A Comparative Study of Access Topologies for Chip-Level Address-Event Communication Channels

Eugenio Culurciello and Andreas G. Andreou, *Member, IEEE*

***Abstract*—We examine channel access algorithms and circuits for intra and inter chip communication channels. Classical access techniques such as arbitration, scanning, ALOHA, and priority encoding are compared by assessing throughput, latency, and power consumption. Our results provide guidance in the design of bio-inspired networks of processors, for efficient transmission of information with limited power consumption and reduced latency.**

***Index Terms*—Accent topologies, address-event, ALOHA, bio-inspired systems, inter-chip communication.**

I. Introduction

THE HUMAN brain's impressive computational abilities, are to a large extent, a result of its ability to process information in a parallel and distributed manner. Neurons actively generate their own output signals when they have salient information to transmit, both at the periphery but also in the central nervous system. Networks of neurons that are massively interconnected and organized in spatial arrangements called maps, are dynamic structures that employ learning and adaptation. The brain uses action potentials or "spikes" to transmit information both in the sensory and central nervous system. Spike communication facilitates robust long-distance communication by means of self-restoring, all-or-none ("digital") signals. Spike trains are temporally sparse, possibly because lower spike rates are more energy-efficient [1]. "Spikes" are not digital signals, however, in the sense of the binary-valued discrete-time signal representation employed ubiquitously in modern information processing machinery. Architecture optimization is accomplished at all levels of the system hierarchy with remarkable results at the end. In a structure with 10^{16} connections and a power budget of 12 W, energy-efficiency is likely to be an optimizing constraint. Asynchronous, on demand information processing enables biological systems to operate effectively and reliably under the physical constraints of wiring complexity and energy supply and heat extraction.

Our recent attempts to endow human engineered systems [2] with brain like functionality has lead to parallel and distributed processing architectures that are being employed to solve problems in machine perception. In such bio-inspired architectures computation and communication resources are shared by individual processor nodes in arrays corresponding to the maps in neural structures. In these neuromorphic architectures, when there is *a priori* knowledge that not all nodes are likely to require computation/communication resources at the same time, a fixed time-slot (synchronous) allocation of resources among all nodes is wasteful. Therefore, in this regime of bursty demand for resources, computation/communication is more efficient if done asynchronously.

In this paper, we employ the mathematical tools and performance criteria developed in the theory of communication in macro-scale systems to analyze access topologies for intra-chip and inter-chip communication. More specifically, our analysis examines the energy efficiency in different access topologies. The foundation for results presented in this paper is the early work by Mortara and Vittoz [3] as well as Boahen's analysis [4] whose focus is on throughput and latency. Complementary to the work presented in this paper is the analysis by Reyneri [5] who examined the merits of pulse signal representations and modulation schemes in neuromorphic systems. The efficiency of the data representation from a data reconstruction perspective has been studied by Apsel and Andreou [6].

In Section II, we begin with a brief discussion of the address event representation (AER), a time division multiple access communication scheme that has been widely used in the neuromorphic very large-scale integration (VLSI) community [7], [8]. To address the power efficiency of the architecture we introduce performance metric that is aimed at maximizing the throughput of the system while minimizing the latency and power dissipation. In Section III, we analyze the throughput and latency of arbitrated and not arbitrated access methods. Power dissipation of the access structures is discussed in Section IV with results and discussion in Section V.

II. AER and Merit Criteria

Inspired by spike communication in the nervous system, Mahowald [7] and Sivilotti [8], proposed AER as a method to communicate information among bio-inspired subsystems and demonstrated its utility in prototyped vision and auditory chips. In an AER system each cell in an ensemble can transmit or receive an event, a discrete-value data structure generated continuously in time (asynchronously). Events are data packets that encode compactly the address of the sender and timing information as well as other relevant to the computational task attributes of the sender such as color, receptive field size, and orientation. Boahen [4] reviews the literature and provides an excellent introduction to the subject matter.

We define the sequence ε of events generated by the sender as a collection of data (x_{gi}) and timing (t_{gi}) information pairs. Events are generated sequentially in time and, therefore, $t_{g0} <$

Manuscript received September 15, 2002. This work was supported by DARPA/ONR under Contract N00014-00-C-0315 and by the National Science Foundation under Grant EIA-0130812.

The authors are with the Electrical and Computer Engineering Department, Johns Hopkins University, Baltimore, MD 21218 USA.

Digital Object Identifier 10.1109/TNN.2003.816385

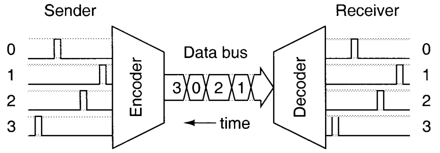


Fig. 1. AER. Events generated by the transmitter are encoded with their address as identifier and communicated through the channel to an external receiver. Events are then routed according to their address.

