11th International "Hiroshima" Symposium on the Development and Application of Semiconductor Tracking Detectors(HSTD11)

In conjunction with 2nd Work shop on SOI pixel Detectors (SOIPIX2017) at OIST , Okinawa, Japan

New Readout Strategies of CMOS Pixel Sensors Dedicated for High Energy Physics Experiments

Xiaomin WEI, Yingjie HE, Bo Li, Wei NIU, Ran ZHENG, Jia WANG, Tingcun WEI, Yongcai HU

School of Computer Science and Engineering, Northwestern Polytechnical University, 710072 Xi'an, P.R. China

weixm@nwpu.edu.cn

MAPS Readout with Binary Tree Search

Since the data in particle images are sparse and demonstrate cluster property. A readout strategy utilizing the idea of binary tree search is proposed for reducing the readout times and readout data.

Readout algorithm: The algorithm is proposed for reading the digital pixel, which is "0" or "1". Our aim is to find the position of "1" in the pixel array. The data are parallel readout in column. The data in a column can be expressed as a vector, PIXEL [0:N-1]. We check if there is "1" in a certain sector of the vector by an OR operation of all the data in the checked sector . The checking sector is depends on the binary tree in Fig.2.

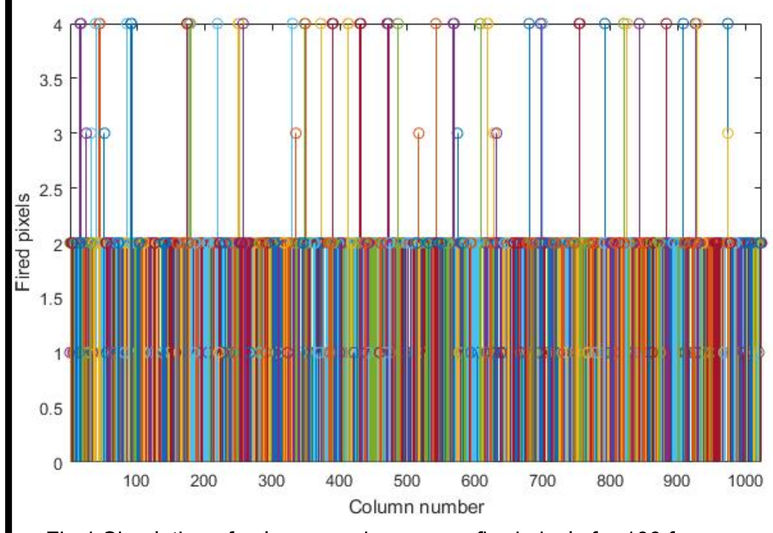

Fig.1 Simulation of column number versus fired pixels for 100 frames, 1024×1024 pixel, cluster size 1~9, average fired pixels cluster 2.2

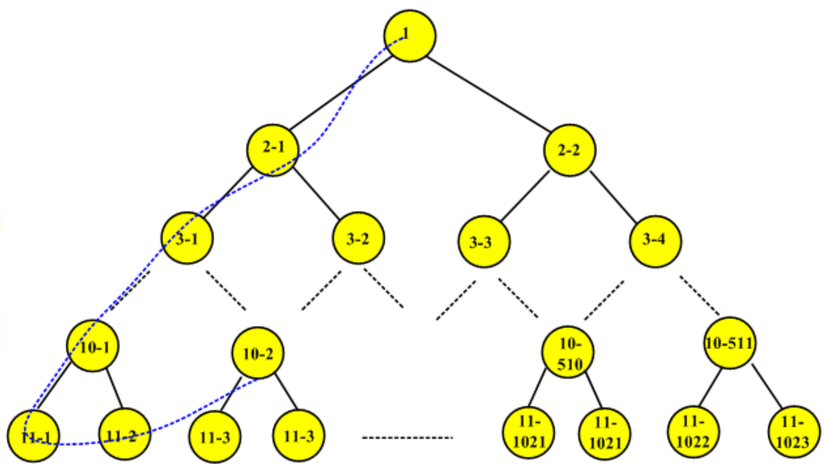

Fig.2 The binary search tree. This is a depth priority search algorithm for a vector length of 1024

