A Low Power, High Speed Readout for Pixel Detectors based on an Arbitration Tree

Farah Fahim, Siddhartha Joshi, Student Member, IEEE, Seda Ogrenci-Memik, Senior Member, IEEE, Hooman Mohseni, Senior Member, IEEE

Abstract— A low-power, high-speed arbitration tree for pixel detector readout is presented. The synchronized, binary tree priority encoder establishes a position dependent priority list at the start of every time frame. Pixels that indicate the presence of data for readout, are sequentially granted access to a shared bus for data transfer to the periphery, without the use of an additional global strobe signal. It can be used for either full frame imaging or zero-suppressed readout, in which case it can simultaneously generate the pixel address. To increase the readout frame rate, the pixel array is subdivided into two halves, which allows interleaved latching of data at the output serializer. The design was implemented in a 65 nm LP-CMOS process for the readout of a 64 \times 64 pixel array. Measurement results demonstrate a deadtime-less, full frame imaging rate of \sim 50 kfps, achieved with a dedicated output for every (32 \times 32) 1024 pixels and for a pixel data packet of 11 bits, with no bit errors detected over 1000 frames. The measured energy per bit is 0.94 pJ.

Index Terms—pixel detector readout, arbitration tree, zero suppression, data sparsification, priority encoder

I. INTRODUCTION

HYBRID pixel radiation detectors typically contain a pixelated sensor layer bonded to a pixelated readout integrated circuit (ROIC). They are used for measuring the properties of incoming radiation in a wide range of applications, including particle tracking in high energy physics, medical imaging, focal plane arrays for astronomy etc. As pixel detectors have evolved over the last few decades, the readout of the pixels has itself evolved from simple analog readouts to fast digital readouts enabling higher data frame rates. Typical in-pixel digital measurements include counting the number of incoming photons [1, 2, 3], measuring the time of arrival of the photon [4, 5] or analog-to-digital conversion of accumulated photons [6, 7, 8] within a given time frame or integration window. It is generally desirable for these detectors to be operated continuously without any deadtime such that when new information is being recorded in the current time frame, simultaneously the previous information is being sent off to the data acquisition system. For deadtime-less operation, the time to read out the frame should be less than or equal to the photon detection and processing time frame. Over time, the number of pixels has increased from a few hundred to greater than a few billion [9], and the sensitive area for the entire detector has increased from few mm² to few m² [10]. As the size and area of the detector increases, so does the power consumption, while the full-frame readout rate decreases. It takes more time and energy to transfer data over longer distance due to larger interconnect capacitance from the central areas of the detector to the periphery. Hence, high speed, low power readout architectures are required.

Applications with low pixel occupancy such as photon correlation spectroscopy [11] or quantum cryptography [12] benefit from data sparsification techniques, such as zero-suppression which eliminate zeros by not transmitting data from pixels with no acquired photons. Data driven zero-suppressed readout is a means of reducing the data bandwidth as well as increasing frame rates, provided that the system is able to move the information off-chip at the same rate as it is being produced.

We propose a low-power, high-speed, reconfigurable readout architecture which allows both full frame as well as zero-suppressed read out, based on an arbitration tree for transferring data from pixel to periphery within a user defined time frame. It is capable of achieving deadtime-less, full frame imaging rates of \sim 50 kfps, with a dedicated output for every 1024 pixels and for a pixel data packet of 11 bits, with no bit errors detected over 1000 frames. The measured energy per bit is 0.94 pJ.

The rest of the paper is arranged as follows: in Section II we briefly discuss pixel detector readout techniques, in Section III we explain the need for our approach and the optimizing potential in other designs. In Section IV, we present our design of a synchronized binary tree priority encoder. The detailed pixel level logic and implementation is shown in Section V. Section VI presents the test results, followed by conclusions in Section VII.

II. PIXEL DETECTOR READOUT TECHNIQUES

Pixel detector readouts using ‘time frames’ enable periodic snapshots of data to be recorded, which allows for deadtime-less operation. Each pixel contains two sets of registers: while one set is processing data in the current time frame, another set is transferring data off-chip from the previous time frame. The traditional method used for full frame digital data readout is by daisy-chaining all data storage registers to create a long shift

Manuscript received May 15, 2019; Sept 12, 2019.

