Contents lists available at ScienceDirect

Nuclear Instruments and Methods in Physics Research A

CrossMark

Low-power priority Address-Encoder and Reset-Decoder data-driven readout for Monolithic Active Pixel Sensors for tracker system

P. Yang [l,*], G. Aglieri [a], C. Cavicchioli [a], P.L. Chalmet [b], N. Chanlek [c], A. Collu [d,e], C. Gao [l],
H. Hillemanns [a], A. Junique [a], M. Kofarago [a,f], M. Keil [a], T. Kugathasan [a], D. Kim [g], J. Kim [h],
A. Lattuca [i,e], C.A. Marin Tobon [a], D. Marras [d,e], M. Mager [a], P. Martinengo [a], G. Mazza [i,e],
H. Mugnier [b], L. Musa [a], C. Puggioni [d,e], J. Rousset [b], F. Reidt [a,j], P. Riedler [a], W. Snoeys [a],
S. Siddhanta [d,e], G. Usai [d,e], J.W. van Hoorne [a,k], J. Yi [h]

[a] CERN, 1210 Geneva 23, Switzerland
[b] MIND, Archamps, France
[c] Suranaree University of Technology, Nakhon Ratchasima, Thailand
[d] University of Cagliari, Cagliari, Italy
[e] INFN, Italy
[f] University of Utrecht, Utrecht, Netherlands
[g] Dongguk and Yonsei University, Seoul, South Korea
[h] Pusan National University, Busan, South Korea
[i] University of Torino, Torino, Italy
[j] Ruprecht-Karls-Universitat Heidelberg, Heidelberg, Germany
[k] Technische Universitat Wien, Vienna, Austria
[l] Central China Normal University, Wuhan, China

ARTICLE INFO

ABSTRACT

Active Pixel Sensors used in High Energy Particle Physics require low power consumption to reduce the detector material budget, low integration time to reduce the possibilities of pile-up and fast readout to improve the detector data capability. To satisfy these requirements, a novel Address-Encoder and Reset-Decoder (AERD) asynchronous circuit for a fast readout of a pixel matrix has been developed. The AERD data-driven readout architecture operates the address encoding and reset decoding based on an arbitration tree, and allows us to readout only the hit pixels. Compared to the traditional readout structure of the rolling shutter scheme in Monolithic Active Pixel Sensors (MAPS), AERD can achieve a low readout time and a low power consumption especially for low hit occupancies. The readout is controlled at the chip periphery with a signal synchronous with the clock, allows a good digital and analogue signal separation in the matrix and a reduction of the power consumption. The AERD circuit has been implemented in the TowerJazz 180 nm CMOS Imaging Sensor (CIS) process with full complementary CMOS logic in the pixel. It works at 10 MHz with a matrix height of 15 mm. The energy consumed to read out one pixel is around 72 pJ. A scheme to boost the readout speed to 40 MHz is also discussed. The sensor chip equipped with AERD has been produced and characterised. Test results including electrical beam measurement are presented.

1. Introduction

Active Pixel Sensors have been widely used for many applications, such as imaging of visible light, X-ray imaging, biotechnology, medicine, astronomy and high energy physics. In the context of high energy particle physics, short lived particles are studied through their decay vertices. Therefore, an excellent tracking system is required to reconstruct and measure track momenta at a high luminosity. Pixel technologies need to achieve high granularity, low power, good radiation tolerance and fast read out. The ALICE Inner Tracking System (ITS) upgrade will replace the present ITS with a new tracker [1] with 7 layers of MAPS [2,3] during the second long shutdown of the LHC in 2018. The chip sensor aims to achieve a spatial resolution of ~ 5 μm (pixel pitch smaller than 30 μm) and a power density lower than 100 mW/cm^2 [2]. The state-of-the-art of MAPS used for particle detectors is represented by the STAR pixel detector, which is equipped with the ULTIMATE chip [4]. The ULTIMATE is an example of the traditional rolling shutter readout architecture. It achieves an integration time of 185.6 μs and a power density of 130 mW/cm^2 [5] with a 928 by 960 pixel array (20.7 μm pitch). The integration time of the

* Corresponding author.
  E-mail address: yangping0710@126.com (P. Yang).

rolling shutter depends on the number of rows for a given readout clock frequency. The frame readout time can be reduced by embedding row-wise parallel readout with increasing the number of column-level discriminators. However, without any kind of data compression inside the matrix, such a reduction in the frame readout time comes with increasing power consumption. Data compression techniques inside the matrix have been studied to improve the readout speed. The main technique is the token-ring architecture with a data-driven and triggered readout. The speed of the token-ring technique is limited by the time that the logic needs to ripple through the whole daisy chain. However it can be improved by implementing a fast priority look-ahead logic, like in the ATLAS readout chip with a pixel array of 80 columns by 336 rows [6]. This allows the readout speed to be independent on the column length by skipping the empty banks. The digital circuitry occupies 22% of the 50 μm by 250 μm pixel size with a power consumption of 4.4 mW/double-columns [7]. To gain more margin on material budget and data readout capability for the ALICE ITS upgrade, the ALPIDE (ALice PIxel DEtector) chip design aims for a power density and an integration time one order of magnitude below the specifications which cannot be achieved by the readout architectures mentioned above. Therefore, a new data-driven readout architecture has been developed with zero-suppression in the matrix. The architecture contains an Address-Encoder and Reset-Decoder (AERD) circuit based on an arbitration-tree with the priority logic, only reading hit pixels. The zero-suppression technique allows a sufficient reduction of the readout time and the power consumption. This paper presents the readout architecture and its implementation. It also suggests means to optimise the circuit, and presents test results and summarises findings.