$t_{g1} < t_{gi}$. On the receiver side, data is reconstructed by operating on the address stream in a way commensurate with the required task. Lazzaro *et al.* have employed both address and timing information in their silicon cochlea computer interfaces [9]

$$\varepsilon = \{(x_{g0}, t_{g0}), (x_{g1}, t_{g1}), \ldots (x_{gi}, t_{gi}), \ldots\}. \quad (1)$$

Often, timing information is eliminated, and the density of the individual addresses hence represent the value of the data. As such the sender employs pulse rate modulation (PRM) coding [5] and temporal integration of the address data at each cell can be used for data reconstruction. The optimal design of such an integrator is discussed in [10].

A typicatation AER system is presented in Fig. 1. Events from a transmitting ensemble generating cells are encoded as sequence of addresses that are transmitted in a channel.

As the number of physical connections from one sensor/computational map, usually a single chip, to another is limited, an algorithm to map the sequence of events generated in the data array into a sequence of multiplexed data in the physical layer of the communication channel (Fig. 2).

Mathematically, the algorithm is the function f_{AC} performing the following mapping:

$$\begin{aligned}\varepsilon' =& \{ (x_0, t_0), (x_1, t_1), \ldots (x_i, t_i), \ldots\} \\ =& f_{\mathrm{AC}} \{ (x_{g0}, t_{g0}), (x_{g1}, t_{g1}), \ldots (x_{gi}, t_{gi}), \ldots\}.\end{aligned}$$

Due to the remapping of originally generated (subscript g) events, the data and timing information is changed

$$\begin{aligned}x_i =& x_{gi} \oplus x_{\mathrm{type}} \oplus x_{\mathrm{chip}} \\ t_i =& t_{gi} + \delta_i.\end{aligned} \quad (2)$$

We now concentrate our attention on how the data is transferred between the transmitting ensemble and the channel. The subsystem on an integrated circuit that allows this transfer is the *access circuit*, while the algorithm describing the behavior of the access circuit is called the *access technique*.

In particular the data portion of the event x_{gi} is modified to include an identifier for the chip that generates the address denoted as x_{chip} and the type of the cell x_{type}; the operator $\oplus$ denotes a bit-vector concatenation. The access circuit also introduces a latency δ_i to the transmission which causes timing mismatch between the original and transmitted on the channel event. The channel is characterized by the channel rate F_{chan} specified at a desired maximum allowed error rate and thus each slot is allocated T_{chan} time. The performance of the access circuitry can be assessed by introducing G, the normalized offered load of input events, and S is the normalized throughput of the communication system in terms of output events. Both these quantities are normalized by T_{chan}. This normalization allows to compare the throughput S to the capacity of the channel directly, since the former is defined as the usable portion of the channel capacity.

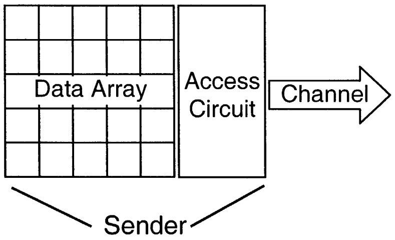


Fig. 2. Sender and channel. Sender comprises a data array and an access circuit.

The performance of the access circuitry is also conditioned by the amount of timing error that it generates in the output sequence of events. The latency δ_i for event i is a function of the access technique and the offered load defined by the average array interevent time $T_{\mathrm{event}} = 1/f_{\mathrm{event}}$ where f_{event} is the event rate

$$\delta_i = h\left(f_{\mathrm{AC}}; T_{\mathrm{event}}\right). \quad (3)$$

When the channel capacity is reached, the access can not service the data originating in the array. When such contention occurs, one event (i) tries to access the channel while the latter is occupied with the transmission of another event ($i{-}1$ or $i{+}1$). A collision can occur when another event is generated in a window $2T_{\mathrm{chan}}$ around the time of generation of event i

$$(t_{g,i} - t_{g,i-1} < T_{\mathrm{chan}})\, OR\, (t_{g,i+1} - t_{g,i} < T_{\mathrm{chan}}). \quad (4)$$

The probability of collision p_{coll} can be computed using the probability of generation p_g of zero events in the interval $2T_{\mathrm{chan}}$

$$p_{\mathrm{coll}} = 1 - p_g\left(0;\ 2T_{\mathrm{chan}}\right). \quad (5)$$

An access algorithm is termed *collision-free* if it is capable to queue events for delayed transmission when the channel is free.

We now proceed to make assumptions on the input data distribution so that we can derive formulas that will help us analyze the advantages and disadvantages of the different access algorithms and circuits. Let us assume that the data array generates events that can be modeled by an independently identically distributed (i.i.d.) Poisson point processes. Since each individual cell produces Poisson distributed events, an ensemble of these cells will result a Poisson distribution as a sum of Poisson point processes. The distribution has the following form:

$$P(k, G) = \frac{G^k}{k!} e^{-G} \quad (6)$$

where $P(k, G)$ is the probability of generating k events in a time frame defined by G, which is the expected number of events per channel cycle time T_{chan}

$$G = \frac{T_{\mathrm{chan}}}{T_{\mathrm{event}}} = N \frac{f_{\mathrm{event}}}{F_{\mathrm{chan}}}. \quad (7)$$

As noted earlier T_{event} is the mean time (inter-event interval) between events in the array of N cells, and f_{event} is the event rate of a single cell. Fig. 3 shows diagrammatically the definition of G.

