Innovative digital architecture for low power-consumption MAPS

Piero Giubilato[1,2], **Davide Chiappara**[1,2], Devis Pantano[1,2], Manuel Da Rocha Rolo[3], Sara Garbolino[3], Andrea Paternò[3]

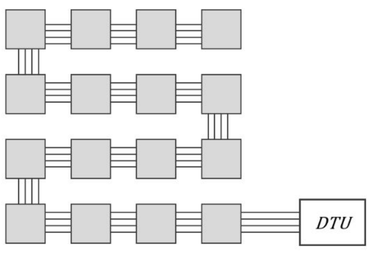
[1] University of Padua - department of physics and astronomy "G. Galilei" - via F. Marzolo, 8, 35131 - Padua, Italy
[2] Italian National Institute of Nuclear Physics (INFN) - Section of Padua - via F. Marzolo, 8, 35131 - Padua, Italy
[3] Italian National Institute of Nuclear Physics (INFN) - Section of Turin - via P.Giuria, 1, 10125 - Turin, Italy

HASH architecture idea

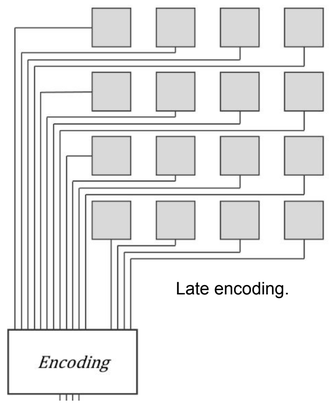
In MAPS, depending on the operating conditions, a significant part of the power consumption may be used to communicate the hit pixel address to the periphery. A common solution employs a **bus shared** among the pixels belonging to the same physical column, where each pixel writes its address, in a **radix-of-two** basis. This approach, and its variations, are an example of **early encoding**, where the pixel transmits its complete address. This approach is usually **not directly scalable**, as the number of communication lines and their topology depend on the matrix size.

A theoretical alternative would be to **directly connect each pixel** to periphery through a one-hot line. This way, the periphery must reconstruct the pixel position by looking at the **physical position** of the wire brining the hit information. Such an ideal arrangement would allow for **simpler pixels**, and **easier addressing** (for reset and configuration). It can be thought as address writing in **radix-of-N**, instead the standard radix-of-two. We refer to this approach as **late encoding**. Clearly, **this approach is impractical**, as modern MAPS pixel density makes it impossible to route all the metal lines such an architecture would require.

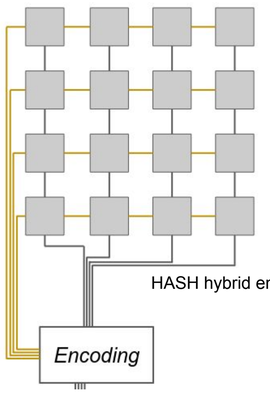
An hybrid approach may use something in between, exploiting **projections**: if there are some **lines shared between pixels**, the address can be communicated through **single bits** and be reconstructed in the periphery. This algorithm is **lossy**; depending on the number of projections, reconstructions might be wrong. By **adding more projections**, the number of wrong cases drops.



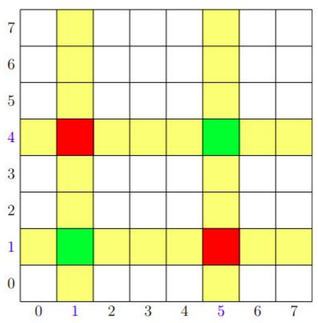
Early encoding.



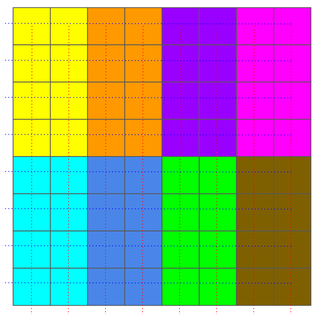
Late encoding.



HASH hybrid encoding.



If 2 projections are used wrong reconstructions are relatively likely (here, 2 hits in green are reconstructed together with two mistakes in red using only the 2 projections along x and y axis).



By adding a third projection (here, represented by the colour), many collisions are solved.

