

# **PRECISE: A 5Gbps Serialiser for Scientific Detectors in a 180nm CMOS Image Sensor Process**

Seddik Benhammadi, Nicola Guerrini, Deividas Krukauskas, Andrea Mifsud and Iain Sedgwick,

## **Published version information**

**Citation:** S. Benhammadi, N. Guerrini, D. Krukauskas, A. Mifsud and I. Sedgwick, "PRECISE: A 5Gbps Serialiser for Scientific Detectors in a 180nm CMOS Image Sensor Process," 2022 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), Italy, 2022, pp. 1-8,

**DOI:** 10.1109/NSS/MIC44845.2022.10399290

© **2024** This version is made available in accordance with the licensing statement contained in the manuscript, under a [CC-BY licence](#) with permission from the authors. Please cite only the published version using the reference above. This is the citation assigned by the publisher at publication. Please check the publisher's website for any updates.

This version is made available in accordance with publisher policies. Please cite only the published version using the reference above. This is the citation assigned by the publisher at the time of issuing the AAM. Please check the publisher's website for any updates.

# PRECISE: A 5Gbps Serialiser for Scientific Detectors in a 180nm CMOS Image Sensor Process

Seddik Benhammadi, Nicola Guerrini, Deividas Krukauskas, Andrea Mifsud and Iain Sedgwick, *Member, IEEE*

**Abstract**—This paper describes a serialiser system tailored for scientific CMOS image sensor applications. The system first structures the data using the Xilinx Aurora 64B/66B protocol, thus enabling rapid system startup, skew compensation without an additional clock output, and transmission of several types of packets. The data is passed on to a 32-to-1 multiplexer working at a maximum data rate of 5 Gbps which is followed by a CML driver for off-chip data transmission. Including a PLL for timing signals generation, the system has been implemented in a 180 nm CMOS Image Sensor technology, occupying 0.265 mm<sup>2</sup> of silicon area, and consuming 185 mW at 5 Gbps. The PLL allows clock signal generation within a frequency range of 1.25 GHz and 5.0 GHz. The full system was successfully tested at data rates ranging from 2.5 Gbps to 5.0 Gbps, with measured bit error rates better than  $10^{-15}$  for data rates up to 4.3 Gbps and  $10^{-13}$  for 5 Gbps.

**Index Terms**—Serialiser, Phase locked loop, Transmitter, CMOS Image Sensor, CML, Line Driver, Aurora protocol, high speed, current-mode multiplexer

## I. INTRODUCTION

**D**ATA volumes in all types of application have been rising for many years, posing a significant challenge for the transmission and storage of this data. To overcome this issue, considerable research in recent years has been devoted to increasing the data rate of CMOS wireline transceivers, which are now capable of extremely high data rates [1], [2].

Modern scientific detectors are also generating extremely high data rates, and this presents a major practical issue as this enormous amount of data needs to be transmitted off the sensor and stored or processed. This issue is typically resolved by dividing the data stream into a number of lanes across the device, thus relaxing the requirements on a single lane. This is typically done using LVDS or other variants of this interface with data rates ranging between 400 Mbps and 2 Gbps per lane [3]–[6]. The problem arises when the number of LVDS lanes required becomes considerable, with 100 lanes or more needed to achieve data rates in excess of 100 Gbps [7], [8]. In scientific

applications in particular, image sensors are often used in environments in which it is preferable to reduce the number of connections between the sensor and its Data Acquisition System (DAQ). This is needed for example, if the sensor is operated in a vacuum, or in a high radiation environment. In order to comply with this need, the number of data lines has to be reduced considerably. This can be done by either reducing the overall data rate of the detectors (through zero suppression, region-of-interest readout or other compression scheme) or by increasing the maximum data rate allowed for every lane. The latter solution is necessary for fields where data compression is not possible or not tolerated by users.

This can present a problem for scientific detectors however, since, for reasons of cost effective area coverage or the use of specialised technologies, these are typically designed in older process nodes [9]–[11], whereas the fastest serialisers use the most recent process nodes - below 28nm is common, and as low as 7nm is reported [12]. Serialisers for scientific detectors at older nodes have, of course, been developed in the past [13]–[17], but tend to run at low data rates, implement custom protocols or still require relatively modern nodes which are incompatible with some detector specific technologies. The 180nm node in particular remains very popular for these applications [18]–[21]. In this work, therefore, we will focus on the design of a high data rate (up to 5Gbps) serialiser for use in these 180nm CMOS Image Sensor technologies.

In this paper we describe the system, its design, and present silicon verified measurements of a test structure with dummy data. The paper is organised as follows: Section II introduces the concept of the serialiser including its main building blocks; Section III details the circuit implementation; Section IV presents the fabricated system; Section V presents silicon verified measurements and Section VI concludes this work.

## II. SYSTEM CONCEPT

The system concept of the serialiser and how it fits in the top-level architecture of a CMOS image sensor are illustrated in Fig. 1. The general idea (Fig. 1a) is to have a number of serialisers spread around the sensor periphery to satisfy its frame rate requirements. In this implementation (Fig. 1b) the serialiser takes 64-bits of data as an input. This can be adjusted on a per application basis. To allow easy integration with DAQ systems based on modern FPGAs, the design implements the 64B/66B Aurora protocol provided by Xilinx [22]. This protocol uses 64B/66B encoding, with an extra two header bits used to define the type of block (or packet) sent over the channel (a packet is a single set of 64 bits sent at once,

A. Mifsud was with the CMOS Sensor Design Group at Science and Technology Facilities Council, Rutherford Appleton Laboratory, Harwell Campus, Didcot, OX11 0QX, U.K.. He is now with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: a.mifsud@imperial.ac.uk).

I. Sedgwick, D. Krukauskas, S. Benhammadi, and N. Guerrini are with the CMOS Sensor Design Group at Science and Technology Facilities Council, Rutherford Appleton Laboratory, Harwell Campus, Didcot, OX11 0QX, U.K. (e-mail: iain.sedgwick, deividas.krukauskas, seddik.benhammadi, nicola.guerrini@stfc.ac.uk).

Manuscript received December 5, 2021.