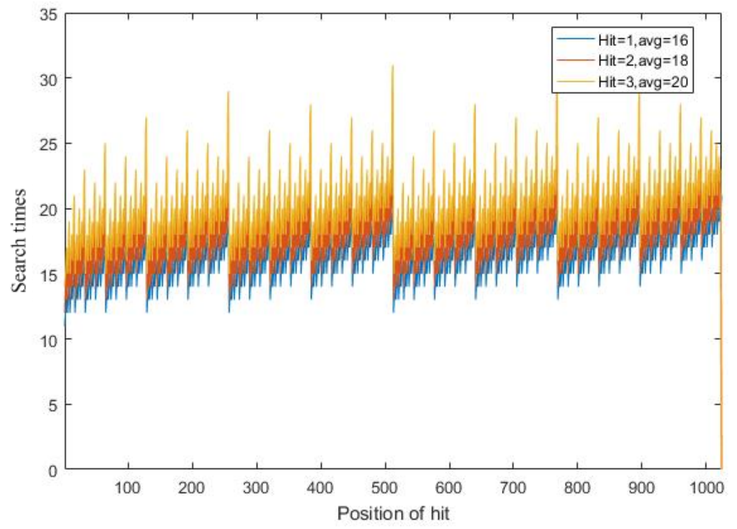

Fig.3 Search times versus position of hit. The average search times are 16, 18, 20 respectively for number of the fired pixel is 1,2,3.

Example of the searching algorithm:

- Checking if there is any "1" in the whole column (PIXEL[0:1023]). If No, stop; if Yes, continue.
- Checking if there is any "1" in the half of the column (PIXEL [0:511] If No, checking PIXEL[512:1023]; if yes, checking PIXEL [0:255].

•Checking in this way until there is no "1".

Simulation results of the readout strategy:

- For the column with more than ten fired pixels, the maximal search time is 80.
- For the column with one hit, the average search times are all less than 20 for 1, 2 and 3 fired pixels.
- For the column with no hit, the search time is 1.

⇒ Since there is no hit in most column, the readout strategy is efficient.

⇒ The data volume may be reduced in some applications if the search path is used to index the hit position.