2. Proposed architecture for the prototype for the ALICE ITS upgrade

The pALPIDEfs (full scale prototype of ALice PIxel DEtector) [8] for the ITS upgrade has been implemented with pixels that consist of a low power (∼40 nW) binary front-end and a data-driven readout circuit. This architecture makes the integration time independent with the array readout time, limiting the pile-up even for a large matrix because of the integration time decided by the shaping time of the front-end. During the prototype development, many different parameters of the architecture were analysed and simulated, in order to find the best compromise between S/N, detection efficiency [9], power and area constraints. A pixel pitch of 28 μm has been chosen and the chip size is 15 mm × 30 mm consisting 512 rows by 1024 columns, as shown in Fig. 1. The front-end works like an analogue memory: it will generate a pulse with a shaping time of about 4 μs. If a strobe is applied during this time after a hit arrives at the pixel, this hit data will be latched in the in-pixel state register as a STATE signal. The AERD circuit which reads and resets the hit pixels is arranged in double columns inside the matrix. The *I/O* signals of the AERD block are described in Fig. 1 and Table 1. The design takes advantage of a gating technique to reduce power: SYNC is a gated clock signal propagated from the chip periphery, used to select the highest priority pixel to be read and reset. A VALID signal is a flag which is activated when there is at least one hit in a pixel.

2.1. Principle of the AERD circuit operation

The AERD readout circuit has been implemented based on an arbiter tree scheme with hierarchical address encoders and reset decoders. The repeated basic logic of the tree contains three units which are shown in Fig. 2. A FAST OR gate chain is used to generate the VALID signal and propagate it to the chip periphery. The address encoder receives as inputs the priority logic outputs to generate the address value of each basic block. The reset decoder is fed by the output of the priority encoder and the SYNC signal from the higher level to generate the SYNC signal for the lower level. That is then used to select the pixel with the highest priority to be read and reset. For a
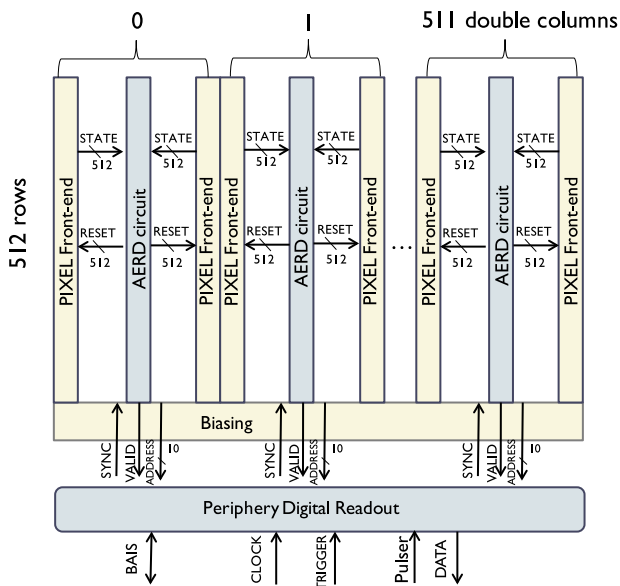

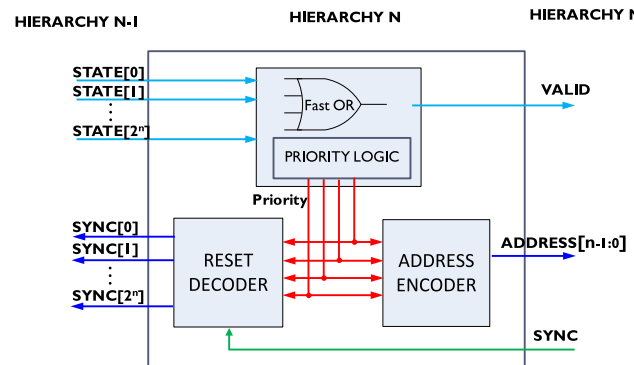**Fig. 1.** The structure of the pALPIDEfs chip.

Fig. 2. The structure of the AERD basic logic block.

Table 1
Description of the AERD *I/O* signals.

Signal	Direction	Description
STATE[1023:0]	Input	Outputs of the pixel state registers, it is high when there is a hit.
SYNC	Input	Synchronous with the clock, it propagates to the highest priority hit pixel.
VALID	Output	Active if at least 1 pixel is hit (FAST OR logic).
ADDR[9:0]	Output	Address of the hit pixel that is being reset.
RESET[1023:0]	Output	Resets the pixel state register on falling edge.

given number of pixels to encode, the number of hierarchical levels, basic logic blocks and routing channels are described by the following equations:

$$N_{Layer} = \log_b(N_{pix}) \tag{1}$$

$$N_{block} = \sum_{i=1}^{\log_b N_{pix}} \frac{N_{pix}}{b^i} = \frac{N_{pix} - 1}{b - 1} \tag{2}$$

$$Routing = [\lceil \log_b(N_{pix}) \rceil - 1] \times b. \tag{3}$$

Here N_{Layer} is the number of the hierarchical levels, N_{block} is the number of basic blocks that are needed, Routing is the number of routing channels of one hierarchical signal such as VALID or SYNC, N_{pix} is the number of pixels covered by the AERD circuit, b is the number of inputs of the basic block. The number of the hierarchical levels decreases with increasing b.

A small implementation area allows a small pixel size which results in a good spatial resolution. The two main area contributors are the routing channels and the transistors. To analyse the best solution, the numbers of routing channels and total transistors are plotted in Fig. 3. The circuit of the basic block gets complicated with increasing b, and the number of the blocks increases significantly when decreasing b to 2. The equation of the transistors has been obtained by summing up the number of transistors in the logic. Because the address value is encoded inside each basic block with a binary output, b should be a power of 2 to obtain the best solution without any wasting bit. For a small b like 4, the number of the transistors is an accurate value; for a large b, the number of the transistors is an approximation, but the trend is correct, and the optimisation for low values of b still holds. Those plots in Fig. 3 reveal that to decode 1024 pixels, $b=4$ is the best choice as it is near the minimum for both number of routing channels
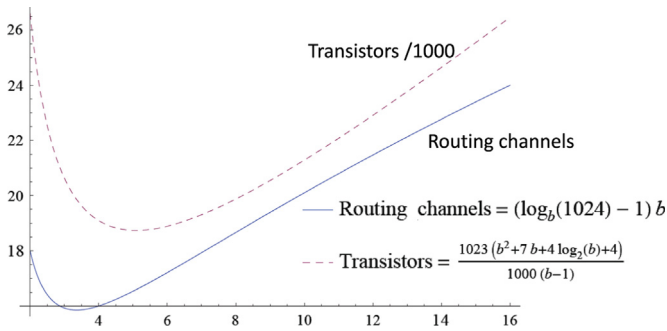

Fig. 3. The optimisation of the tree hierarchical structure.

and transistors. The implementation of the logic circuit is shown in Fig. 4. (a) The priority logic generates four outputs and only one is active during a readout cycle. (b) The address encoder is a combinatorial circuit with a tri-state output. A SYNC signal (Table 1) enables/disables the output states of each block to control the address bus. (c) The reset happens at the falling edge of the SYNC signal, which is implemented with NOR gates fed by the outputs of the priority logic.