This document was prepared by using the resources of the Fermi National Accelerator Laboratory (Fermilab), a U.S. Department of Energy, Office of Science, HEP User Facility. Fermilab is managed by Fermi Research Alliance, LLC (FRA), acting under Contract No. DE-AC02-07CH11359.

F. Fahim is with the ASIC Development Group of the Electrical Engineering Department of the Particle Physics Division at the Fermi National Accelerator Laboratory, PO Box 500, MS 222, Batavia, IL 60510, USA, (telephone: +1 630 840 8423, fax: +1 630 840 2950, farah@fnal.gov)

F. Fahim, S. Joshi, S. Ogrenci-Memik, and H. Mohseni are with the Department of Electrical and Computer Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, USA.
register connected to one or multiple external I/Os and clocked at a high frequency to obtain off-chip data transfer [13, 14]. However, this method has the disadvantage of high-power consumption, both from distributing the readout clock signal to every register and from clocking at high frequency. This method can be modified to allow a frame-based zero-suppressed readout by adding logic which skips a pixel with no data, and adds a ‘flag’ which identifies the position of the skipped pixels [15].

Several methods such as data node based, network based or a combination of these architectures [16,17] can be used to move packet based, zero-suppressed data from the shift register at one end of the column or matrix to the periphery at the opposite end of the ROIC, and subsequently off-chip. The data are essentially moved from the shift register in one pixel to another, possibly with additional registers per group of pixels to alleviate congestion. Several algorithms which give priority to data that has been waiting the longest exist [18], which are easier to implement when the data to be transferred contain timestamps.

One of the simplest methods of data-driven readout is a token passing scheme, where the first pixel to grab the token is enabled to transmit data on a shared bus [19]. The main disadvantage in the token passing scheme is that the readout speed is limited by the time it takes for the token to circle through a given bank of pixels. This limitation can be overcome by adding a fast look-ahead logic [20, 21, 22].

In other readout methods which employ statistical time division multiplexing, transmission nodes are continuously monitoring the availability of the data bus for transmission. To avoid collisions when two nodes try to access the bus at the same time, one node is arbitrarily granted access to the bus. The other node waits for a random time duration before attempting to transmit again [23].

In neuromorphic designs which use address event representation schemes, readout constraints are even more stringent as these do not operate using time frames but instead require an event driven approach. Since a predetermined list of pixels is not available, arbitrators are required at every node to avoid race conditions when new pixels need to transmit data [24, 25, 26].

Although we are discussing readout techniques for pixel detectors, it is potentially applicable to other scenarios which require sparse data readouts. This includes access and transfer of data from content addressable memories [27, 28, 29], sparsifying output in tracking-trigger ASICs [30], population count circuits [31] and networking applications [32].

III. Binary Tree Priority Encoder for Address Generation and Pixel Data Transfer

In zero-suppressed readout techniques the address of the pixel along with its data is required to reconstruct a position dependent map. The significant advantage of a binary tree priority encoder over the techniques discussed in section II is that it can simultaneously generate the pixel address while transferring data with minimal additional circuitry. A binary-tree priority encoder, first implemented in [33], was originally used as an address generator. The concept can be extended to allow the selected pixel to access a common bus for transferring its data to a peripheral data transmitter, as shown in Fig. 1. The binary tree operates on an address-based priority, whereby pixels with higher address have higher priority. When a pixel has valid data, it asserts a request signal readRequest, which propagates through the binary tree, cascading down from the pixel to the periphery. After reaching the root node of the tree, the same signal is propagated back on the reverse path as an asynchronous acknowledge signal selectPixel to select the pixel with the highest priority and simultaneously create the pixel address. However, typically a global, independent, strobe signal readStrobe is required by the periphery to latch the contents of the selected pixel and its address to the data-output register. It subsequently disables the readRequest signal, removing the pixel’s access to the bus.