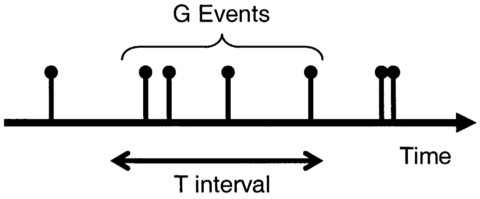


Fig. 3. Event generation and load G in a time interval T.

A measure of the average event rate or offered load is given as a mean of the inter-event timing between the generation of observed events by equation

$$\begin{aligned} T_{\text{event}} &= \lim_{i\to\infty} \frac{(t_{g,1}-t_{g,0})+\cdots+(t_{g,i}-t_{g,i-1})}{i} \\ &= \lim_{i\to\infty} \frac{t_{g,i}-t_{g,0}}{i}. \end{aligned}$$

This quantity can be calculated by observing a large number of input events.

A. Merit Criterion

In this paper, we examine the performance of the access technique by exploring the channel utilization or throughput S [4], determining the theoretical expected latency μ_{sys} for a given load G and the power P_c (in Watts) used by the access system.

It is then possible to estimate the best access technique for each application by computing the quality metric

$$Q = \max_{f_{\text{AC}}} \left(\frac{S(G, f_{\text{AC}})}{P_c(G, f_{\text{AC}})\mu_{\text{sys}}(G, f_{\text{AC}})} \right). \tag{8}$$

Maximizing the metric in the above equation leads to the identification of the optimal access technique for a given load. Note that since all the above quantities are dependent on the offered load G, the input distribution greatly affects the choice of the optimal access technique.

III. Channel Access Algorithms

In this section, we compute throughput and theoretical expected latency for four access algorithms: priority encoder, the ALOHA-derived, the arbitration tree, and sequential scanning.

A. Sequential Scanning

The scanning register access algorithm employs ripple counters to repetitively sample the activity of a population of asynchronous transmitters and communicate in a predetermined sequence the event of each active transmitter. This access is synchronous and is externally controlled by a sampling clock, independently of the event rates in the array. This access algorithm is particularly useful when sampling uniformly randomly distributed signal, or when events are not clustered in time or any spatial dimension. In fact when the data is uniformly distributed the entropy of the input channel is maximized. The mean time between two scans T_{sr} of the same cell can be written as

$$T_{sr} = T_{\text{chan}} \cdot N. \tag{9}$$

In general, for perfectly symmetric transmitters and receivers T_{sr} of a scanning register system will have very little dispersion. This because we are dealing with bit-parallel systems, and the dispersion is due only to the difference in rising time of the gates. This hypothesis might not be true for cluttered transmitting systems with high collision rate, or for some serial AER system, although the length of the packets remains the same (address size does not vary). An average transmission occurs in a time equal to T_{chan}: the minimum time between a request from the transmitter and the reception of an acknowledge from the receiver chip (handshaking).

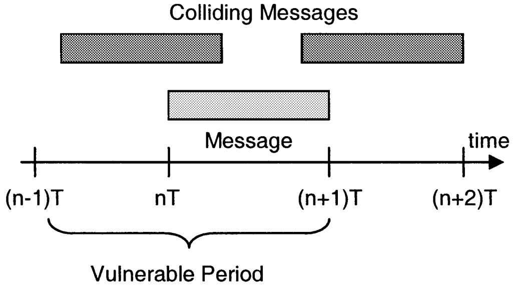


Fig. 4. Vulnerable period for collision. Here $T = T_{sr}$.

Because of the topology of scanning access, the latency between requests and acknowledges from the receiver will have little variability. This is because a ROM table decoder is generally used. Also acknowledge signals are transmitted as soon as an address is decoded. Therefore, the standard deviation of the channel latency will be much less that the communication cycle time [11], or as follows:

$$\sigma_{\text{chan}} \ll T_{\text{chan}}. \tag{10}$$

The service statistics of the scanning registers are thus practically deterministic.

This access scheme allows no collision between events, since only the selected cell can use the channel, while all other events will be queued, but not discarded. This latency in the queue will determine the amount of data lost while scanning. We can calculate the amount of data lost by computing the collision probability.

For an event to be successfully transmitted (no collisions), no other event must be generated in a timeframe of amplitude $2T_{sr}$; thus the rate involved in the calculation of p_{coll} is $2G$. Fig. 4 represents the vulnerable period for collisions when an event is sent at time *nT* $_{sr}$ with $T = T_{sr}$. Its amplitude is $2T_{sr}$ (from $(n-1)T_{sr}$ to $(n+1)T_{sr}$).