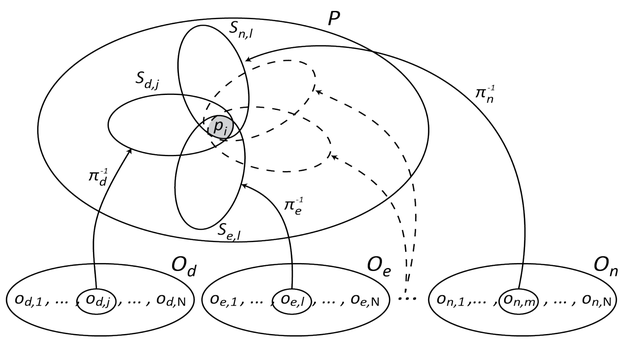
Mathematical justification

A thorough preliminary study has been done to find the **best projections** to use for this purpose.

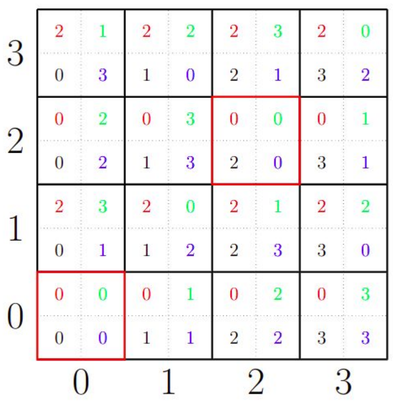
A formalism has been created, electing the projections that **maximize the reconstruction capability** by looking at the couterimage of the projected bits.

Pairs of projections are ranked as:

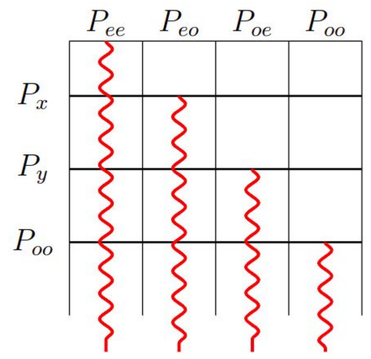
- **Optimal** when a single hit is always reconstructed correctly by using only these two projections;
- **Suboptimal** when it is possible to make a mistake trying to reconstruct a single hit;
- **Trivial** when the two projections provide the very same information.



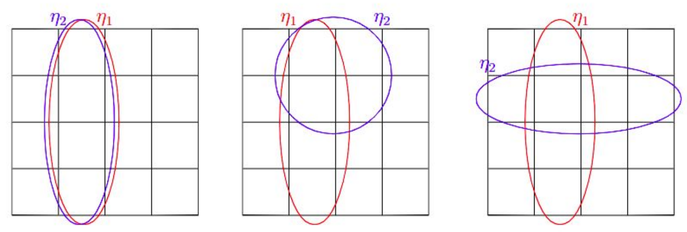
Mathematical representation of optimal projections using counterimage of projected bits.



Several projection representation in a 4x4 matrix. The numbers show the projected value, while colors represent different projections. Looking at red addresses we see that green and blue projection are not optimal.



When the matrix is even, only three optimal projections can be chosen, selecting from projections on row (P_y), columns (P_x) and diagonal with even and odd parameters (P_{eo} etc.).



A figure showing, respectively, trivial, suboptimal and optimal pairs of projections.

The Goal

- Reduce power consumption
- Maintain good rate capability
- Have a scalable architecture for big matrices

The boundary conditions

- Sparsification hypothesis (low hit population)
- Rate not too high

The method

- A lossy algorithm to transmit address data
- A periphery capable to avoid crashes when out of its application regime

HASH Architecture

The Solution

- Scalable architecture - can easily be built arbitrarily large
- Rate capability demonstrated by using pessimistic time constraints
- Simple pixels - low power consumed
- Its implementation can change depending on physical constraints (add projections)
- Theorized in different flavours:
  - In-pixel memory
  - Clustering algorithm
  - Loss prone but faster

The future

The architecture has been thoroughly studied by using Monte Carlo simulations. The rate performances are good for many scientific applications (space physics, also some high energy physics). Power consumption estimation is promising. We foresee to complete place and route and a tentative silicon run.

Device implementation

A **fixed number of projection equal to 4** was chosen, and the whole **digital architecture was developed, simulated and verified** in system-verilog.

A "smart" pixel layout has been developed, so that **the pixel layout (gates and metals) is in fact identical for each pixel in the matrix**, making it easy to generate large area sensors. The only elements changing from pixel to pixel is in fact a fixed layout set of vias, which are programmatically enabled (via is actually present) or disabled (via is omitted) to determine the pixel function in the matrix.