The block diagram of the tree structure to decode 16 pixels as an example is shown in Fig. 5: The VALID signal propagates from the lower hierarchical level of the arbiter tree to the top. If pixel 4 is hit, through the FAST OR chain, VALID will be active. During the readout phase, while the VALID signal is active, a synchronous signal (SYNC) propagates back into the hit pixel to read its address. Combined with priority logic, at the lowest hierarchical level of the tree, the SYNC signal resets only the pixel with the highest priority during the same clock cycle after the address of that pixel has been read. During the propagation of the SYNC signal, also the ADDRESS[3:0] of the pixel being reset propagates down to the End of Column (EoC). The ADDRESS lines are managed by tri-state buffers, which are enabled when SYNC is high, and in high impedance when SYNC is low. Therefore, to ensure that the value of ADDRESS is available at the EoC, half a SYNC cycle is set aside for the propagation of the ADDRESS signals. On the falling edge of the SYNC, the state register of the read pixel is reset, and when SYNC is low, the new configuration of VALID and internal signals propagate such that the next pixel is going to be read in the subsequent SYNC cycle. The address value is encoded in each level to decrease the power consumption and save logic area [10] as described in Section 2.4.

2.2. Implementation and post-simulation results

2.2.1. Timing sequence

Fig. 6 shows the timing diagram of the circuit, which indicates the delays that were considered to select the working frequency of the circuit.

0. The first pixel in the double-columns is hit and the VALID is asserted after a delay D.

1. The ADDRESS starts to be encoded when SYNC changes to high. The recovery delay B is the time required from the rising edge of the CLK for the address to stabilise. It is calculated at the EoC, and therefore it includes the propagation delay of the CLK to the pixel, and the propagation delay of ADDRESS down to the EoC. Delay B must be smaller than $T/2$, where T is the period of the CLK signal.

2. The pixel state register of the highest priority pixel is reset on the falling edge of the SYNC signal. The removal delay A is the time
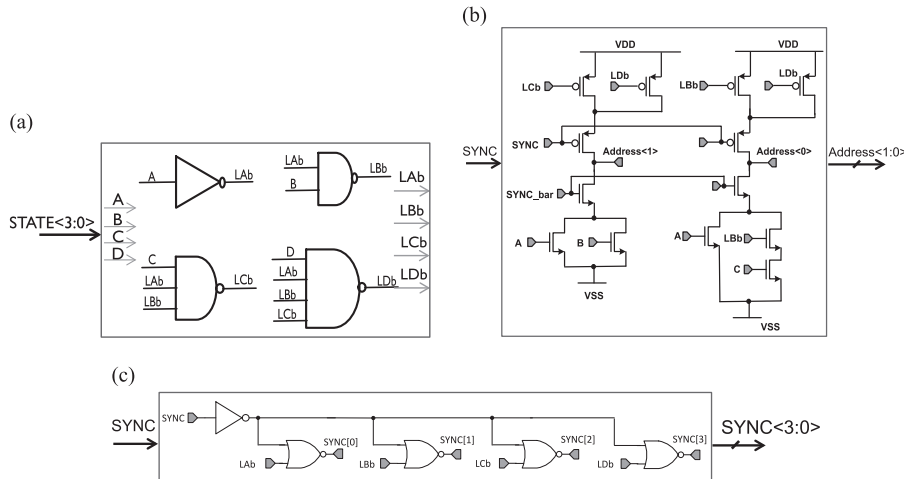

Fig. 4. The implemented circuit of the basic block, corresponding to Fig. 2: (a) priority logic of four inputs; (b) combinatorial address encoder logic; and (c) reset decoder logic.

Fig. 5. Two hierarchical levels of AERD to decode 16 pixels.

Fig. 6. Readout timing diagram of the AERD.

required from the falling edge of the CLK signal as seen from the EoC until the pixel is reset.

3. The ADDRESS value is sampled at the falling edge of the CLK. The sampling window duration is $T/2 - B + A$.

4. Delay C is the time required for readout of the last hit pixel in the double-columns, from the falling edge of the CLK to the falling edge of the VALID signal. Delay C must be smaller than T, to avoid an additional readout cycle.

Assuming a duty clock cycle of 50%, these parameters set the top limit of the CLK frequency as demonstrated by the following equation:

$$f_{max} = \frac{1}{T_{min}} = \max\left[\frac{1}{2B}, \frac{1}{C}\right]. \tag{4}$$

2.2.2. Delay analysis

The estimated loads in this technology are $R = 4.3\ k\Omega$, $C = 3.7\ pF$ for a 15 mm long wire. Without any buffers, the propagation delay is simply estimated as $t_d = 0.7 \times RC = 11.11\ ns$. To optimise the delay where N_{buf} is the number of buffers, $t_{d_{inv}}$ is the average propagation delay of one inverter, the minimum delay t_{dopt} can be calculated as

$$t_{dopt} = 0.7 \times \frac{RC}{N_{buf}} + t_{d_{inv}} \times N_{buf} \times 2. \tag{5}$$

Assuming $t_{d_{inv}} = 200\ ps$, then $N_{buf} = 8$ is the number of buffers needed. The minimum propagation delay equals:

$$t_{dopt} = 3.4\ ns \tag{6}$$

The timing of the AERD basic block has been fully characterised in the digital flow, the average internal delay t_{davg} of the signals to propagate through each of the hierarchical levels has been characterised as approximately 800 ps. The worst case delay of the read out occurs when the first row of pixels is hit, as this results in the largest loads on the ADDRESS and SYNC drivers. Therefore, the time required

to read out one ADDRESS can be calculated as

$$t_{min} = t_{dopt} \times 2 + \text{Layer} \times t_{davg} = 3.4 \text{ ns} \times 2 + 5$$
$$\times 800 \text{ ps} = 10.8 \text{ ns}. \quad (7)$$

This calculation excludes the gate capacitance and the Process–Voltage–Temperature (PVT) variations and reveals that it is difficult to work at 40 MHz if just half a CLK cycle (12.5 ns) is used to enable and propagate the ADDRESS.

