Multiresolution Image Sensor

Sabrina E. Kemeny, *Member, IEEE*, Roger Panicacci, Bedabrata Pain, Larry Matthies, and Eric R. Fossum, *Senior Member, IEEE*

***Abstract*— The recent development of the CMOS active pixel sensor (APS) has, for the first time, permitted large scale integration of supporting circuitry and smart camera functions on the same chip as a high-performance image sensor. This paper reports on the demonstration of a new 128 × 128 CMOS APS with programmable multiresolution readout capability. By placing signal processing circuitry on the imaging focal plane, the image sensor can output data at varying resolutions which can decrease the computational load of downstream image processing. For instance, software intensive image pyramid reconstruction can be eliminated. The circuit uses a passive switched capacitor network to average arbitrarily large neighborhoods of pixels which can then be read out at any user-defined resolution by configuring a set of digital shift registers. The full resolution frame rate is 30 Hz with higher rates for all other image resolutions. The sensor achieved 80 dB of dynamic range while dissipating only 5 mW of power. Circuit error was less than −34 dB and introduced no objectionable fixed pattern noise or other artifacts into the image.**

***Index Terms*— Focal plane array, image processing, imager, multimedia, sensor.**

I. Introduction

FOR a variety of image processing tasks, such as biological vision modeling, stereo range finding, pattern recognition, target tracking, and transmission of compressed images, it is desirable to have image data available at varying resolutions to increase processing speed and efficiency. The user can then obtain a frame of data at the lowest resolution necessary for the task at hand and eliminate unnecessary processing steps. The multiresolution image data is usually generated through an image pyramid approach (implemented in software), and has been used extensively in recent years [1]–[4]. Typically, each image level is a low-pass filtered and down-sampled version of the prior level, although block averaging and down sampling can also be used to generate the pyramid [5]. In software, construction of the multiresolution pyramid is often the most computationally intensive and time consuming portion of the image processing task. For applications where power consumption is of concern, the power consumed by the processor while performing this task can be critical. These problems become especially severe for image processing tasks performed on large format imagers (e.g., 1024 × 1024) that are read out at video rates (30 frames/s).

CMOS active pixel sensor (APS) technology allows the integration of support electronics and smart camera functions onto the same chip as a high-performance image sensor [6]. The integration of support electronics such as timing and control, correlated double sampling, and analog to digital conversion leads to fewer components, thus increasing system robustness while reducing system mass and cost. The implementation of programmable multiresolution readout allows unprecedented camera functionality which eases the performance requirements of downstream image processing. The CMOS APS technology also enjoys other advantages over its charge-coupled device (CCD) counterpart such as ultra low power performance (50–100× lower than comparable CCD systems) and increased radiation hardness [7]. CMOS APS architectures [Fig. 1(a)] allow x–y addressability of the array for windows of interest and sparse sampling readout of the array. Unfortunately, sparse sampling the array, for example, by reading out every fourth pixel of every fourth row, reduces the amount of image data by a factor of 1/16 but introduces aliasing into the image. In the multiresolution sensor, regions of the array are averaged together (block or kernel averaging) and read out [Fig. 1(b)], leading to data reduction without aliasing effects.

The multiresolution CMOS APS is a 128 × 128 photogate array that is programmable to read out any size $n \times n$ block of pixels (kernel). Each kernel value represents the average of all the pixel values in its region. Averaging is done in the column readout circuitry so that the average value is based on a full resolution image. Combining the sensor's X–Y addressability with programmable resolution, the device can achieve true electronic zoom capability. In a standard digital camera, electronic zoom is achieved by mapping each pixel in a small area of interest to several display pixels. In contrast, the multiresolution sensor allows one to read out a small area of interest at a higher resolution than the full frame such that each pixel may be mapped to an individual display pixel much like an optical zoom lens allows one to capture more detail in a small area. This capability can also be used to increase processing speed of tracking algorithms where course resolution image data can be quickly read out and processed to determine an area of interest followed by read out of the area of interest at a higher resolution in the subsequent frame as illustrated in Fig. 2.