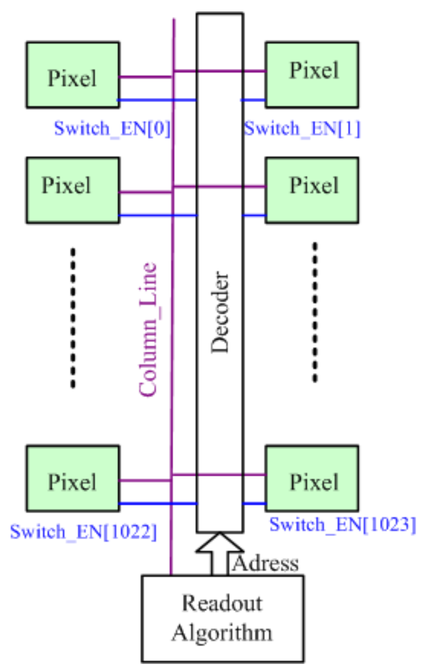

Fig.4 Readout architecture of one column

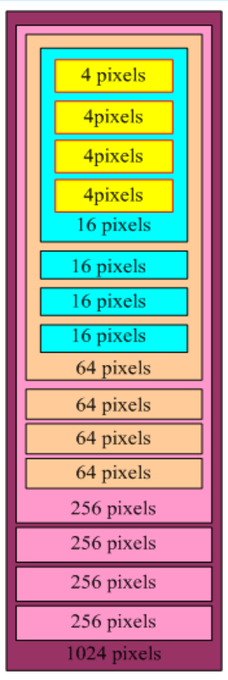

Fig.7 Hierarchical decoder

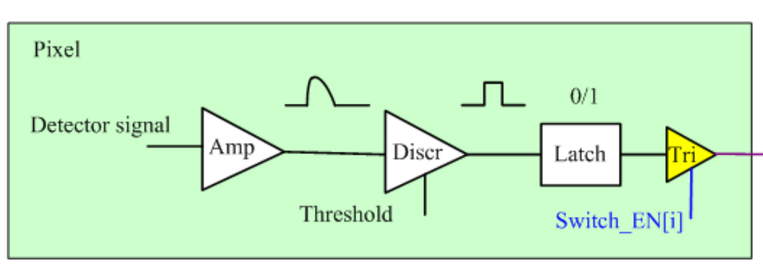

Fig.5 Readout scheme

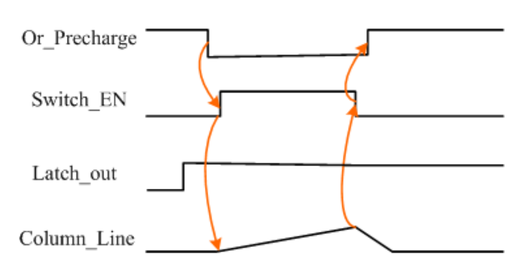

Fig.6 Readout timing

Readout circuits:

- The readout algorithm is realized in the bottom of a column. Combing with a decoder, the pixel output can be switched to the column line or not to the column line. (Fig.4)
- In the pixel, a triple state logic is required as the switch to the column line.
- The search result can be obtained by a logic OR of all the selected pixels. This can be realized in the column level by the triple state logic and a NMOS transistor with proper timing. (Fig.5 and Fig.6)
- The decoder is realized with an hierarchical architecture (Fig.7). The bottom level is for 4 pixels. The switch control signals are 0000,1111,1100,0011,1000, 0100, 0010, and 0001.

Compressive Sensing Readout Applied in CMOS Pixel Sensors

Compressive Sensing for MAPS

Suppose the signal to be measure is x, which is an N ×1 column vector. If x is sparse, or x can be expressed as sparse signal under a basis Ψ, then x can be recovered by sampling times M (M<<N). The process is shown as Fig.1.

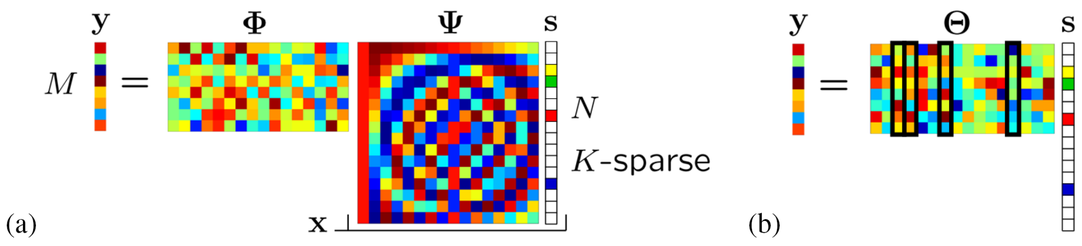

Fig.8 Compressive sensing measurement, y= Φx= Φ Ψs=Θs

The signals in MAPS is spare in both column and row. We take identity matrix as Ψ, and then Φ Ψ=Θ, and x=s. As a result, y= Θx= Φx. The realization flow is shown in Fig. 8. The key problems are to find an proper observation basis and the data reconstructing algorithm. Fig. 11 and Fig. 12 presents the simulation results for different observation basis and different reconstructing algorithm.

Finding an observation basis Φ which is incoherence with Ψ and satisfies Restricted Isometry Property. → Obtaining the measured results y by observing x with Φ → Reconstructing x with y under the constrain of Φ and Ψ