If the transmission is errorless the output distribution of events in time will match the input one. The effect of errors on the output distribution is to distort the portion of the distribution with events with very low interevent timing. If the transmitting element uses an integrating capacitor to store its data, the capacitor could saturate before it is sampled (if the input dynamic range is high). The amount of data loss can be written as p_{coll}

$$\begin{aligned} p_{\text{coll}} &= 1 - p(0, 2G) = 1 - e^{-2G} \\ G &= \frac{T_{sr}}{T_{\text{event}}}. \end{aligned}$$

Although the scanning register access does not generate output data collision, we can still interpret a slow scanner as a system that induces collision within each cell, since it does not service them fast enough and data saturation or clipping occurs. Collision probability has been utilized in effectively

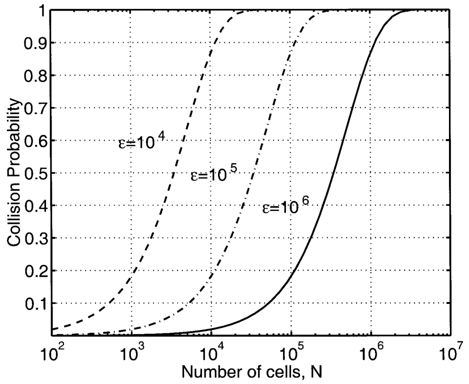


Fig. 5. Collision probability (p_{coll}) in sequential scanning as a function of N and $\epsilon = f_{\text{event}}/F_{\text{chan}}$ $(f_{\text{event}} = 100\text{ Hz})$.

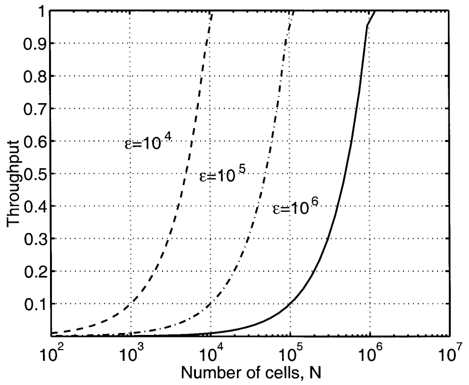


Fig. 6. Scanning registers access throughput as a function of N and $\epsilon = f_{\text{event}}/F_{\text{chan}}$ $(f_{\text{event}} = 100\text{ Hz})$.

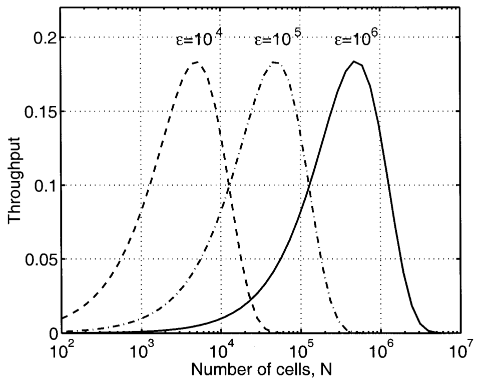


Fig. 7. ALOHA access throughput as a function of N and $\epsilon = f_{\text{event}}/F_{\text{chan}}$ $(f_{\text{event}} = 100\text{ Hz})$ 100 Hz).

in neuromorphic systems [12] to implement pulse-stream sigmoids.

Fig. 5 presents the collision probability as a function of the number of N. Having a faster channel (constant N) decreases the errors because of the high capacity, while increasing N(constant capacity) augments the channel utilization and, therefore, the errors.

Since there is not loss of data due to collisions, the maximum throughput for the scanning register access technique in (events/second) is given

$$S = G = \frac{T_{sr}}{T_{\text{event}}} = N f_{\text{event}} T_{\text{chan}}. \tag{11}$$

This function will increase until data loss occurs, and at that point saturates to the bandwidth of the channel, as can be seen in Fig. 6. Note that when the throughput is maximized, timing errors occur. Therefore, the scanning register access should be used only *near* and not at the maximum throughput.

The average latency of the scanning register access can be estimated to be one half of the maximum value or

$$\mu_{\text{sys}} = \frac{T_{sr}}{2}. \tag{12}$$

This quantity is already proportional to the number of cells in the sensory system through the input load G. Although scanning provides high throughput, it does only at a certain condition, when the spiking rate T_{event} is adapted to the scanning rate for the whole array (synchronous multiplexing). But AER systems generate very high dynamic range signals [13] that would be truncated if adaptation or automatic gain control is used. Therefore, if loss of data can be accepted and or originate in reduce dynamic range stimuli, then scanning systems can be adapted to give the maximum throughput from a given sensory system.

