3D-Stacked SPAD Sensor with In-Pixel Multi-Frame Storage for Photon Counting and Time Resolved Applications

Tarek Alabbas
Ouster UK Ltd. | +44 (0) 131 563 9078
125 Princes Street, EH2 4AD,
Edinburgh, UK
tarek.alabbas@ouster.io

Yiyang Liu
The University of Edinburgh
King's Buildings, West Mains Road,
EH9 3FF, Edinburgh, UK
y.liu-305@sms.ed.ac.uk

Neil Calder
Ouster UK Ltd.
125 Princes Street, EH2 4AD,
Edinburgh, UK
neil.calder@ouster.io

Robert Henderson
The University of Edinburgh
King's Buildings, West Mains Road,
EH9 3FF, Edinburgh, UK
robert.henderson@ed.ac.uk

***Abstract*— We present a 3D-stacked BSI SPAD sensor architecture with high memory storage capacity enabling multi-frame and multi-bin photon counting and time resolved applications. The sensor utilizes an in-pixel SRAM macro with a SPAD driven precharge-read-increment-write operation based on a compact linear feedback shift register (LFSR) implementation. This allows for a 21.5µm pixel pitch with 16 memory locations and 18-bits each in a 65nm bottom tier process.**

I. Introduction

The high photon detection efficiency (PDE), timing resolution and low dark count rate (DCR) of modern day SPADs make them suitable candidates for a variety of intensity and depth imaging applications. Many publications have demonstrated various photon counting [1,2,3,4,5] and time of flight [6,7,8,9] architectures with recent focus on 3D-stacked technologies for optimal SPAD and circuitry performance [10].

A common theme among all architectures is addressing the storage memory required in-pixel. This is typically constructed of logic cells such as d-type flip-flops (DFFs). Photon counting pixels focus on reducing the pixel size for array scalability. Generally, they employ a limited number of bits coupled with off focal plane sub-frame accumulation and/or time to saturation (TTS) techniques to boost the dynamic range (DR) [1,2,4,5]. Direct time of flight (DTOF) pixels focus on the efficient usage of limited memory resource for storing time stamps [7] or partial histogram data [6,8,9]. Commonly, they employ gating or zooming techniques to capture information over the full measurement range.

In this work, we present a pixel employing a high density SRAM macro to provide either a multi-frame store for intensity imaging or a multi-bin histogram for DTOF. SRAM offers more than 10× storage capacity compared to a standard cell DFF in the same technology node. However; unlike DFFs, SRAMs require peripheral circuitry to perform precharge-read-increment-write (PRIW) operations which offsets their area efficiency. Thus, we utilize an asynchronous SPAD event-driven PRIW operation implemented via an in-pixel timing generator to minimize this overhead. Furthermore, a compact realization of the increment logic is implemented via an implicit 18-bit linear feedback shift register (LFSR) with XNOR feedback.

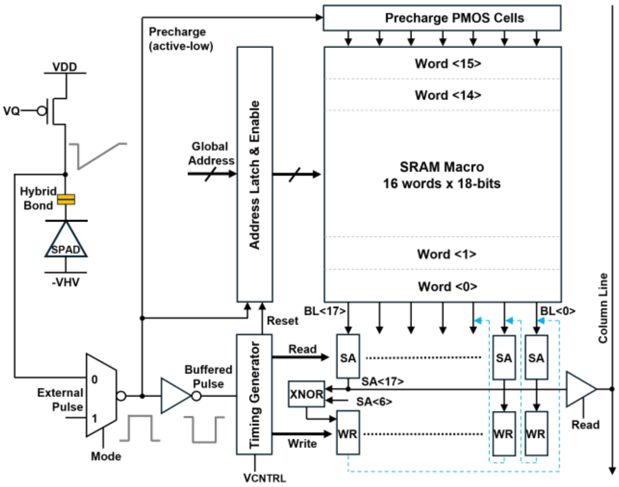

Figure 1. Pixel block diagram.

The SRAM word addressing is operated at pixel row level allowing reprogrammable array operation. When operated at different time-scales, the memory can serve different applications. At nanosecond time-scales, coarse time resolved information can be collected and stored in-pixel where memory locations serve as time bins. At microsecond or millisecond time-scales, high speed photon counting enables observing fast occurring phenomena where memory locations serve as a burst or a frame storage before readout is initiated.

