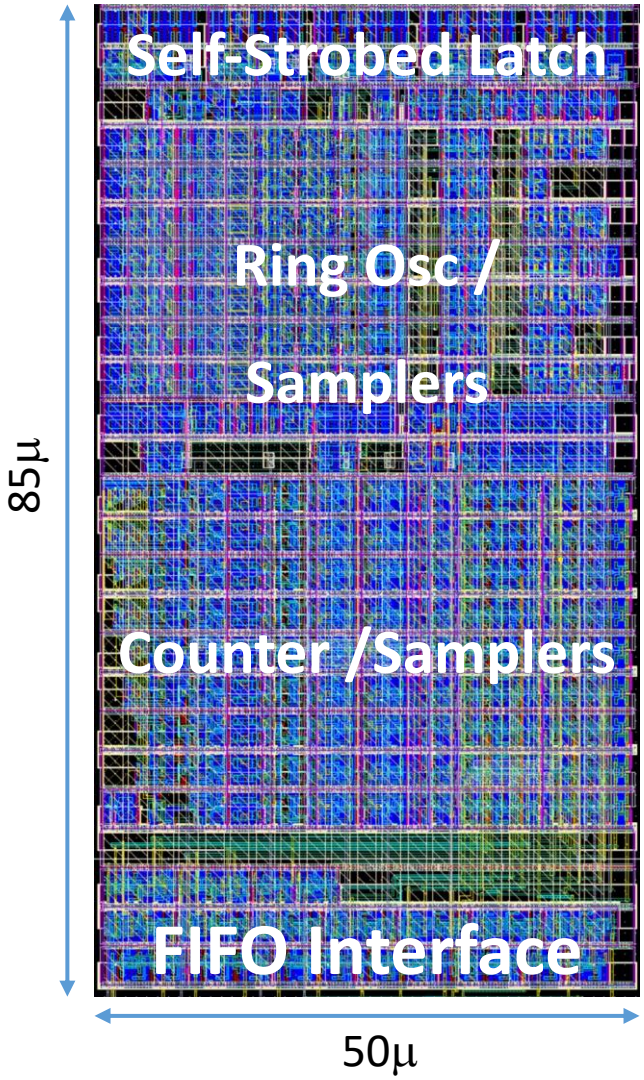

Figure 5.16: Layout of the TDC

The value of the Pos3 and Pos10 outputs are used to control the mux as shown in Figure 5.15. If Pos3 = 1 (Pos10 = 0), when the sampling event occurred, the 1 → 0 transition must have occurred between E4 and E10 which means the transition occurred in the pink part of the cycle in the figure.

Since this is well after the counter transitioned, its values were stable, and the MUX control signal selects Cnt0TS – Cnt8TS.

On the other hand, if Pos3 = 0 and Pos10 = 1, the 1 $\rightarrow$ 0 transition must have occurred between E11 and E3, which are the blue and green parts of the cycle, which includes the time the counter is changing, but doesn't include the time the phase delay count changes. In this case, the MUX selects the phase-delayed signals, Cnt0TS_F – Cnt8TS_F, which affords the greatest timing margin in sampling the counter value while it is stable. If the phase delayed counter data is used, we need to distinguish two different regions. If the TS_MR occurs after E11 and before E0, the blue region, the value of the phase shifted counter value is correct. If TS_MR occurs after E0, but before E4, the green region, the stable, phase shifted counter value is wrong, since it should have incremented. So, in this region we externally increment the counter value by 1.

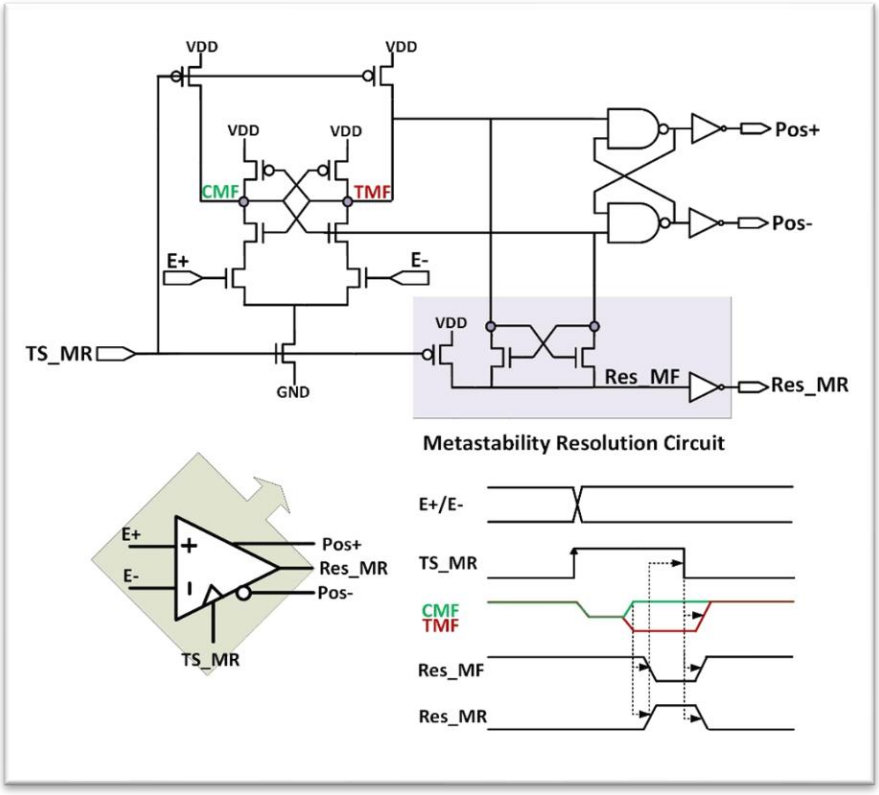

Figure 5.17: Fine-sampler with Metastability Resolution Circuit

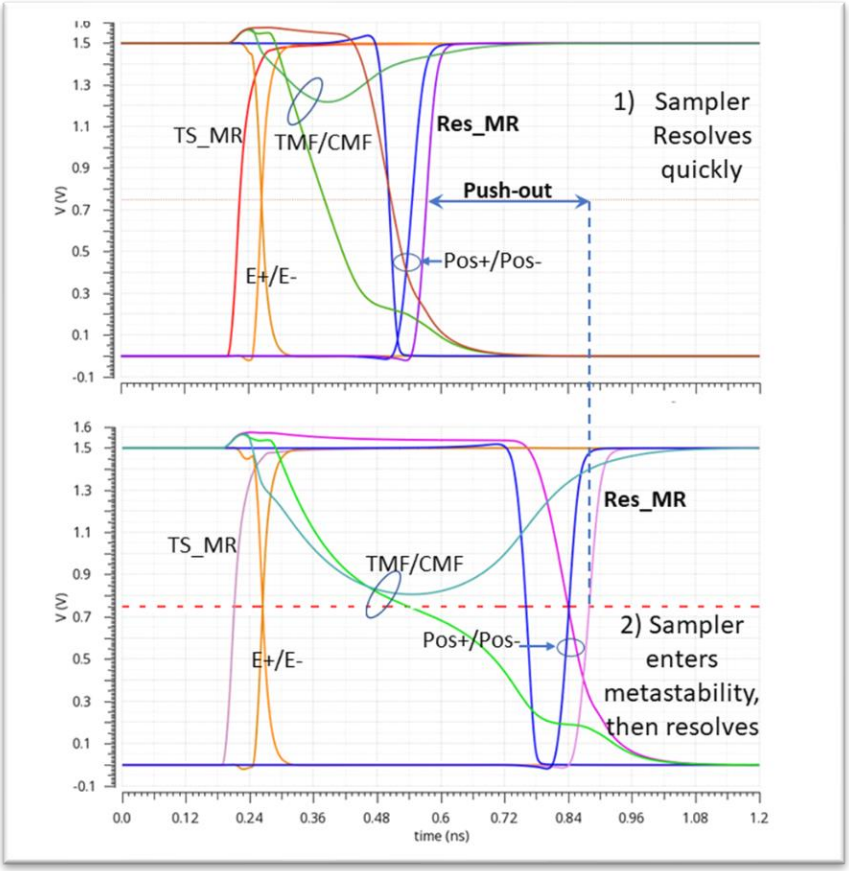

Figure 5.18: Waveforms for Metastability Resolution Circuit

The layout of the TDC is shown in Figure 5.16. The width of the cell is 50μm since it needs to match the pixel pitch. At the top edge, which connects to the pixel column are the 5 self-strobed latches. The RO, samplers and encoder sit underneath the latches and are located above the counter. The lower edge which connects to the FIFO, carries the spatiotemporal wires to the FIFO to be written.

5.4.1 Sampler Metastability and Arbitration

The interface between the pixel column and the TDC includes an arbiter circuit which allows the transfer of a new event to the TDC only if the TDC has resolved the timing of an earlier event. Based on the timing relation of TS_MR to the inputs of the 7 samplers, E0/E7, E1/E8, …, E5/E12, E6/E13, samplers take disparate times to resolve. The correct timing of an event may only be determined once all samplers have valid outputs. As shown in Figure 5.17, a metastability resolution circuit, accompanies each of the fine-samplers. It works as follows. Prior to assertion of TS_MR, nodes CMF, TMF, and the monotonically falling signal, Res_MF are pre-charged to VDD.