B. ALOHA-Based

The simplest asynchronous access algorithm is the one where each cell is allowed to access the channel as soon as an event arises. This access is event driven: events themselves access the output bus without any external intervention. This access scheme is the basis of the Ethernet network protocol (IEEE-802), and it is called ALOHA access protocol [14] without retransmissions. The ALOHA protocol has a limited throughput of only 18% of the channel maximum capacity, because of all the cells transmitting at will. Fig. 7 shows the throughput of the ALOHA channel; notice the peaking of the throughput function. Higher event generation rates allow for more cells to participate in the channel before the maximum throughput is obtained. The channel should be always used at its peak throughput to maximize the use of its capacity. Note that the absence of external multiplexing logic, makes the ALOHA service time statistics almost deterministic. [15] presents a version of ALOHA with collision detection. A more restrictive access type, carrier sense multiple access (CSMA) 1-persistent, allows cell to still transmit as they need, but also require them to wait for the channel to be free, impeding collision and loss of data. This type of access reaches channel throughput of 53%, more than twice as much as the ALOHA. If the purpose is to make an efficient use of the channel bandwidth, an important issue especially for high N, this access technique represents an undesirable choice [16]. Given the above-mentioned Poisson distribution of the input inter-event interval, we can calculate the probability of an access collision p_{coll} in a pure ALOHA channel as follows [14]:

$$\begin{aligned} p_{\text{coll}}(T_{\text{chan}}) &= 1 - p(0, 2G) = 1 - e^{-2G} \\ G &= \frac{T_{\text{chan}}}{T_{\text{event}}} \end{aligned}$$

with the throughput of the channel given by

$$S = Ge^{-2G} = \frac{1 - p_{\text{coll}}}{2} \ln\left(\frac{1}{1 - p_{\text{coll}}}\right). \tag{13}$$

An expression for the throughput S of the CSMA 1-persistent is given by [17]

$$S = \frac{Ge^{-G}(1+G)}{G+e^{-G}}. \tag{14}$$

Note that for an event to be successfully transmitted, no other event must be generated in the interval $2T_{\text{chan}}$, thus, the rate involved in the calculation of p_{coll} is $2G$. Fig. 4 also represents the vulnerable period in the ALOHA system with $T = T_{\text{chan}}$. The interval is again $2T_{\text{chan}}$ (from $(n-1)T_{\text{chan}}$ to $(n+1)T_{\text{chan}}$) as seen for scanning registers.

In a slotted ALOHA access protocol, the events are allowed to access the channel only in discrete-time slots. The performance of the channel, in terms of throughput is expressed by the following relation in terms of the rate G or the probability of collision [14], [17]:

$$p_{\text{coll}}(T_{\text{chan}}) = 1 - p(0,G) = 1 - e^{-G} \tag{15}$$
$$S = Ge^{-G}. \tag{16}$$

The performance is 37%, or twice better than pure ALOHA.

Since the vulnerable period for collision is halved the performance improves. But slotted ALOHA is a synchronous system, therefore, suffers for additional latency due to synchronization.

CSMA access technique samples the channel before transmitting, and communicates the event as soon as the channel is found free. This allows to increase the throughput of the channel to up to 53%, although only if the event transmission takes much longer than the interval required to sample the channel. In neuromorphic microsystems that are assembled at the board using parallel transmission of data, the sense time and transmission time are very similar, therefore, the performance drops to the case of slotted ALOHA [16].

Note that the subtle differences between these access protocols depend only on the hardwired algorithm used for the design of the access circuitry. More specifically, in the CSMA the p-persistence parameter [14] cannot be determined uniquely. This is because sensing the channel involves sensing the transmitter's request. Furthermore, for this reason, collision of events occur only in a small time frame determined by the handshaking of transmitter and receiver, and, therefore, has a much less catastrophic impact than in packet transmission. Long packets take several times the interval of a single parallel transmission (which looks like a single bit packet) and, therefore, constitute a broader window for event to collide.

In any of the examined cases (as in [3], [15], and [18]) the throughput of the channel is less than half of the bandwidth of the channel, suggesting the necessity of finding more efficient ways to access the channel.

The latency μ_{sys} of the ALOHA system is dependent on the collision rate

$$\mu_{\text{sys}} = \frac{1}{1-p_{\text{coll}}} T_{\text{chan}} \tag{17}$$

while for the p-persistence access system with probability p of delayed transmission is given by

$$\mu_{\text{sys}} = \frac{1}{(1-p_{\text{coll}})\,p} T_{\text{chan}}. \tag{18}$$

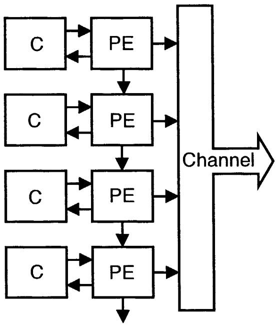


Fig. 8. PE access topology for an ensemble of C.