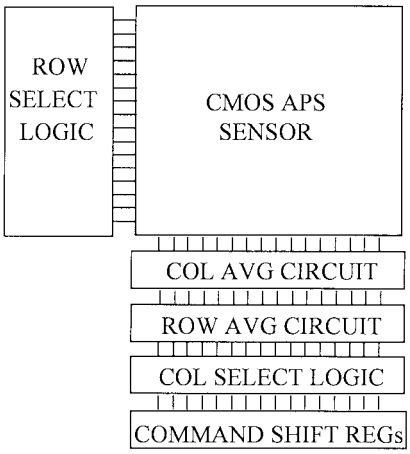
Details of the multiresolution sensor operation are discussed in Section II. Section III presents the test results from the fab-

Manuscript received September 30, 1996; revised January 31, 1997. This paper was recommended by Guest Editors B. Sheu, C.-Y. Wu, H.-D. Lin, and M. Ghanbari. This work was performed by the Center for Space Microelectronics Technology, Jet Propulsion Laboratory, California Institute of Technology and was sponsored in part by the Jet Propulsion Laboratory Director's Discretionary Fund (DDF) and the National Aeronautics and Space Administration, Office of Space Access and Technology.

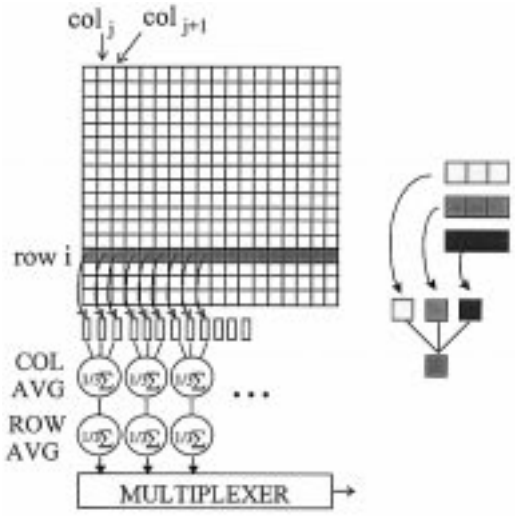
S. E. Kemeny, R. Panicacci, and E. R. Fossum are with Photobit, La Crescenta, CA 91214 USA.

B. Pain and L. Matthies are with the Jet Propulsion Laboratory–California Institute of Technology, Pasadena, CA 91109 USA.

Publisher Item Identifier S 1051-8215(97)05896-5.



(a)



(b)

Fig. 1. (a) Programmable multiresolution CMOS active pixel sensor architecture and (b) example of column's functional configuration for 3 × 3 block averaging. (Actual neighborhood size is programmable.)

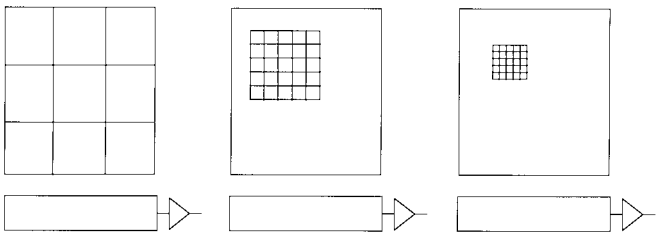


Fig. 2. Sensor X-Y addressability and multiresolution readout allows the user to zoom into an area of interest with increased resolution.

ricated chip. Finally, applications of the sensor are discussed in Section IV.

II. Design and Operation

A. Design Overview

The sensor contains a 128 × 128 photogate pixel array similar to previous APS arrays demonstrated at the Jet Propulsion

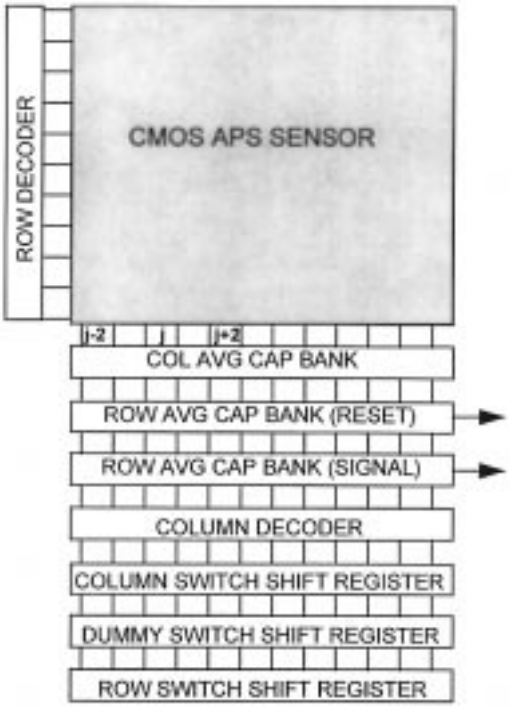


Fig. 3. Multiresolution image sensor block diagram.