The delays have been estimated with a simulation where there are two events to be read out, the first pixel (0) and the last pixel (1023). These delays are calculated with all the parasitic resistances and capacitances which are extracted from the layout. Table 2 shows the delays on different corners. From Eq. (4), the operating frequency of the circuit was chosen to be 10 MHz. An improvement scheme to boost the speed up to 40 MHz is proposed in Section 3.

2.2.3. Layout view

AERD is composed of 341 basic logic blocks and buffers. Fig. 7 shows the layout of 4 pixels. The pixel area is entirely occupied by the sensor, the front-end circuit and a fraction of the AERD. They occupy 15%, 23% and 62% respectively. The irregular shape of the AERD provides more space for the collection electrode and aims to decrease the input node capacitance [11]. The distance between power lines and the rest of the circuit has been increased beyond the minimum design rule value to improve the yield for this large chip.

2.3. Race condition of this asynchronous AERD circuit

Asynchronous circuits are generally affected by critical races or hazards due to the internal gate delays. The possible races or

Table 2
Delays A–D on the 3 corners (time unit ns) FF: fast corner, $V_{DD}=1.98$ V, 0 °C; TT: nominal corner, $V_{DD}=1.8$ V, 27 °C; SS: slow corner, $V_{DD}=1.62$ V, 125 °C.

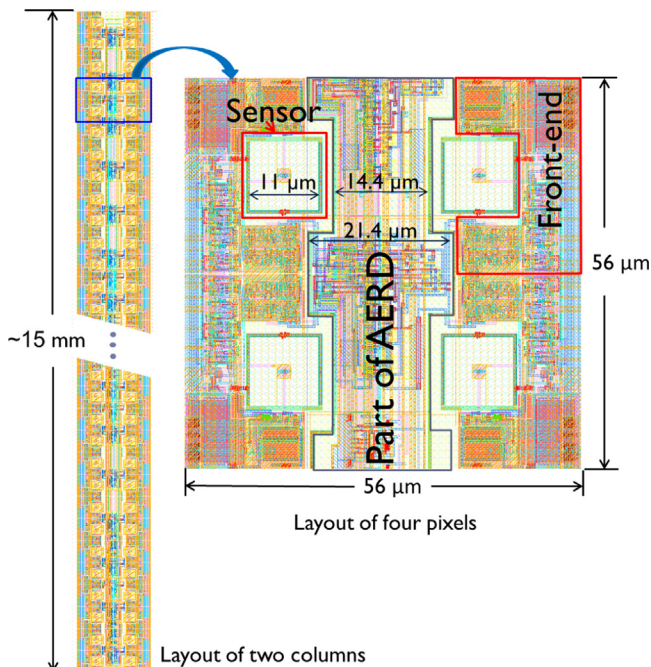| DELAY	FF	TT	SS
A	7.1	9.2	14.3
B	11.6	14.8	22.4
C	12.6	16.5	26.7
D	3.2	3.9	5.6

Fig. 7. Layout of view of a 4-pixel block.

hazard conditions of this architecture occur during the decoding phase. As shown in Fig. 8(a), four NOR gates are used to generate the SYNC signals for the lower hierarchical levels. The NOR gates are fed by an inverted SYNC signal and the active-low output of the priority logic. It has a chance to reset more than one pixels at the same falling edge when the gate delay of the priority logic is smaller than the RC delay of the inverted SYNC. As shown in Fig. 8 (b), if LAb and LDb are active, there are hits on the highest priority pixel (STATE A) and the lowest priority pixel (STATE D) of four elements. Delay_D3 is the time difference between SYNC_D0 and SYNC_D3 while Delay_state is the NAND gates delay of the priority logic (Fig. 4). If $Delay\_D3 > Delay\_state$, a glitch will occur on SYNC [3] and STATE D will also be reset at the same falling edge of the SYNC as STATE A. To prevent this race condition, $Delay\_D3$ must be smaller than $Delay\_state$. Because the pixel pitch is only 28 μm, $Delay\_D3$ is negligible compared to the logic gate delay.

2.4. Power consumption analysis

2.4.1. Address encoder

A hierarchical address encoder is used in the AERD rather than the traditional one encoding the address at the lowest hierarchical level [10] to save dynamic power consumption. Full custom combinatorial encoder logic is used inside every basic block instead of the traditional logic for address generation to save area and transistor count (reduced from 14 to 13).

2.4.2. Power consumption

A rough estimate of the matrix power consumption for the rolling shutter and the AERD scheme is carried out below. Three different matrix architectures are considered, including the rolling shutter with in-pixel comparator scheme, the rolling shutter with column-level comparator scheme and the AERD readout with in-pixel comparator scheme, and their respective power consumption is expressed in Eqs. (8)(a)–(c). The front-end is active row by row for the rolling shutter scheme and always active for the AERD readout scheme. Eqs. (8)(d) is the most probable value of the signal amplitude. For simplicity in the calculation of the dynamic power consumption in the case of the rolling shutter we assumed that only one pixel is hit per particle. This does not have a significant impact as the power is dominated by the static power in that case. Moreover to calculate the dynamic power for the rolling shutter architecture we assume only one clock signal is distributed into the pixel array:

$$P_{rscol} = \underbrace{P_{FErscol} \times N_{col}}_{\text{static power}} + \underbrace{\frac{C_L \times dv^2 \times N_{col} \times N_{row} \times Hit\_density}{T_{clock} \times N_{row}}}_{\text{dynamic power for loading data}}$$
$$+ \underbrace{\frac{C_L \times N \times V_{DD}^2 \times N_{col}}{T_{clock} \times N_{row}}}_{\text{dynamic power for clock distribution}} \quad (a)$$

