

# MORTON (Z) SCAN BASED REAL-TIME VARIABLE RESOLUTION CMOS IMAGE SENSOR

*Evgeny Artyomov, Yair Rivenson, Guy Levi, Orly Yadid-Pecht, Senior Member, IEEE*

The VLSI Systems Center, Ben-Gurion University, Beer-Sheva 84105, Israel

## ABSTRACT

An image sensor architecture with an alternative image scan method, based on Morton (Z) order, is presented. This scan, compared to the conventional row (raster) scan, enables faster and efficient mean (average) computation of square image blocks. Digital averaging is used and the pixel data is read out with either the original resolution, a  $2 \times 2$  or a  $4 \times 4$  block averaging. A test chip of  $128 \times 128$  array has been implemented in  $0.35 \mu\text{m}$  CMOS technology, has 15% fill factor, is operated by a 3.3V supply and dissipates 30mW at video rate

## 1. INTRODUCTION

Most of existing image sensors use row (raster) scan for readout [1]. This scan is best for applications where all pixels have to be transmitted out of a chip without additional spatial processing. In the case that spatial processing is required (for mean (average) or median computation, for example), multiple additional storage elements are required (digital or analog) outside the pixel matrix or inside the pixel.

Various architectures for variable resolution image sensors via pixel averaging were proposed in the literature [2]-[7]. In [2] resolution reduction is achieved by charge sharing via a capacitor bank. Multiple additional capacitors are added to the chip in order to save signal from multiple rows of the sensor. In [3] a passive switch-capacitor circuit is used for pixel averaging. The main disadvantage of this design is the large pixel that incorporates additional capacitors for sample and hold. The same disadvantage exists in [4]. Averaging is performed inside the sensor array, which needs additional transistors inside the pixel. As a result, the pixel consists of two PMOS and six NMOS transistors. The approach presented in [5] uses currents for averaging. However, only simulation results are presented. In [6], multiple resolution is achieved by integrating the "Signal" and "Reset" separately from different rows with the help of switch-capacitor integrators. In the dynamically reconfigurable vision CMOS image sensor [7], windows of higher resolution can be concentrated on the object in the scene. External control for window position and size is required.

All the variable resolution imagers mentioned above are capable of changing resolution over the whole sensor or in regions of interest via external control. Multiple elements have to be added inside the pixel in some architectures [3], [4], that increase pixel size and/or decrease fill factor. Much circuitry is required outside the pixel matrix in the architectures presented in [2], [6], [7]. These problems are mostly caused by the kind of scan, which is used in image sensors – the row (raster) scan. This scan is most suitable and simple for image transmission and storage, in the case where all pixels have to be transmitted. Other types to order pixels during readout also exist, Morton (Z) and Peano-Hilbert are two examples [8], [9]. These orders are also used for image compression [10].

In this work, we present a new image sensor architecture and implementation, based on the Morton scan, for fast and efficient mean computation of square pixel blocks ( $2 \times 2$ ,  $4 \times 4$ , ...), that can be used for variable resolution, motion detection, tracking, compression etc in image sensors.

## 2. MORTON ORDER – GENERAL DESCRIPTION

Fig. 1 shows four different types of spatial ordering methods [8], [9].

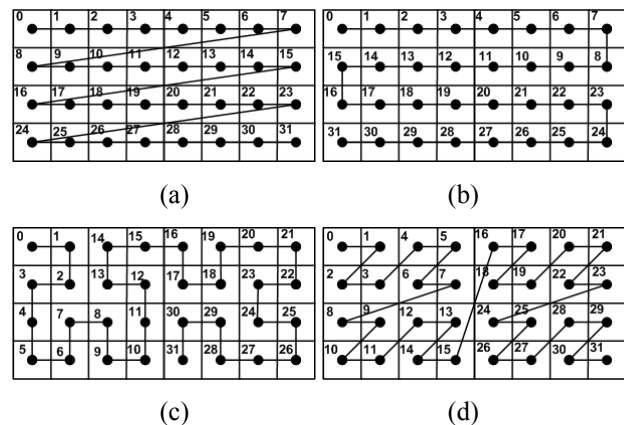


Figure 1. Different space ordering methods

- a) Row (Raster) order, b) Row prime order,
- c) Peano-Hilbert order, d) Morton (Z) order

As can be seen from these images, in Morton and Peano-Hilbert orders, pixels, which have to be read-out sequentially, are concentrated in blocks. Block clusters of  $2 \times 2$ ,  $4 \times 4$ ,  $8 \times 8$  ... pixels, can be easily extracted, since pixels in these blocks are transmitted one after another (row ordering does not possess this valuable feature because pixels are transmitted serially row after row). This feature can be handy for spatial image processing, such as resolution reduction. In order to reduce image resolution by a factor of two, the mean of four pixels (a  $2 \times 2$  block) has to be calculated. With these orderings (Morton and Peano-Hilbert), it can be done in a simple, straightforward way, without requiring multiple storage elements. Pixels are summed and divided during read-out in a sequent manner, i.e. for instance pixels 0, 1, 2 and 3 can be added and divided by 4; pixels 4, 5, 6 and 7 can be processed in the same way etc. This calculation can be expanded to blocks of sizes  $4 \times 4$ ,  $8 \times 8$  etc.

