A Zero Suppression Micro-Circuit for Binary Readout CMOS Pixel Sensors

A. Himmi[1], G. Dozière[1], O. Torheim[1,2], Ch. Hu-Guo[1], M. Winter[1]

1- Institut Pluridisciplinaire Hubert Curien, 23 rue du Loess, 67037 Strasbourg, France
2- Dept. of Physics and Technology, University of Bergen, Norway

Contact: Abdelkader.Himmi@ires.in2p3.fr

Abstract

The EUDET-JRA1 beam telescope and the STAR vertex detector upgrade will be equipped with CMOS pixel sensors (MAPS) allowing to provide high density tracking adapted to intense particle beams. The EUDET sensor Mimosa26, is designed and fabricated in a CMOS-0.35µm Opto process. Its architecture is based on a matrix of 1152 x 576 pixels, 1152 column-level analogue-to-digital conversion by discriminators and a zero suppression circuitry. This poster focused on the data sparsification architecture, allowing a data compression factor ranging from 10 to 1000, depending on the hit density per frame. It will be extended to the final sensor for the STAR upgrade.

OUTLINE

METHOD

RESULTS

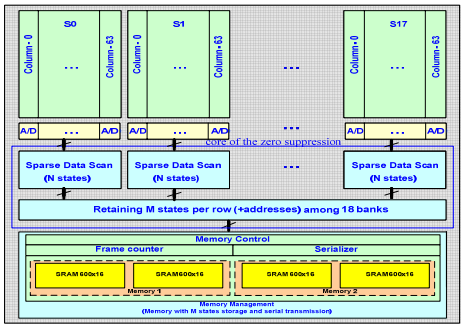
Block diagram of the sensor read-out architecture

Pixel array : 576x1152 pixels
Readout row by row. The row is divided into 18 groups

Analog to digital conversion at the bottom of each column (Discriminator or ADC)

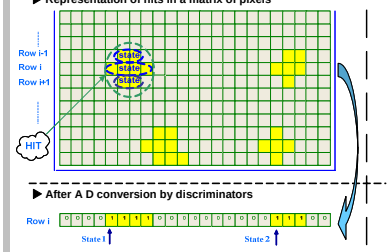
Zero suppression algorithm :

Memory which stores hits informations and serial transmission



View illustrating the encoding of the digital pixels

▶ Representation of hits in a matrix of pixels



▶ After A D conversion by discriminators

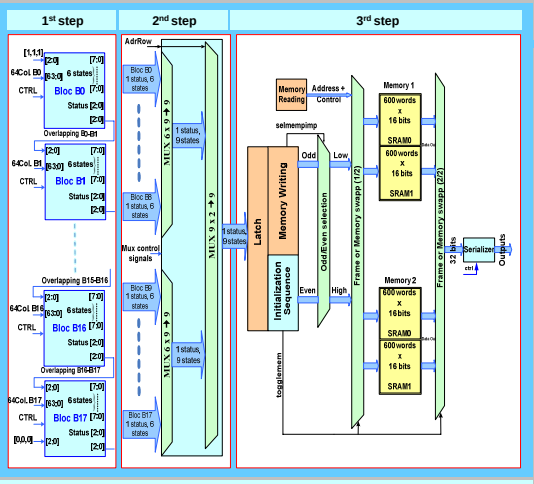
State: Column address of the 1st pixel+ 2 bits code

▶ The state format: column address of the first hit pixel, followed by 2 bits encoding the number of contiguous pixels in the group delivering a signal above threshold.

▶ A maximum number of 4 contiguous pixels is accounted for.

▶M states by row can be processed. This limit was derived from a statistical study based on the highest occupancy expected in the pixel array.

Readout Chain: Hit finding algorithm



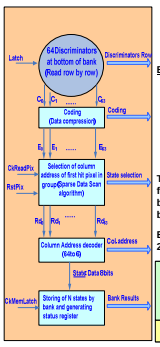
1st step: Sparse data scan

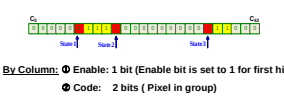
▶ Zero suppression is based on row by row sparse data scan readout and organized in pipeline mode in three steps

▶ 1152 column terminations are distributed over 18 banks, each bank connected to 64 column level discriminators outputs

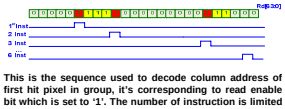
▶ Based on a sparse data scan algorithm to find hit pixels (discriminator output = "1")→ Up to 4 contiguous pixel signal above threshold will be encoded in a 2 bits state word following by address of the 1st pixel

▶ Find up to N (=6) states with column addresses per bank



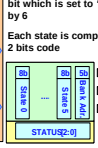


By Column: Φ Enable: 1 bit (Enable bit is set to 1 for first hit)
Φ Code: 2 bits (Pixel in group)

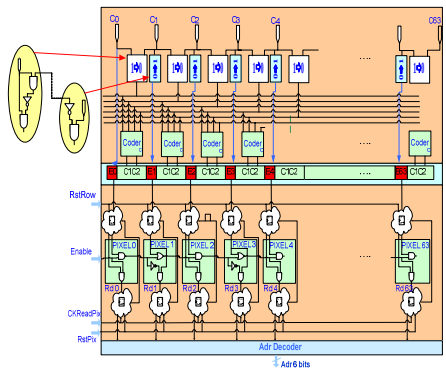


This is the sequence used to decode column address of first hit pixel in group, it's corresponding to read enable bit which is set to '1'. The number of instruction is limited by 6