$$P_{rsin} = \underbrace{P_{FErsin} \times N_{col}}_{\text{static power}} + \underbrace{\frac{C_L \times V_{DD}^2 \times N_{col} \times N_{row} \times Hit\_density}{T_{clock} \times N_{row}}}_{\text{dynamic power for loading data}}$$
$$+ \underbrace{\frac{C_L \times N \times V_{DD}^2 \times N_{col}}{T_{clock} \times N_{row}}}_{\text{dynamic power for clock distribution}} \quad (b)$$

$$P_{AERD} = \underbrace{P_{FEAERD} \times N_{row} \times N_{col}}_{\text{static power}}$$
$$+ \underbrace{\frac{C_L V_{DD}^2 \times \frac{1}{2} N_{col} \times N_{row} \times Hit\_density \times Add\_avg}{T_{clock} \times N_{row}}}_{\text{dynamic power for loading data}} \quad (c)$$

$$dv = 60 \text{ μv}/e^- \times 80 \ e^-/\text{μm} \times 18 \text{ μm} = 86.4 \text{ mV} \quad (d)$$
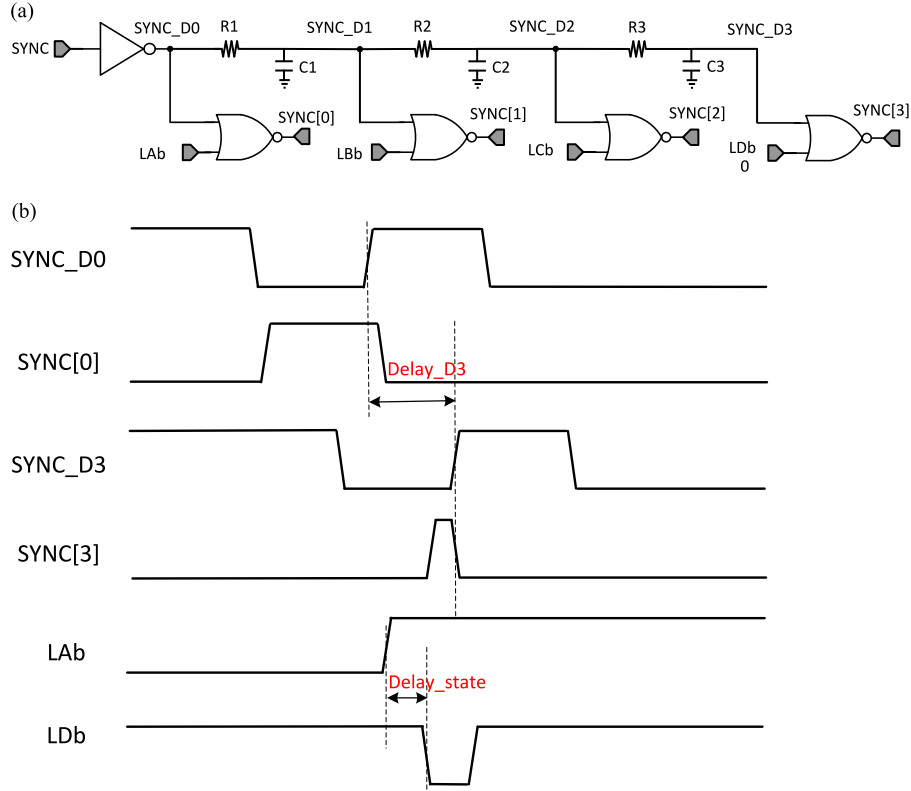
$$(8)$$

Fig. 8. Operational timing programme of the reset decoder scheme: (a) reset decoder circuit of the AERD basic logic block and (b) timing analysis of the racing condition of the AERD.

where $P_{FErscol}$ and P_{FErsin} are the front-end power consumption of the rolling shutter with column-level comparator and with in-pixel comparator, N_{col} is the number of columns in the array, N_{row} is the number of rows in the array. *Hit_density* is the ratio between the number of hit pixels and the total number of pixels. *Add_avg* is the average number of activated address lines of the AERD readout structure. C_L is the load of the column line, $C_L \times N$ is the capacitive load of the clock signal, expressed as a factor N times the load C_L of the column line. V_{DD} is the power supply, dv is the amplitude of the analogue output signal between different rows of the rolling shutter scheme, T_{clock} is the readout clock cycle, P_{FEAERD} is the power consumption of the AERD front-end. In Eqs. (8)(c), a factor $\frac{1}{2}$ is used because the AERD readout circuit is arranged in double-columns. An example of the front-end gain is 60 μv/e⁻ [4], 80 e⁻/μm means the particle generates 80 hole/electron pairs in 1 μm silicon thickness. 18 μm is the epi-layer thickness of the standard TowerJazz CIS process. The dynamic power consumption of the rolling shutter scheme included two parts: one is the hit data processing, the other one is the power consumption of the clock.

We use the chip dimension of the pALPIDEfs for a comparison in more detail. $V_{DD}=1.8$ V, $N_{row}=512$, $N_{col}=1024$, N is about 3 for the matrix we consider. The AERD architecture has 10 address lines to readout 1024 pixels, using *Add_avg*=6 as an average number of the activated lines during the readout. In this case the power consumption is consistent with the parasitic simulation results. This approximate number 6 is reasonable and includes 5 for the activated address lines, 1 for the SYNC gated clock signal and the VALID signal propagation. For the chip dimensions we are using for comparison, the load of the clock signal is $C_L \times 3$ for the rolling shutter scheme. The examples of the front-end power consumption can be found in MISTRAL, ASTRAL [12] and ALPIDE [8], where $P_{FErscol}=20$ μW, $P_{FErsin}=200$ μW and $P_{FEAERD}=40$ nW. Taking into account that in the inner tracking detector layers of the vertex particle detector at LHC, the hit pixel density is in order of a few per thousand, therefore we took the *Hit_density* range
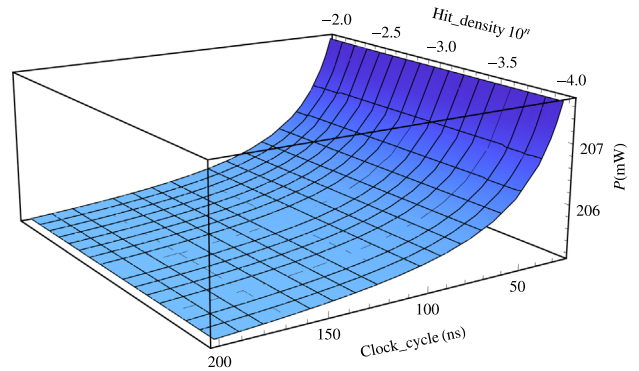