C. Priority Encoder (PE)

The PE algorithm allows any cell, identified by an ordering number, to communicate at any time provided that the channel is free (fixed priority), similarly to the ALOHA-based protocols. It is an event-driven asynchronous scheme. The PE access technique has the limitation of the ALOHA protocol, since it is only a modified biased version of it. Priority here refers to the case when more than one request is received at the beginning of a cycle. In this case only the lowest numbered cell (C) will be granted access to the channel, while all other requests will be queued (Fig. 8). The PE circuit is implemented by a cascade of static logic elements that needs to have input data maintained stable until the end of the cycle. This can always be accomplished by a proper asynchronous circuit that will not withdraw the input signal before receiving an acknowledge signal.

Two possible implementation of the PE channel can be analyzed: the first version uses no buffer between PE and asynchronous channel output. The priority is used only in case two or more channels try to access the channel simultaneously: in this case, the higher priority channel will win. Nevertheless, since there is no buffering of the inputs to the PE, those signal are able to change even within a cycle, thus generating erroneous result. In fact if a higher priority channel requests while a lower one is awaiting an acknowledge from the receiver, it will disable the lower priority and receive the acknowledge itself. The receiver will randomly get one of the two address and discard the other. In a second and worse case, if the inhibition signal from the higher cell is not received fast enough by the lower cell, the receiver would get as input the logical OR of the two cell's addresses.

The second possible implementation requires buffering of the inputs and disabling any input change in the buffer until the previous cycle is terminated. At that point all the requests will be arbitrated by the PE cascade and a winner will result. This scheme does not allow the receiver to randomly discard one of the conflicting events, but still suffers from the second type of error mentioned above. In fact, there will still be glitches at the output of the PE during its settling time, and these pulses can result in spurious events if the receiver is fast enough to detect them. Since the optimal AER channel makes no hypothesis on the speed of the receiver, these errors must be carefully taken into account.

Note that a buffered PE will act exactly as a CSMA 1-persistent protocol and will exercise priority on his inputs only if they happen to be coincident in time. Since coincidence in time for modern circuits means windows of approximately 1ns or less, an extremely low number of events will coincide and, therefore, the importance of priority decreases (at least for low event rates).

For both implementations is, therefore, essential to calculate the probability of the collision of two or more events which results into erroneous output data or dump of information. As shown in the previously examined topologies, the calculation of collisions depends on the statistics of the input signal and the topology of the AER circuit. In the following discussion, we will use Poisson distributed ensembles of event-generating cells.

The topology addressed in this study is of a single array of cells for the auditory channel and an array (rows) of array (columns) employed in vision systems. Each array is arbitrated separately and in the vision case rows requests are processed with an OR operator. Because of the differences in the visual and auditory sensory systems, they will be analyzed separately.

1) Vision Priority Encoder: For VLSI artificial vision circuits, the event-generating population is an ensemble of equal Poisson distributed cells.

The unbuffered PE for visual communication is simply an ALOHA access protocol, where transmission is further skewed because of the processing of the priority. But generally, when events are sparse and collision is low, which is the desired configuration for this type of PE, then the system behaves as a pure ALOHA.

The collision probability p_{coll} and the channel throughput S are given by

$$\begin{aligned} p_{\text{coll}}\left(T_{\text{chan}}\right) =& 1 - p\left(0, 2G\right) = 1 - e^{-2G} \\ S =& Ge^{-2G}. \end{aligned} \tag{19}$$

The channel throughput results in only 18% of the total channel capacity.

Buffered PE, recall a CSMA 0-persistent access protocol [19], and in fact behaves according to the same principles: the channel throughput is then given by

$$S = \min(1, G). \tag{20}$$

The latency of this system can be computed as done in the ALOHA access system.

2) Auditory Priority Encoder: We will now consider a silicon cochlea [20], an auditory sensory system that has employed a PE at its output [21] The silicon cochlea is a frequency analysis system that spans a certain auditory frequency band. In the filter-bank implementation, each filter looks at a portion of the band and extracts information from that band. Because of the processing of the silicon cochlea each event-generating cell (here called auditory channel) will have a different Poisson parameter (mean and standard deviation) that is directly dependent on the auditory channel examined. Thus the G parameter depends on the center frequency of the filter in each auditory channel. Note that auditory channels are assumed independent, since they measure different portions of the spectrum of the input auditory (here speech) signal, but they all share the same AER output channel.

For a Poisson distributed event-generation the individual channels rates in the time interval T_{Pecyc} (similarly to T_{chan}) can be rewritten as follows (similarly to ALOHA):

$$\begin{aligned} p_{sgi}\left(T_{\text{PEcyc}}\right) =& p\left(0, 2G_i\right) = e^{-2G_i} \\ G_i =& \frac{T_{\text{PEcyc}}}{T_{\text{event,i}}} = f_{\text{event,i}} \cdot T_{\text{PEcyc}} \end{aligned}$$

where p_{sgi} is the probability of successful generation (no collision). $F_{\text{event,i}}$ is the center frequency of the ith auditory channel.