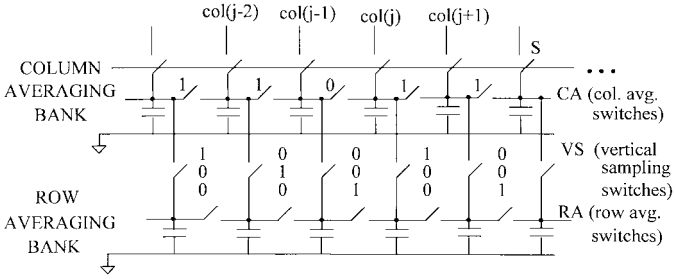


Fig. 4. Ideal switch and capacitor model for six columns configured for 3 × 3 block averaging.

the array is a network of capacitors to store pixel reset and signal levels. The column circuitry also contains an additional capacitor and a set of switches to the adjacent column to perform averaging on any size square array of pixels called a kernel (rectangular kernels are also possible). Resolution of the sensor is set by the size of the kernel. Large kernels sizes are set for low resolution imaging requirements. The X–Y addressability of the sensor allows the user to zoom into areas of interest.

Fig. 3 shows a block diagram of the sensor. A decoder at the side of the array selects a row of pixels for readout. Each pixel is controlled by a photogate signal enabling readout of integrated charge, a reset signal, and select signal to enable the buffered pixel data to drive the column output line. Column parallel circuitry at the bottom of the array samples the addressed row of pixel data simultaneously. The kernel size determines how a set of shift registers in the column circuits are configured. These shift registers control how the columns containing stored readout data are averaged

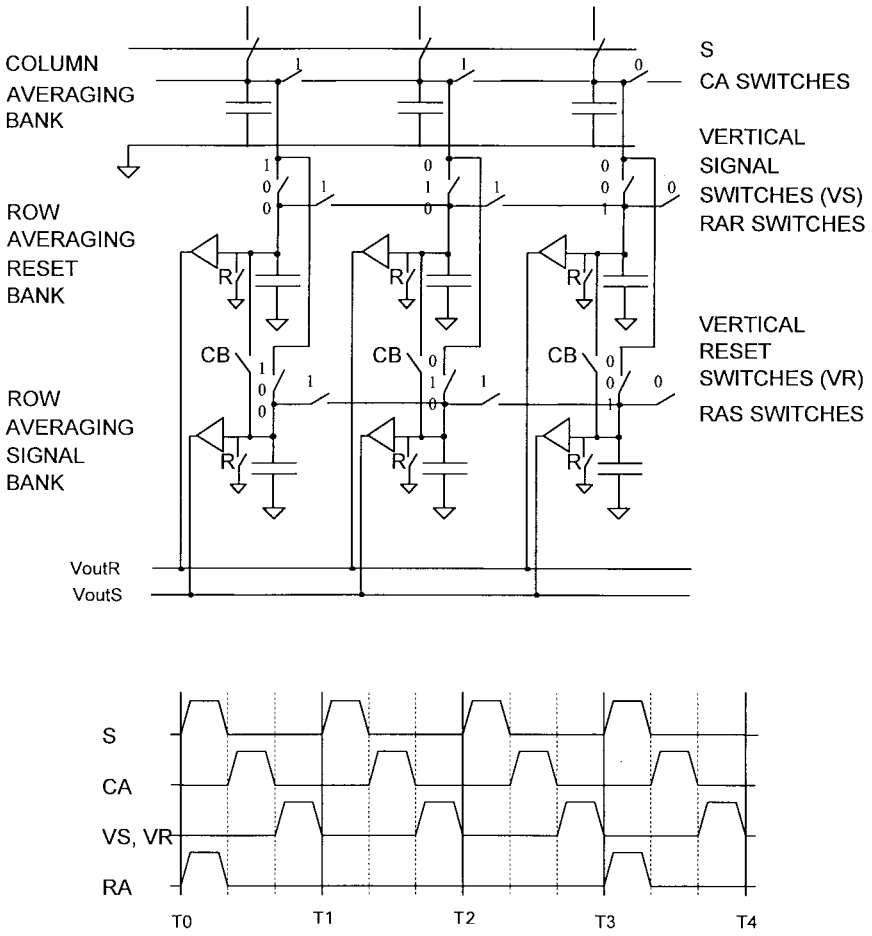


Fig. 5. Multiresolution column processing circuitry for three columns.

processing. A second decoder at the bottom of the array controls which columns containing the processed data are read out. The sensor's differential output circuitry performs correlated double sampling (CDS) to suppress pixel kTC noise, $1/f$ noise, and fixed pattern noise.