For the purpose of open access, the author(s) has applied a Creative Commons Attribution (CC BY) license to any Accepted Manuscript version arising

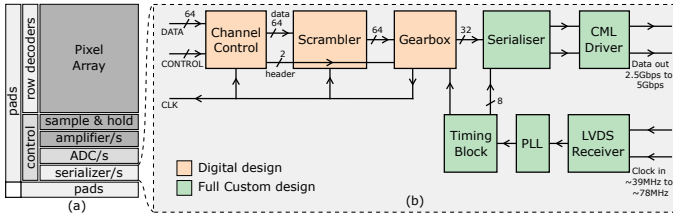


Fig. 1. System concept showing: (a) a typical architecture of a symmetrical high data rate image sensor; (b) a detailed block diagram of the system presented in this work.

referred to as a block in [22]). There are ten types of blocks which can be split into two main categories: (1) data packets, which are used to send information over the channel and (2) control packets, which are used to extend the features available to the channel. Some of these features include: (1) idle packets, which are not visible to the user on the other end, (2) clock compensation packets, which are used by the receiving end to compensate for any clock skew in real time, and (3) separator packets, which can be used as a signal to the user that the end of a frame has been reached (a frame is a number of packets). These features are particularly useful in image sensor applications as: (1) serialisers can run continuously, avoiding any significant impact on the supply voltages; (2) the on-chip clock signal does not need to be transmitted to the receiving end, thus requiring only two differential pins for the data; (3) separator packets can be used to segment image sensor into frames. Another important aspect is that this protocol can accommodate data rates up to 25 Gbps for high end FPGAs [23], thus facilitating scaling up the data rate further in the future.

The serialiser which implements the 64B/66B Aurora protocol requires several blocks. Firstly, there is a control block which initialises the channel, and adds the two control bits to the 64-bit data packets. This is then passed on to the scrambler block whose function is to ensure DC balance of the datastream by restricting the length of consecutive 1s or 0s. It is more convenient for the design of the last stage of the serialiser if the number of bits to transmit is presented in blocks equal to  $2^N$ , where  $N$  is in the range of  $[0..6]$ . In this work therefore, a gearbox architecture is placed between the scrambler and the final serialiser stage. This takes a 66-bit output from the scrambler and converts it into 32-bits blocks which can be easily serialised by a binary architecture. As a result, the input to the scrambler is paused for half a clock cycle every 33 cycles to cater for the accumulated 2 bits from the previous 32 cycles. The final stage is a 32-to-1 serialiser, followed by a CML driver to send data to the external DAQ. The clock and other control signals are generated using a Phase Locked Loop (PLL) and a timing block respectively.

### III. CIRCUIT IMPLEMENTATION

The system presented in this paper is a single channel implementation of an Aurora based serialiser as the first step towards a multi-channel implementation. It is designed as a test structure with access to several parts of the design to enable in-silicon testing and verification of several parts of the IP.

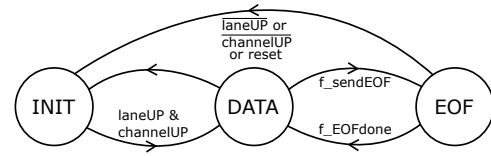


Fig. 2. State machine implemented in the channel control Verilog block.

During the design of this chip the architecture shown in Fig. 1 was split into two main blocks: (i) Low speed circuits that were implemented using a digital design flow; and (ii) High speed circuits that make use of a full-custom analogue design flow. Both blocks were integrated in a single chip using an analog-on-top design flow.

#### A. Low speed circuits

These consist of the channel control block, scrambler and gearbox, which were all designed using Verilog, and then mapped to a standard cell based layout using a digital design flow.

1) *Channel Control*: The channel control block was designed to implement the sequence of control and data packets specified in the Aurora protocol. The control packets aid the receiver in: (1) aligning to the start / end of each block, and (2) compensate for the skew between the lanes in the case of a multi-lane architecture. To initialise a channel, a particular sequence of control packets is repeated until two input control signals (shown as laneUP and channelUP in Fig 2) go high. This means that the logic on the receiving end of the transmission is ready to start receiving data, and the control block moves to the next state, DATA. The last state, EOF, is used to send a separator packet (also called End Of Frame packet). This is a control packet which can contain up to 7 bytes of information, for instance serialiser address, and row number. This packet is sent after every  $R$  data packets, defining a frame in the Aurora 64B/66B protocol.

2) *Scrambler*: The scrambler is simply a 58-bit Linear-Feedback-Shift-Register (LFSR) with a feedback polynomial implemented using XOR gates (Fig. 3). The feedback polynomial, defined in the 64b/66b Aurora protocol specification [22], is:

$$G(x) = 1 + x^{39} + x^{58} \quad (1)$$

where  $G(x)$  is the result, and  $x^k$  is the bit at location  $k$  in LFSR containing 58 bits, each of which is initially set to 0. Thus, in case of a serial scrambler, this would result in scrambled data being generated as:

$$SD[t] = D[t] \oplus LFSR[38][t] \oplus LFSR[57][t] \quad (2)$$

where  $SD$  is the scrambled data bit,  $D$  is the input data bit to LFSR,  $LFSR[38]$  is the 39<sup>th</sup> bit of the LFSR and  $LFSR[57]$  is the 58<sup>th</sup> bit of the LFSR at any given time  $t$ . Unfolding equation 2 as a function of time enables it to be parallelised [24]. For instance, time points  $[t+1]$  and  $[t+2]$  result in equations 3 and 4 respectively.

$$\begin{aligned} LFSR[38][t+1] &= LFSR[37][t], \\ LFSR[57][t+1] &= LFSR[56][t]; \end{aligned} \quad (3)$$

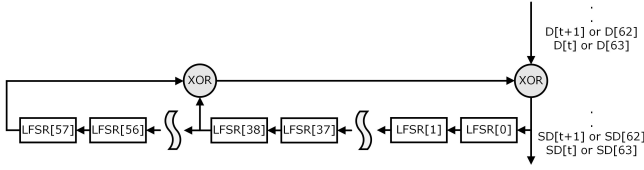


Fig. 3. A block diagram of a serial scrambler (1 bit per clock cycle).

$$\begin{aligned} LFSR[38][t+2] &= LFSR[36][t], \\ LFSR[57][t+2] &= LFSR[55][t]; \end{aligned} \quad (4)$$

Although the clock frequency at this stage is low enough to enable the scrambler to unfold equation 2, it is too fast for serial scrambling to meet overall system requirements. As a result, scrambling is performed in a single clock cycle rather than 64 clock cycles, by using an XOR function of LFSR[38], LFSR[57] and data input values that would be present during a given clock cycle. For example,

$$\begin{aligned} SD63 &= D63 \oplus LFSR[38] \oplus LFSR[57], \\ SD62 &= D62 \oplus LFSR[37] \oplus LFSR[56], \\ SD61 &= \dots \end{aligned} \quad (5)$$

3) *Gearbox*: The gearbox is the last stage of the low speed circuitry. It is needed in order to translate the 66 bit wide output of the scrambler block into chunks whose width is a power of two (32 bits wide in our case). A width equal to a power of 2 is desirable because it simplifies the design of the following block of the serialiser. Since this is a high speed block, simplification of its design leads to improved performance and data rate. The width of the gearbox buffer is selected as follows. The least common multiple of the 66 bit input width and the 32 bit output width is 1056. However, a 1056 bit buffer would be very large, and we plan for a situation where multiple serialisers exist on one chip, so reducing silicon area is important. An alternative to this long buffer is to implement it as a circular buffer of width 132, similar to [24].