In the case of a high number of equally spaced frequency bins (as in the Fourier transform) it is possible to consider the signal as equally distributed amongst the filter frequency. Therefore, the center frequency $f_{\text{event,i}}$ is the mean rate. However in silicon cochleas [20], [21] the filter bank center frequencies are logarithmically spaced, therefore, the offered load provided by each auditory channel is in a logarithmic relation with the neighboring channels: $f_{s,i} = 2f_{s,i-1}$.

When the asynchronous channel is handshaking with a receiver, the time interval T_{Pecyc} is called simply T_{cycle} and can be decomposed into the following:

$$T_{\text{cycle}} = T_{\text{req}} + T_{\text{tx}} + T_{ack}. \tag{21}$$

where the request time T_{req} is the combination of the PE settling time and the generation of the request, the transmission time T_{tx} is the time constant of the physical wiring and the acknowledge time T_{ask} is the time required to the receiver to respond to a request.

The probability of a successful generation p_{sg} of an event (without collision) from an auditory channel striving to access the AER communication channel can be written as follows:

$$p_{sg} = \exp\left(-2\sum_{i=1}^{N}\sum_{j=1}^{i-1} G_j\right). \tag{22}$$

This probability is conditional to the probability that cell i produced an event at time 0 and that no single cell j is generating events in the time interval $\{-T_{\text{Pecyc}}, T_{\text{Pecyc}}\}$. Note here that the PE blocks events having lower priority than the event under consideration (i), therefore, the upper limit on the summation on j. The event-generation time interval can either be the closed-loop cycle time between sending a request and the reception of the receiver's acknowledge (unbuffered PE), or the inhibition settling time (buffered version). Thus, the same formula (above) can be used to determine what portion of the data will be discarded and how often erroneous data will be generated.

Note that the collision with high frequency auditory channels is dominant in this system. Priority should, therefore, be given to slower auditory channels. Collisions occur if the total handshaking cycle time (unbuffered PE) lasts longer than the time constant of the highest frequency auditory channel (inverse of center frequency of the auditory channel).

The resulting throughput for this access technique results

$$S = Gp_{sg}. \tag{23}$$

Fig. 9 shows a plot of the throughput of the PE system with respect to the number of cells and the channel transmission speed.

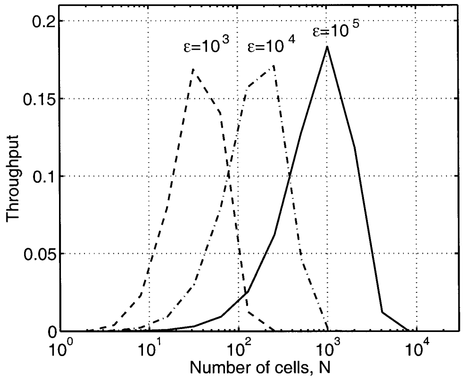


Fig. 9. Priority encoder access throughput as a function of N and $\epsilon = f_{\mathrm{event}}/F_{\mathrm{chan}}$ ($f_{\mathrm{event}} = 100$ Hz).

The N axis is limited to powers of 2 in Fig. 9, therefore, the plot appears to have low resolution. Throughput declines sooner for slower service rates. In this formula, the cumulative offered load G is defined as

$$G = \sum_{i=1}^{N}\sum_{j=1}^{i-1} \quad G_j = \sum_{i=1}^{N}\sum_{j=1}^{i-1} T_{\mathrm{PEcyc}} f_{\mathrm{event,j}}. \tag{24}$$

The graphs in Fig. 9 resembles very closely a pure ALOHA channel. In fact the PE block only events with lower priority, but is not able to block all the rest of events, that also provoke collisions. The highest throughput is again 18% of the channel capacity.

Because of the close similarities with ALOHA, the latency of the channel μ_{sys} can be computed in a similar manner.

D. Arbitrated Access

Since free for all access of the channel to the transmitting cells results in low communication efficiency, the use of special event-driven arbitrating algorithms allows to take advantages of the special properties of the input signal distribution. Instead of partly discarding colliding events it is preferable to queue them in order to obtain higher channel throughput. The arbiter is a digital tree circuit [16] or an analog cascade [22] that grants population of asynchronous transmitting cells access to the channel depending on the signal value using an analog winner takes all or timing (AER bistable digital arbitration). Arbitration requires some additional time to resolve a winning cell and, therefore, lengthens the cycle time and introduces longer latency. These effects result in interevent interval degradation. Important quantities to assess the performance of an arbitrated channel are latency, relative timing errors, and the way these quantities vary with the cell population size N.

An arbitrated channel can be modeled as a statistical waiting line by means of queueing theory techniques. Queues are differentiated by their input and service statistics. Other authors have proposed exponential and deterministic service time statistics, these do not necessarily apply to the case of arbitration implemented for parallel transmission AER VLSI systems. In case the transmissions are serial and the packet length (data x_i) is variable (like computer networks). This scenario will not occur in a serial AER system, since the address word always has the same length. The the exponential service time distribution is a better model. Therefore, the arbitrated channel can be described, to a first approximation [4], by a $M/G/1$ queue [23], [24], with an exponential (M) probability distribution of interarrival time of input events that are assumed to be Poisson distributed. In addition, the queue has unlimited buffer space, since cells can be stalled. Service times and statistics can be initially approximated with a deterministic distribution, since the delay introduced by switching a large logic circuit is much longer than the maximum delay that can be introduced by the variations of rising time of single logic gates.