On assertion of TS_MR, the differential voltage between E^+ and E^-, determines the time it takes for CMF and TMF to separate; one falling to ground, the other recovering back to VDD. In fact, Res_MF remains at VDD, for as long as $|CMF - TMF| < V_{TN}$. Upon resolution of meta-stability, $|CMF - TMF| \geq V_{TN}$, Res_MF discharges to ground, and Res_MR asserts.

Simulation waveforms for two operating conditions are included in Figure 5.18. In the first case, due to the relation between the sampler clock (TS_MR) and the differential inputs, the sampler is far from metastability; nodes CMF and TMF resolve quickly and Res_MR asserts indicating that the sampler outputs POS+ and POS- are stable. In the second case, due to the relationship between the sampler clock (TS_MR) and the differential inputs, the sampler enters metastability; CMF and TMF both approach mid-supply level; after a while they do separate, one approaching ground, the other VDD, indicating that metastability is resolved, and as a result, Res_MR asserts. [27] A comparison of the two cases shows a significant delay of the outputs and a greater push-out of the Res_MR signal under metastability conditions.

These signals are used by the TDC arbiter, shown in Figure 5.19, to determine when the sample period is done and TS_MR should be reset. When Res_MR for all samplers asserts, Resolved_MR transitions high. The rising edge of this signal, grounds TS_MR, which places the samplers in pre-charge, while holding their present data. In response, Resolved_MR, itself, transitions low, at which point, the TDC becomes responsive to new events from the associated column.

5.4.2 TDC Dead Time and Metastability Signatures

The latency of this pipe-stage; i.e., from assertion of Event_MR to de-assertion of Resolved_MR, determines the dead-time, or repetition period, of the TDC, which is roughly about 900ps (post-extracted, nominal), on condition that the fine-samplers do not encounter metastability.

The effective sensor column dead time or the repetition period is defined as the shortest time between two column events that are distinguishable and can be timestamped. It is determined by the larger of the dead times (DT) of the pixel column and the TDC; that is $t_{DT} = \max\{t_{DT|COL}, t_{DT|TDC}\}$ in which case the effective column detection rate equals $1/t_{DT}$. In the case of this design, the pixel column dead time is 980ps while that for the TDC is 900ps, that is the effective dead time

[27] These figures only simulate to Res_MR asserting, and don't include the circuitry that uses this information to reset TS_MR.

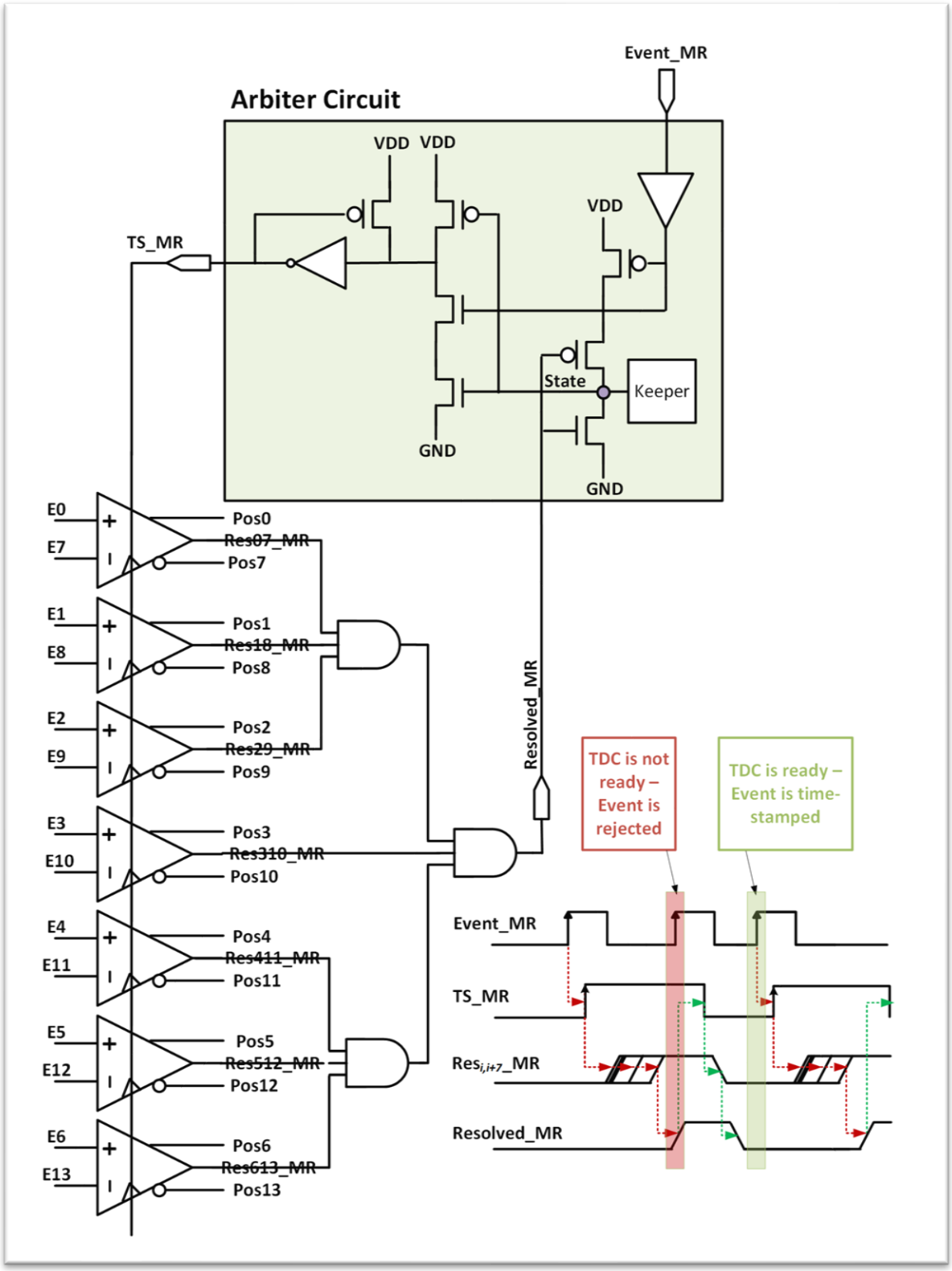

Figure 5.19: TDC Readiness and Arbitration

is 980ps (detection rate ~ 1GHz) unless one of the TDC samplers enter metastability which will extend the dead time.

While our Res_MR and mux selection of the counter output remove metastability issues in sampling the timing of the event, like all arbiters, it too can become metastable. Like the column case, this occurs when Event_MR arrives just when the arbiter is getting ready to re-arm, when Resolved_MR has fallen.

Event_MR and Resolved_MR both need to be low to drive the State node high. Once State is high, the arbiter is activated for new events. The problem situation occurs when the pixel column attempts to timestamp an event while the State node is transiting high, that is just after Resolved_MR fell (indicating the TDC is ready). As Event_MR transitions high, the drive pulling the State node to VDD turns off, leaving the voltage at a non-binary value. The keeper will eventually drive the voltage to a 0 or 1 value, but it might take some time to resolve. Fortunately, while in the metastable state, TS_MR remains low, and only asserts if the keeper resolves the State value to 1. Thus, in the unlikely case that metastability happens, its effect is to delay the launch of the TS_MR pulse, which only corrupts the timing of that event but doesn't cause any systematic error. In particular, if TS_MR fires, it will always be a well-formed pulse, like the situation for the pixel drivers.

5.5 The FIFO

Each pixel-column has a dedicated FIFO, comprising eight 18-bit wide entries of 13 temporal and 5 spatial data bits. The depth of the FIFO was set to handle the maximum expected number of detectable events impinged on a column during an exposure period, and can easily be changed in the design. Figure 5.20 shows a block diagram of the FIFO design which is mostly an array of 8 x 18 SRAM cells for storage. The design of the FIFO was challenging, since we needed to fit the FIFO into the pixel pitch, and even with the densest standard SRAM cell, 18 cells wouldn't fit in 50μm. In order to make this possible, 2 SRAM cells were laid out together as shown in Figure 5.21. By increasing the vertical height of the cell, we can put the widest part of the SRAM, the cross coupled NMOS and NMOS access devices on top of each other, which enables us to reduce the width of the cell. While this cell is less dense overall, it allows the design to fit into the required column bit. The increase in area is not a significant issue, since the number of registers in each FIFO is not that large. Using this technique the effective SRAM width is 1.7μm leaving 19.4μm for the layout of the word line decoder and driver.

