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, Tingyun WEI, Yongcai HU
School of Computer Science and Engineering, Northwestern Polytechnical University, Xi’an, P.R. China
weimu@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 dependent on the binary tree in Fig.2.

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 if there is any “1” in the whole column (PIXEL[0:1023]).

For the column with no hit, the search time is 1.
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 more than ten fired pixels, the maximal search time is 80.

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 columns, 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.

Readout circuits:
- The readout algorithm is realized in the bottom of a column. Combining 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 output. The decoder is realized with an hierarchical architecture (Fig.7). The bottom level is for 4 pixels. The switch can be studied in advance.

Simulation results of the measurement:
- The compressive sensing readout in both column and row directions should be studied in advance.

Compressive Sensing for MAPS

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

The signals in MAPS are sparse in both column and row. We take identity matrix as \( \Phi \), and then \( \Phi \Psi x = s \). As a result, \( y = \Phi x = \Phi \Psi s \). 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.

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.

Compressive Sensing Readout Applied in CMOS Pixel Sensors

The design of readout circuits should be considered in the future.