In such a circular buffer, the buffer must be loaded and read out multiple times before a whole cycle is completed and the input and output side are re-aligned. In our case 16 input cycles are needed -  $16 \times 66 = 1056$ , and correspondingly, 33 output cycles are needed -  $33 \times 32 = 1056$ . A modulo counter on either side of the buffer is used to address the relevant chunk of the data, so that the same bit is not read and written simultaneously. This is shown in Figure 4 and the gearbox operates as follows:

- Data enters the serialiser in 66 bit wide chunks at the input frequency  $F_{in}$
- The data passes through a multiplexer driven by a modulo 2 counter. This places data alternately in the first 66 and last 66 bits of a 132 bit internal storage register.
- Concurrently, data is read from 32 bit chunks of the register, with addresses selected by a modulo 33 counter. Since 32 and 132 are not integer multiples, this leads to some cases where the 32 bit chunk concatenates the end of the buffer with the beginning. This data is temporally continuous since if the end of the buffer contains data

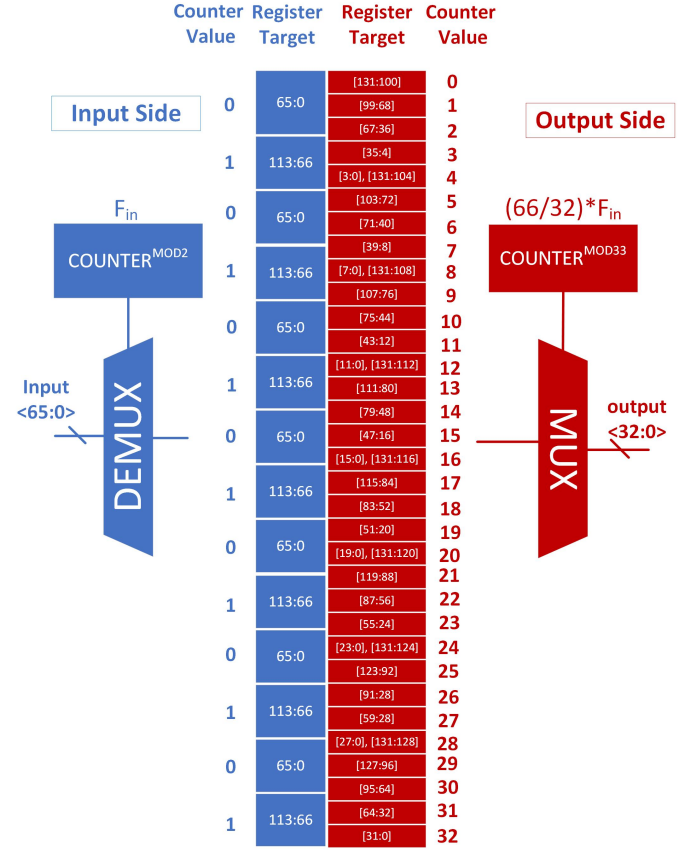


Fig. 4. Outline of the gearbox, showing the input 66 bit bus side in blue, and the output 32 bit bus side in red.

from load cycle N, the start will contain data from load cycle N+1.

- Once 16 input cycles and 33 output cycles are complete, the counters are again aligned, and the process repeats.

### B. High speed circuits

Following the gearbox, we have a serialiser stage that is split into two sub-stages, and then a CML driver which drives the pads and the external routing beyond.

1) *Serialiser*: The serialiser is split in two stages, a 32-to-8 stage, followed by an 8-to-1 stage as illustrated in Fig. 5a. Given that the clock frequency of the first stage is less than 1 GHz, static scan flip-flops were used in a parallel-in serial-out configuration. On the other hand, emphasis was put on routing to reduce parasitic capacitance and resistance, and avoid any mismatches between all clock paths. The stage that follows is an 8-to-1 current-mode multiplexer with active inductors to speed up the transitions [25], [26]. Instead of using a single select transistor, the circuit uses two MOSFETs, enabling the pulse width of the select signals to be larger than a single bit time, which relaxes the requirements for the timing block. The multiplexer (refer to Fig. 5c) also converts the CMOS rail-to-rail data signal to a differential signal with lower swing to prevent the subsequent stage, the CML driver, from operating out of saturation.

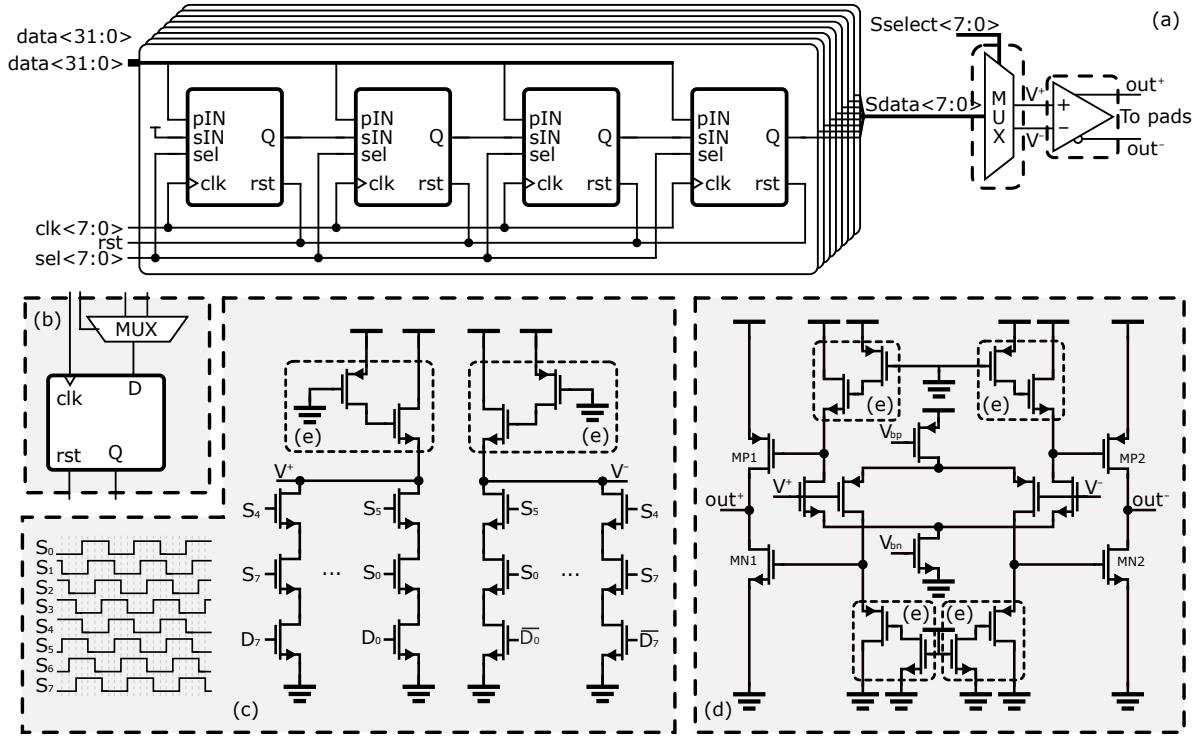


Fig. 5. Analogue circuitry of the two serialiser stages showing: (a) a top-level circuit; (b) a scan flip flop used for the 32-to-8 serialiser; (c) the 8-to-1 current-mode multiplexer and its timing signals; (d) the current mode line driver; (e) active inductors used in the design to sharpen the transitions.

