Design of on-chip data sparsification for a mixed-mode MAPS device

Alessandro Gabrielli on behalf of the SLIM5 Collaboration
INFN-Bologna & Physics Department University of Bologna
Viale Berti Pichat 6/2 40127 Bologna Italy
Tel: +39-051-2095052 Fax: +39-051-2095297
Email: alessandro.gabrielli@bo.infn.it URL: http://www.bo.infn.it/~gabriell

Abstract

The device described in the paper is built up of a bidimensional matrix of MAPS, already designed and fabricated in the past by the SLIM5 Collaboration, and of an off-pixel digital readout sparsification. The readout logic is based on std-cells and implements an optimised token-like technique. It is aimed at overcoming the readout speed limit of future large-matrix pixel detectors for particle tracking, by matching the requirements of future HEP experiments. In particular, the readout architecture extends the flexibility of the MAPS devices to be also used in first level triggers on tracks in vertex detectors.
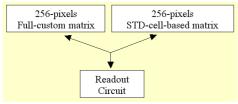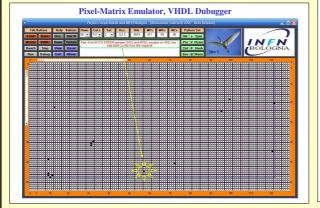
Conclusions

The paper describes the design of a mixed-mode ASIC that implements a matrix of MAPS cells along with a digital readout sparsification circuit. The design has been carried out within the SLIM5 Collaboration and it is aimed at overcoming the readout speed limit of future large-matrix pixel detectors for particle tracking, by matching the requirements of future HEP experiments. In particular, the readout architecture extends the flexibility of the MAPS devices to be also used in first level triggers on tracks in vertex detectors.

A brief description

Basically, let us say that the when the matrix has some hits (pixels that detect an over-threshold charge), it is swept from left to right and, at each clock period, all the hits present in a column of pixels, from 1 to 8, can be readout. This operation starts as long as a hardwired readout queue has free locations to temporarily store the information of the hits. In fact, at the hits' coordinates is associated a time mark (time-stamp) and the overall formatted data are either sent to the output port, or temporarily stored in a FIFO-like memory in case the output port is busy. Thus, in principle, the architecture can readout the matrix up to 8 hits at a time in case they belong to the same column and can send the formatted data to the output but, at the same time, the output port can accept only one hit information at a time and this is why a queuing system is necessary. Moreover, the global architecture might be considered as a circuit that can run in two different operating modes, called custom-mode and digital-mode. If fact, it can be connected to an actual full-custom matrix of MAPS or to a digital matrix emulator composed of standard cells. In the first case the pixels may only be switched on via striking particles while in the second case the digital matrix must be loaded during an initial slow-control phase. The two different implementations share the same matrix's I/O pins but can be selected and activated only one at a time. For both modes, before running, a slow-control phase is required to load an internal configuration. In particular, 16 mask signals should be provided to select the MPs which are to be read and which are not, for examples in case they are too noisy or broken. Default mask, after a reset phase, is all-at-1, meaning no-mask. Moreover, it must be selected which of the two operating modes is wanted and, consequently, which matrix is to be enabled. The default mode, after a reset phase, is the digital-mode. In addition, only for the digitally emulated matrix, 256 registers should be loaded to simulate a given charge injection over the silicon area. Default registers, after a reset phase, are all-at-0, meaning no hits. The readout circuit operates in the same manner for the two modes. Figure below shows a sketch of the operating modes of Apsel3D.

The circuit is a digital architecture for a sparsified readout that interfaces with a matrix of 256 Monolithic Active Pixel Sensor (MAPS). It is the base for a prototype of a mixed-mode ASIC, namely Apsel3D. It readouts and sparsifies the hits of a matrix of 256 pixels. Once read, the hits are switched off. The matrix is divided into regions of 4 x 4 single pixels thus, 256 pixels are clustered into 16 groups of 16 pixels each, here-in-after named macropixels (MPs). In addition, the matrix is arranged in 32 columns by 8 rows of single pixels or, from a different viewpoint, in 8 columns of MPs, called MCs, by 2 rows of MPs, called MRs.

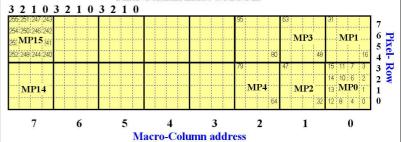## Pixel-Matrix Emulator, VHDL Dubugger

The matrix organization

The MAPS cells have been fabricated and tested in the past years with different layout flavours and now this ASIC includes readout capabilities. Plus, the mixed-mode design approach extends the performance of the chip as both the matrix and the readout logic have been developed separately and, lately, integrated together. In particular the matrix of MAPS has been described with a VHDL-Verilog model and used as a macro-cell block within a bigger digital design. The global place-and-route has been also digitally designed. The design is implemented via the STM 0.13μm CMOS digital technology. Let's give a brief functional description of the readout logic. The matrix of pixels is grouped into regions of 16 pixels (macro-pixels). Each region is activated and frozen via dedicated wires while the reset and readout phases are carried out via wires that are shared over the whole matrix. The dedicated wires may also be masked, one by one, in case they would be too noisy or burned, to avoid reading not-significant hits at any readout phase. At a given time, for example when a bunch-crossing signal is provided, the columns of macro-pixels that own at least one hit are seen via fast-or wires and frozen. The coordinates of these columns are associated with a time-stamp, whose buffers are located outside the matrix. Then, one at a time, the columns are enabled and the macro-pixels' output data are written on a readout bus. Once the column of pixels has been readout it is set free. Then the process moves to another column. The readout phase involves one column of macro-pixels at a time and this leaves the rest of the matrix free and capable of detecting new hits. Thus, the matrix may own hits, along with their personal time-stamps, belonging to different bunch-crossings. During the readout phase the matrix is swept and all the hits belonging to a given bunch are readout, reset and set free. The process continues till all the macro-pixels have been readout. All these capabilities have been designed into a single die composed of a large full-custom matrix of pixels that is readout via a standard-cell based digital circuit.

This is valid for both custom-mode and digital-mode. The entire matrix composed of 256 pixels is to be interpreted as follows:

- 8 MCs, addressed from left to right, range from 7 to 0,

- 8 rows of pixels, addressed from top to bottom, range from 7 to 0,

- 4 columns of pixels inside each MP, from left to right, range from 3 to 0.

In this view, each pixels is identified by a MC, a column inside the MC, and a pixel row. By converting these coordinated in digital logic it turns out 3+2+3 bits, i.e. 8-bits altogether which address exactly 256 pixels. This is the way the addresses are sent to the readout output port.

Pixel-Column inside a MC/MP

The Apsel3D layout

This is the picture of APSEL3D ASIC, designed with STM 130nm 6M Technology.

The whole layout dimension is: 2343.56μm × 1379.24μm.

The pitch of the pads is: 120.54μm on the Left, Bottom and Right sides.

The pitch of the pads is: 114.16μm on the Top side