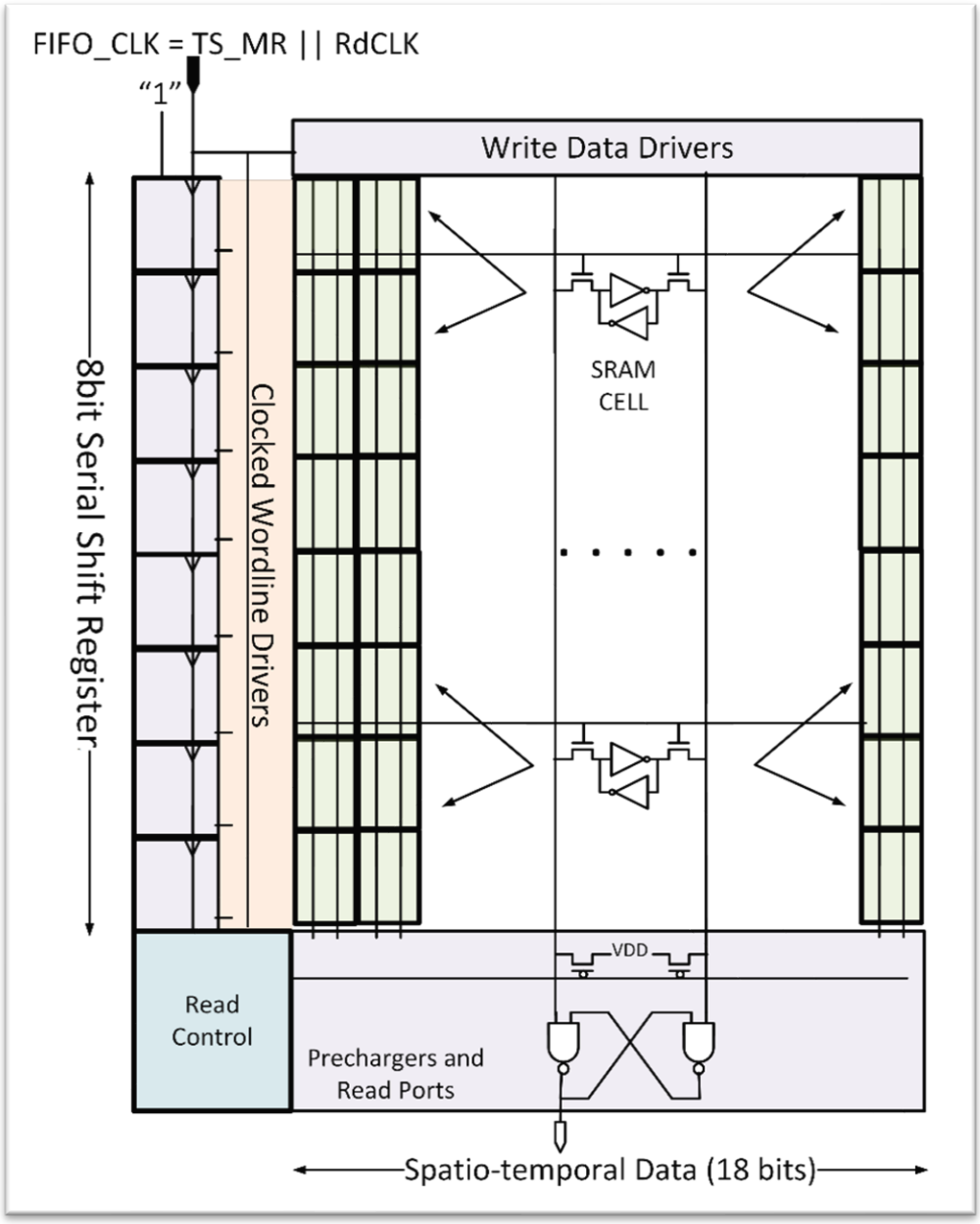

Figure 5.20: The FIFO Block Diagram

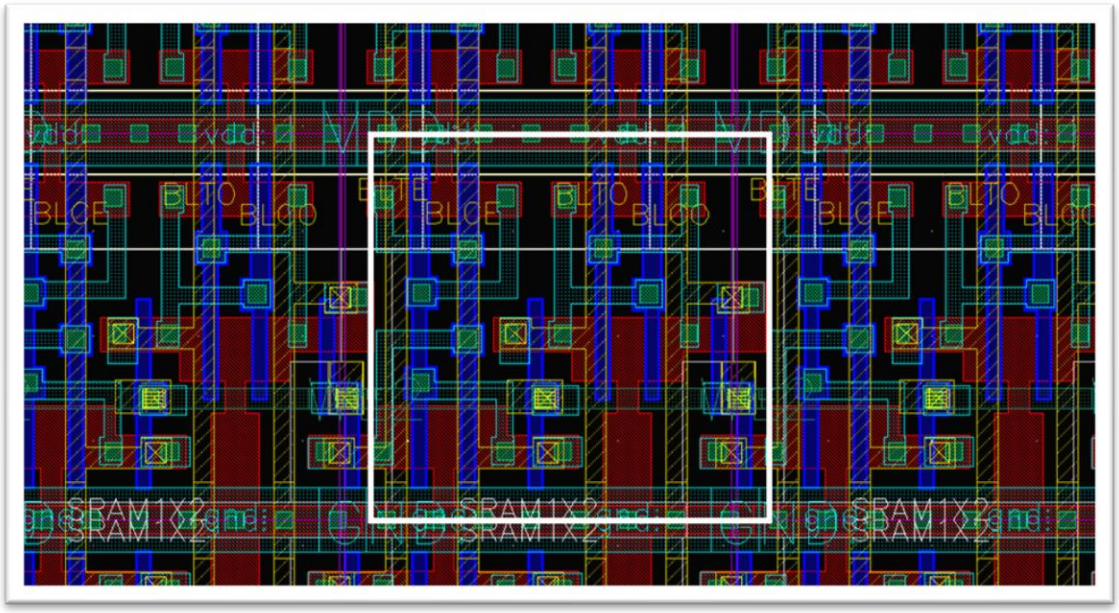

Figure 5.21: SRAM Layout. The Inset includes 2 SRAM Cells

The next task is controlling the word-lines and bit-lines of the SRAM so the correct data is stored in each FIFO entry. This control requires some care, since the timing of the event signal, TS_MR, and the spatiotemporal data is shifted in time, and the system both needs to be able to write new data into the FIFO, and read that data out of the FIFO. As TS_MR is the timing reference signal for writes, and RdCLK is the reference for reads, these signals are ORed together to create the FIFO_CLK. The word-line selection is controlled by a 9bit shift-register which stores a single "one" that indicates the current active entry in the FIFO. During reset, FIFORst_L = 0, the flip-flop not connected to a word-line (the 9th register) is set high, while all the others are set low, so after reset, no word-lines are active.[28]

Each rising edge of FIFO_CLK advances the shift-register, so after the first TS_MR arrives, the first word-line is driven high. It remains high until the next event occurs, which lowers the first word-line and raises the second. The critical edge on a write is the falling edge of the word-line since the bit-line values at the falling edge are stored in the cell. To provide good hold margin, the spatiotemporal data from the TDC is latched when FIFO_CLK is high before being driven onto the bitlines, so it can only change when this clock is low. This works well since this data will take some

[28] Using a shift register is smaller than using a counter and word-line decoder.

time to become valid to allow the flops sampling the values to settle (and because of meta-stability, settling may take some time). In fact, it is not guaranteed to be valid until right before TS_MR falls. Of course, this means that right after the clock arrives the wrong data will be driven into the FIFO which is ok, since the word-line will remain high, ensuring that finally the correct data is written into the cell. This set of operations are shown in Figure 5.22. The write operation is enabled by WrEn_H.

