AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY

Algorithm for an asynchronous approximation of a center of gravity for charge sharing compensation in pixel detectors' readout circuits

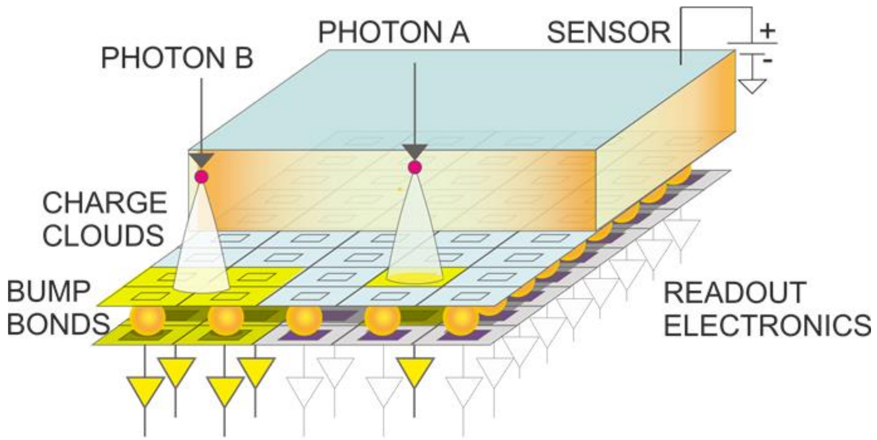
Piotr Otfinowski[1)], Grzegorz Deptuch[1,2)], Piotr Maj[1)]

[1)]AGH University of Science and Technology, Krakow, Poland
[2)]Fermi National Accelerator Laboratory, Batavia, IL, USA

TWEPP 2018 Topical Workshop on Electronics for Particle Physics, 17-21 September 2018, Antwerp, Belgium

Outline

Charge sharing

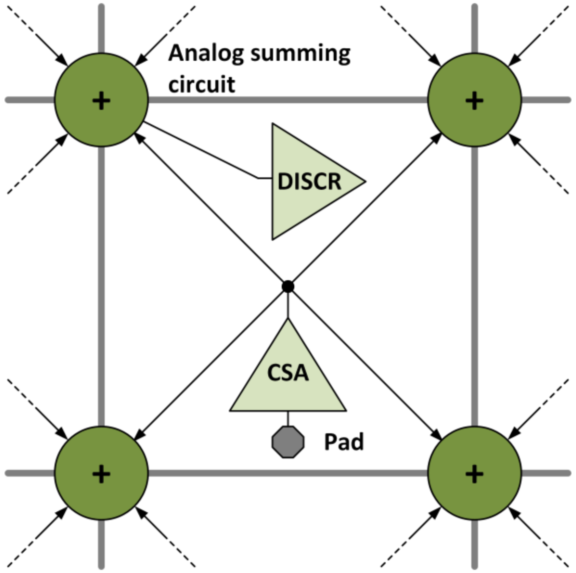


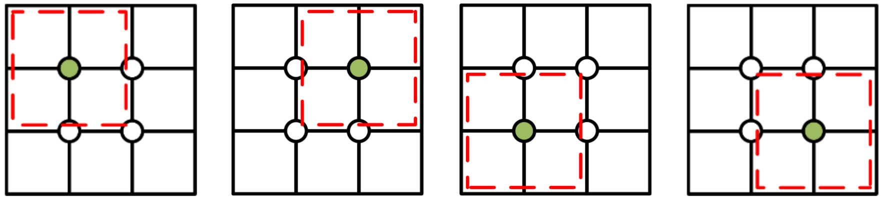
- Charge cloud generated by a photon is subjected to the diffusion process.
- It spatial density can be modelled by a Gaussian distribution with sigma:

$$\sigma = d\sqrt{\frac{2kT}{eV_B}}$$

where d - detector thickness, k – Boltzmann constant, T absolute temperature, e – elementary charge and V_B - detector bias voltage.