![Figure 1: Pixels with valid data assert a request signal (readRequest), which propagates through the binary tree. An acknowledge signal (selectPixel) on its reverse path selects the pixel with the highest priority for readout and simultaneously creates the pixel address. A global, independent strobe signal (readStrobe) tells the periphery to latch the contents of the selected pixel to the data-output register and disables the readRequest signal to remove the pixel’s access to the bus.](image-url)

The binary-tree priority encoder has several advantages: the high-speed output data transfer clock is localized to a short serial data-output register in the data transmitter, instead of being distributed across the matrix of thousands of pixels. However, this method has several limitations. Firstly, it uses two entirely independent paths, one for selection of pixel for readout and the other to latch data and deselect the pixel. The delays through these paths are different, hence to maintain data integrity the readStrobe period needs to always be based on the worse case skews and delays. Secondly, there is a large capacitance on the common bus, requiring a long time for data to settle before being valid for readout.
With reference to Fig. 2, data are latched at posedge of readStrobe, and at its negedge, the current pixel’s access to the readout bus is terminated. The next pixel is then selected automatically by the combinational logic of the binary tree. For a given readout speed, based on all the delays through the two paths, once a pixel is given access to the bus, the maximum time available for data to settle is:

\[ t_{setting} = t_p - t_{on} - t_{d1(readStrobe)} - t_{skew1} - t_{d2(selectPixel)} - t_{skew2} \]

where \( t_p \) is the readStrobe period, \( t_{on} \) is when readStrobe is high (typically much longer than a register’s hold time, \( t_{hold} \)), \( t_{d1} \) is the propagation delay for the independent readStrobe from the periphery to the pixel, with \( t_{skew1} \) the pixel position-dependent uncertainty in resetting the pixel, \( t_{d2} \) is the propagation delay of the selectPixel through the arbitration tree, \( t_{skew2} \) is the pixel position dependent uncertainty in selecting the next pixel. \( t_{d1} \) and \( t_{d2} \) are using two different signal networks with different propagation delays. The maximum readout rate is therefore limited by the worst case propagation delay and skew. Moreover, readStrobe and selectPixel are not derived from a single on-chip clock, and use two entirely different propagation paths. For data to be valid, the data bus settling time should be less than or equal to \( t_{setting} \).

The implementation presented in [33] was subsequently optimized in [34] by minimizing the number of transistors required for the logic. The Address-Encoder and Reset-Decoder (AERD) proposed in [35], also uses a binary tree arbiter to generate pixel addresses. Instead of the selectPixel signal, a sync signal is generated by ANDing the readRequest signal with a synchronous clock signal used for pixel selection. The positive edge of the clock is used to latch the address at the periphery. At the negative edge of the clock the pixel is disabled and the next pixel is selected. The disadvantage of this scheme is its clock duty cycle inefficiency, half the clock period is used for pixel selection and the other half for generating the address.

We propose a solution that also does not require a global strobe signal based on a different paradigm [36, 37] by optimally utilizing the clock period, which achieves faster operation as explained in the following section.

### IV. Synchronized Implementation Of Bus Arbitration

A synchronized binary tree priority encoder (SB-PE) has been developed to overcome the challenges presented in section III. In this implementation, the pixel’s access to a shared bus is “synchronized” and entirely controlled by the output data transmitter without requiring a global readStrobe signal.

The concept of the priority encoder is shown in Fig. 3. The SB-PE behaves like a commuter switch selecting the pixels one after another based on their address-dependent priority. A series of synchronous pulses (readOutControl) are sent through the binary tree: the first pulse reaches the pixel with highest priority \( \text{Pix}_a \), the falling edge enables this pixel and the rising edge disables it, the commuter opens the switch with \( \text{Pix}_a \) and closes the switch to \( \text{Pix}_d \), therefore the next pulse reaches \( \text{Pix}_d \) and so on. When all the pixels with data have been readout, the commuter switch defaults to the last pixel in the tree, \( \text{Pix}_z \). However, since it no longer has any valid data, its data output will be ‘0’. \( \text{Pix}_z \) is continuously read out till a frame change occurs and a new priority list is created. Alternatively, it could be gated to stop readout. This eliminates the problem of the two independent data paths selectPixel and readStrobe selecting and transferring the data from the pixel to the data transmitter. Furthermore, the duty cycle of readOutControl signal can be optimized to achieve higher operating speeds.