For our implementation, we have chosen Morton order due to its simplicity of generation and decoding in hardware. In addition, Morton order possesses a valuable feature - the actual coordinates of the pixel can be extracted easily from the pixel index in the Morton sequence [8]. For example, the pixel with index 21 (image size -  $8 \times 8$  pixels) has a binary representation (010101). By partitioning this binary number into two parts (only odd and even digits) the row and column coordinates of the pixel can be extracted, respectively (000 = 0, 111 = 7).

### 3. SYSTEM ARCHITECTURE

#### 3.1 OVERALL SYSTEM ARCHITECTURE

The general architecture of the image sensor is presented in Fig. 2. It can be divided into three parts: a Morton-scan sensor, an A/D converter and a computational part. The Morton-scan sensor enables sequential reading of pixel data according to the Morton order. The computational part is a digital logic for the mean computation. It consists of a Full Adder of 11 bit and 12 registers. The size of the averaging block is changed by controlling the reset time of the registers and shifting of the result by 0, 2 or 4 bits accordingly. The presented configuration enables currently the computation of  $2 \times 2$  and  $4 \times 4$  blocks. In order to compute the mean of larger blocks, the digital part has to be modified accordingly. The control of the block size for averaging is currently done manually outside the chip and the resolution is changed for the entire frame. In the future, adaptability to the scene content can be inserted.

Since Morton scan requires sequential read-out of pixels from different rows of the sensor, the conventional sensor architecture has to be modified. At the same time, we do not want to complicate the structure significantly. These requirements are met by the architecture presented in Fig.2.

The Morton scan sensor architecture consists of an active pixel sensor (APS), a commutation circuit, two sample and hold (S/H) circuits, row and column block decoders, a column decoder and a multiplexer (MUX). The presented architecture allows selection of  $4 \times 4$  pixel blocks with the help of the row and column block decoders. Two groups of S/H circuits are required to enable simultaneous sampling of the pixel data to one of the circuits and the data transmission out of the chip from the other. Each group of S/H circuits consists of 16 circuits only, for each pixel from the block. The commutation circuit is actually an analog multiplexer that connects different output lines of the pixel columns to the appropriate S/H circuit. The multiplexer on the output chooses one of the S/H circuits for data transmission.

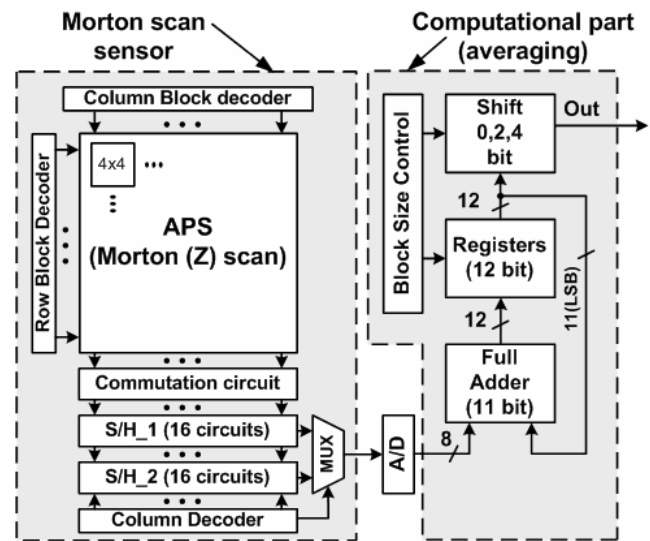


Figure 2. Real-time variable resolution image sensor – general block diagram

While one  $4 \times 4$  block is sampled by the first (S/H\_1) circuit, the pixel values from the previous block are transmitted out of the chip from the second (S/H\_2) circuit and visa versa. This is done via the column decoder and the MUX. This process continues block after block without wasting time for pixel sampling. Since 16 pixels are sampled to the S/H circuits in parallel, they are ordered during transmission out of the chip with the help of the column decoder, to keep the Morton order at the pixel level. A timing diagram of the process is shown in Fig. 3.