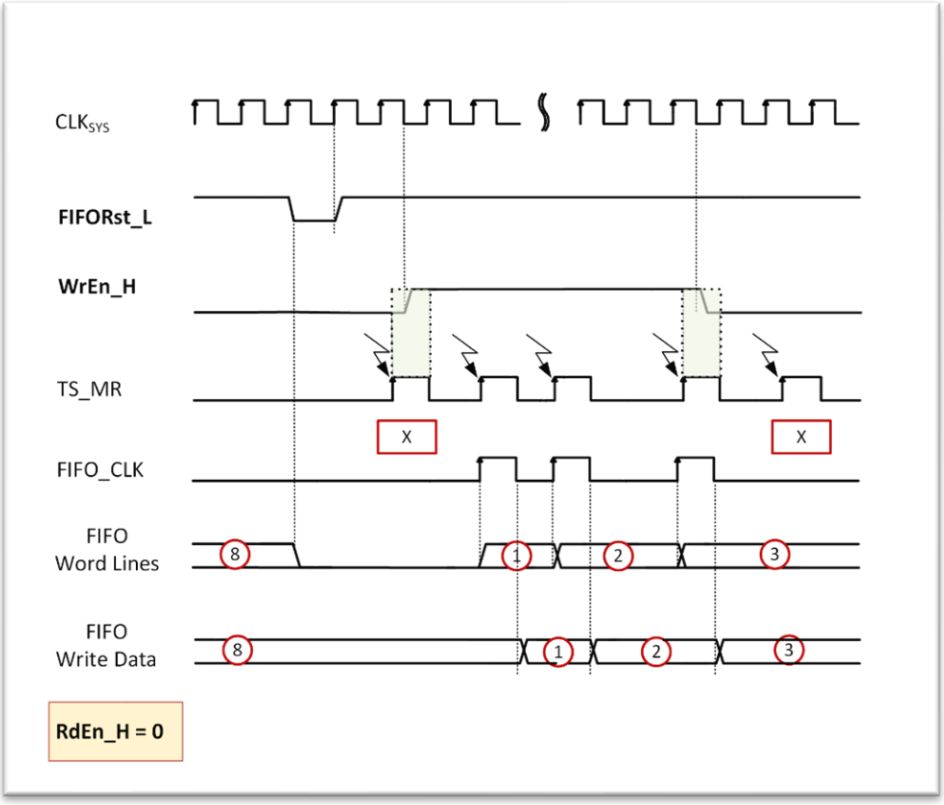

Figure 5.22: FIFO Write Timing Diagram

After the data is captured it can be read out by disabling the write pathway, and asserting RdEn_H, as shown in Figure 5.23. Upon entering read mode, the bit-lines are all precharged. Pulsing FIFORst_L resets the word-line pointers so no word-lines are asserted. Each subsequent assertion of RdCLK advances the FIFO pointer by one. During reads the shift register output is ANDed with FIFO_CLK before driving the word-line. So after the first RdCLK, the word-line to the first FIFO entry will pulse high, and the SRAM value will be driven onto the bit-lines. It should be noted that while data of a selected row are read out on the positive phase of RdCLK, it must hold stable when the FIFO enters precharge on the negative phase of the clock. This is accomplished by using a

cross-coupled NAND structure which connects to the differential monotonically falling bit-lines as seen in the figure. The contents of the FIFOs are read out, entry by entry, in 8 cycles of RdCLK.

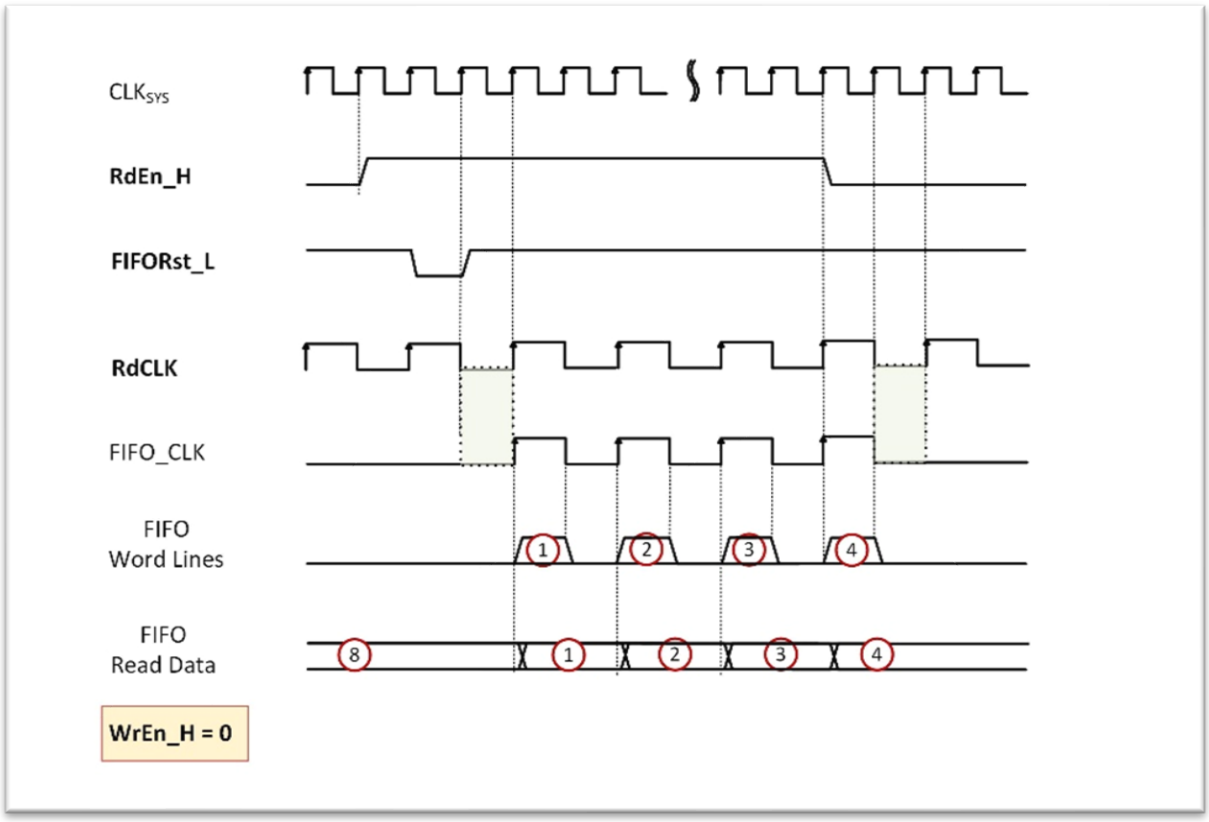

Figure 5.23: FIFO Read Timing Diagram

Unlike the other blocks of the imager, the FIFO does not use post-charge circuits, so it doesn't generate any internal timing signals. Instead it uses the FIFO_CLK as the only timing signal, and all operations are synchronous with this timing signal. Thus, the FIFO doesn't have any intrinsic metastability issues, and since it's input clock will always be well-formed (no runt pulses) there are not special cases that need to be considered.

Without additional circuity, testing the FIFOs would be a challenge – in the current design we don't have direct control of the data that can be written into the SRAM. To fix this issue, a test data generator was added to the design. A small control circuit generates two values, B0 and B1 which are driven to all the bits. The data generators for the even bits can select between their input and B0, while the odd bits can select between their input and B1. By controlling the values of B0 and B1 on each test cycle, all FIFOs can contemporaneously be written with one of 4 patterns, blank (all zero), opaque (all 1), checker-board and reverse checker-board in 8 cycles of a system-clock derived, bypass clock. The functionality of the FIFOs is subsequently verified by entering the FIFO

read operation to compare the read and write data for equivalence. The data generator also can be used to clear out the old FIFO data between acquisitions if desired.

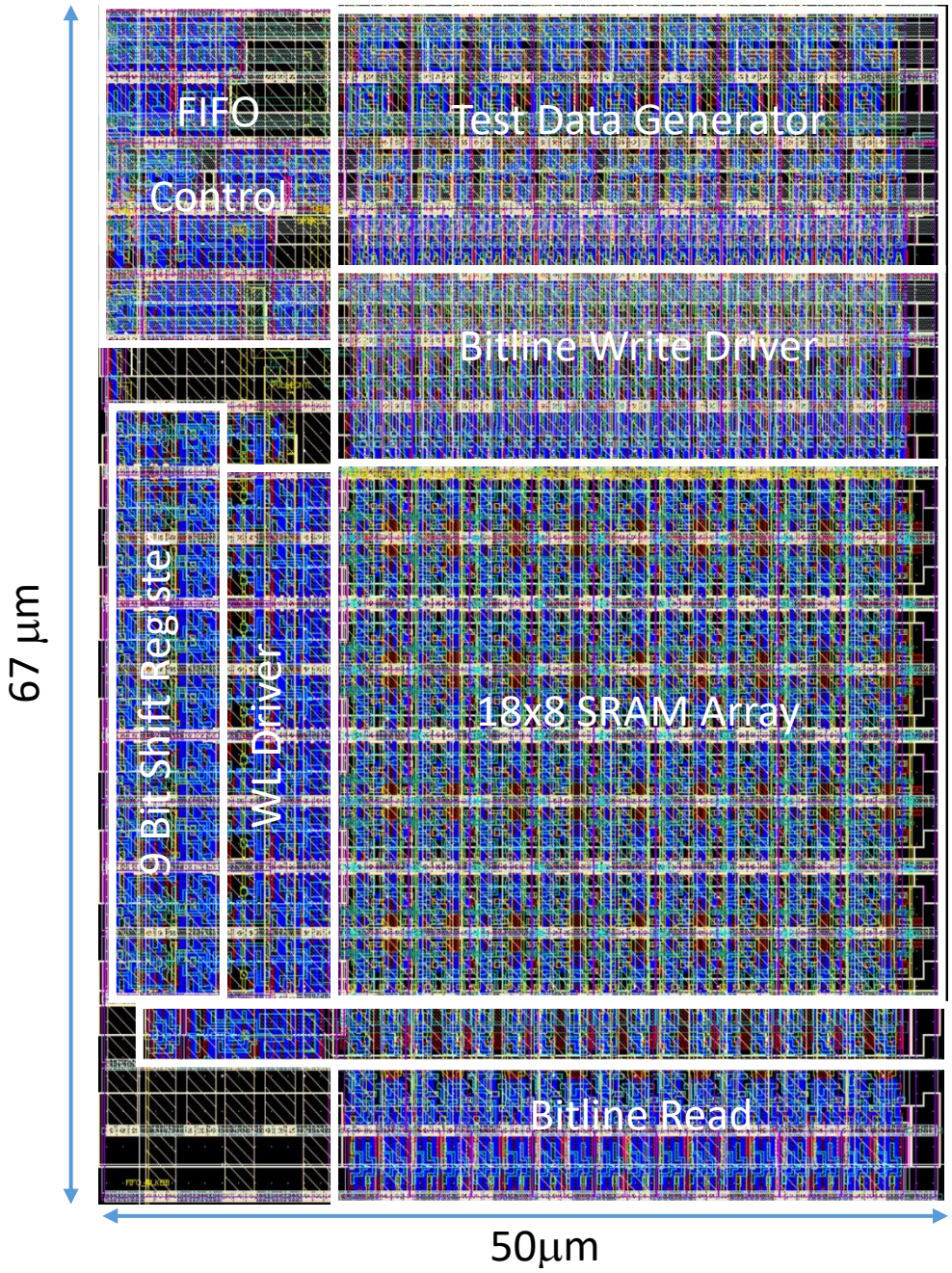

Figure 5.24: FIFO Layout

Figure 5.24 shows the layout of the completed column FIFO. Using the x2 memory cell layout enables the complete design to fit in the 50µm pixel pitch. The total height of the block is around

67μm, which is only slightly larger than a pixel. Doubling the FIFO storage capacity would increase the area of the block by less than 50%.