![Fig. 2 Timing diagram showing data of pixel [N-1] being latched by readStrobe and subsequently pixel [N] being enabled by selectPixel [N] showing the various delays and pixel position-dependent skew. Data are latched at the rising edge of readStrobe and the pixel is subsequently disabled at the falling edge. After the current pixel is disabled, the selectPixel signal enables the next pixel based on its priority.](image)

![Fig. 3 Priority encoder behaving as a commuter switch and splitting the readOutControl signal into pulses for each individual pixel based on the priority list. When all pixels with data have been read out, the switch continuously chooses \( \text{Pix}_z \) (the last pixel in the array).](image)
The uncertainty in pixel selection time is position dependent. The shortest propagation delay is between two adjacent pixels which share the same parent node of the binary tree. Conversely, pixels with no shared path exhibit the longest delay. With reference to Fig. 4, the selection between the 1st and 2nd pixels with hits is an example of shortest delay, while the subsequent selection of the 3rd pixel is an example of the longest delay.

The worst-case uncertainty can be estimated based on the number of levels in the binary tree. Defining a single propagation step between nodes as a “hop”, the worst case selection delay for an array of \(2^N\) pixels requires 2N hops. For a matrix of \(2^{10}\) pixels and a gate delay of \(\sim 100\) ps, this translates to 20 hops, requiring few nanoseconds. Since the readOutControl signal is generated by the data transmitter which also latches the data, the two signals use the same data paths. Moreover, selectPixel is derived from readOutControl and hence their skews are similar. With reference to Fig. 5, the maximum time available for the readout bus to settle is:

\[
\tau_{setting} = \tau_p - \tau_{on} - t_{d2(selectPixel)} - t_{skew2}
\]

where \(\tau_p\) is the readOutControl period, \(\tau_{on}\) is when readOutControl is ‘high’, \(t_{d1}\) and \(t_{d2}\) are the propagation delays of readOutControl to arrive at two successive pixels through the arbitration tree, \(t_{skew1}\) and \(t_{skew2}\) are the corresponding position-dependent uncertainties in releasing and enabling a pixel respectively. Data are latched at pose of readOutControl. \(t_{d1}\) and \(t_{skew1}\) do not appear in this equation, provided \(\tau_{on} > t_{d1} + t_{skew1}\), so that a given settling time can sustain a higher readout rate. \(t_{d1}\) must be greater than \(t_{hold}\), but this is easily achieved in modern processes.

Fig. 4 The binary tree priority encoder is shown for eight pixels. Pulses generated from the data transmitter (readOutControl) are broadcast only to the selected pixel by using the path enabled by readRequest from the pixel of highest priority to the data transmitter in the opposite direction. The next pulse reaches the pixel with the next highest priority, and so on.

Fig. 5. Timing diagram of the proposed pixel selection scheme. Pixel [N-1] is latched and subsequently disabled at the posedge of readOutControl and the pixel [N] is selected at the negedge. readOutControl is generated from serializerClk, based on the number of bits in a data packet. \(t_{d1}\) and \(t_{skew1}\) are now absorbed in \(t_{on}\), so that there is longer time for data to settle for a given readout rate.
readOutControl is derived from the data output serializerClk, and its period \( t_p \) is set equal to the time it takes to transfer a data packet off-chip. Since the data is latched at the periphery any of the pose of the serializerClk, which occur before the pose edge of readOutControl can also be used to latch the data for additional safety margin at the cost of shorter \( t_{setting} \).

To evaluate the advantages of the proposed architecture, it was compared with a serial shift register as well as with the original priority encoder [33]. The comparison was performed at the Register Transfer Logic (RTL) implementation level using area and power information from the normal Vt standard cell libraries for both 65 nm and 130 nm CMOS processes for the readout of 1024 pixels, each with 10 bits of data to be read out at 400 Mbps. The analysis is approximate but conservative, since the power consumption for the priority encoder does not account for the increased number of buffers required for the clock tree distribution of the readStrobe signal, since the pixel area is larger than the synthesized area of the readout. Table 1 clearly highlights the benefit of the SB-PE in terms of both power and area. As expected, irrespective of the readout scheme, both power consumption and area decrease as the technology scales. For an 8% marginal increase in area the SB-PE has around 50 to 80 times lower power consumption than the serial shift register. The elimination of buffers required for clock tree synthesis of the readStrobe signal effectively reduces the power of the SB-PE by almost a factor of two compared to the standard priority encoder.