2) *CML driver*: The chosen configuration for the CML driver is that of a class AB device [25], with a symmetrical input stage that feeds differentially to a push-pull output stage architecture as shown in Fig. 5d. Just like the 8-to-1 multiplexer, it makes use of active inductors on both ends to sharpen the signal transition. This also has the added benefit of keeping the transistors in the output stage ( $MP1, MP2, MN1, MN2$ ) in saturation as the voltage at the gate will always be higher than  $V_{DD} - V_{TH}$  and  $V_{TH}$  for the PMOS and NMOS transistors respectively.

3) *PLL*: The PLL shown in Fig. 6 was added to the system to synthesise a new clock at the correct frequency for the required data rate. Using a PLL avoids the need for designing a multi-GHz clock receiver. In this case, for 5 Gbps, a 2.5 GHz clock is required. To mitigate the frequency limitations of a typical single-loop voltage controlled oscillator, a 3-stage VCO was used, where each stage is made up of a dual-loop Park-Kim delay cell [27]. Additionally, the first 2 stages of the frequency divider make use of dynamic flip-flops due to the range of frequencies in use. These are then followed by 3 additional static flip-flops. Post-layout extracted simulations show that the PLL can operate for clock outputs within a frequency range of 1.6 to 4.0 GHz which is ideal for this application.

4) *Timing block*: The timing block takes the PLL's output clock and a reset signal as its inputs, and synthesizes the select signals in Fig. 5c. This is done by progressively dividing the clock by 2, and generating true and complementary versions. Thanks to the relative logic delay introduced between the true and complementary versions at each step, this leads to a set of 8 signals, offset from each other by a value approximately

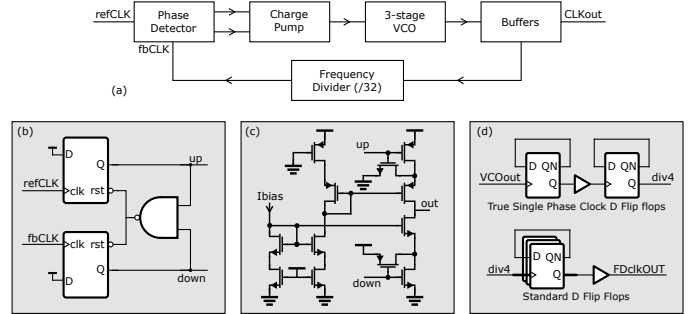


Fig. 6. The PLL block showing: (a) a top-level block diagram; (b) the phase detector; (c) the charge pump; (d) the frequency divider making use of TSPC flip flops.

equal to the logic delay of a single gate. Given that the typical FO4 of a 180nm process is between 80 - 100 ps [28], [29], this gives a bit period appropriate for speeds around 5Gbps.

The advantage of this scheme over alternatives such as a DLL is its simplicity and lower power consumption. The disadvantage is that the bit period, and thus the width of the eye, is set by the logic delay. Slowing down the clock does not lead to the eye growing in width - it remains the same size even at slower speeds. Improving the eye at slower speeds would require increasing the length of the logic delay. Mis-match in the elements also leads to a closing of the eye.

To allow some tuning of the delay, the delay element includes a transmission gate, so that the delay of one of the branches can be adjusted by varying the gate voltages. This is shown in Fig. 7.



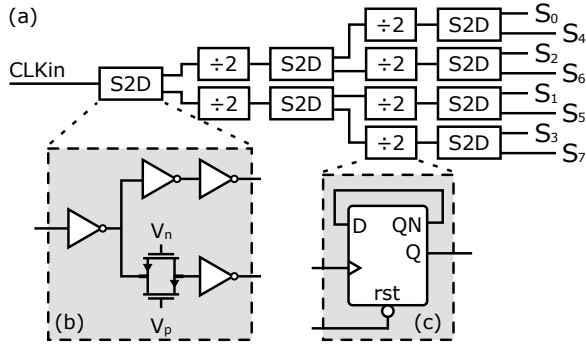


Fig. 7. Timing block circuitry showing: (a) a top-level block diagram; (b) a single-ended to differential converter circuit; (c) a frequency divide by two circuit using a dynamic flip-flop.

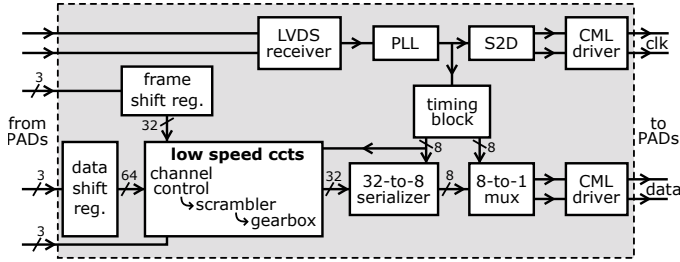


Fig. 8. Block diagram of the fabricated test structure. It includes two paths: (top) clock path; (bottom) data path.

#### IV. FABRICATED DEVICE

The high speed readout system has been implemented in the TowerJazz  $0.18\mu\text{m}$  6-metal CMOS image sensor technology. The low speed digital circuits were synthesized and implemented with the digital design flow. The system was configured and fabricated in a test structure as a complete system where both the data path, and the clock path can be checked as shown in Fig. 8. Fig. 9 shows a microphotograph and a layout with an annotated floorplan of the fabricated test structure.

#### V. MEASURED RESULTS

The fabricated structure was tested in two configurations to characterise both the PLL and the serialiser structures within the test setup available, which is illustrated in Fig. 10. A re-timer, DS110DF111, was used on the COB (Chip-On-Board) to reduce the track length driven by the CML driver to around 1 cm, including a through via. This enabled the authors to focus on verifying the operation and limits of the serialiser and PLL internal circuits whilst minimising external effects.