Charge sharing compensation – Fractional charge summing





- Signals from four adjacent pixels are summed at analog summing circuits to recover full charge.
- As charge collection regions are overlapping, if the charge cloud is smaller than a single pixel, it is always completely collected by at least one summing circuit.

Trade-offs:

- Summing signals from 4 inputs – 2 x electronic noise
- 4-time larger charge collection area – 4 x counts per pixel
- Necessity for an additional allocation algorithm

Charge sharing compensation – Existing solutions

ASIC	Institute	Technology	Year
Medipix 3RX	CERN	130 nm	2013
X-Counter	XCounter	180 nm	2013
PIX45	AGH	40 nm	2013
MiniVIPIC	AGH/Fermilab	130 nm	2014
Pixie III	PIXIRAD	160 nm	2015
Chase Jr	AGH	40 nm	2017

Algorithm idea

- A pixel with a maximum charge collected typically lies in the center of a group of pixels, that collected fractional charges from a charge shared event.
- If only a delineated group of pixels that received fractional signal can be identified (an object), a pixel that should register an event can be found by an iterative minimization procedure.
- The functionality can be entirely implemented in the digital domain

Algorithm for asynchronous approximation of a

Center **O**f **G**ravity **I**n a **T**emporal **O**bject – COGITO

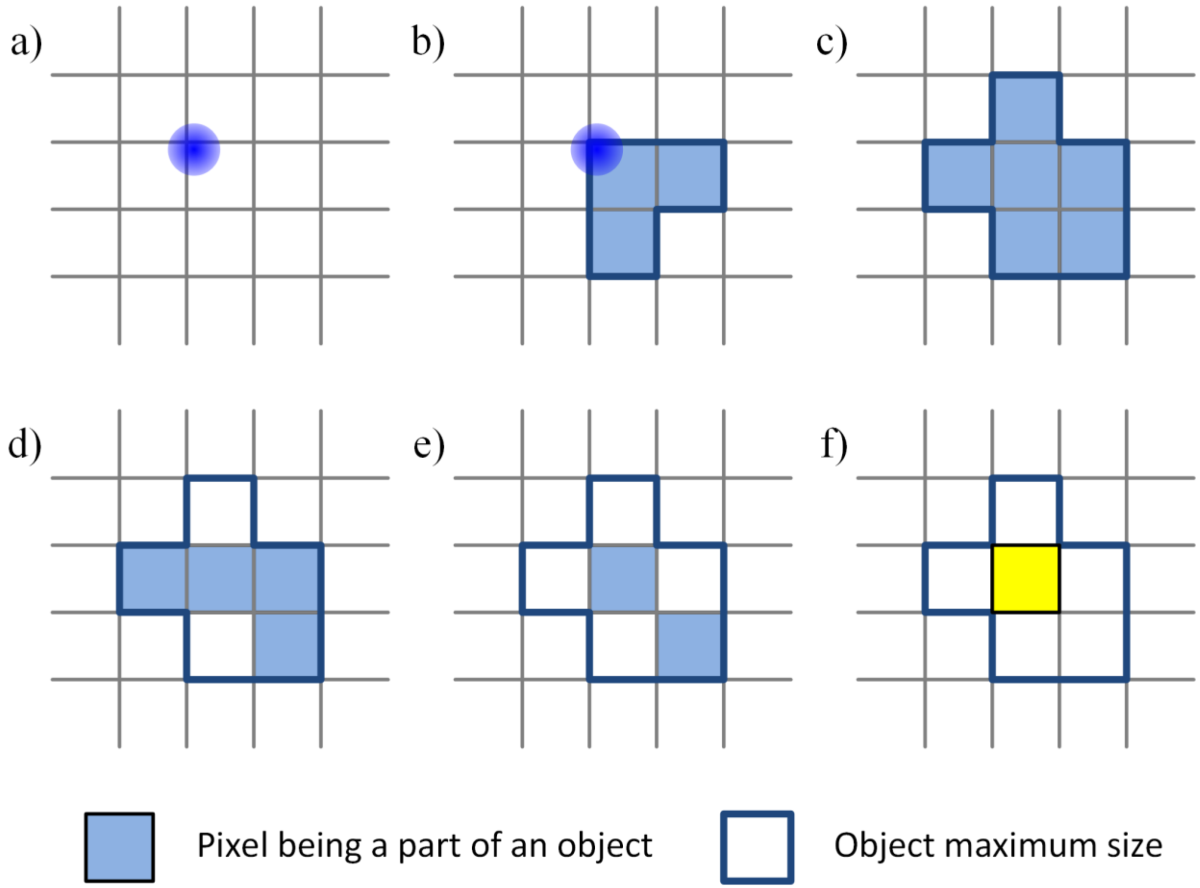