Row pixel data is read onto a column averaging capacitor with switches to its neighboring columns that are subsequently enabled resulting in averaged column data for that row (Fig. 4). Averaged column data for that row is stored on a second bank of capacitors in one of the columns of the kernel. Data from successively read out rows is stored in each of the remaining columns in the kernel. Shift registers in the readout circuitry are configured according to kernel size to determine which switches are enabled to perform averaging and where the averaged column data is stored.

Once all n rows of the kernel are read, they are averaged by connecting the second bank of capacitors together and mixing the charge. A shift register configured to enable dummy switches to correct for switch feedthrough effects is also included. Both reset and signal levels are processed for a kernel so that correlated double sampling and double-delta sampling operations can be performed.

Operation will be illustrated by stepping through the sequence for 3×3 block (kernel) averaging (Fig. 4). In this deselected, denoted by bit 0 over the switch), while the other switches are closed (i.e., selected, denoted by bit 1 over the switch). Pixel signals are sampled onto the column averaging capacitors by globally pulsing (closing) the signal select switches (S). Charge redistributes such that the voltage on each capacitor in each block of three capacitors V_{i_ker} is the same such that

$$V_{i\_ker} = \frac{1}{n} \sum_{k=1}^{n} V_{j-k}$$

where n is the horizontal size (number of columns) of the block average (kernel), V_{j-k} the pixel voltage value of the $(j–k)$th column, and V_{i_ker} is the average value for the ith row in the kernel. These kernel row averages are then sampled onto the first capacitor in the n-capacitor block of the row averaging bank of capacitors. Column averaging is repeated with the next pixel row $(i + 1)$ and these new $V_{(i+1)\_ker}$ kernel averages are sampled onto the second capacitor in the n-capacitor blocks of the row averaging bank of capacitors. The process is repeated until all n rows have been processed and n samples have been collected in n adjacent capacitors in the row averaging bank. The temporal switching sequence (digital pattern) is shown for the 3×3 kernel case (Fig. 5). After the n-samples (V_i, V_{i+1}, ..., $V_{i+(n-1)}$) have been collected

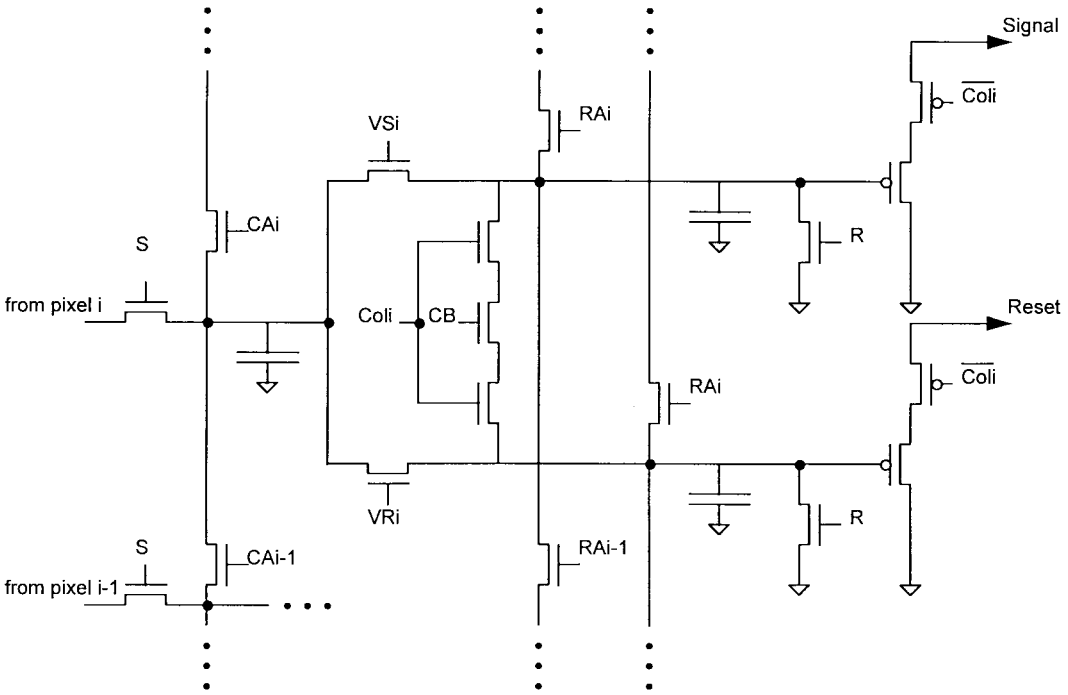


Fig. 6. Transistor-level schematic of column circuit. Capacitors are poly-diffusion linear capacitors.