##### A. PLL

The PLL can be tested separately from the complete serialiser by wire-bonding the die in a special arrangement. This arrangement connects a CML driver which is driven by the PLL directly to the input of the re-timer on the PCB. The re-timer itself was used to check that the PLL was oscillating at the programmed frequency. This has the added advantage of making the whole process automated. The data rate at which the re-timer is locked can be set via a serial interface. Initial

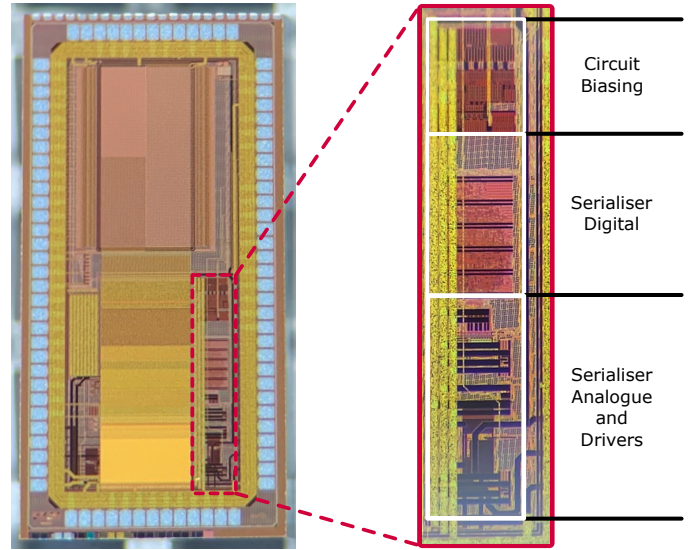


Fig. 9. Fabricated high speed serialiser in a  $0.18\mu\text{m}$  CMOS image sensor technology. Shown are: (a) micrograph of the full test structure, including CIS; (b) layout and annotated floorplan of the serialiser and associated circuitry.

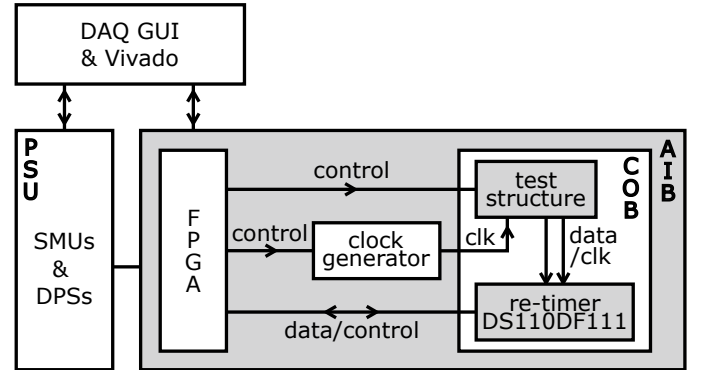


Fig. 10. Test setup used to verify operation of test structure: COB - Chip On Board; AIB - Application Interface Board; SMU - Source Measurement Unit; DPS - Digital Power Supply.

testing showed a failure to lock at the intended frequencies. Several parameters controlling the PLL were therefore swept in order to locate the cause of the issue. The swept parameters were the following:

- 1) VDD\_PLL\_A18 - Analog Voltage Supply of PLL VCO
- 2) VDD\_PLL\_D18 - Digital Voltage Supply of PLL internal circuitry
- 3) IBIAS\_PLL - PLL charge pump current

Varying these parameters provided an essential tool to check which circuit / IP was the limiting factor at that particular output frequency. Shmoo plots resulting from these tests are shown in Fig. 11. It is evident from these results that as the frequency increases, the analog VCO supply requires an increase in voltage. A similar effect occurs to a lesser extent on the digital PLL supply. Upon investigation it was found that this was due to supply and ground track resistance between the pads and the PLL, totalling  $16\Omega$ ,  $16\Omega$  and  $3\Omega$  for VDD\_PLL\_A18, VDD\_PLL\_D18 and VSS respectively. Increasing frequency was therefore leading to an increased current and increased

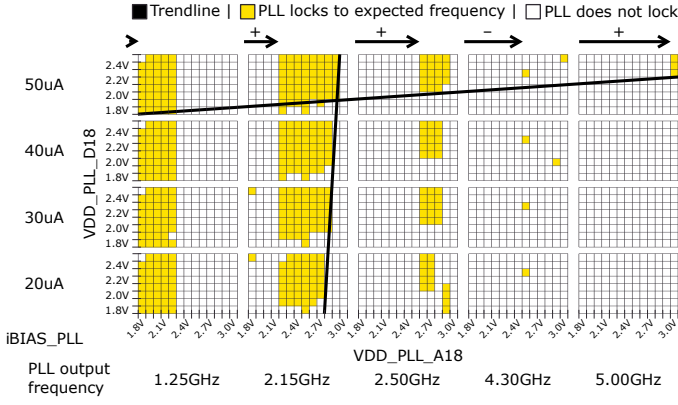


Fig. 11. Shmoo plots of the PLL's response under different conditions for VDD\_PLL\_A18, VDD\_PLL\_D18, iBIAS\_PLL and expected PLL output frequency. Yellow implies that the PLL's output was locked to the expected frequency whilst white implies that this was not successful.

TABLE I

EXPECTED INTERNAL VOLTAGES VALUES FOR DIFFERENT FREQUENCIES.

Freq. (GHz)	Ext. VDDA (V)	Ext. VDDD (V)	VDDD Cur. (mA)	VDDA Cur. (mA)	VSS Cur. (mA)	Int. VDDD (V)	Int. VDDA (V)
1.25	2.2	1.8	0.6	13.0	56.8	1.6	1.8
2.15	2.2	1.9	0.9	18.5	62.4	1.7	1.7
2.50	2.5	2.1	1.2	24.0	68.1	1.9	1.9
4.30	2.5	2.3	1.9	26.1	70.4	2.1	1.9
5.00	3.0	2.3	1.4	31.4	75.7	2.0	2.3

$$R_{VDDD} = R_{VDDA} = 16\Omega; R_{VSS} = 3\Omega$$

Ext. - External; Int. - Internal;

VDDA - VDD\_PLL\_A18; VDDD - VDD\_PLL\_D18

supply voltage droop, requiring an increase in the corresponding supply to compensate for the higher droop as listed in Table I. From these results it can be seen that, even though the external analog supply is increasing, the estimated internal value remains constant at approximately the nominal operating range of the 1.8V devices until around 4.3GHz, validating the theory that the power supply routing is the limiting factor. However, this is not the case for the digital supply, given that it exceeds this range at frequencies lower than 4.3GHz. This suggested that a different limitation existed in the digital circuit. Upon further investigation, it was found that the output of the PLL was connected to a track of significant length (in the mm range) which led to the deterioration of this high speed clock as it travelled to the following block. In the tests reported in this paper, these layout issues were overcome using the boosted power supplies already described, and in future chips they can be easily rectified by more carefully routing these supplies and signals.