Algorithm operation principle

1. Maximization phase: pixels collecting a charge cloud are identified and logically organized into one bordered group, hereinafter called an ‘object’.

2. Minimization phase: the object formed in the preceding phase undergoes iterative minimization, i.e. shrinks towards its center.

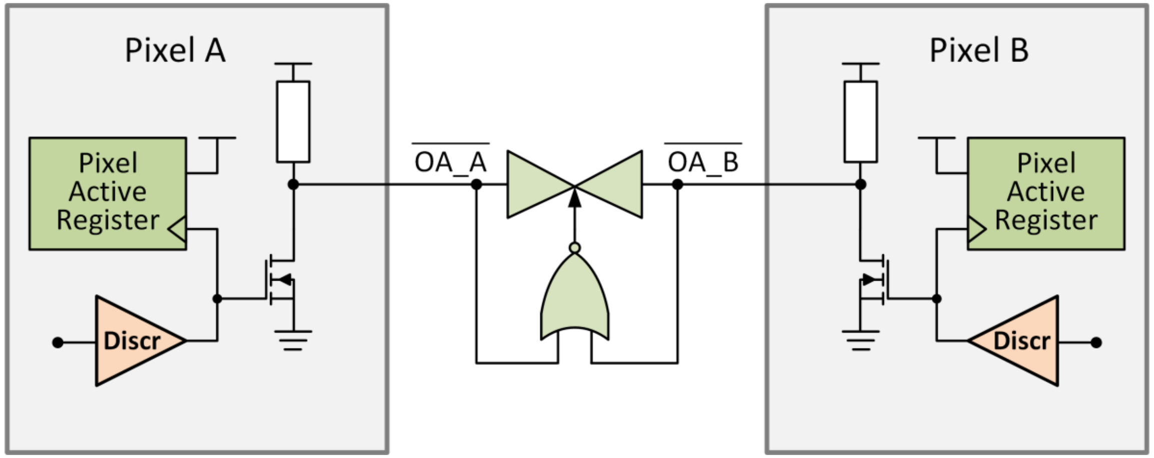
3. Resolution phase: a single pixel estimating the center of gravity of the object is selected.

Algorithm operation – example



a) occurrence of an event, b) start of the maximization phase, c) the final object, d) minimization phase, e) end of the minimization phase, f) resolution phase.

Implementation details – maximization phase



The purpose of the maximization phase is building of an object:

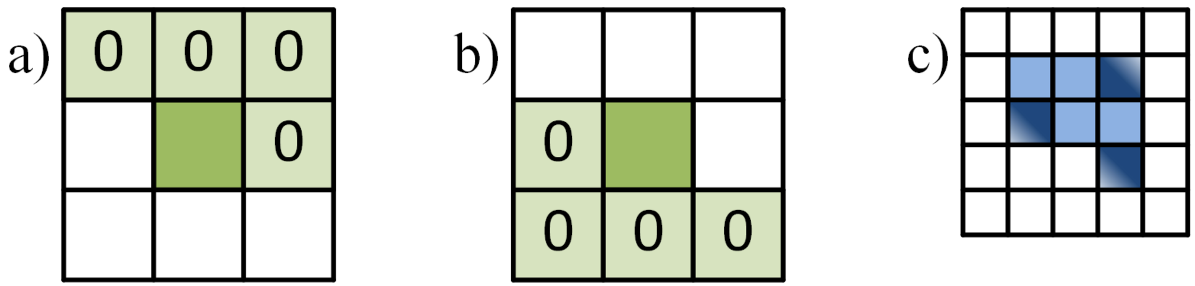
- When signal in a pixel crosses a threshold, its OA signal is pulled low.
- If two adjacent pixels have OA signal active at the same time, the signals are connected by a transmission gate.
  Both pixels form an object.
- As long as at least one pixel has its discriminator over a threshold, OA signal for all pixels belonging to an object is pulled low.
- When discriminators of all pixels in an object return below a threshold, the OA signal gets pulled-up by the resistors in the entire object at the same time, which also ends this phase.