Fig.9 The Realization of compressive sensing in MAPS

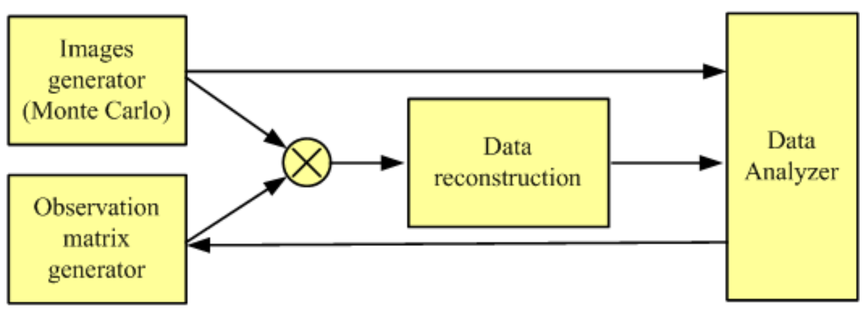

Fig.10 The simulation system of compressive sensing MAPS

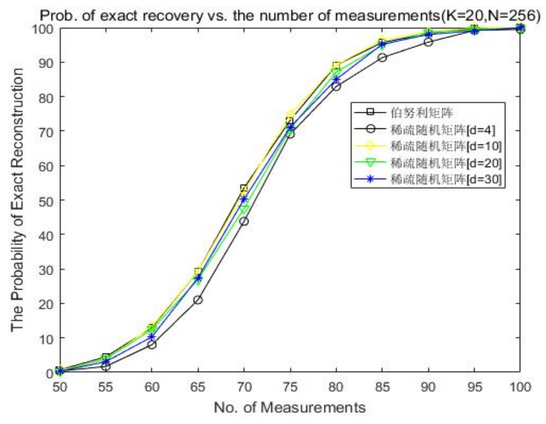

Fig.11 Simulation of observation matrix

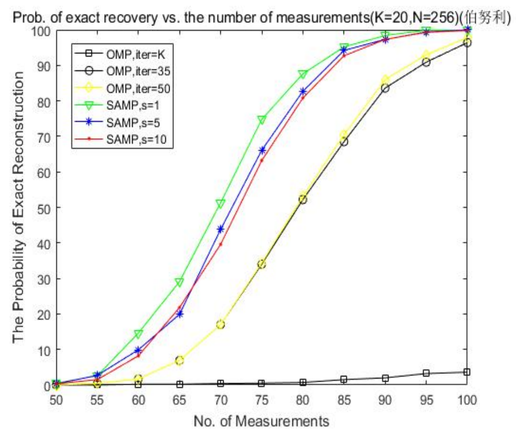

Fig.12 Simulation of reconstruction algorithm

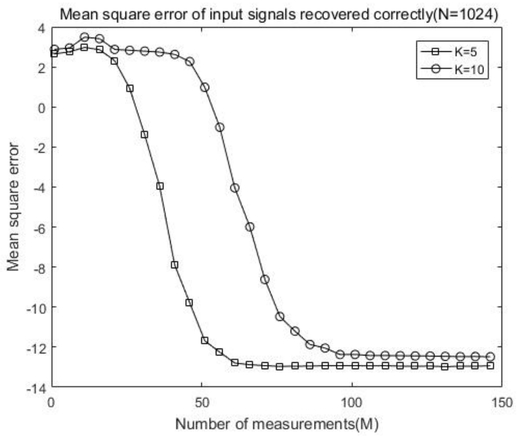

Fig.13 Simulation of required sampling time

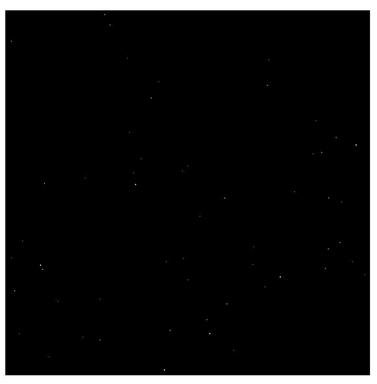

Fig.14 Reconstruction of images with 60 hits in a pixel array of 1024 ×1024

Simulation results

- In these simulation, the column is parallel read out. The image can be well reconstructed with sampling times of 50, 60,70, 80 for 20 hits, 40 hits, 60 hits and 300 hits, respectively. This readout strategy is efficient for the hit density is relative higher.
- The compressive sensing readout in both column and row directions should be studied in advance.
- The design of readout circuits should be considered in the future.