It is worth noting that during all these sweeps, the CML driver had its own (separated) supply held constant at 1.8V. From the results, we can therefore conclude that the driver is capable of transmitting an alternating signal up to a maximum of 5GHz under certain conditions (rail-to-rail input, track characteristics, wire bond, alternating input, etc).

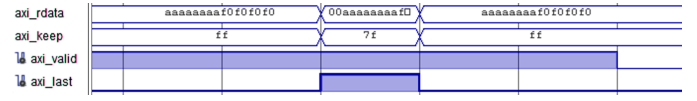


Fig. 12. Internal FPGA signals during normal operation. These have been accessed, viewed and triggered through an internal instantiated Xilinx Vivado debug core.

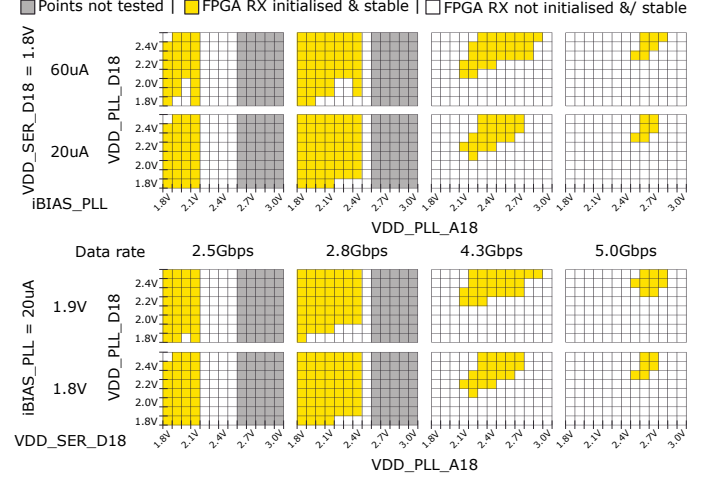


Fig. 13. Shmoo plots of the serialiser's response under different conditions for VDD\_PLL\_A18, VDD\_PLL\_D18, VDD\_SER\_D18, iBIAS\_PLL and expected serialiser output data rate. Yellow implies that the Aurora receiver in the FPGA has gone through the initialisation procedure and received data for a few  $\mu$ s, whilst white implies that this was unsuccessful. Grey areas have not been tested.

## B. Serialiser

After the functionality of the PLL had been verified, operating conditions established and the reasons for the boosted power supplies understood, testing moved on to the serialiser. This was divided into two categories; (1) functional testing, where the Aurora state machine and other associated logic is verified and (2) bit-error-rate testing, where the level of bit errors is recorded for a given input pattern.

1) *Functional Testing*: The serialiser was tested at a data rate of 2.5Gbps to verify the functionality of the channel. Initialisation was judged to be successful upon acknowledgment from the receiver (inside the FPGA) that the channel had been initialised successfully. End-of-frame packets were also generated at the transmitter and received and decoded by the FPGA as demonstrated by signal axi\_last in Fig. 12. This was followed by testing at various data rates to check that the Aurora 64B/66B initialisation sequence was successful and that the channel was stable over a few  $\mu$ s. As for the PLL tests, this was done to get a snapshot of the system, and use the parameters swept to understand the limitations of the circuitry as the data rate increased. This is illustrated in Fig. 13. We can observe that the points at which the channel and receiver are stable are a subset of those in Fig. 11 for which the PLL was stable. For more accurate information on the stability and robustness of the channel, BER testing was then performed for a number of points and data rates.

2) *BER Testing*: For test purposes, the chip contains a 64 bit register which can be loaded with a test pattern. During

TABLE II  
MEASURED BIT ERROR RATE AT DIFFERENT DATA RATES.

Data Rate (Gbps)	2.5	2.8	4.3	5
IBIAS_PLL	20 $\mu$ A	20 $\mu$ A	20 $\mu$ A	20 $\mu$ A
VDD_PLL_A18	1.8 V	2.2 V	2.60 V	2.80 V
VDD_PLL_D18	1.8 V	2.0 V	2.45 V	2.60 V
VDD_SER_D18	1.8 V	1.8 V	1.80 V	2.00 V
# of packets	5.20E+13	1.53E+12	2.35E+12	6.82E+12
# of correct bits	3.30E+15	9.78E+13	1.50E+14	4.37E+14
# of incorrect bits	0	0	0	81
BER	< 3E-16	< 1E-14	< 7E-15	1.85E-13

transmission, the FPGA checks if the loaded pattern is correctly received, and from this information the BER can be calculated. The results of this BER testing are noted in Table II. These numbers were all calculated inside the receiving FPGA based on the incoming data stream. Additionally, an eye measurement was performed at several data rates to get a better understanding of the output at various data rates. Fig. 14 shows the eye diagram information obtained from the re-timer present on the COB at a selection of data rates. Clean eyes are achieved at all rates. The eye can be seen to be off centre and not to have grown above 150ps at 2.8Gbps. This is due to the width of the eye being controlled by the logic delay as explained in section III-B4.

## VI. CONCLUSION

This paper has presented a fully-integrated serialiser system designed for scientific detectors in 180nm processes. The serialiser combines the Xilinx Aurora 64B/66B interface with high speed circuitry to reach data rates up to 5 Gbps in a 180 nm CIS technology.

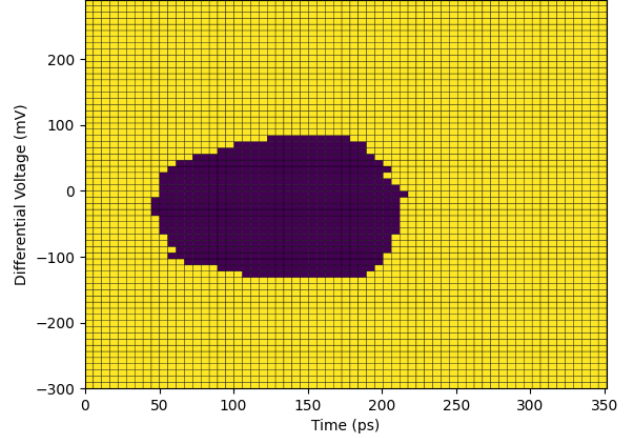
Fabricated as a test structure in a CIS 180 nm technology, the system occupies 0.265 mm<sup>2</sup> and consumes 185 mW of power at a data rate of 5 Gbps. The PLL's output frequency range of operation was verified to be between 1.25 GHz and 5.0 GHz with increasing voltage supplies (as the operating frequency increases) to compensate for significant IR losses in the supply routing. The full system was confirmed to operate at data rates ranging from 2.5 Gbps to 5.0 Gbps, with measured bit error rates better than 10<sup>-15</sup> bits up to 4.3 Gbps and 10<sup>-13</sup> at 5 Gbps.