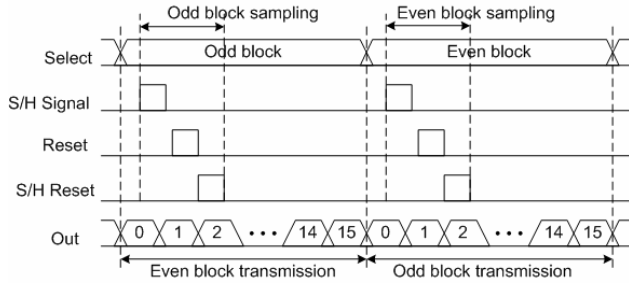


Figure 3. Timing of the block sampling and pixel transmission

### 3.2. PIXEL ARRAY ARCHITECTURE

In order to allow simultaneous reading from the four rows of the sensor (a 4×4 block) the conventional three-transistor photodiode pixel was changed.

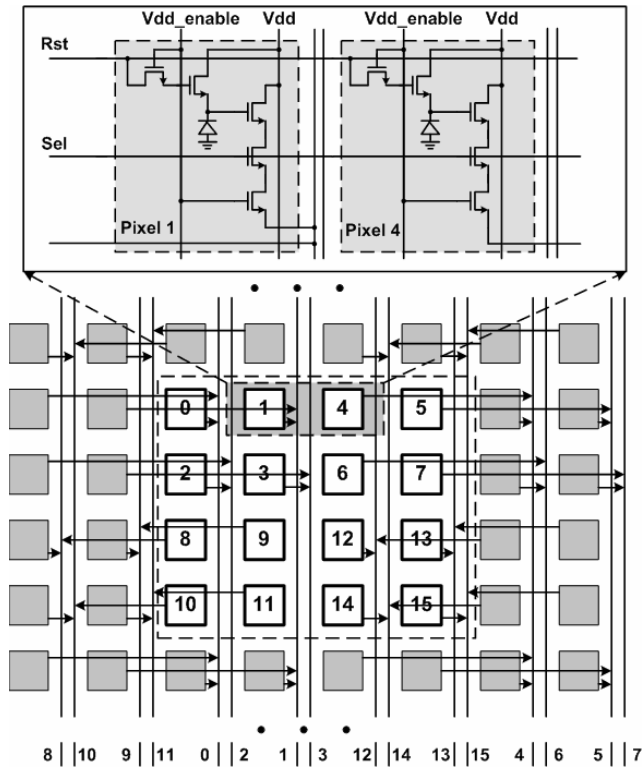


Figure 4. The pixel array architecture

A schematic description of the proposed pixel is presented in Fig.4. This pixel is a modified version of the pixel presented in [11], where additional transistor is added. The presented architecture allows selective “Reset” and “Select” of the desired group of pixels.

The architecture of the pixel array part is also presented in Fig.4 (only output signal lines are shown). Compared to the

conventional pixel array, additional vertical and horizontal wires are added to the pixel matrix. The presented array architecture efficiently utilizes lines from neighboring columns, allowing 4×4 blocks read-out with the addition of only one additional output line per column. The straightforward approach requires four output lines per column. In our design, two output lines are required per column, but each pixel is connected only to one output line. In addition, each output line can be utilized by two neighboring blocks. The required block is chosen by applying “Vdd\_enable” = “1” to the power pin of the four selected columns and “select” or “reset” to the appropriate four rows. Neighboring 4×4 blocks cannot be read out simultaneously since they share the common output lines.

In Fig. 4 an example of a read-out procedure for a one 4x4 block is also shown. Pixels that have to be read-out are numbered from 0 to 15. The active output lines for read-out are marked with the appropriate pixel index, which is being read. As can be seen from the figure, information, presented on the output lines, does not correspond to the Morton order. To keep the required order, pixels are reordered during transition out of the sensor with the help of the column decoder.

### 4. RESULTS

The test chip realizing the presented architecture was fabricated in a 0.35μm, double poly, n-well TSMC CMOS technology. It incorporates a 128×128 APS Morton scan array and a successive approximation A/D converter. Table 1 summarizes the chip attributes. The chip photomicrograph is shown in Fig. 5.

Table 1. Chip attributes

Technology	0.35μm
Voltage supply	3.3V, 4V
Sensor array size (pixels)	128×128
Pitch width	7μm
Fill factor	15%
Chip size	9 mm <sup>2</sup>
Frame scanning frequency	30 Hz
Power dissipation	30 mW
Averaging size (pixels)	1×1, 2×2, 4×4
Mean computation	Digital
Output	Digital, 8 bit

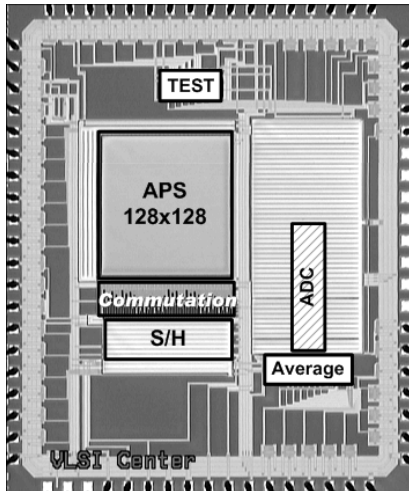


Figure 5. Chip photomicrograph

Examples of the chip output for different resolutions are shown in Fig. 6.