II. Architecture

Figure 1 shows the overall pixel block diagram which includes an SRAM macro of 16 words by 18-bits each. The high bit depth allows for harnessing the native SPAD dynamic range. Above the SRAM macro there are PMOS precharge cells to ensure the internal bit-lines are charged up before addressing. Below, there are differential sense amplifiers (SAs) to sample the state of the bit-lines when the SRAM is addressed. The SAs then feed the tri-stated write drivers that can write back into the addressed word. SA<6> and SA<17> also feed an XNOR gate that performs the LFSR code shift.

A global 16-bit one-hot code is broadcast across the chip during operation to provide an address to the SRAM macro. The rate of change of the global code depends on the target application and can vary between nanoseconds and milliseconds for coarse time resolved and multi-frame modes respectively.

The following sections explain the SPAD driven pixel operation in more detail.

A. SPAD Front End

The SPAD front end is minimalistic by design and includes a thick oxide PMOS device for passive quench and recharge. The moving node (cathode) is interfaced directly with the thin oxide logic (MUX). While this limits the SPAD's excess bias to 1.2V (VDD) and therefore the pixel's sensitivity, this does provide an area advantage. This is deemed an acceptable trade-off due to the high PDE of modern SPADs.

The deadtime of the SPAD is controlled by a global VQ bias voltage at the gate of the PMOS transistor. VQ is set to provide ~7ns deadtime as a compromise between dynamic range and the timing margin required for the SPAD driven memory operation.

The SPAD device itself resides fully on the top tier of the stacked sensor and is connected to the bottom tier circuitry through a single hybrid-bond site.

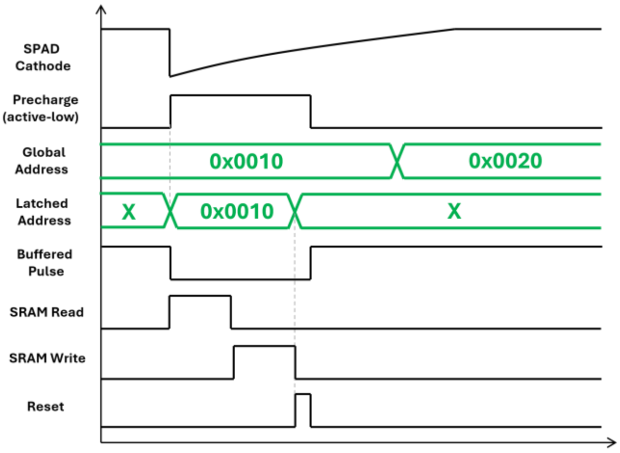

Figure 2. Example pixel timing diagram.

B. Pixel Timing

Figure 2 provides an example timing of the pixel operation. Before a SPAD event occurs, the SRAM memory is idle and the differential bit-lines are held in precharge state. Meanwhile, the row driven global address code is toggling over time.

Once a SPAD event occurs, the bit-lines are released from precharge and the global one-hot code is sampled into the pixel latches such that one memory word is addressed.

Simultaneously, the buffered SPAD pulse is passed into an in-pixel timing generator constructed from delay cells with a global delay control voltage V_{CNTRL}. The timing generator then generates three consecutive pulses within the SPAD deadtime to trigger the memory operation.

First, a read pulse is generated allowing for the SRAM bit-lines to develop a differential voltage which is sampled by the sense amplifiers onto local capacitors. The sampled state represents the previous LFSR value stored in the SRAM and only needs to be held in the capacitors for the duration of the write operation (few nanoseconds). The values in SA<6> and SA<17> are used by an XNOR gate to generate the next '1' or '0' bit to be shifted into the LFSR.

Second, a write pulse is generated which activates the differential tristate write drivers to write back the new LFSR code into the same addressed memory word. The write operation itself performs the LFSR code shift implicitly as every write driver is hard wired onto the next SRAM cell bit-lines. For example, write driver <0> is wired onto bit-lines <1>, write driver <1> is wired onto bit-lines <2> and so on. The only exception is write driver <17> which unlike the rest takes in the XNOR gate output as its input and is wired back onto bit-lines <0>. This wiring pattern is demonstrated in Fig.1 as blue dashed lines.

Finally, and after the write operation is complete, a reset pulse is generated to release the address latch such that no memory word is addressed. The pixel is now ready for subsequent SPAD events to trigger the same precharg-read-increment-write cycle again.

C. LFSR Counter

An LFSR counter architecture was chosen for two main reasons. Firstly, it provides a more compact implementation than the traditional binary adder. This is evident by the use of a single XNOR gate as opposed to 18 half adders for the 18-bit word length. The implicit shift operation which requires no physical registers is also area efficient. Secondly, the LFSR shift operation is not dependent on the settling time of a long adder chain and is therefore faster.