Fig. 9. The matrix power consumption of the rolling shutter readout with column-level comparator.

from 0.01 to 0.0001 to do the calculation. The results of the average matrix power consumption of these three different architectures are shown in Figs. 9 and 10 which has a big difference in vertical scales. Those 3D plots indicate that the matrix power consumption of the column-level comparator architecture is much higher than the other two in-pixel level comparator architectures. This is because the analogue signal of the pixels needs much more energy to be transferred to the end of the matrix without any signal loss at the same speed as transferring digital signals. And the column-level comparators consume additional power beside the pixel array. The matrix power consumption of the ALPIDE and the rolling shutter scheme with in-pixel comparator is comparable, but the readout for the ALPIDE architecture can be much faster because of the AERD zero-suppression readout circuit. For the rolling shutter with in-pixel comparator architecture, an additional zero-suppression circuit is needed in the
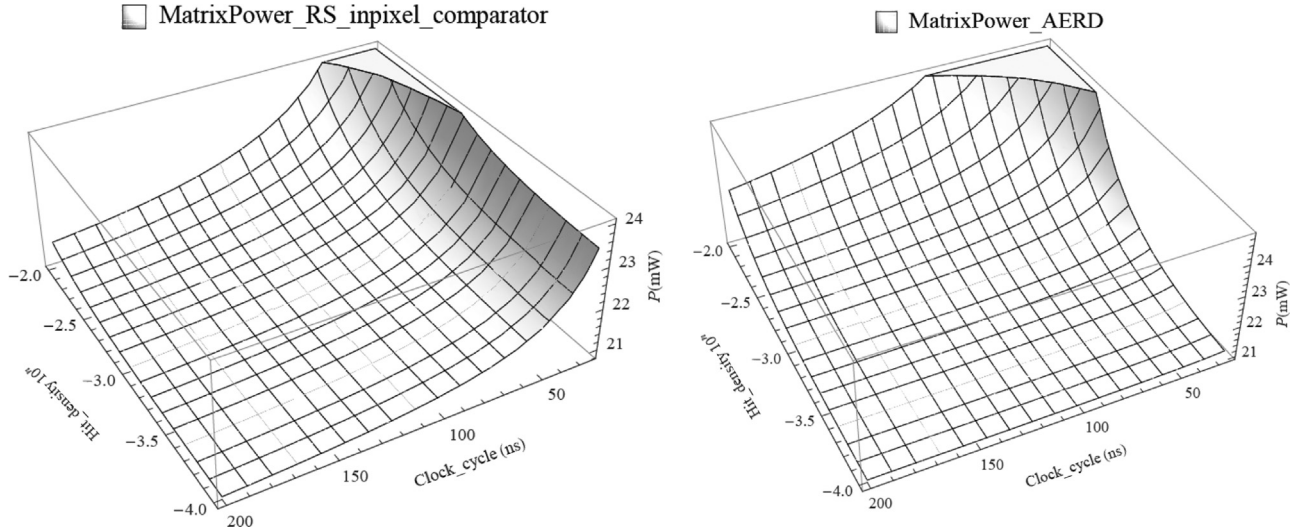
Fig. 10. Comparison: the matrix power consumption of the AERD readout versus the rolling shutter readout.
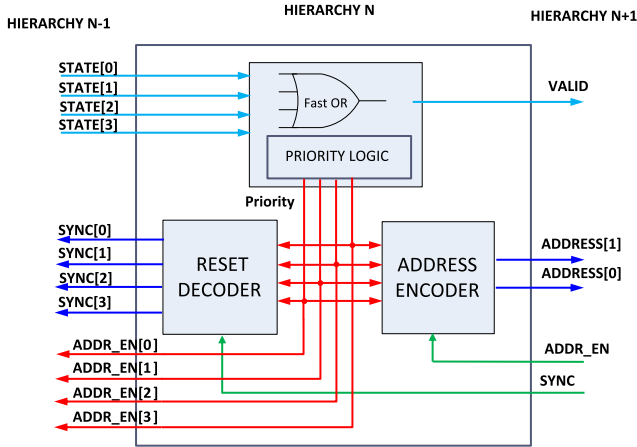

Fig. 11. The proposed structure of the basic AERD block to boost the speed.

chip digital periphery. Because of the low power front-end and the zero-suppression technique of the AERD readout, the matrix power consumption of the ALPIDE is still smaller than the rolling shutter with in-pixel comparator scheme when the readout speed is faster than 7 MHz at a hit density level lower than 0.1%.

A power consumption below 24 mW is achievable when the hit density is extremely low (less than 0.1%) at a high readout speed of 40 MHz. According to the simulation results with parasitic capacitances and resistances of the pALPIDEfs chip, the dynamic readout power consumption to transmit the data to the chip periphery is approximately 720 μW/hit. The energy used to read out one hit is around 72 pJ.

3. Proposed scheme to improve the AERD readout speed

3.1. Proposed new AERD structure to improve the speed

If we can use the full clock cycle rather than half the clock cycle to generate and propagate the address value to the EoC, this would improve the readout speed sufficiently to run at 40 MHz (Fig. 11). The main difference of the new logic block is that the outputs of the priority logic are used to enable the address encoder for the lower hierarchical levels (Fig. 2). An example to decode 16 pixels is shown in Fig. 12, the address encoder of the top level is always active because there is only one block on the top and no conflicts

on the ADDRESS bus. More than one outputs of the ADDR_EN can be active because all the blocks operate in parallel starting from the second level of the tree. To guarantee that just the pixel with the highest priority is enabled to encode the address avoiding conflicts on the ADDRESS bus, additional OR gates are needed to manage the priority path on ADDR_EN signals. The ADDR_EN iteration logic is expressed as follows:

$$ADDR\_EN\langle N-1\rangle = ADDR\_EN\langle N+1\rangle \ OR \ ADDR\_EN\langle N\rangle \qquad (9)$$

Here $ADDR\_EN\langle N-1\rangle$ is the input ADDR_EN signal of level N-1, $ADDR\_EN\langle N+1\rangle$ is the input ADDR_EN signal of level N+1, $ADDR\_EN\langle N\rangle$ is the input ADDR_EN signal of level N.