Implementation details – minimization phase

During the minimization phase, an object shrinks. Pixels leave an object in a specified order, by resetting their Pixel Active (PA) registers.

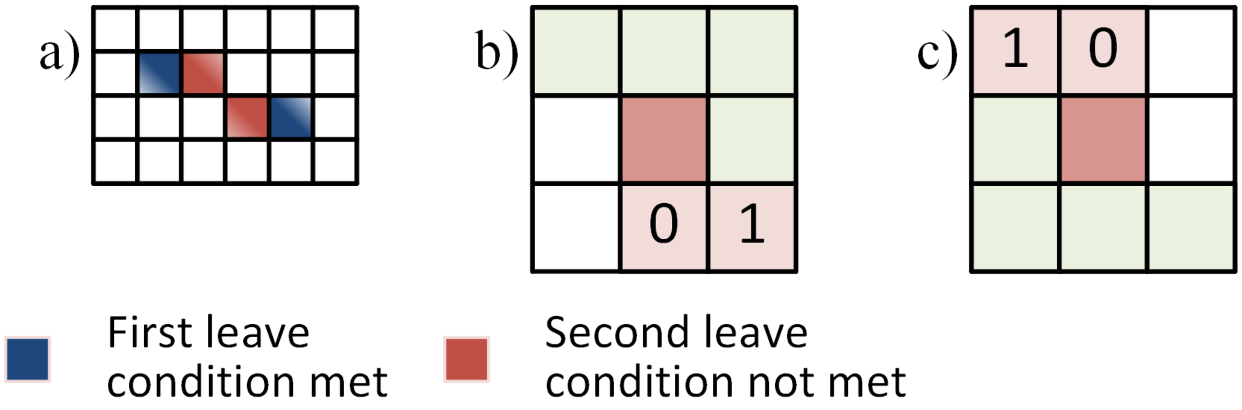
Each pixel asynchronously checks its location, based on the PA signals from its eight neighbors. If its position matches a certain pattern, a pixel is chosen to leave an object. This operation is similar to applying bitmasks to a binary image.

The first leave condition results in directional thinning of an object.
It is defined by bitmasks:



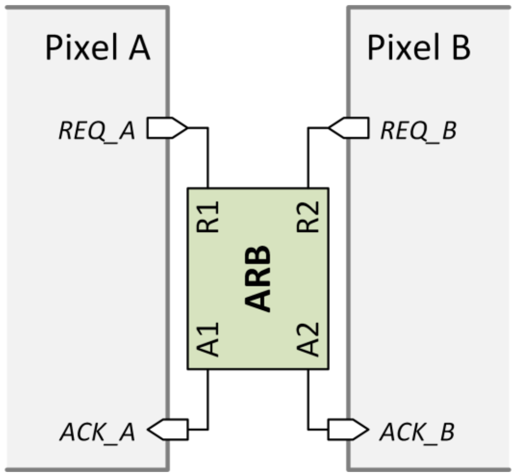
Implementation details – minimization phase

Applying the first leave condition alone can result in splitting of an object into two parts in some cases. This leads to the introduction of the second leave condition, which complements the first one:



Implementation details – minimization phase

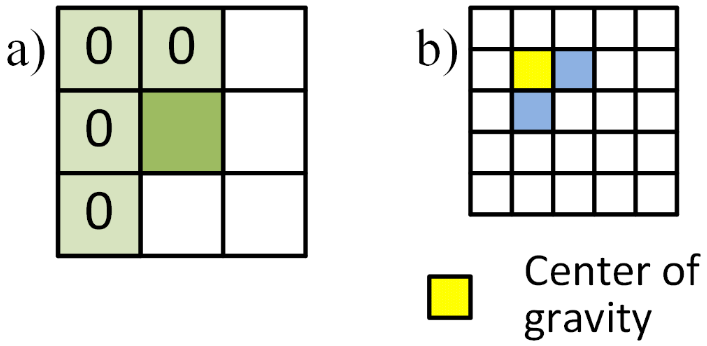
A pixel meeting the leave conditions generates the Leave Request signal, but does not leave an object instantaneously. Before it can do so, it needs to check if any of its neighbors is not selected to leave too. To deal with such a situation, asynchronous arbiters are present between each pair of neighboring pixels to determine the precedence.



The complete procedure for a pixel to leave an object is based on a 4-phase asynchronous handshake protocol.

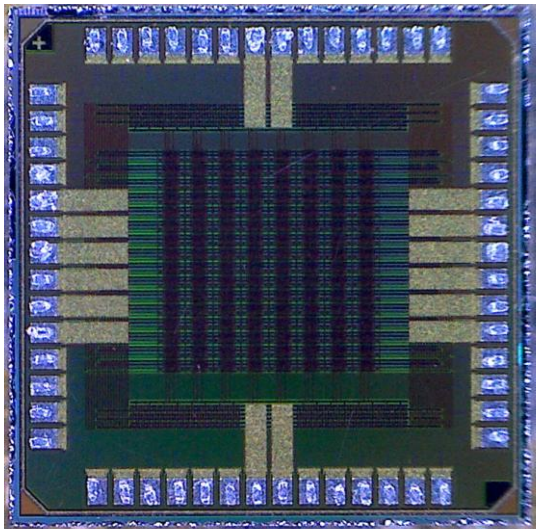
Implementation details – resolution phase

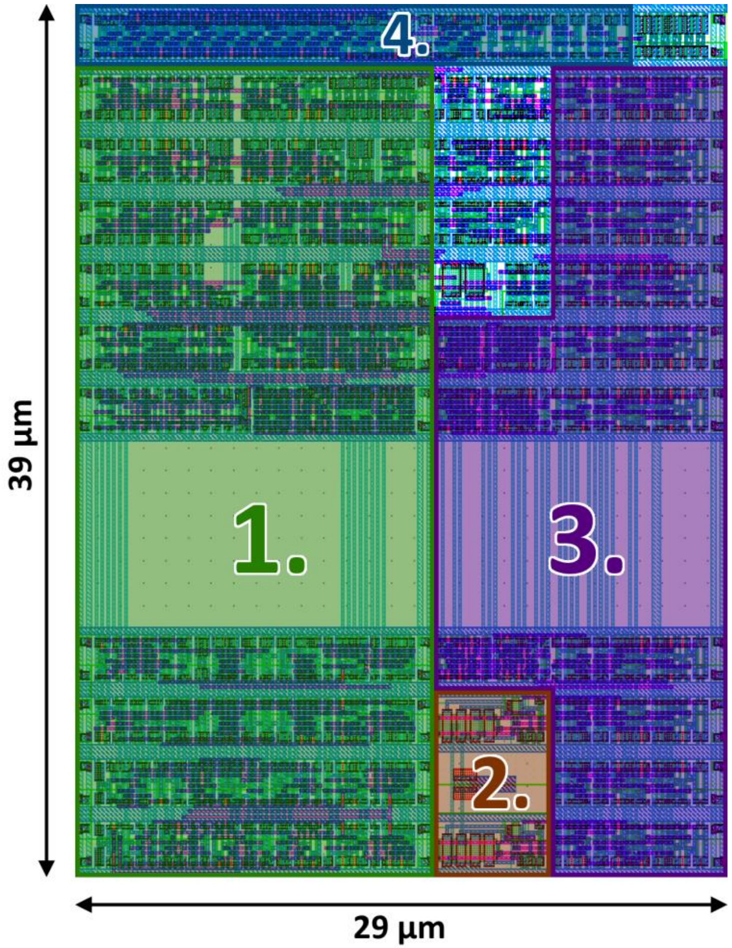
- When the object size is equal to at most 2 × 2 pixels, the minimization phase is over.
- A pixel approximating the center of gravity is chosen as the one in the upper-left corner of the remaining object.