The LFSR counter does have some limitations however. Ideally the pixel memory values should be reset to a known state before integration and then the code is shifted pseudo-randomly every time a SPAD event occurs. Hence, the final resulting code is not readily intelligible and requires decoding to a decimal equivalent which could be an overhead for the readout system, specially when the word length is large. The LFSR also has a forbidden state which could cause the pixel to lock up. For an XNOR design, an all ones state would render the LFSR locked up.

Unfortunately, such anomalies are seen in captured images as fixed pattern noise of a small number of pixels which are believed to have an incorrect timing behaviour where the precharge state (all ones) gets sampled into the sense amplifiers therefore locking the LFSR code.

D. Readout and Reset Mechanism

To read out the integrated values in the SRAM memory, the pixel needs to be configured in readout mode. This is done by setting the global Mode signal high to switch the pixel input to accept external pulses rather than the SPAD output. The Read signal also needs to be set high in the typical row by row rolling shutter fashion.

Once a row is selected for readout, the global address is set to select the first memory word. To commence the readout process an external pulse is then injected into the pixel triggering an SRAM PRIW. This results in the previous word value to be sampled into the SAs.

At this point the value of SA<17> is buffered by the single tri-state read driver onto the column readout line and can be sampled at the edge of the array into the column parallel readout pipeline. Injecting another pulse results in another PRIW operation but due to the shift operation the previous value in SA<16> now appears in SA<17> and so gets buffered onto the column output line. Repeating this process for a total of 18 pulses results in a full readout of the original LFSR code that was stored in the selected memory location.

Once the word readout is complete the write drivers can be forced to write all zeros into the selected memory word to reset it using a row driven reset pulse (not shown in Fig.1).

Similarly, the global address is updated to select the next memory word and another 18 pulses are injected to complete the readout of the new address. This mechanism is repeated a total of 16 times per row to read out the full memory macro before the next row is selected. As discussed, the LFSR codes can now be decoded into their equivalent decimal counts.

This serial readout economises 18 tri-state drivers that would otherwise be necessary to read the word in parallel. It does not limit the sensor's frame rate due to a combination of the in-pixel frame storage and the small overall number of pulses per row when compared to the operation of an ADC in a typical image sensor.

III. Demonstrative Results

To demonstrate the sensors functionality, several images were captured in different modes of operation.

A. *Photon Counting*

The sensor was set to capture 16 consecutive frames of 100µs exposure time each and the individual frames were stored in-pixel. The frames were then externally summed together. Figure 3 shows the resulting image of letter figures in room conditions.

Several image artefacts were observed in the image in the form of spuriously high counts or very low (including 0) counts. This is attributed to timing issues in the pixel operation. For example, an incomplete read or write operation can result in a corrupted LFSR code, and due to its random nature even a corruption of a single bit can result in a very high decoded value. Another suspected source of corruption is metastability in sampling the global address in-pixel resulting in more than one SRAM words being addressed during the address transition.

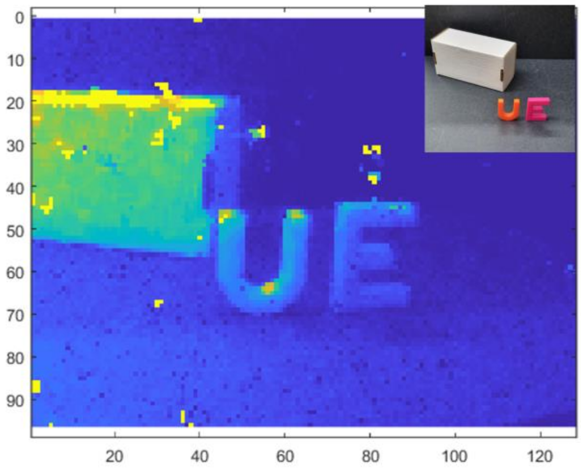

Figure 3. Photon counting image captured by the sensor. A course filter for the high corrupted pixel counts was applied. Camera photo of the scene is shown inset.

We believe a solution to the pixel's timing issues is to replace the asynchronous memory operation with a clocked front end at the expense of area and clock power.

The high counts in Fig.3 were coarsely filtered out. However, some clusters can still be seen.

B. *Global Shutter Operation*

To demonstrate the sensor's global shutter operation, the rate of change of the global address was set to 1ms and 16 frames were captured and stored in-pixel of a rotating fan in room conditions. Figure 4 shows raw frames number 2, 4 and 6 with the letter 'E' in the middle of the fan clearly visible as the fan rotates.