For the place and route phase, an **automatic routing algorithm** has been integrated with the standard placement tools (Synopsis): for each pixel the algorithms determines the correct vias setting accordingly to its position in the matrix,so to automatically implement the hash-code generating architecture across the whole matrix.

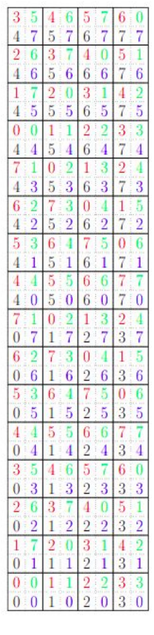
The periphery was **completely developed, simulated, and tested** for various possible faults. In the current implementation, all addresses are tested, and wrong addresses give an empty signal accordingly.

Many further flavours (e.g. without address testing, with self-reset, or with **possible fake hits** are envisioned and can be created by changing only slightly the current implementation.

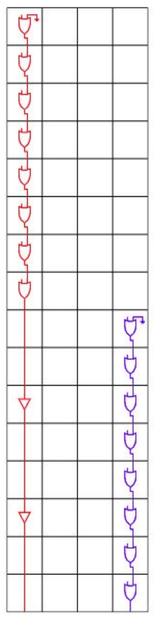
The automata of the periphery readout and reset machine in the current implementation. This flavour is hyper safe and reliable, prevents any digital collision, but could be further optimized in terms of speed.

Matrix	Pixel pitch	Aspect ratio	Hash line speed	Data line speed
A	50 µm	512x512	26 ns/cm	7 ns/cm
B	50 µm	128x2048	10 ns/cm	7 ns/cm
C	10 µm	128x2048	26 ns/cm	7 ns/cm

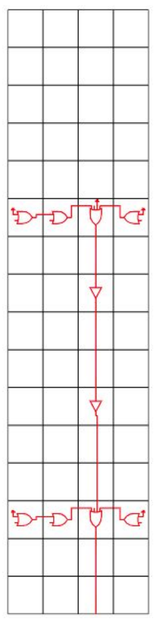
Three matrices, with different aspect ratio and pixel size, are foreseen. Line speed deduced from technological libraries (110 nm)



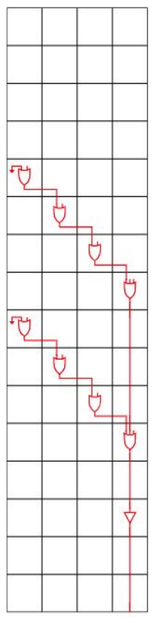
Hash projections in current implementation



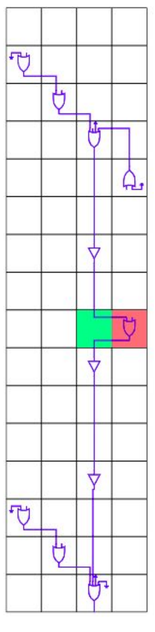
Guide for connection of P_x projection



Guide for connection of P_y projection



Guide for connection of a diagonal projection



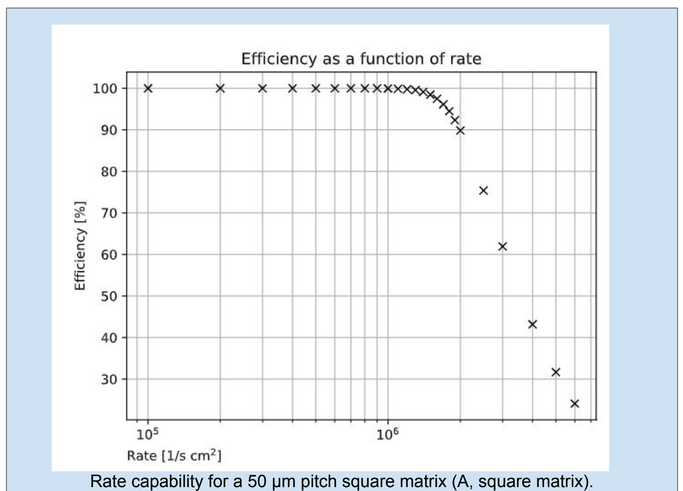
In most diagonal bits, a peculiar operation, named hook, is needed