Table 1. Comparison between readout schemes based on power consumption and area, for 65 nm and 130 nm technology nodes for 1024 pixels each with 10 bit of data transmitting at a rate of 400 Mbps. Total per pixel area reported is based on 10-bit storage/shift register.

<table>
<thead>
<tr>
<th>Readout Type:</th>
<th>Serial shift register</th>
<th>Priority encoder</th>
<th>Synchronized priority encoder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Consumption 65 nm</td>
<td>51.20 mW</td>
<td>~1 mW</td>
<td>0.63 mW</td>
</tr>
<tr>
<td>Power Consumption 130 nm</td>
<td>146.98 mW</td>
<td>~6 mW</td>
<td>3.24 mW</td>
</tr>
<tr>
<td>Total Area - 65 nm</td>
<td>211.7 ( \mu m^2 )</td>
<td>~240.0 ( \mu m^2 )</td>
<td>229.5 ( \mu m^2 )</td>
</tr>
<tr>
<td>Total Area - 130 nm</td>
<td>600.0 ( \mu m^2 )</td>
<td>~710.3 ( \mu m^2 )</td>
<td>674.1 ( \mu m^2 )</td>
</tr>
</tbody>
</table>

V. PIXEL LEVEL IMPLEMENTATION

The pixel level implementation of the logic which interfaces with the readout is shown in Fig. 6. The operating time frame is defined by a global, user-defined frameClk. As mentioned earlier for deadtimeless operation, the pixel includes two sets of data registers, one for storing data from the current frame and the other for sending data out from the previous frame. At the rising edge of the frameClk, the registers are toggled by changing switch positions if new data is available for readout. Simultaneously, the readRequest signal is asserted to indicate that the pixel has data.

For full frame imaging additional pixel logic is used to ‘set’ the readRequest signal for all pixels. When the selectPixel signal is ‘low’, the tristate buffer is enabling pixel access to the data bus. Subsequently the rising edge of the selectPixel signal resets and cancels the readRequest.

For triggered zero suppressed readout applications, such as those implemented for front-end readout of particle detectors at the LHC, the scheme as shown in Fig. 6 should be complemented by a memory buffer of sufficient depth to accommodate trigger latency. Depending on the pixel size and trigger latency, such memory can be placed in the periphery or in each pixel. The selection and transfer of data from memory would require additional logic for matching the trigger bunch crossing ID with the time stamp of the stored data.

Fig. 6 Conceptual circuit level implementation of a pixel with two sets of data registers for current and previous time frames. A readRequest signal is activated (high) at the start of a frame if pixel has data or in frame based readout. The falling edge of the selectPixel signal allows the pixel to access the common data bus and the rising edge disables the pixel and resets (low) the readRequest signal. A global signal changes the time frame, which cycles the data storage registers for readout.

Fig. 7 Priority encoder for four pixels showing the propagation of readRequest from pixel to the periphery, the pixel selection using the readOutControl signal uses the opposite path through the arbitration tree to reach only one pixel at a time. The address generation logic is also shown, which can be omitted if implementation is only for full frame readout.
The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.

A. Distribution of signals and placement of cells

A digital-on-top design methodology is used to assemble the readout architecture of the ROIC.

The switches in Fig. 4 are implemented as combinational logic gates as shown in Fig. 7. The circuit associated with a priority encoder for four pixels shows that, the higher order address bits can be computed and settle faster than the lower order bits.

If only full frame imaging is required, the readout architecture uses a cascade of alternating leafNOR and leafNAND cells without any address generation circuitry. Hence, for a matrix of 1024 pixels, 512 leafNAND cells and 511 leafNOR cells are required, no additional buffers are added.
VI. EXPERIMENTAL RESULTS

The synchronized binary tree priority encoder has to date been implemented in two ROICs. The first is a 192 × 192 pixel array [39] in 130 nm LP-CMOS, with both full frame as well as zero-suppressed readout capabilities. The second is a 64 × 64 pixel array in 65 nm LP-CMOS [7], exclusively with full-frame readout, which does not require the address generation logic. The test results presented in this section are for this second ROIC, which is the only one available so far. For deadtimeless operation a full frame readout speed of ~ 40 kfps or higher needs to be demonstrated which is equal to the time it takes for a single ADC conversion cycle. This full-frame rate is achievable if the output serializer is operating at 400 MHz. The goal of testing was to determine the speed, reliability and power consumption of the readout architecture.