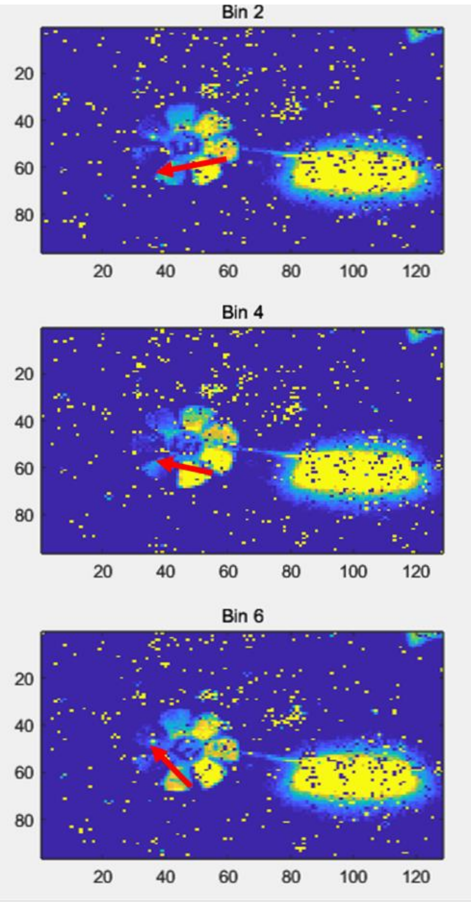

Figure 4. Three selected 1ms frames of a rotating fan demonstrating global shutter operation. Red arrows highlight the change in the letter 'E' orientation as the fan rotates.

C. *Burst Imaging*

To demonstrate the sensor's temporal resolution, another set of images were acquired of an LED flickering at 60kHz. The change rate in the global address was set to 1µs effectively resulting in 16 back to back 1µs frames to be stored in-pixel. This is equivalent to 1Mfps burst mode operation.

Figure 5 shows the 16 captured frames where the LED's temporal change in intensity is clearly visible.

IV. Summary and Conclusion

We presented a compact pixel architecture based on a SPAD driven SRAM macro operation for various imaging applications. The basic functionality of the sensor was demonstrated and pitfalls of the design were highlighted.

A micrograph of the 128×96 3D-stacked SPAD BSI SPAD sensor is shown in Fig.6 while Table 1 provides a comparison to the state of the art in terms of the sensor's features and memory capacity.

ACKNOWLEDGMENT

The authors would like to thank Ouster for supporting this work and Boan Fang for his lab assistance.

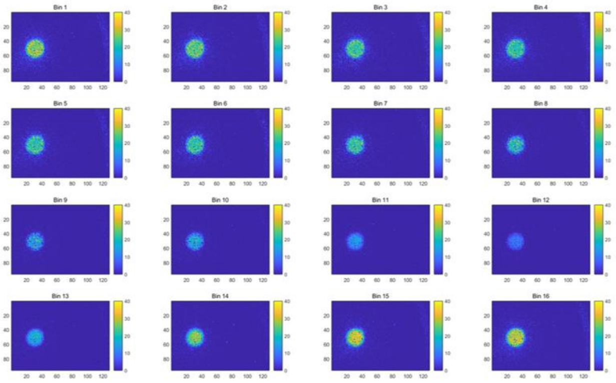

Figure 5. 16 frames at 1Mfps burst mode imaging of a 60kHz flickering LED. Each memory (i.e.bin) represents a 1µs slice in time.

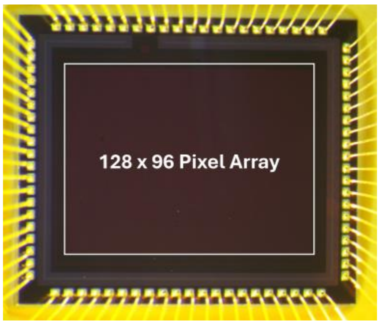

Figure 6. Micrograph of the 3.7mm x 3.1mm 3D-stacked BSI SPAD sensor.

REFERENCES

[1] T. Al Abbas et al., "A 128×120 5-Wire 1.96mm2 40nm/90nm 3D Stacked SPAD Time Resolved Image Sensor SoC for Microendoscopy," Symposium on VLSI Circuits, Kyoto, Japan, pp. C260-C261, 2019.