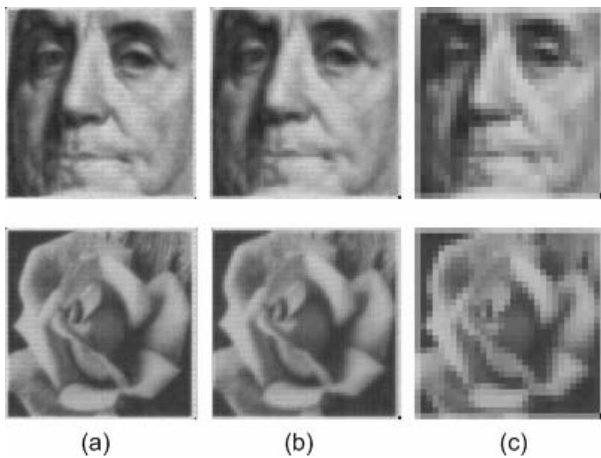


Figure 6. Chip test results: (a) full resolution, (b)  $2 \times 2$  averaging, (c)  $4 \times 4$  averaging

## 5. CONCLUSIONS

An architecture for a variable resolution sensor based Morton scan has been presented. Reduction of the resolution is performed during read-out from the sensor with the help of digital hardware. Since averaging is performed after CDS, additional noise reduction is achieved. This architecture will allow building of various smart image sensors that utilize variable resolution, without additional extensive hardware and power requirements.

The main disadvantage of the proposed architecture is that output is not standard and an appropriate decoder may be

required on the receiver site. However, there are various applications that can benefit from the presented architecture: cellular phones, space applications, surveillance, tracking etc. Such applications have tighter requirements on the sender site than on the receiver site, and can benefit much of the use of this proposed architecture.

## ACKNOWLEDGMENTS

The research described in this paper was funded by the Israeli Ministry of Science. The authors also thank Nir Goren for the help provided during chip testing.

## REFERENCES

- [1] O. Yadid-Pecht, R. Ginosar and Y. Shacham-Diamand, "A random access photodiode array for intelligent image capture", *IEEE Trans. on Electron Devices*, Vol. 38, No. 8, pp. 1772 - 1781, Aug.1991.
- [2] R. P. Panicacci, , S. Kemeny, L. Matthies, B. Pain, and Eric R. Fossum "Programmable multiresolution CMOS active pixel sensor", in *Proc. of SPIE, Solid State Sensor Arrays and CCD Cameras*, 1996, Vol. 2654, pp. 72-81.
- [3] Y.Ohtsuka, I.Ohta, K.Aizawa, "A new image sensor with programmable spatially variant multiresolution readout capability", *IEEE Workshop on CCD and Advanced Image Sensors*, Nevada, Jun. 2001, pp.40-43.
- [4] F. Saffih, R. Hornsey , "Multiresolution CMOS image sensor", *Technical Digest of SPIE Opto-Canada*, Ottawa, Ontario, Canada , May 2002, p.425.
- [5] J. Coulombe, M. Sawan and C. Wang, "Variable resolution CMOS current mode active pixel sensor", *IEEE ISCAS*, Geneva, Switzerland, May 2000, pp. II-293.
- [6] Z. Zhou, B.Pain, E.Fossum, "A CMOS imager with On-Chip Variable resolution for Light-Adaptive Imaging" in *Proc. IEEE ISSCC*, San-Francisco, Feb. 1998, pp.174-175.
- [7] G. Yang, C. Sun, C. Wrigley, and B. Pain, "Dynamically reconfigurable vision CMOS image sensor", *Proceedings of First IEEE International Conference on Sensors*, June 2002, Vol. 1, pp. 21-26.
- [8] H.Samet, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, 1990.
- [9] H.Samet, *Applications of Spatial Data Structures – Computer Graphics, Image Processing and GIS*, Addison-Wesley, 1990.
- [10] R. Pajarola, P. Widmayer, "An Image Compression Method for Spatial Search", *IEEE Transactions On Image Processing*, Vol. 9, No. 3, March 2000.
- [11] O. Yadid-Pecht, B. Pain, C. Staller, C. Clark, E. Fossum, "CMOS Active Pixel Sensor Star Tracker with Regional Electronic Shutter", *IEEE J. Solid State Circuits*, Vol. 32, No. 2, pp. 285-288, February 1997.