Prototype IC

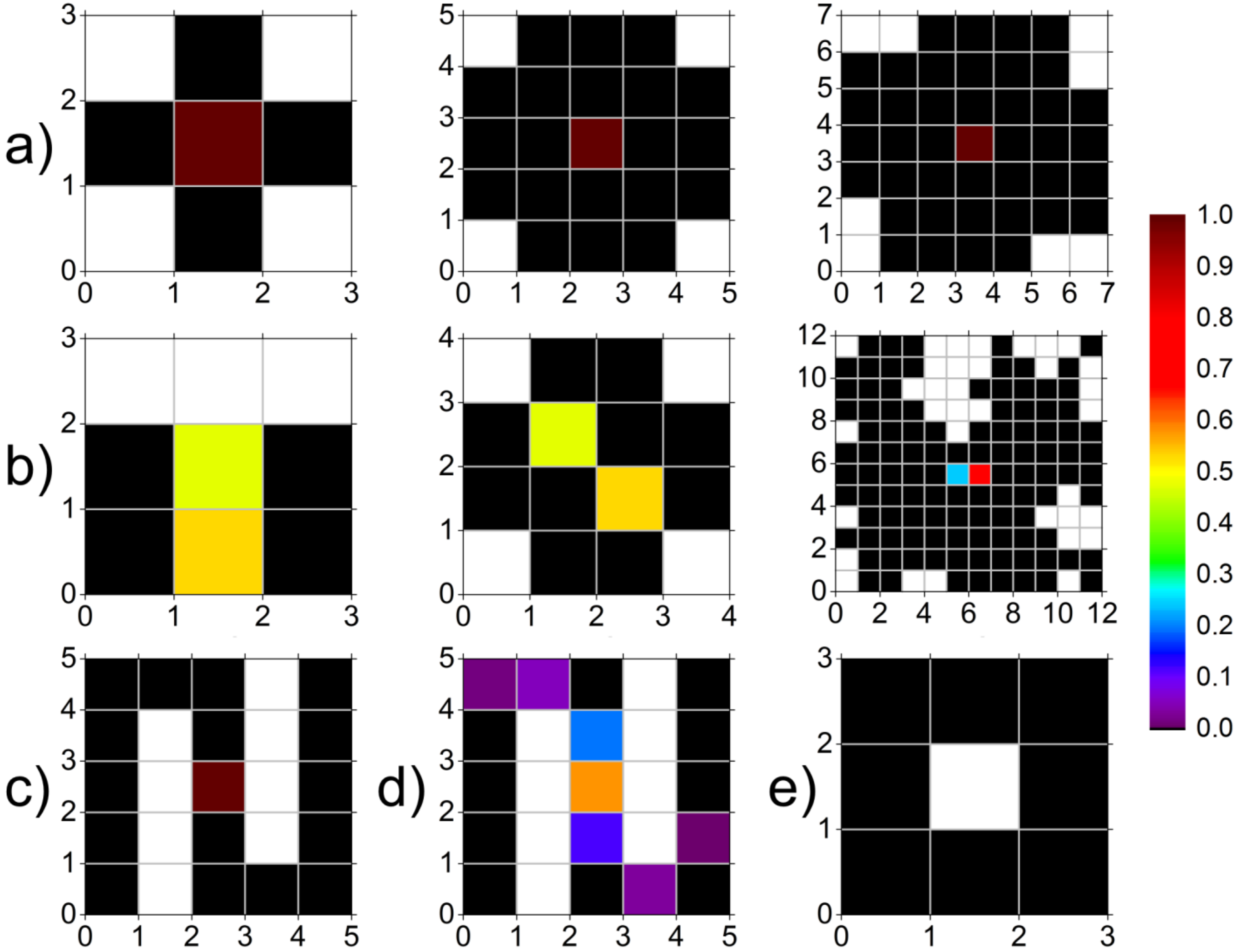
- Prototype integrated circuit has been designed and manufactured in a 55 nm CMOS process.
- The chip has a size of 1.57 × 1.57 mm^2 and it includes an array of 16 × 16 pixels with 50 µm pixel pitch.
- The circuit contains only the digital part, responsible for the algorithm operation. A digital multiplexer, connected to fifteen test signals, imitates discriminators' outputs of the preceding analog front-end.