[2] J. Ogi et al., "7.5 A 250fps 124dB Dynamic-Range SPAD Image Sensor Stacked with Pixel-Parallel Photon Counter Employing Sub-Frame Extrapolating Architecture for Motion Artifact Suppression," IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, pp. 113-115, 2021.

[3] K. Morimoto et al., "3.2 Megapixel 3D-Stacked Charge Focusing SPAD for Low-Light Imaging and Depth Sensing," IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, pp. 20.2.1-20.2.4, 2021.

[4] Y. Ota et al., "A 0.37W 143dB-Dynamic-Range 1Mpixel Backside-Illuminated Charge-Focusing SPAD Image Sensor with Pixel-Wise Exposure Control and Adaptive Clocked Recharging," IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, pp. 94-96, 2022

[5] T. Takatsuka et al., "A 3.36 µm-pitch SPAD photon-counting image sensor using clustered multi-cycle clocked recharging technique with intermediate most-significant-bit readout," IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits), Kyoto, Japan, pp. 1-2, 2023.

[6] R. K. Henderson et al., "5.7 A 256×256 40nm/90nm CMOS 3D-Stacked 120dB Dynamic-Range Reconfigurable Time-Resolved SPAD Imager," IEEE International Solid-State Circuits Conference - (ISSCC), San Francisco, CA, USA, pp. 106-108, February 2019.

[7] P. Padmanabhan et al., "7.4 A 256×128 3D-Stacked (45nm) SPAD FLASH LiDAR with 7-Level Coincidence Detection and Progressive Gating for 100m Range and 10klux Background Light," IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, pp. 111-113, 2021.

[8] D. Stoppa et al., "A Reconfigurable QVGA/Q3VGA Direct Time-of-Flight 3D Imaging System with On-chip Depth-map Computation in 45/40nm 3D-stacked BSI SPAD CMOS," International Image Sensors Workshop, 2021.

[9] C. Zhang et al., "A 240 × 160 3D-Stacked SPAD dToF Image Sensor With Rolling Shutter and In-Pixel Histogram for Mobile Devices," in IEEE Open Journal of the Solid-State Circuits Society, vol. 2, pp. 3-11, 2022.

[10] M. Wojtkiewicz et al., "Review of Back-Side Illuminated 3-D-Stacked SPADs for Time-of-Flight and Single-Photon Imaging," in IEEE Transactions on Electron Devices, vol. 71, no. 6, pp. 3470-3477, June 2024.

	[1] VLSI 2019	[2] ISSCC 2021	[3] IEDM 2021	[4] ISSCC 2022	[5] VLSI 2023	[6] ISSCC 2019	[7] ISSCC 2021	[8] IISW 2021	[9] JSSC 2022	This Work
Pixel (macro) pitch	8µm	12.24µm	6.39µm	9.585µm	3.36µm	38.4µm	~79µm	50µm	32µm	21.5µm
Pixel structure	Single SPAD	Single SPAD	Single SPAD	Single SPAD	Single SPAD	4x4 SPAD Macro-pixel	16x8 SPAD Macro-pixel	4x4 SPAD Macro-pixel	2x2 SPAD Macro-pixel	Single SPAD
Resolution (SPADs)	128x120	160x264	2072x1548	960x960	748x448	256x256	256x128	320x240	240x160	128x96
3D-Stacked	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES
Technology node (bottom-tier)	40nm	40nm	40nm	40nm	22nm	40nm	22nm	40nm	65nm	65nm
Bits per pixel	14	14	11	14	8	16x14	86	32x12	32x8	16x18
Pixel memory type	Logic cells	Logic cells	Logic cells	Logic cells	Logic cells	Logic cells	Logic cells	Logic cells	SRAM	SRAM
Pixel storage density	0.22 bits/µm²	0.093 bits/µm²	0.27 bits/µm²	0.15 bits/µm²	0.71 bits/µm²	0.15 bits/µm²	0.014 bits/µm²	0.15 bits/µm²	0.25 bits/µm²	0.62 bits/µm²
Photon counting multi-frame storage	NO	NO	NO	NO	NO	NO	NO	NO	NO	YES
Photon counting HDR technique	Off focal plane accumulation	TTS + off focal plane accumulation	Counter limited	TTS	Clustered CLK recharge + off focal plane accumulation	Counter limited	Counter limited	Counter limited	Counter limited	SPAD native
DTOF technique	n/a	n/a	n/a	n/a	n/a	16-bin Histogram	TDC codes	16-bin Histogram	32-bin Histogram	16-bin Histogram

Table 1. Comparison to the state of the art.