5.6 Imager Peripheral Circuits

In addition to the FIFO data generator to facilitate testing of the imager, the chip contains a number of circuits outside of the imager core. To enable reading the event data, the read outputs of the 48 FIFOs are multiplexed together onto a set of 18 high-speed output pins per each subarray. This allows the user to select either a single column to read, or to read out the entire chip. The 8 entries of the 48 FIFOs are contemporaneously read in 8 read clock cycles, producing 18 x 48bits of data per cycle. If the user is interested in all the data, it can be sequentially driven off-chip in 48 IO clock cycles per FIFO entry for a total of 384 clock cycles. The serialized data is accompanied by the IO clock for the source synchronous operation.

We use a JTAG port to control the chip. Apart from the required JTAG and high-speed output pins, a few other pins are used to control the operation of the part. In addition to supply pads, VDD, VSS, and V_{SPAD}, the sensor accepts a master clock (CLK_{SYS}) which sets the I/O speed, and a voltage to control the SPAD quench time, V_{HOLD}[29], as inputs.

5.7 Scaling to Larger Arrays

While the circuits presented in this chapter enable a 48x64 imager, scaling to larger arrays poses a few challenges. Increasing the number of pixels in the row direction is straightforward. Each column operates independently, so increasing in this direction doesn't affect performance;[30] the number of TDCs and FIFOs simply scale with the number of columns. Controller signal lines associated with electrical control of the pixel array and the FIFOs see bigger loads and require larger drivers. Increasing the number of pixels in the column direction is far more challenging. This

[29] The only legacy analog voltage level on the chip. It can be replaced with digital starved inverters.

[30] The only challenge is in distributing the electronic calibration signal to all the cells in a row, but there are many methods to deal with this issue. The simplest is to use a low resistance top level metal for this signal, and drive it from both sides of the array.

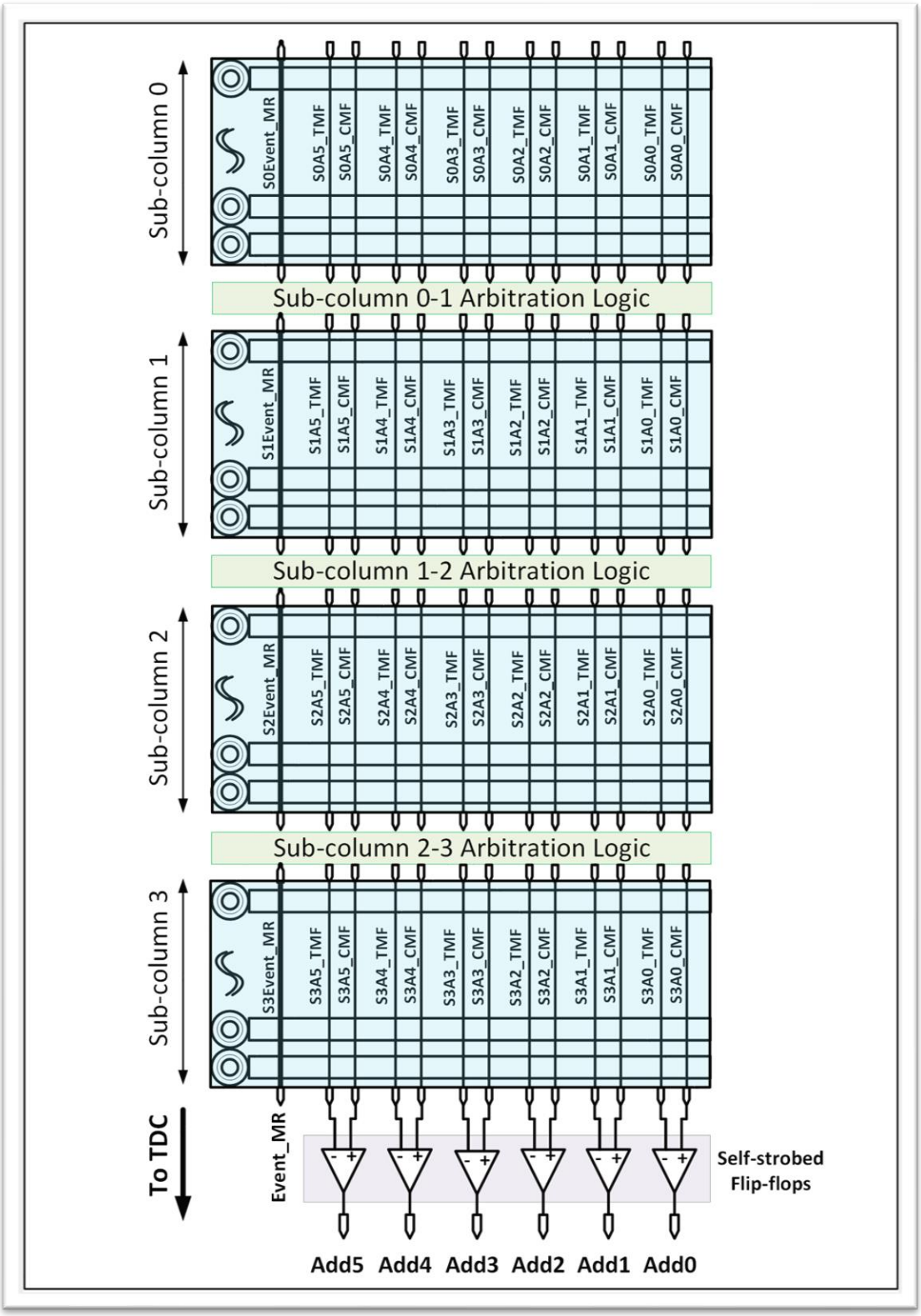

Figure 5.25: Segmented Column Architecture

is principally because of a linear increase in self-loading and the quadratic increase in the RC of the column address lines A*i*_CMF/A*i*_TMF.

To enable scaling the number of rows in each column, columns are segmented into shorter sub-columns, with a type of repeater between the segments. Figure 5.25 shows the arrangement of a 64-pixel column in four sub-columns separated by the arbitration and transfer logic. An event in a sub-column can be transferred to the next sub-column if it arrives while that column is currently quiescent[31] waiting for an event. This decision is made by using an arbiter like the one used in the TDC.

The arbiter circuit, shown in Figure 5.26, has 4 states:

1. So long as $S_{(i+1)}$Event_MR is asserted, subcolumn (i+1) is busy; node Ready_H is driven to ground and the arbiter input signal $S_{(i)}$Event_MR from the preceding sub-array is ignored. The monotonically falling output of the arbiter, Transfer_CMF, which enables data transfer is de-asserted and held at VDD.
2. Once $S_{(i+1)}$Event_MR de-asserts, (i+1) sub-column (i+1) enters its quiescent state.
3. In the quiescent state, Ready_H is held at ground by a keeper, as long as $S_{(i)}$Event_MR is high. If the latter is low, or tansitions low, Ready_H is high or transitions high. This arms the arbiter, making column (i+1) receptive to data from column (i).
4. From this point on, an assertion of $S_{(i)}$Event_MR grounds Transfer_CMF, which enables the transfer of spatiotemporal data from sub-column (i) to sub-column (i+1).

The arbiter, in combination with the circuit in Figure 5.27 which shows the data-transfer logic between sub-columns (i) and (i+1) functions as the Event-detection Circuit in each pixel. If the column is quiescent, and an event needs to be transmitted, the event information is driven onto the differential address lines. Figure 5.28 gives the timing diagram for data transfer from column (i) to column (i+1) when the latter is in the quiescent state. The SPAD firing in sub-column (i), drives the differential address lines, which in turn causes $S_{(i)}$Event_MR to assert. If the arbiter is ready, this transitions flows through the arbiter and causes Transfer_CMF of sub-column (i+1) to be

[31] A sub-column (m) is in the quiescent state when its dual-rail monotonic address lines, A_m_CMF/A_m_TMF, are held at VDD, and its timing signal, $S_{(m)}$Event_MR is de-asserted.

driven to ground, enabling the NOR gates, through which the logical levels of $A[5:0]_{(i)}$_T/CMF are driven unto the sub-column (i+1) address lines, $A[5:0]_{(i+1)}$_T/CMF transferring the event information onto the next block of columns.