A high-Vt standard cell library was used for this implementation to reduce the power consumption by a factor of two compared to the normal-Vt library. The chip is divided into four quadrants with one dedicated output pin for every 1024 pixels, with one SB-PE per 512 pixels to interleave pixel readout from the left and the right halves of the matrix, as presented in Fig. 10.

![Pixel array divided into two halves for interleaved ping-pong pipelining of data from pixel to the central data control and transmitter block, which contains an output serializer for serial data transfer and also generates the readOutControl signal derived from the serializerClk for pixel selection.]

![Highlighting of the physical placement of the output data buses from two halves of the matrix. Interleaving the two allows the data bus a longer time to settle ensuring data integrity, while also reducing the total bus capacitance by a factor of two.]

![Readout power consumption vs. power supply voltage for 1024 pixels with the serializer clock operating at 200 MHz]

![Readout power consumption vs. serializer clock frequency for 1024 pixels for the nominal digital power supply of 1.2V]

When considering the speed limitations of the data readout architecture, the worst-case occurs when all the lines of the 11-bit shared parallel data output bus, which transfers data from the pixel to the periphery, transition from ‘0 to 1’ or ‘1 to 0’ for every consecutive data transfer cycle.

For this test, the ROIC is configured such that 12.5% of the pixels contain ADC data corresponding to all 1’s, with each of them followed in the full-frame readout sequence by a pixel with no ADC data. With the range bit set to 1, this causes the...
output bus to switch at every readout cycle from 11'b11111111111 to 11'b10000000000 for 25% of the frame. Fig. 12 captures such transitions at the output of the ROIC, for a serializer clock frequency of 400 MHz.

The power consumption of the ROIC power supplies for analog, digital and LVDS sections can be independently monitored. The digital power consumption includes the readout power as well as ADC digital control logic. The power consumption of the readout is measured by recording the power during the full-frame readout and by subtracting the dynamic power corresponding to the ADC conversion (which can be independently measured by turning off the data readout serializerClk).

Power consumption is plotted against supply voltage shown in Fig. 13 as well as serializer clock frequency shown in Fig. 14, which both demonstrate good agreement between measurements and simulations. Simulations include post-layout parasitic capacitance per address line of up to 1 pF. The nominal operating voltage is 1.2 V. The readout fails progressively when lowering supply voltage: at 0.85 V only half of the chip is working, with only one active matrix of 1024 pixels, while no output is observed at 0.8 V.

![Fig. 15 Bit Error Rate vs. output serializer clock frequency, performed at 1.2 V power supply. No bit errors were detected up to 550 MHz. Bit Error Rate vs. power supply voltage, performed at 200 MHz output serializer clock frequency. No bit errors were detected above 1 V power supply. The BER Test is based on min/max switching of the data bus used to transfer data from pixel to periphery.](image)

To measure the output data bit error rate (BER), 1000 frames were analyzed. Each frame consists of 11286 bits of data, starting with a 22-bit header [10101010101010000000000000], followed by 11-bit data per pixel from 1024 pixels. The output serializer is operated for a range of frequencies from 100 MHz to 800 MHz, and the number of errors for 10 million bits is determined. These measurements do not use a pseudo random bit stream (PRBS) but instead are based on the worse case test pattern generated by toggling ten lines of the pixel data bus to min/max values for consecutive data transfer cycles. No bit errors were observed up to 550 MHz, at 1.2 V power supply, after which BER increases as shown in Fig. 15. These measurements were also performed for a fixed clock frequency of 200 MHz, and varying the digital power supply voltage from 900 mV to 1.4 V. No bit errors were observed for supply voltages above 1.0 V.

In this implementation for full-frame readout, which does not include address generation, the energy per bit measurement at the nominal operating voltage of 1.2 V and at a serializer output frequency of 400 MHz is 0.94 pJ/bit. This design reads 1024 pixels of 50 × 50 μm², distributed across an area of 1.6 mm × 1.6 mm optimized by the digital-on-top, place and route design assembly. It compares favorably with other binary tree readouts [35].