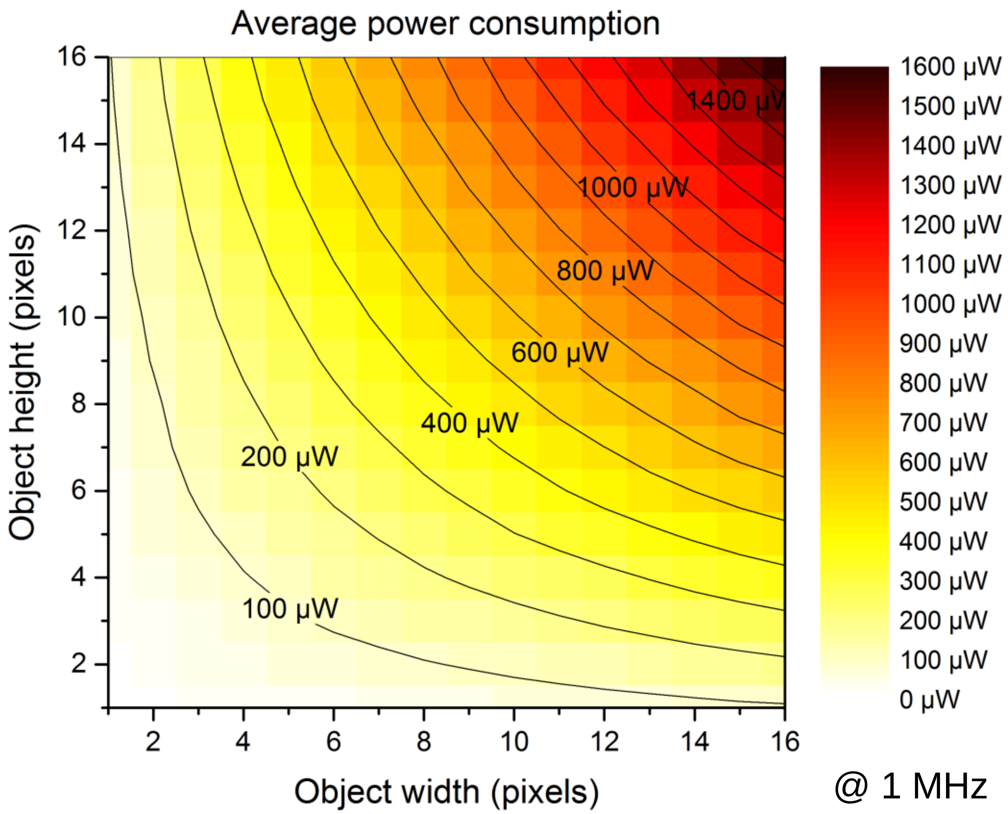
1. COGITO algorithm logic,
2. pullup circuits,
3. configuration registers,
4. input signal multiplexer.