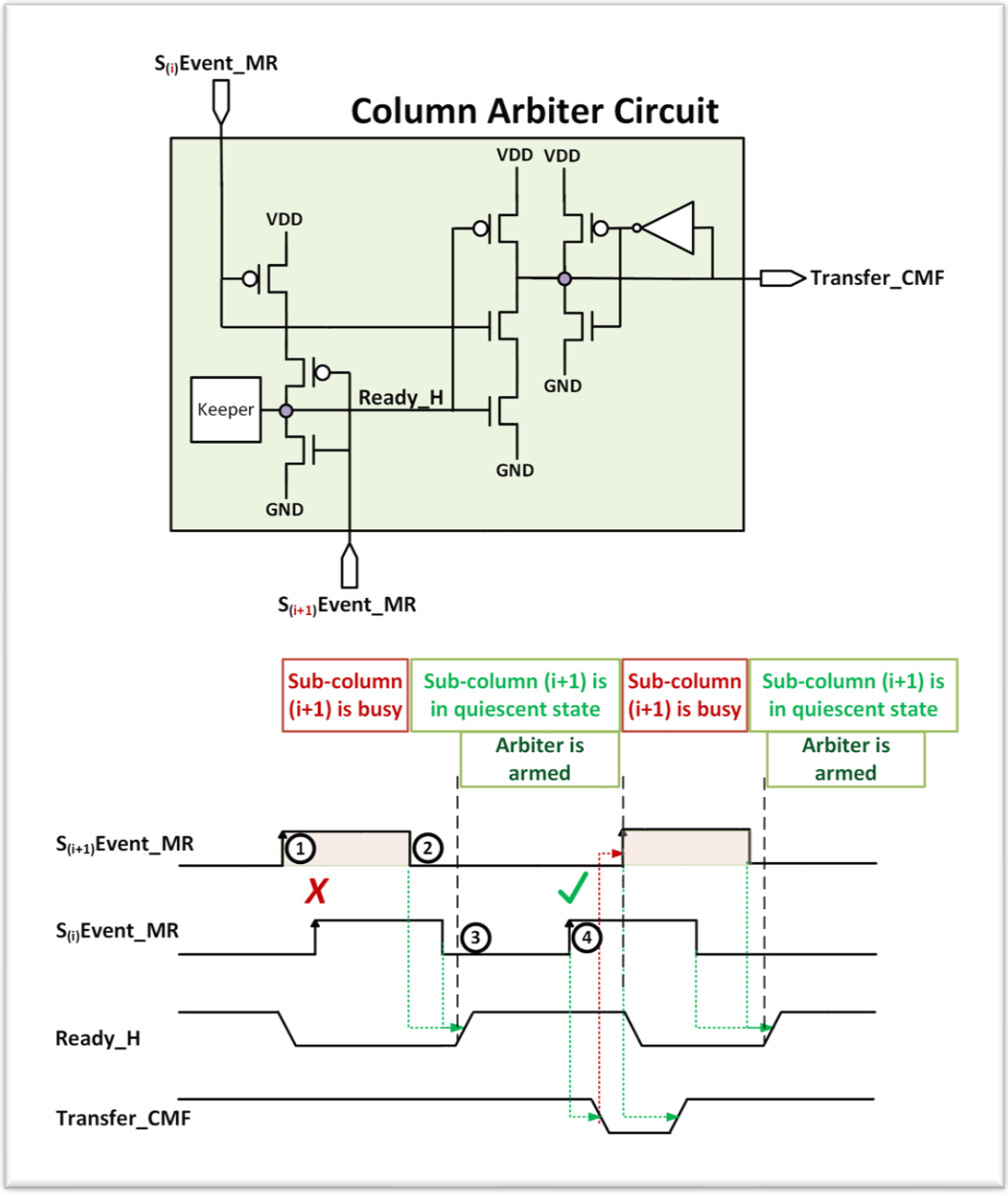

Figure 5.26: Intra-column Arbitration Circuit

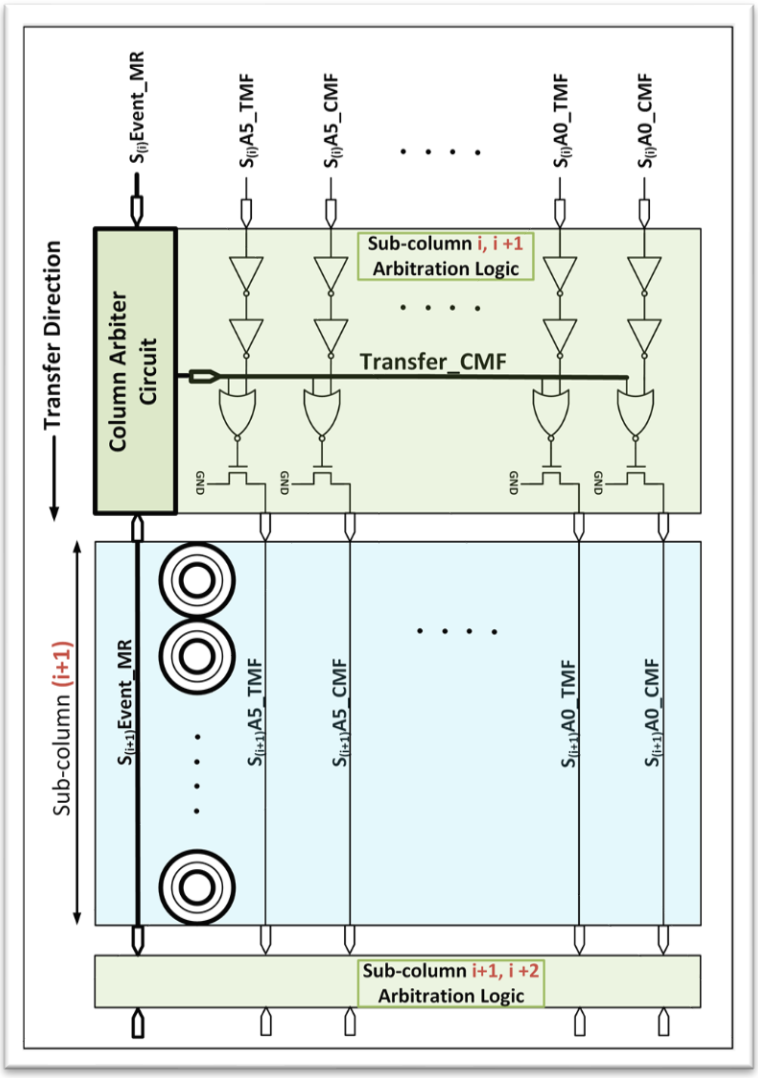

Figure 5.27: Intra-column Arbitration and Data Transfer Logic

Driving the address lines initiates the assertion of $S_{(i+1)}$Event_MR, which returns Transfer_CMF to VDD, and ends transfer of data. The two sub-columns, then independently, return to their quiescent states. Notice that in this architecture the cycle time of the overall system is set by the cycle time of a local sub-column section. While the latency through the array increases for the pixels farther away from the TDCs, the throughput does not depend on the total numbers of rows in the design.

Using this architecture should allow each column to maintain a photon capture rate of around 1G photons/sec, but requires a method to calibrate out the variable delay from the photon detection to the TDC where the event is timestamped. Fortunately, our calibration hardware can easily provide the needed input. Since the chip can activate any row to provide an electronic time reference, each row can be used to "start" a measurement, with the row closest to the TDC always used to end the

measurement. The difference in counts between measurements for different rows can be used to "correct" the timing of SPAD arrivals for each row, removing the internal delay from the data.