Ongoing work is focusing on: (1) improving the layout integration of all individual blocks in the system reducing IR losses and RC constants, (2) developing additional features to aid the DAQ on the receiving end of the CIS, and (3) integrating multiple instances of this system within a single chip to produce a multi-lane system.

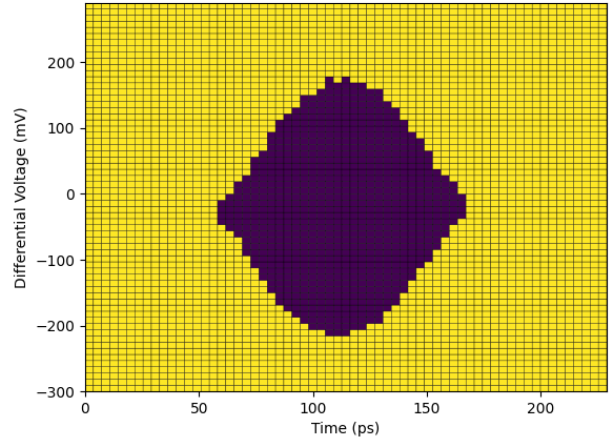
## ACKNOWLEDGMENTS

The authors would like to thank Thomas Gardiner for useful discussions and advice on the serialiser design. They would also like to thank aSpect Systems GmbH for the design and delivery of the test system.

Eye Diagram for Link Running at 2.8Gbps



Eye Diagram for Link Running at 4.3Gbps



Eye Diagram for Link Running at 5.0Gbps

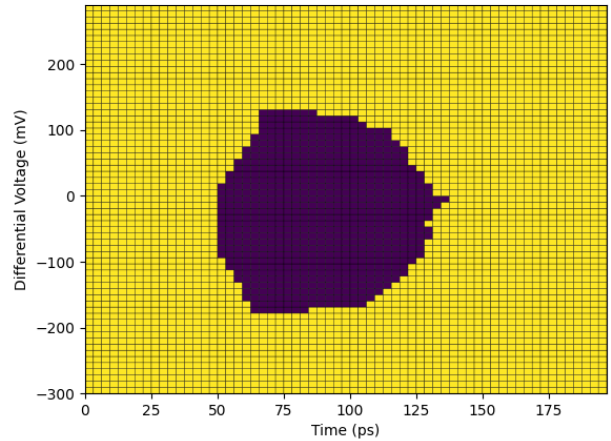


Fig. 14. Eye diagram from re-timer, DS110DF111, for link running at 2.8, 4.3 and 5 Gbps.

## REFERENCES

- [1] B.-J. Yoo, D.-H. Lim, H. Pang, J.-H. Lee, S.-Y. Baek, N. Kim, D.-H. Choi, Y.-H. Choi, H. Yang, T. Yoon, S.-H. Chu, K. Kim, W. Jung, B.-K. Kim, J. Lee, G. Kang, S.-H. Park, M. Choi, and J. Shin, "6.4 A 56Gb/s 7.7mW/Gb/s PAM-4 Wireline Transceiver in 10nm FinFET Using MM-CDR-Based ADC Timing Skew Control and Low-Power DSP