3.2. Timing sequence

Separating the two phases of address encoding and reset decoding generates two improvements: first a full clock cycle can be used for address propagation, and second the loads of some internal signals are smaller. The timing is as follows, see Fig. 13:

0. The first pixel in the double-columns is hit and the VALID is asserted after the delay D.

1. The ADDRESS starts to be encoded when ADDR_EN signal is active. The new priority logic is operated automatically after resetting the previous pixel. The delay B is the time required from the rising edge of the STATE for the ADDRESS to stabilise at the EoC. Delay B must be smaller than T, where T is the period of the CLK signal.

2. The pixel state register of the highest priority is reset at the falling edge of the SYNC signal. The removal delay A is the time required from the falling edge of the CLK signal at the EoC to the pixel be reset.

3. The ADDRESS value is sampled at the falling edge of the CLK signal.

4. Delay C is the time required in correspondence to the readout of the last hit pixel in the double-columns, from the falling edge of the CLK to the falling edge of the VALID signal. Delay C must be smaller than T to avoid an additional readout cycle. The load of the SYNC signal is much smaller than before by separating the encoding and decoding of the readout, therefore reducing delay C.

These parameters set the new top limit of the CLK frequency:

$$f\_max = \frac{1}{T\_min} = \max\left[\frac{1}{B}, \frac{1}{C}\right]. \qquad (10)$$
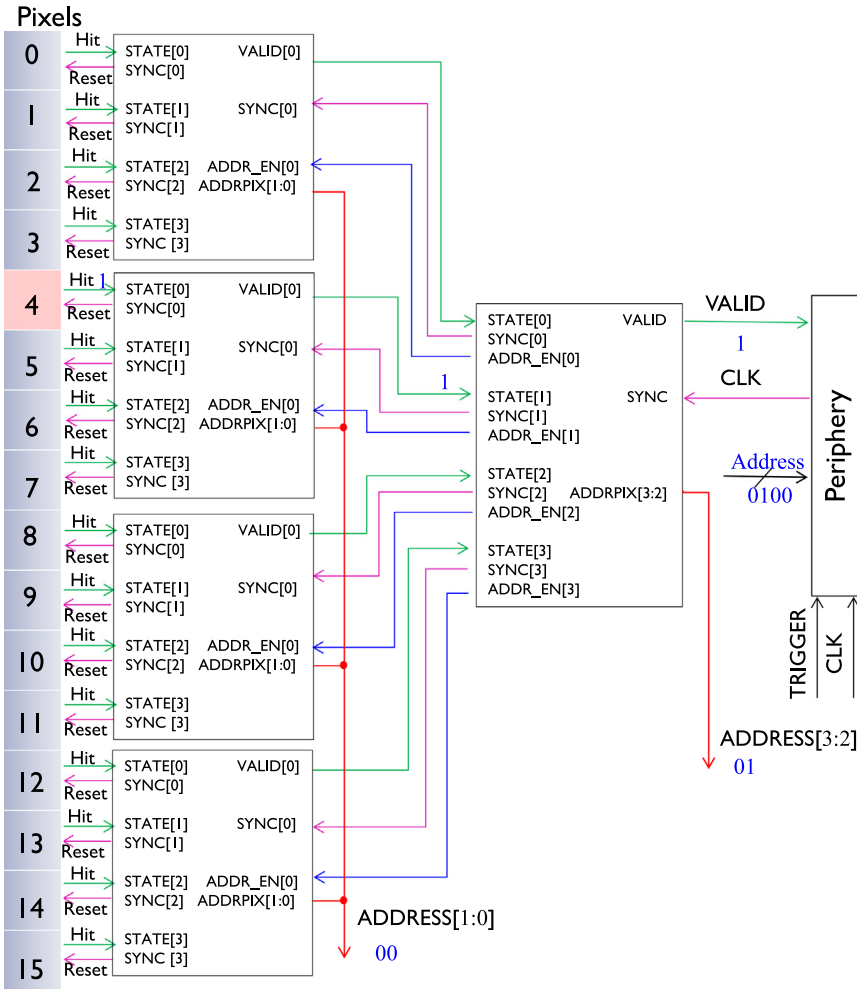
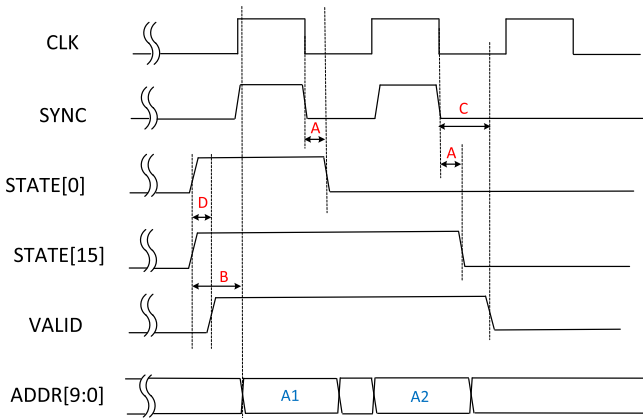**Fig. 12.** AERD scheme to decode 16 pixels with speed improvement.

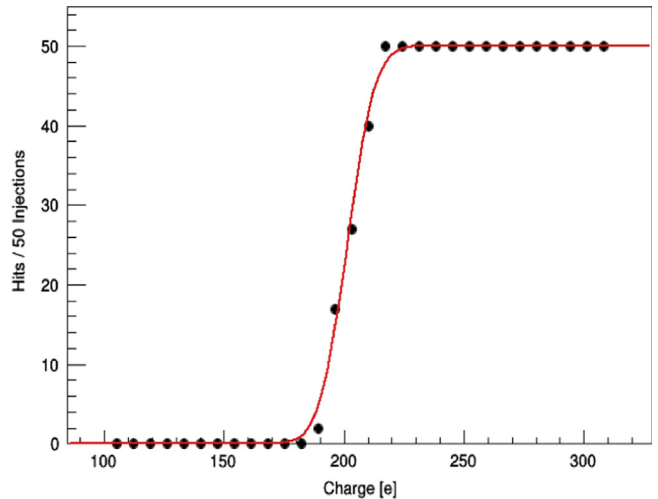**Fig. 13.** The timing diagram of the fast speed AERD read out.