Each state is composed of 6 bits Column address and 2 bits code



▶ Maximum 6 States registers of 8 bits
▶ Bank address register of 5 bits
▶ Status register of 3 bits
(number of states by bank)



2nd step: States Multiplexer

▶ States Multiplexer reads out the outcomes of the 1st step in 18 banks and keep up to M (=9) states

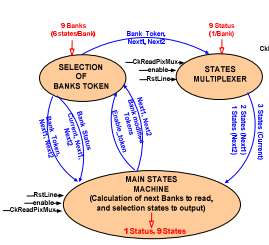
▶ This block is constituted of 3 sub-blocks:

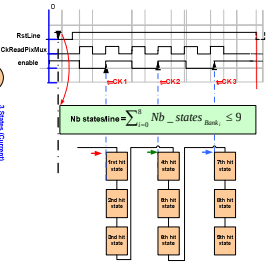
☒ 2 identicals modules (MUX 6 x 9 → 9) which extract each one 9 states and 1 status for half row

☒ 1 module (MUX 2 x 9 → 9) which retains 9 states and a states from these 2 modules

▶ Mux6x9To9 algorithm read 9 hits states at maximum in 3 steps

▶ At each step, 3 hits states can be latched at maximum





$$Nb\ states\ line = \sum_{i=0}^{8} Nb\_states_{Bank_i} \leq 9$$

▶ The row states format includes:

- Row address
- Status register
- States column addresses
- 1 overflow bit

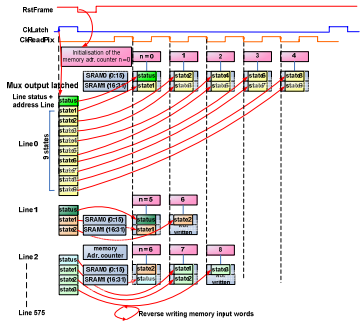
3rd step: Memory management

▶ Store the outcomes of the 2nd step to a memory

▶ Memory is composed of 2 IP's buffers (4 SRAM: 600 x 16 bits each) to ensure the continuous read-out

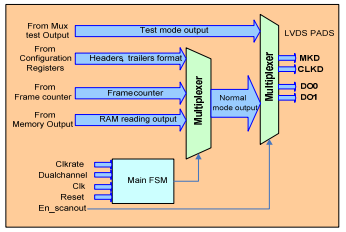
▶ During the current frame, the writing mode uses 2 SRAM's and the reading mode works with 2 others SRAM's

▶ During the next frame, the 2 modes (reading/writing) are swapped and this process is repeated at each frame



▶ Optimization of memory space

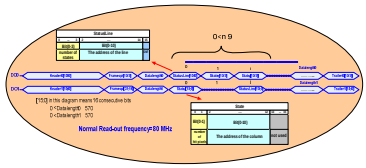
▶ Serial transmission by LVDS data output at 80 MHz



Data generated by Mimosa26 :

▶ Header	→ 16 bits / output
▶ Frame counter	→ 16 bits / output
▶ Data length	→ 16 bits / output
▶ Data	→ Max = 570 x 16 bits / output
▶ Trailer	→ 16 bits / output

▶ Total stream size per output : 9216 bits = 576 W16 = 1152 W8



▶ For each line with hit : one Status/line followed by up to 9 States

▶ The following data stream is generated

▶ Status/Line word

- Address of line
- Number of States (9 Max, overflow flag if > 9)

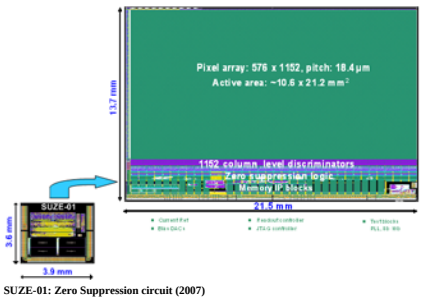
▶ States list – One state = consecutive pixels at 1 in the line

- Column address of the first pixel at 1
- Number of pixels at 1

Digital Output Sensor

MIMOSA26 Layout: 1st Sensor with Integrated Zero Suppression

AMS C35B4: 0.35 µm OPTO techonology (15 µm EPI), submitted on Dec.2008



SUZE-01: Zero Suppression circuit (2007)

▶ Purpose: reduce the raw data flow of MAPS

Test Sensor with Integrated Zero Suppression

▶JTAG Configuration & testability, real time reading with LVDS bus.

All parameters are set via JTAG interface using graphic user interface

▶The ASIC includes an embedded structure of test:

It generates internally a matrix constituted of 278x2 lines pattern (input data)

The architecture can be tested entirely or by blocks

▶The platform NXI National Instruments reads the output data stream at 160 Mbits/s.

▶Results: All the features of this architecture were tested successfully :

Encoding of the hit: location and geometry.

Limits of the data compression system.

3 patterns tested 7 millions times without errors

Robustness test: 199 frames x 10 000 random patterns test at 80 MHz without errors.