**VII. CONCLUSION**

The synchronized, binary tree priority encoder readout for pixel detectors is presented. It is a low power, high speed bus arbitration technique for data transfer from pixels to periphery. It is suitable for both full-frame readout as well as zero-suppressed readout. This technique minimizes switching activity across the matrix by allowing only one single pixel to be active at a time. Furthermore, it eliminates the need to synchronize between the two independent signals for pixel selection and deselection. Simulations and measurements show the viability of the design and demonstrate significant power saving. Unlike [35], this readout is not organized as a thin long column which is susceptible to large sidewall capacitance from closely packed metal tracks but instead by subchips [39]. The subchip approach is particularly suited to take advantage of data transfer through the backside of the chip utilizing low capacitance, small diameter through silicon vias [40] for future large chip implementation. The full chip assembly process of segmenting subchips based on resource optimization is elaborated in [41]. If several parallel output ports are not available, a hierarchical nested approach could be used. Encoding the address value at each level can further decrease the power consumption and save area [42]. Furthermore, the digital-on-top place and route assembly can easily scale the architecture for implementation to different pixel geometries and technologies.

Although we demonstrate our technique for pixel detectors, it is potentially applicable to other scenarios which require sparse data readouts. This includes access and transfer of data from content addressable memories, sparsifying output in tracking-trigger ASICs, population count circuits and networking applications.

**ACKNOWLEDGEMENTS**

We are indebted to Fermilab colleague and Group Leader Grzegorz Deptuch for suggesting the improved form of a priority-encoder-based readout controller for use in pixel detectors. We also appreciate the support provided by the Fermilab ASIC Design Group staff and Particle Physics Division management in supporting this work. We acknowledge contributions from Scott Holm, Fermilab and Lukasz Kadalbowski, AGH-UST Kraków for developing the readout system to test the chip described in this paper.
Farah Fahim received the Ph.D. degree in Electrical Engineering from Northwestern University, Evanston, IL, USA, in 2019. She received her bachelors and masters from the University of Limerick, Ireland. She worked at Rutherford Appleton Laboratory, UK from 2005 - 2009. She joined Fermi National Accelerator Laboratory in 2009 and is currently a senior engineer. She holds 5 issued U.S. patents on large area pixel detectors. Her contributions and research interests include pixel detectors and reconfigurable systems.

Siddhartha Joshi is a Ph.D. candidate at the Electrical and Computer Engineering Department of Northwestern University, Evanston Illinois USA. He received his bachelors from the Birla Institute of Technology and Science, Pilani. He has worked at Fermi National Accelerator Laboratory and Intel Corporation during his studies. His contributions and research interests include associative memory design and optimization, design automation and reconfigurable systems.

Seda Ogrenci-Memik (M’98–SM’01) is a Professor at the Electrical and Computer Engineering Department of Northwestern University, Evanston Illinois USA.

Her contributions span a variety of areas including design automation, high-level synthesis, reconfigurable systems, thermal-aware design, and thermal management of high performance systems. She was a Distinguished Visiting Professor at the School of Computer Science of the Complutense Universidad de Madrid in 2010 and a visiting faculty at Koç University in 2010.

Hooman Mohseni received the Ph.D. degree in electrical engineering from Northwestern University, Evanston, IL, USA, in 2001.

He then joined Sarnoff Corporation, where he was a Member of Technical Staff leading several government, domestic, and international commercial projects. He joined Northwestern University as a faculty member in 2004. He is the Director of Bio-Inspired Sensors and Opto-electronics Lab, and Northwestern’s Solid State and Photonics Initiative. He received the Young Faculty Award from Defense Advanced Project Agency in 2007. He was selected by NSF as a US delegates in U.S.-Korea Nanomanufacturing Exchange program in 2007, and USJapan Young Scientist Exchange Program on Nanotechnology in 2006. He received National Science Foundation’s CAREER Award in 2006. He has served as the Advisory Board, the Program Chair, and the Co-chair in several major conferences including IEEE Photonics, SPIE Optics and Photonics, and SPIE Security and Defense. He has published more than 120 peer-reviewed articles in major scientific journals including Nature, Nano Letters, Small, and ACS Nano. He holds 14 issued U.S. and International patents on novel optoelectronic devices and nanoprocessing. He has presented more than 51 invited and keynote talks at different commercial, government, and educational institutes. He is a Fellow of SPIE and a Fellow of Optical Society of America.