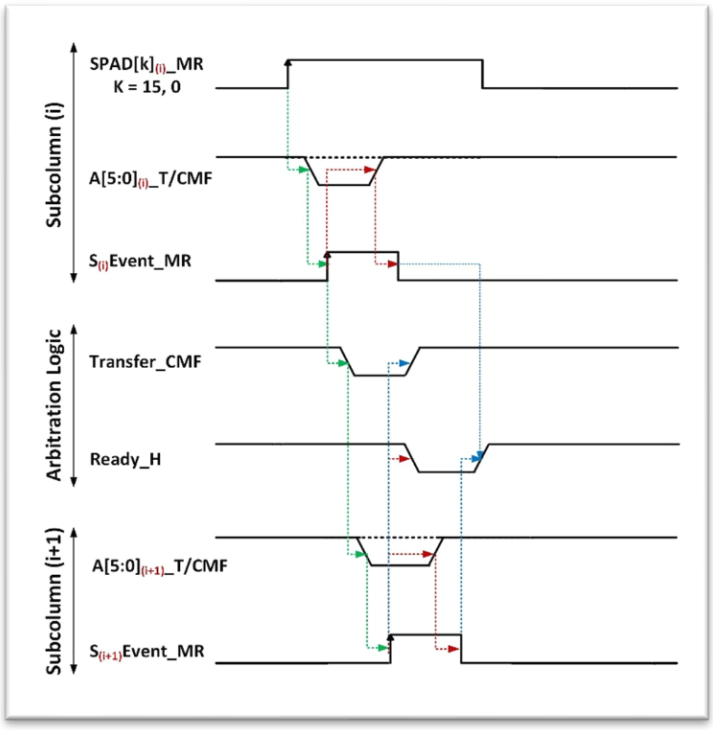

Figure 5.28: Data transfer Waveforms

5.8 Summary

In this chapter we presented the design and implementation of a scalable, all-digital, column-parallel 64x48 pixel Flash image sensor with a 1GHz column detection rate. The IC does not use any analog circuitry or high-speed clocks. Using self-resetting circuits, the design seamlessly detects the arrival of photonic events, and with proper timing returns the column to its quiescent state, ready to accept a new event. A new event falling within the dead time of a first event is discarded while spatiotemporal information of the first event is preserved. The TDCs do not use replica-biasing to lock their ring-oscillator frequencies to a reference clock, instead, their frequencies are measured by timestamping precisely spaced pulses whose temporal information is used to determine the oscillation frequency. The design is scalable to larger arrays by dividing the sensor columns to shorter sub-columns and using arbitration to buffer and transfer the spatiotemporal information in a sub-column to a neighboring downstream sub-column if it is ready. In this manner large image sensor array can be constructed with performances comparable to smaller counterparts.

Conclusion

We started the thesis by presenting the principles governing LiDaR imaging and the Time Correlated Single Photon Counting (TCSPC) as a method to estimate distance of objects. These imaging systems require a highly-sensitive, low-jitter, photon detector, which is usually accomplished using Single Photon Avalanche Diodes, or SPADs as a photodetector. SPADs need some supporting circuits that can use passive RC components, but adding active devices yields a design that is easier to control.

After introducing the basic LiDaR system, we showed that the signal to noise ratio was simply the ratio of the irradiance caused by the pulsed laser illuminator, to intensity of the background irradiance. Even with high illumination levels, a SPAD will not detect a photon during each pulse; this means that creating a timing estimate with good fidelity requires accumulating the results from a large number acquisitions.

In Chapter 3 we addressed the statistical nature of light which follows the Poisson distribution. Based on the SPAD first photon postulate, i.e., a SPAD may only fire once during an acquisition, we arrived at the Exponential distribution which describes the probability of detected photons over time. We then derived a Mixture Exponential distribution to include the influence of background illumination on photon detection. Since each exposure causes at most a single detected photon, a large number of exposures are required to capture a scene. We constructed a simulator matching these photon statistics which creates histograms comprising multiple exposures defining a single frame. We then built multi-frame histograms of depth error and analyzed the influence of background illumination, *SNR* and the number of exposures on the depth accuracy of a recovered object. These results showed that at high *SNR*s (higher than 3) we need around 3 photons in each fine time bin to maximize performance, and that number nearly doubles as the *SNR* drops to 1. For a fixed background illumination, this means at high *SNR*, the number of exposures is inversely proportional to *SNR*, but grows rapidly as *SNR* approaches 1. Even when tight distance accuracy isn't needed, the number of exposures can't be reduced too far to avoid problems with false detections. More distant objects make these issues worse, since SPADs only detect the first photon they encounter. This effect gets worse as noise photon flux increases.

Based on the insights gained from these analyses, we studied time-gating as a technique to improve the recovery of distant objects. While time-gating precludes visibility of near objects, an improved recovery technique which includes time-domain exponential filtering of data has the ability of exposing a whole distant scene without compromising long-range recovery accuracy.

Finally, we explored how illumination scanning can be used to reduce the extra acquisitions needed in low *SNR* situations. By only partially illuminating a scene, it is possible to increase the laser photon flux, λ_L, and move to a higher *SNR* operation point. Moving from *SNR* = 1 to *SNR* > 3 reduces total number of laser photons that the sensor needs to collect by 2 times, which, in an environment where the average laser power is constrained reduces the acquisition time by 2 times as well.

In Chapter 4, based on the multinomial distribution, we presented a mathematical formulation for depth uncertainty, as well as conditions leading to ***false-detections***. The formulation agrees with the simulated results to within 5%. To gain insight into conditions affecting the uncertainty in depth measurement, an upper bound Gaussian approximation to the multinomial distribution was derived. The upper bound formulation sheds light on the relation between *SNR* and N_{Exp} for a fixed error-spread, σ_{Err}.The accuracy of this estimation is to within 25% when compared to the simulated results.

Finally, in chapter 5 we presented the design and implementation of a scalable, all-digital, column-parallel 64x48 pixel Flash imager with a 1GHz column detection rate. The IC does not use any analog circuitry or high-speed clocks. Using self-resetting circuits, the design seamlessly detects the arrival of photonic events, and with proper timing returns the column to its quiescent state, ready to accept a new event. A new event falling within the dead time of a first event is discarded while spatiotemporal information of the first event is preserved. The TDCs do not use replica-biasing to lock their ring-oscillator frequencies to a reference clock, instead, their frequencies are measured by timestamping precisely spaced pulses whose temporal information is used to determine the oscillation frequency. The design is scalable to larger arrays by dividing the sensor columns to shorter sub-columns and using arbitration to buffer and transfer the spatiotemporal information in a sub-column to a neighboring downstream sub-column if it is ready. In this manner large image sensor array can be constructed with performances comparable to smaller counterparts.

Appendix: First Photon Postulate and the Probability of Avalanche

As we have described previously, there is only a small probability that a photon impinging a SPAD would trigger an avalanche. With this in mind, *The Postulate* states that, once the first photon in a stream is detected; i.e. it triggers an avalanche, subsequent photon arrivals are ignored[32].

Given a photon flux, λ, and the SPAD *PDP*, one can determine the probability that a certain number of photons in a photon-stream triggers an avalanche. The *conditional probability* that one out of k photons is detected can be calculated as:

$$P(Avalanche|Ph = k)$$

$$= 1.PDP + (1 - PDP)PDP + (1 - PDP)(1 - PDP)PDP + \cdots + (1 - PDP)^{(K-1)}PDP$$

$$= PDP.\left(1 + (1 - PDP) + (1 - PDP)^2 + \cdots + (1 - PDP)^{(K-1)}\right) = 1 - (1 - PDP)^k$$

It can be seen that if the SPAD $PDP = 1$, $P(Avalanche|Ph = k) = 1$, which means that for any number of photons impinging a pixel, avalanche will occur with the 1st photon in the stream.

Using the underlying Poisson Distribution in concert with the above probability, we can derive the probability of avalanche for any number of photons in a stream:

$$P(Avalanche) = \sum_{k=1}^{\infty} P(Avalanche|Ph = K).P(Ph = k) = \sum_{k=1}^{\infty}\left(1 - (1 - PDP)^k\right)P(Ph = k)$$

$$P(Avalanche) = \sum_{k=1}^{\infty}\left(1 - (1 - PDP)^k\right)\frac{(\lambda.\Delta t)^k.e^{-\lambda.\Delta t}}{k!}$$

Employing the Taylor series expansion for $e^{-\lambda.\Delta t}$, we arrive at:

$$P(Avalanche) = 1 - e^{-PDP.\lambda.\Delta t}$$

The above equation is the familiar CDF for the Exponential Distribution obtained from the underlying Poisson Distribution. Figure A. 1 includes the PDF and the CDF (Probability of Avalanche) for 3 flux levels from a *single* light source over time.

---

[32] Recall that once a SPAD is triggered it becomes unresponsive to subsequent impinging photons.

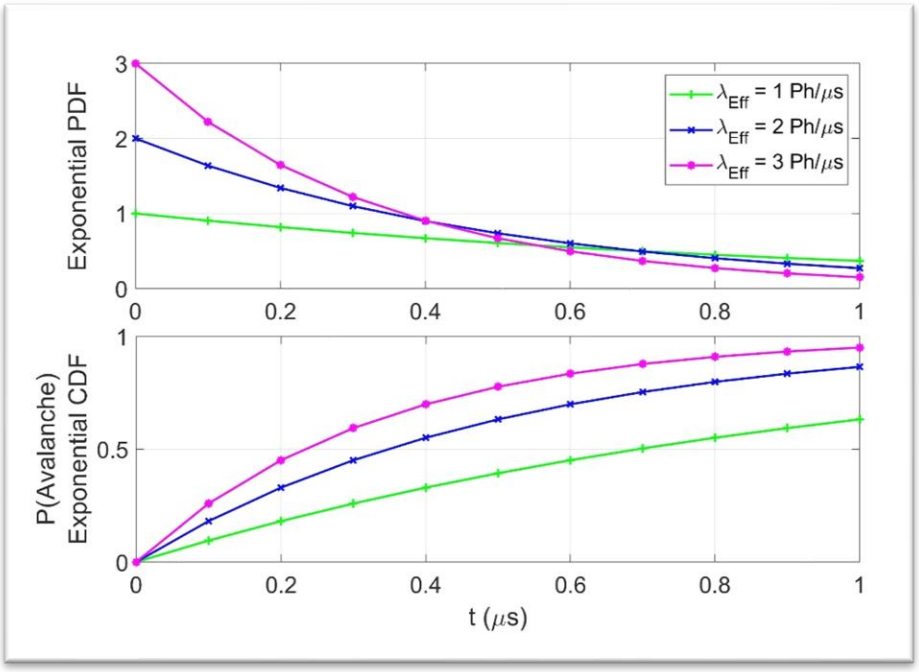

Figure A. 1: Exponential PDF and CDF – P(Avalanche)

References

[1] C. McCormick, “Stereo Vision Tutorial - Part I,” mccormickml.com, 2014.