- with Approximate Multiplier,” in *2020 IEEE International Solid-State Circuits Conference - (ISSCC)*, 2020, pp. 122–124.
- [2] P. Upadhyaya, C. F. Poon, S. W. Lim, J. Cho, A. Roldan, W. Zhang, J. Namkoong, T. Pham, B. Xu, W. Lin, H. Zhang, N. Narang, K. H. Tan, G. Zhang, Y. Frans, and K. Chang, “A Fully Adaptive 19–58-Gb/s PAM-4 and 9.5–29-Gb/s NRZ Wireline Transceiver With Configurable ADC in 16-nm FinFET,” *IEEE Journal of Solid-State Circuits*, vol. 54, no. 1, pp. 18–28, 2019.
  - [3] R. Aerts, S. Domer, J. Fong, K. Vanda, R. Rubacha, B. Harris, T. Geurts, M. Innocent, and R. Aerts, “1.4Gpix/s, 2.9Mpix CMOS image sensor for readout of holographic data memory,” in *International Image Sensor Workshop, IISW*, 2007, p. 4.
  - [4] P. Centen, S. Lehr, S. Roth, J. Rotte, F. Heizmann, A. Momin, R. Dohmen, K.-H. Schaaf, K. J. Damstra, R. van Ree, and M. Schreiber, “A 4e-noise 2/3-inch global shutter 1920x1080P120 CMOS-Imager,” in *International Image Sensor Workshop, IISW*, 2013, p. 4.
  - [5] G. Meynants, X. Wu, S. Van hoogenbemt, T. De Ridder, P. De Dit, K. Ruythooren, and K. Van Esbroeck, “A 700 frames/s 2MPixel global shutter image sensor with 2Me- full wekk charge and 12um pixel pitch,” in *International Image Sensor Workshop, IISW*, 2015, p. 4.
  - [6] J. Rotte, A. Defernez, R. Visser, R. van Ree, K. Damstra, P. Centen, A. Birman, D. Veinger, and S. Louwsma, “A 2.5 $\mu$ m 9.5 Mpixel high frame rate CMOS imager with hybrid output multiplexer and 58Gb/s datarate,” in *International Image Sensor Workshop, IISW*, 2019, p. 4.
  - [7] T. Arai, T. Yasue, K. Kitamura, H. Shimamoto, T. Kosugi, S. Jun, S. Aoyama, M. Hsu, Y. Yamashita, H. Sumi, and S. Kawahito, “A 1.1- $\mu$ m 33-mpixel 240-fps 3-d-stacked CMOS image sensor with three-stage cyclic-cyclic-sar analog-to-digital converters,” *IEEE Transactions on Electron Devices*, vol. 64, no. 12, pp. 4992–5000, 2017.
  - [8] A. Krymski, “Image Sensor with Panel Readout and Serialization using Multiple PLLs,” in *International Image Sensor Workshop, IISW*, 2019, p. 4.
  - [9] G. Contin, “The maps-based its upgrade for alice,” 2020. [Online]. Available: <https://arxiv.org/abs/2001.03042>
  - [10] A. Marras, J. Correa, S. Lange, V. Vardanyan, T. Gerhardt, M. Kuhn, F. Krivan, I. Shevyakov, M. Zimmer, M. Hoesch, K. Bagschik, F. Scholz, N. Guerrini, B. Marsh, I. Sedgwick, G. Cautero, D. Giurelli, G. Iztok, R. H. Menk, M. Scarcia, L. Stebel, T. Nicholls, W. Nichols, U. K. Pedersen, P. Shikhaliev, N. Tartoni, H. Hyun, S. Kim, K. Kim, S. Rah, A. Dawiec, F. Orsini, G. Pinaroli, A. Greer, S. Aplin, A. D. Jewell, T. J. Jones, S. Nikzad, M. E. Hoenk, F. Okrent, H. Graafsma, and C. B. Wunderer, “Characterization of the Percival detector with soft X-rays,” *Journal of Synchrotron Radiation*, vol. 28, no. 1, pp. 131–145, Jan 2021. [Online]. Available: <https://doi.org/10.1107/S1600577520013958>
  - [11] A. Allahgholi, J. Becker, A. Delfs, R. Dinapoli, P. Göttlicher, H. Graafsma, D. Greiffenberg, H. Hirsemann, S. Jack, A. Klyuev, H. Krüger, M. Kuhn, T. Laurus, A. Marras, D. Mezza, A. Mozzanica, J. Poehlsen, O. Shefer Shalev, I. Sheviakov, B. Schmitt, J. Schwandt, X. Shi, S. Smoljanin, U. Trunk, J. Zhang, and M. Zimmer, “Megapixels @ Megahertz – The AGIPD high-speed cameras for the European XFEL,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 942, p. 162324, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168900219309106>
  - [12] J. Im, K. Zheng, C.-H. A. Chou, L. Zhou, J. W. Kim, S. Chen, Y. Wang, H.-W. Hung, K. Tan, W. Lin, A. B. Roldan, D. Carey, I. Chlis, R. Casey, A. Bekele, Y. Cao, D. Mahashin, H. Ahn, H. Zhang, Y. Frans, and K. Chang, “A 112-Gb/s PAM-4 Long-Reach Wireline Transceiver Using a 36-Way Time-Interleaved SAR ADC and Inverter-Based RX Analog Front-End in 7-nm FinFET,” *IEEE Journal of Solid-State Circuits*, vol. 56, no. 1, pp. 7–18, 2021.
  - [13] C. Grace, E. Fong, D. Gnani, T. Stezelberger, and P. Denes, “A 24-Channel Digitizer With a JESD204B-Compliant Serial Interface for High-Speed Detectors,” *IEEE Transactions on Nuclear Science*, vol. 68, no. 4, pp. 426–433, 2021.
  - [14] M. Idzik, K. Swientek, T. Fiutowski, S. Kulis, and D. Przyborowski, “A 10-bit multichannel digitizer ASIC for detectors in particle physics experiments,” *IEEE Transactions on Nuclear Science*, vol. 59, no. 2, pp. 294–302, 2012.
  - [15] Q. Sun, C. Hu-Guo, K. Jaaskelainen, I. Valin, X. Fang, Y. Zhang, M. Winter, and Y. Hu, “The development of on-chip serial link transmitter for maps,” *IEEE Transactions on Nuclear Science*, vol. 57, no. 2, pp. 543–549, 2010.
  - [16] J. Wang, L. Guan, Z. Sang, J. Chapman, T. Dai, B. Zhou, and J. Zhu, “Characterization of a serializer ASIC chip for the upgrade of the ATLAS muon detector,” *IEEE Transactions on Nuclear Science*, vol. 62, no. 6, pp. 3242–3248, 2015.
  - [17] T. Gardiner, “An Ultra-Fast 10Gb/s 64b66b Data Serialiser Backend in 65nm CMOS Technology,” in *Proceedings of Topical Workshop on Electronics for Particle Physics — PoS(TWEPP2018)*, vol. 343. SISSA Medialab, 2019, p. 088. [Online]. Available: <https://pos.sissa.it/343/088/>
  - [18] C. E. Assim Boukhayma, Antonino Caizzone, “A cmos image sensor pixel combining deep sub-electron noise with wide dynamic range,” *IEEE Electron Devices Letters*, vol. 41, no. 6, pp. 880–883, 2020.
  - [19] F.-D. Luis Miguel C. Freitas, “A cmos image sensor with 14-bit column-parallel 3rd order incremental sigma-delta converters,” *Sensors and Actuators A: Physical*, vol. 313, 2020.
  - [20] C. Accarino, M. Al-Rawhani, Y. D. Shah, D. Maneuski, S. Mitra, C. Buttar, and D. R. S. Cumming, “Low noise and high photodetection probability spad in 180 nm standard cmos technology,” in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–4.
  - [21] I. Berdalovic, R. Bates, C. Buttar, R. Cardella, N. E. Plaja, T. Hemperek, B. Hiti, J. van Hoorne, T. Kugathasan, I. Mandic, D. Maneuski, C. M. Tobon, K. Moustakas, L. Musa, H. Pernegger, P. Riedler, C. Riegel, D. Schaefer, E. Schioppa, A. Sharma, W. Snoeys, C. S. Sanchez, T. Wang, and N. Wermes, “Monolithic pixel development in TowerJazz 180 nm CMOS for the outer pixel layers in the ATLAS experiment,” *Journal of Instrumentation*, vol. 13, no. 01, pp. C01023–C01023, jan 2018. [Online]. Available: <https://doi.org/10.1088/1748-0221/13/01/c01023>
  - [22] Xilinx, “Aurora 64B/66B Protocol Specification,” Oct. 2014.
  - [23] Xilinx, “Aurora 64B/66B v11.2 LogiCORE IP Product Guide,” Apr. 2018.
  - [24] W. Ruan and Q. Hu, “A 0.18um CMOS transmit physical coding sublayer IC for 100G Ethernet,” *Journal of Semiconductors*, vol. 37, no. 3, p. 035005, mar 2016. [Online]. Available: <https://dx.doi.org/10.1088/1674-4926/37/3/035005>
  - [25] Jean Jiang and Fei Yuan, “A new CMOS class-AB transmitter for 10 Gbps serial links,” in *The 2004 47th Midwest Symposium on Circuits and Systems, 2004. MWSCAS '04.*, vol. 3, 2004, pp. iii–319.
  - [26] J. Jiang and F. Yuan, “A new CMOS current-mode multiplexer for 10 gbps serial links,” *Analog Integrated Circuits and Signal Processing*, vol. 44, no. 1, pp. 61–76, 2005. [Online]. Available: <https://doi.org/10.1007/s10470-005-1615-0>
  - [27] Chan-Hong Park and Beomsup Kim, “A low-noise, 900-mhz VCO in 0.6um CMOS,” *IEEE Journal of Solid-State Circuits*, vol. 34, no. 5, pp. 586–591, 1999.
  - [28] C. Grecu, P. P. Pande, A. Ivanov, and R. Saleh, “A scalable communication-centric SoC interconnect architecture,” in *International Symposium on Signals, Circuits and Systems. Proceedings, SCS 2003. (Cat. No.03EX720)*, 2004, pp. 343–348.
  - [29] A. Stillmaker and B. Baas, “Scaling equations for the accurate prediction of CMOS device performance from 180nm to 7nm,” *Integration*, vol. 58, pp. 74–81, 2017.