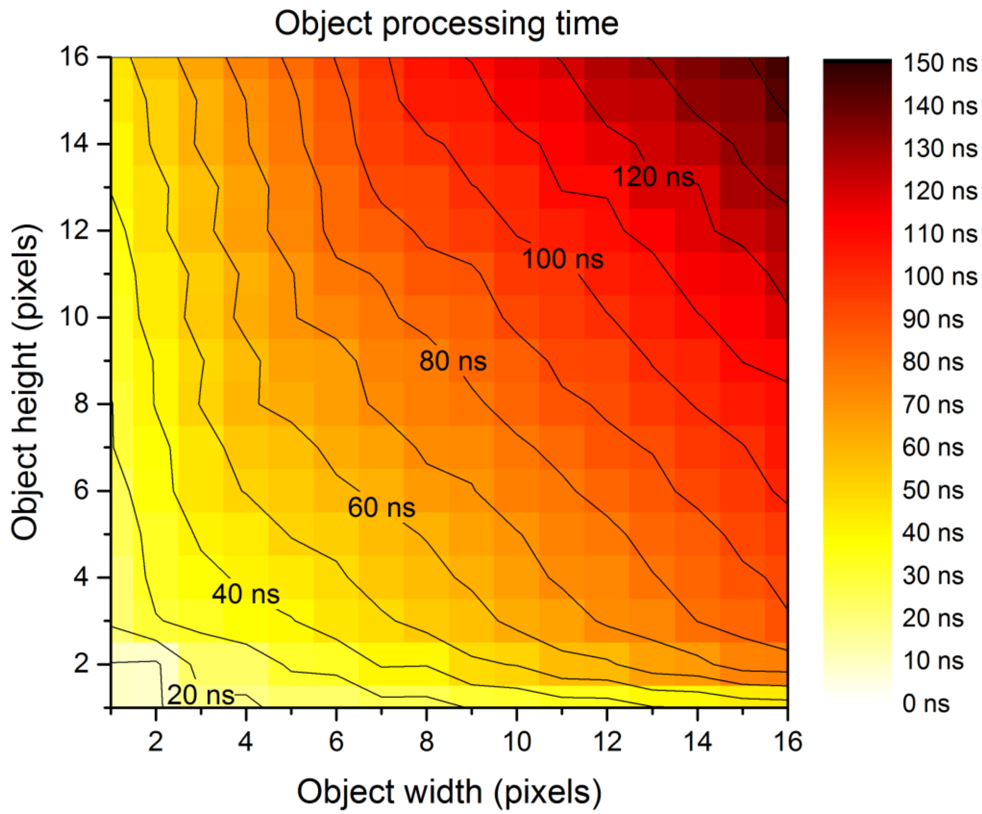
Measurement results



Measurement results

Average power consumption (of an entire array) and processing times for objects of rectangular shapes, with widths and heights ranging from 1 to 16 pixels.





COGITO Algorithm - summary

- Versatile - accepts binary information about simultaneous occurrence of an event, occupying certain area of an array and compacts this information to a single pixel located as close as possible to its center of gravity.
  Analog front-end and following digital back-end blocks can be application-specific.

- Fully asynchronous operation – based only on discriminator outputs. No additional strobe or clock signals required.

- Can be implemented directly in the sensor readout circuit, allowing for very fast image processing.

AGH

References

1. P. Otfinowski et al., “Asynchronous Approximation of a Center of Gravity for Pixel Detectors' Readout Circuits,” IEEE Journal of Solid State Circuits, vol. 53, iss. 5, 2018

2. R. Ballabriga, et al., "The Medipix3RX: a high resolution, zero dead-time pixel detector readout chip allowing spectroscopic imaging", 2013 JINST, 8, C022016, pp 1-15.

3. R. Bellazzini et al., “PIXIE III: a very large area photon-counting CMOS pixel ASIC for sharp X-ray spectral imaging,” JINST 10 C01032, Jan. 2015.

4. A. Krzyzanowska, et al., “Characterization of the Photon Counting CHASE Jr., Chip Built in a 40-nm CMOS Process with a Charge Sharing Correction Algorithm Using a Collimated X-Ray Beam,” IEEE Trans. Nucl. Sci., vol. 64, no. 9, 2017

5. G. Deptuch et al., “An Algorithm of an X-ray Hit Allocation to a Single Pixel in a Cluster and Its Test-Circuit Implementation,” IEEE Trans. Circuits Syst. I, vol. 65, no. 1, 2018.

6. P. Maj et al., “Measurements of matching and noise performance of a prototype readout chip in 40 nm CMOS process for hybrid pixel detectors,” IEEE Trans. Nucl. Sci., vol. 62, no. 1, 2015.