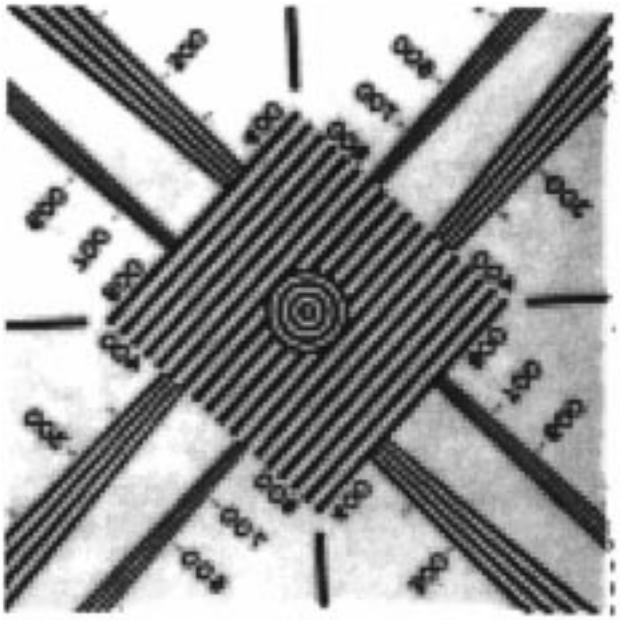
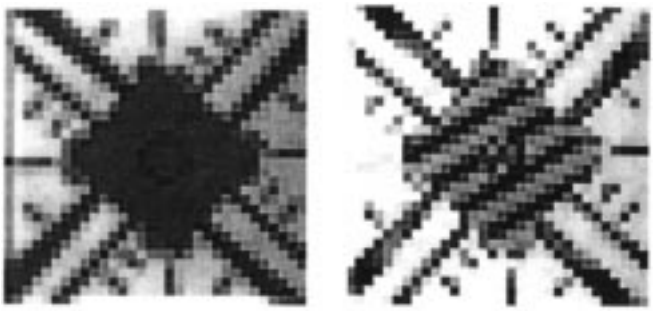


Fig. 7. Sensor's full resolution image (128 × 128).



(a) (b)

Fig. 8. (a) 4 × 4 block averaged time and (b) 1/4 subsampled image (no averaging).

switches with the same pattern used for the column averaging switches, resulting in the final block average

$$V_{ker} = \tfrac{1}{2} \sum_{i=1}^{m} V_{i\text{-}ker}$$

where m is the vertical size (number of rows) in the kernel. The constant factor of 1/2 arises from charge sharing between the column and row averaging capacitors when the column

for the first 3 × 3 kernel

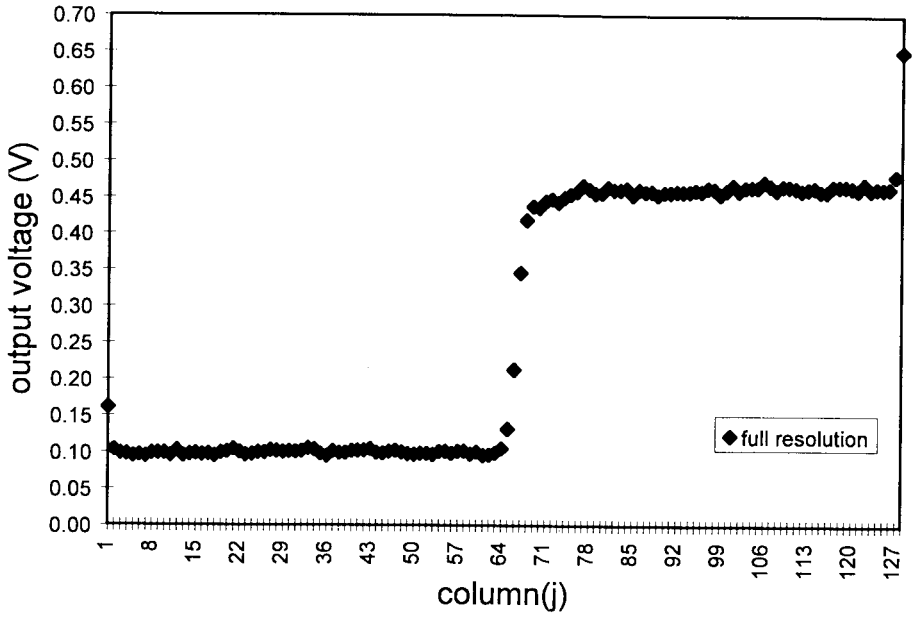
$$\begin{aligned} V_{ker\,0} = 1/2\{1/3[1/3(V_{0,0}+V_{0,1}+V_{0,2}) \\ +1/3(V_{1,0}+V_{1,1}+V_{1,2}) \\ +1/3(V_{2,0}+V_{2,1}+V_{2,2})]\}. \end{aligned}$$