[2] https://oceanservice.noaa.gov/facts/lidar.html.

[3] J. W. Horton, “Fundamentals of Sonar,” U. S. Naval Institute, Annapolis, MD., 1957.

[4] www.bom.gov.au/australia/radar/about/what_is_radar.shtml.

[5] M. Wang, “The State of Solid-State 3D Flash LiDAR,” medium.com/@miccowang, 2018.

[6] S. Bellisai, *et al.*, “Indirect time-of-flight 3D ranging based on SPADs,” Proceedings of SPIE, 2012.

[7] J. Kvam, ”Time of Flight: Principles, Challenges, and Performance,” ST Technology Tour, 2017.

[8] S. Lee, “A Time-of-Flight Range Sensor Using Four-Tap Lock-In Pixels with High near Infrared Sensitivity for LiDAR Applications,” MDPI.com, 2019.

[9] S. Bush, “3D Lidar Sensor from Panasonic,” Electronics Weekly.com, 2017.

[10] D.V. O’Connor, D. Phillips, “Time-Correlated Single Photon Counting,” Elsevier Inc, 1984.

[11] http://www.everyphotoncounts.com/device-timing.php.

[12] D. Renker, “Geiger-mode avalanche photodiodes, history, properties and problems,” Nuclear Instruments and Methods in Physics Research, 2006.

[13] A. Biber, *et al*, “Avalanche Photodiode Image Sensor in Standard BiCMOS Technology,” IEEE Trans. Electron Devices, 2000.

[14] A. Rochas, *et al*, "Low-noise Silicon Avalanche Photodiodes Fabricated in Conventional CMOS Technologies," IEEE Trans. Electron Dev., 2002.

[15] home.deib.polimi.it/cova/elet/Articoli%20e%20presentazioni/2013SPADlab_SSN.pdf.

[16] F. Zappa, *et al*, "Fully-Integrated Active-Quenching Circuit for Single-Photon Detection," ESSCIRC, 2002.

[17] J. Richardson, *et al*, "Dynamic Quenching for Single Photon Avalanche Diode Arrays," International Image Sensor Workshop, 2007.

[18] A. Ghioni, *et al*, "Planar Silicon SPADs with 200μm Diameter and 35ps Photon Timing Resolution," Proceedings of SPIE, 2006.

[19] S. Cova, *et al*, "Avalanche Photodiodes and Quenching Circuits for Single-photon Detection," Applied Optics , 1996.

[20] X. Hesong, *et al*, "Design and characterization of a p+/n-well SPAD array in 150nm CMOS process," Optical Express, 2017.

[21] A. Lacaita, *et al*, "On the Bremsstrahlung Origin of Hot-carrier-induced Photons in Silicon Devices," IEEE Transaction on Electron Devices, 1993

[22] P. F. McManamon, *et al*, "Review of LaDaR: a historic, yet emerging, sensor technology with rich phenomenology," SPIE, 2012.

[23] J. Kostamovaara, *et al*, "On Laser Ranging Based on High-Speed/Energy Laser Diode Pulses and Single-Photon Detection Techniques," IEEE Photonics Journal, 2015.

[24] K. Ito, *et al*, "System Design and Performance Characterization of a MEMs-Based Laser Scanning Time-of-Flight Sensor Based on a 256x64-pixel Single Photon Imager," IEEE Photonics Journal, 2013.

[25] Safety of laser products - Part 1: Equipment classification and requirements, (2nd ed.) International Electrotechnical Commission. 2007.

[26] A. Weiner, Ultrafast Optics," Wiley, 2009.

[27] PCO-7114-50-4 Laser driver.

[28] C. Niclass, *et al*, "Design and characterization of a 256x64-pixel single-photon imager in CMOS for a MEMs-based laser scanning time-of-flight sensor," Optical Express, 2012.

[29] P. A. Gagniuc, *Markov Chains: From Theory to Implementation and Experimentation*, John Wiley and Sons, 2017.

[30] A. McCarthy, *et al*, "Long-range time-of-flight scanning sensor based on high-speed time-correlated single-photon counting," Applied Optics, 2009.

[31] I. Bar-David, "Communication Under Poisson Regime," IEEE Transactions on Information Theory, 1969.

[32] A.M. Wallace, *et al*, "Detecting and Characterizing Returns in a Pulsed Ladar," IEE Proceedings on Line, 2006.

[33] S. Hernandez-Marin, *et al*, "Multilayered 3D LiDAR Image Construction Using Spatial Models in a Bayesian Framework," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008.

[34] W. K. Chen (*ed.*), "The Electrical Engineering Handbook," Academic Press, 2004.

[35] P. Gray, P. Hurst, S. Lewis, R. Meyer, "Analysis and Design of Analog Integrated Circuits," Wiley, 2009.

[36] J. C. Cooper, "The Poisson and Exponential Distributions," Applied Probability Trust, 2005.

[37] https://blogs.sas.com/content/iml/2013/07/22/the-inverse-cdf-method.html.

[38] C. Toth, *et al*, "Peak Detection from Full Waveform LiDaR Data," Semantics Scholar, 2011.

[39] Ocular Robotics, RobotEye RE05, 2D Spinning Sensor.

[40] Carnegie Robotics, MultiSense SL, 2D Spinning Sensor.

[41] B. R. Martin, "Statistics for Physical Science," Academic Press, 2012.

[42] C. Veerappan *et al.*, "A 160 × 128 Single-photon Image Sensor with On-pixel 55ps 10b Time-to-Digital Converter," ISSCC Dig. Tech. Papers, 2011.

[43] C. Nickass, *et al.*, "A CMOS 64x48 Single Photon Avalanche Diode Array with Event-Driven Readout," ESSCIRC Dig. Tech. Papers, 2006.

[44] S. Lindner, *et al.*, "Column-parallel Dynamic TDC Reallocation in a SPAD Sensor Module Fabricated in 180 nm CMOS for Near Infrared Optical Tomography," Proc. Int. Image Sensor Workshop, 2017.

[45] A. Carimatto, *et al*., "A 67,392-SPAD PVTB-compensated Multi-channel Digital SiPM with 432 Column-parallel 48ps 17b TDCs for Endoscopic Time-of-Fight PET," ISSCC Dig. Tech. Papers, 2015.

[46] S. Mandai, *et al*., "A 780 × 800 μm^2 Multichannel Digital Silicon Photomultiplier with Column-parallel Time-to-Digital Converter and Basic Characterization," IEEE Trans. Nucl. Sci., 2011.

[47] R. J. Proebsting, "Speed Enhancement Techniques for CMOS Circuits," US Patent 4,985,643.

[48] A. Chandrakasan, *et al.*, "Design of High-Performance Microprocessor Circuits," 2000.

[49] C. Zhang, *et al*, "A 30-frames/s, 252 × 144 SPAD Flash LiDAR with 1728 Dual-Clock 48.8-ps TDCs, and Pixel-Wise Integrated Histogramming" IEEE Journal of Solid State Circuits, 2019.

[50] M. Perenzoni, *et al*, "A 64×64-Pixels Digital Silicon Photomultiplier Direct TOF Sensor With 100-MPhotons/s/pixel Background Rejection and Imaging/Altimeter Mode With 0.14% Precision Up To 6 km for Spacecraft Navigation and Landing," IEEE Journal of Solid State Circuits, 2016.

[51] J. M. Pavia, *et al.*, "A 1 × 400 backside-illuminated SPAD sensor with 49.7 ps resolution, 30 pJ/sample TDCs fabricated in 3D CMOS technology for near-infrared optical tomography," IEEE J. Solid-State Circuits, 2015.

[52] A. Chandrakasan, *et al.*, "Design of High-Performance Microprocessor Circuits," 2000.

[53] B. Markovic, *et al.*, "A High-Linearity, 17 ps Precision Time-to-Digital Converter Based on a Single-Stage Vernier Delay Loop Fine Interpolation," IEEE Transactions on Circuits and Systems, 2013.

[54] J. Richardson, *et al.*, "A 32x32 50ps Resolution 10 bit Time to Digital Converter Array in 130nm COMS for correlated Imaging," Custom Inttegrated Circuits Conference, 2009

[55] F. Villa, *et al.*, "CMOS Imager With 1024 SPADs and TDCs for Single-Photon Timing and 3-D Time-of-Flight," IEEE Journal of Selected Topics in Quantum Electronics, 2014.

[56] C Niclass, *et al*, "A 128x128 Single-Photon Image Sensor With Column-Level 10-Bit Time-to-Digital Converter Array," IEEE Journal of Solid State Circuits, 2008.