Fig. 14. The threshold with S-curve of pALPIDEfs.

The simulation results of these delays with estimated loads of this silicon process on the nominal corner are $B=12.6$ ns, $A=8.5$ ns, $C=9.3$ ns. From our experience with simulation including parasitics, the delay B will be smaller than 1.5×12.6 ns which is approximately 19 ns, thereby establishing the feasibility of this architecture at 40 MHz.

3.3. Area estimation of the improved version

All the hierarchical blocks are arranged in one column, creating a shortage of vertical connection channels. The routing lines therefore dominate the area inside the AERD block more than the logic gates [2]. A "standard" layout with all hierarchical connection nets over it is created, and allows more efficient connections on the top level. Some of the nets are very short thereby reducing the overall area penalty. This is the reason that the metal layers of M2–M4 do not fully fill the minimum width of the shape as shown in Fig. 7. There are 16 additional nets in the proposed improved architecture, but if the routing
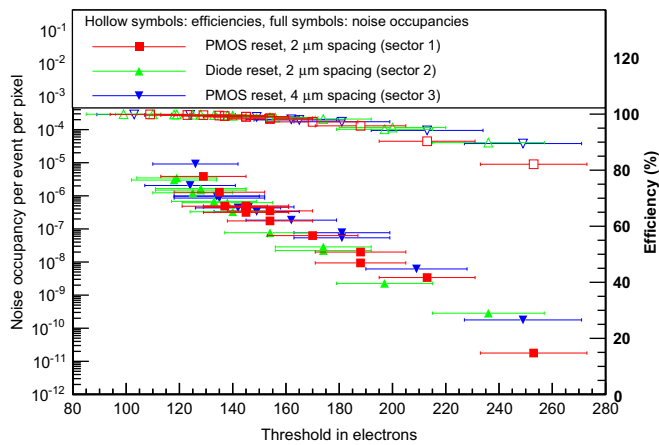
Fig. 15. Beam test results of efficiency and noise occupancy.

channels are used more effectively by "overlapping" short nets in the same channel, the routing area will fit inside the same pixel pitch or even can be reduced. To summarise, the proposed improved architecture incurs a very low penalty thereby improving speed without using extra silicon.

4. Test results of the pALPIDEfs

The pALPIDEfs only has digital outputs with the AERD operating at 10 MHz to read the digital data from the matrix. The testing performs charge injection with increasing amplitude at the input node of the front-end to study its threshold and noise distribution. The typical response obtained by this scan takes the shape of an S-curve which describes the channel response: at very low values for the input signal, the front-end never detects a hit and at very high values it always does. The 50% point corresponds to the threshold and the slope at this point corresponds to the noise. For a pure Gaussian noise, the S-curve can be fitted by an error function to extract the threshold and noise value. A good S-curve shape is obtained as shown in Fig. 14, and the threshold value is comparable with the results from the simulation. Fig. 15 shows the beam test results of the detection efficiency and noise occupancy at CERN PS using 6 GeV π-particles. The detection efficiency is 99% when the threshold is below 160 e^-. After masking 20 noisy pixels, the noise occupancy is below 10^{-5} at a threshold

higher than 110 e^-. These results satisfy the ALICE ITS upgrade specifications. The irradiated devices are currently under test. However, previous irradiation tests on smaller scale prototypes satisfied the ALICE requirements of 10^{13} n_{eq}/cm^2 [13].

5. Conclusion

As an alternative to the traditional rolling shutter readout architecture, the Address Encoder and Reset Decoder data-driven readout architecture has been implemented and tested, and shows significant advantages both in terms of readout time and power consumption. This allows the pALPIDEfs development to reduce integration time and power consumption well below the specifications for the ALICE ITS upgrade. The test results from the fabricated prototypes show that the AERD circuit operated at 10 MHz meets or exceeds specifications. Furthermore, this paper has shown that the readout speed can be boosted from 10 MHz to 40 MHz through the separation of the decode and reset phases without any extra silicon area.

References

[1] L. Musa, et al., Journal of Physics G 41 (2014) 087002. http://dx.doi.org/10.1088/0954-3899/41/8/087002.
[2] W. Snoeys, et al., Journal of Instrumentation 8 (2013) C12041. http://dx.doi.org/10.1088/1748-0221/8/12/C12041.
[3] C. Cavicchioli, et al., Nuclear Instruments and Methods in Physics Research A765 (2014) 177.
[4] Ch. Hu-Guo, et al., Journal of Instrumentation 4 (2009) P04012. http://dx.doi.org/10.1088/1748-0221/4/04/P04012.
[5] L. Greiner, et al., Nuclear Instruments and Methods in Physics Research A650 (2011) 68.
[6] I. Peric, et al., IEEE Transactions on Nuclear Science NS-46 (1999) 166.
[7] Tomasz Hemperek, et al., Digital architecture of the new ATLAS pixel chip FE-I4, in: IEEE Nuclear Science Symposium Conference Record, N13-179, 2009.
[8] P. Yang, et al., MAPS development for the ALICE ITS upgrade, Presented at PIXEL 2014, Niagara Falls, Canada, September 2014.
[9] S. Senyukov, et al., Nuclear Instruments and Methods in Physics Research A730 (2013) 115.
[10] J. Georgiou, et al., Electronics Letters 42 (3) (2006).
[11] J. Van Hoorne et al., The upgrade of the ALICE inner tracking system-status of the R&D on monolithic silicon pixel sensors, Presented at Third International Conference on Technology and Instrumentation in Particle Physics, June, 2–6, 2014, Beurs van Berlage, Netherlands.
[12] F. Morel, et al., Journal of Instrumentation 9 (2014) C01026. http://dx.doi.org/10.1088/1748-0221/9/01/C01026.
[13] H. Hillemanns et al., Radiation hardness and detector performance of new 180 nm CMOS MAPS prototype test structures developed for the upgrade of the ALICE Inner Tracking System, in: Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), IEEE, 2013.