These kernel values are then scanned out of the array by reading out every nth capacitor in the row average bank. The row averaging capacitors are then reset (circuitry not shown) and the process is repeated to generate the next row of kernels.

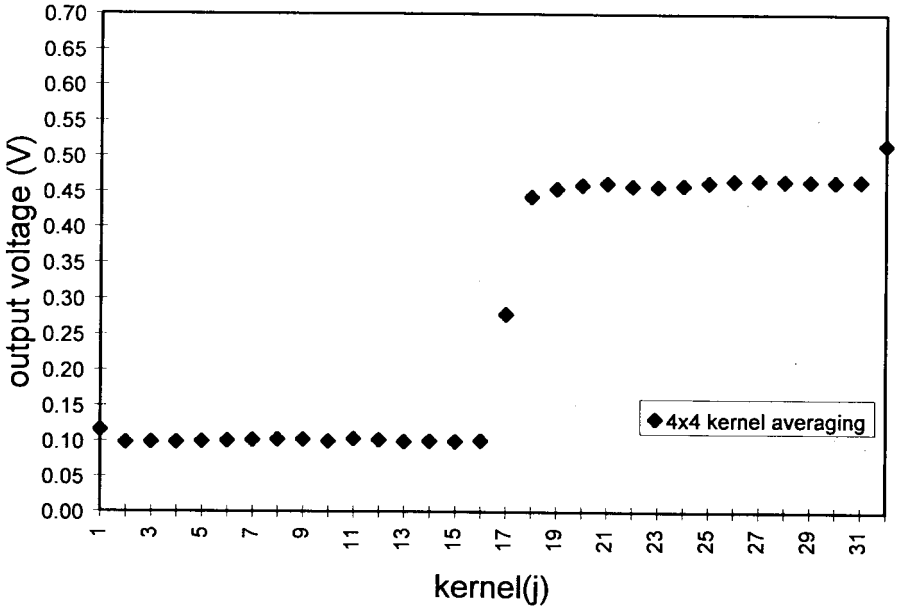
Note that in the configuration described above, kernels must be either square or rectangular, where the number of rows must be less than or equal to the number of columns.

B. Column Processing Circuitry

Shown in Fig. 5 is the actual column parallel circuitry for three columns. There is one bank of column averaging



(a)



(b)

Fig. 9. (a) Full resolution sensor output for one row of pixels and (b) 4 × 4 kernel output from sample image.

than the single bank shown in Fig. 4). One bank stores the row average pixel reset levels, and the other stores the row average pixel signal levels in order to perform on-chip double correlated sampling. The column averaging bank is used sequentially to horizontally average together the kernel row reset levels followed by the signal levels. The kernel reset switch to ground is shown as well as the column buffer amplifier for generating $V_{out}R$ and $V_{out}S$. The buffer amplifier is only enabled when the column is selected for readout.

The digital patterns shown are an example of the timing for a 3 × 3 kernel. They are generated by gating the output of the configuration shift registers and the timing signals shown in Fig. 5. The global timing signals are CA (enable column onto row averaging capacitor), and VR (sample reset onto row averaging capacitor). Each of these global signals is gated with the output of one of the two configuration shift registers. The CA and RA signal are gated with the output of the same shift register (CARA shift register). The VS and VR signals are gated with the output of the second shift register (VSVR shift register).

The transistor-level schematic of the column circuit is shown in Fig. 6. The signals CA_i, RA_i, VS_i, and VR_i are the outputs from the corresponding global signals gated with the shift register output bit for that column. The buffer amplifier is a p-channel source follower. The CB signal is part of the double delta sampling readout scheme as reported in [9] used

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.