Performances

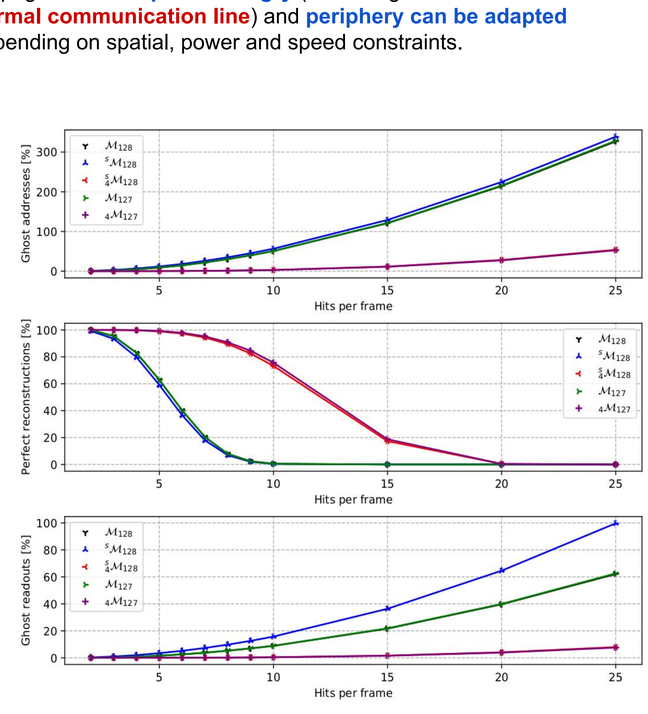
Several **Monte Carlo simulations** were run to validate this architecture:

- In an idealized condition, the reconstruction capability, changing the number and the type of projections;
- In a real physical condition, using RTL simulation environment, the digital logic was validated and issues were sought;
- In a replicated condition, using ad-hoc developed C++ software, high statistics was simulated to compute the efficiency of the device in various conditions and various flavours.

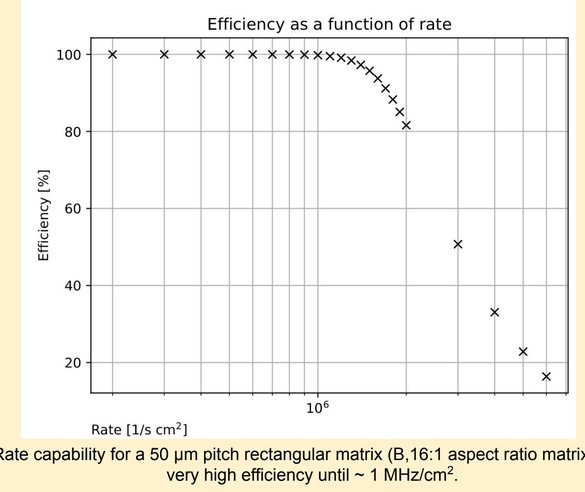
From a **power point of view**, pixels are **extremely simple**, hash propagation is **not power hungry** (a hashing line consumes ~ **twice a normal communication line**) and **periphery can be adapted** depending on spatial, power and speed constraints.



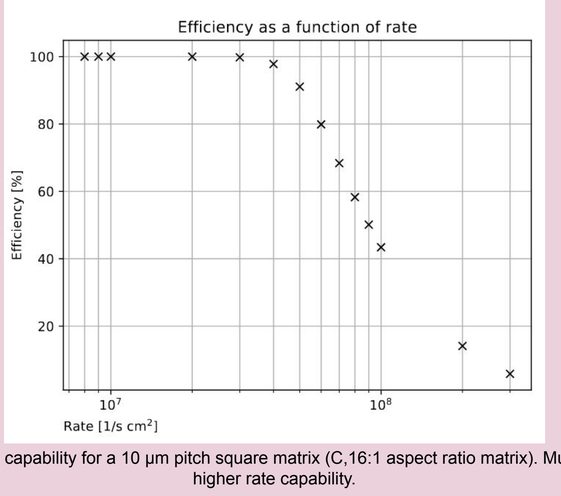
Rate capability for a 50 µm pitch square matrix (A, square matrix). Very high efficiency until ~ 1 MHz/cm^2.



Mathematically, the performance of a set of projections can be computed by looking at the percentage of wrong addresses (first plot), the percentage of perfect reconstruction (second plot) and the percentage of ghost readout when address are evacuated (third plot). Here five different matrices, with slightly different sizes, three or four projections and suboptimal sets of projections.



Rate capability for a 50 µm pitch rectangular matrix (B,16:1 aspect ratio matrix). Still very high efficiency until ~ 1 MHz/cm^2.



Rate capability for a 10 µm pitch square matrix (C,16:1 aspect ratio matrix). Much higher rate capability.