The analysis that follows is due to Boahen and the main results repeated here for completeness. Using results for an $M/G/1$ queue [17], the time spent in the queue can be expressed using the famous Pollaczek-Khinchin formula [4], [23]. This result predicts that the moments of the time spent in the queue w_q depend on the moments of the service time x

$$\overline{w_q} = \frac{\lambda\overline{x^2}}{2(1-G)} \quad \sigma_w^2 = \overline{w_q}^2 + \frac{\lambda\overline{x^3}}{3(1-G)} \tag{25}$$

λ is the arrival rate of events per second (T_{event}), while μ is the processing rate per second or service time (can be also expressed as $1/x$). The ratio of the two is

$$\rho = \frac{\lambda}{\mu} = G \tag{26}$$

which corresponds to the throughput of the queueing system. G can be also defined as $G = \lambda T_{\mathrm{chan}} = \lambda/F_{\mathrm{chan}}$ and $x = T_{\mathrm{chan}}$.

The moments of the number of cycles spent in the queue m are described by

$$\overline{m} = \frac{\overline{w_q}}{T_{\mathrm{chan}}} = \frac{G}{2(1-G)} \quad \sigma_m^2 = \frac{\overline{w_q^2} - \overline{w_q}^2}{T_{\mathrm{chan}}^2} = \frac{2}{3}\overline{m}. \tag{27}$$

The resulting value of the variance is valid only if the service statistics have moments $x^n = T_{\mathrm{chan}}^n$, as a result of choosing a deterministic distribution for the service times.

Introducing the latency μ for an ensemble N of cells as the allowed time interval between generation and reception of events, a latency e_μ (this corresponds to the μ_{sys} of other access techniques) can be calculated as follows:

$$e_\mu = \overline{w} = \frac{GT_{\mathrm{chan}}}{2(1-G)} = \frac{G(2-G)}{N(1-G)}. \tag{28}$$

To obtain the second term, we assumed that at least half of the events (sparse activity) must be transmitted in the time specified by the latency μ. In that case, $\mu/T_{\mathrm{chan}} = N/2G$ holds.

Finally the throughput is given by the rate G, since there are no collisions in an arbitrated channel. More interestingly, it can be expressed as a function of the latency (solving it for G)

$$\begin{aligned} S =& G = \frac{T_{\mathrm{chan}}}{\overline{w}} = \frac{(1-G)}{G} \\ =& N\left(\frac{e_\mu}{2} + \frac{1}{N} - \sqrt{\left(\frac{e_\mu}{2}\right)^2 + \frac{1}{N^2}}\right). \end{aligned}$$

The arbitration circuit is an asynchronous pipelined queueing system of $M/G/1$ type. The Pollaczek-Khinchin mean-value formula allows for an estimation of the total time an event has to spend in the system before being serviced

$$T = T_{\mathrm{chan}}\left(1 + \frac{G}{2\,(1-G)}\right). \tag{29}$$

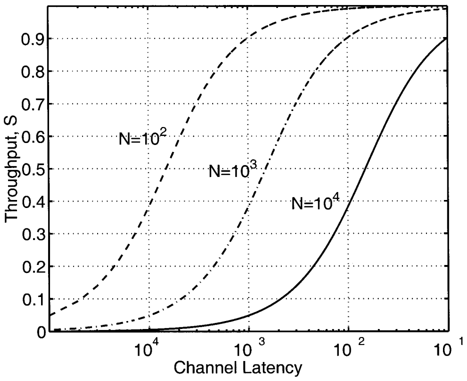


Fig. 10. Arbitrated throughput as a function of the channel latency with $e_\mu = 10^{-4}s$.

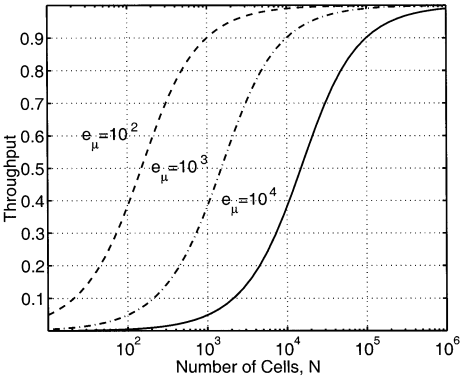


Fig. 11. Arbitrated throughput as a function of the number of spiking cells.

where T_{chan} is the average service time of the queue. We can express the timing error e_μ due to arbitration as

$$e_\mu = \frac{T_{\mathrm{chan}}}{\mu}\left(1+\frac{G}{2\,(1-G)}\right) = \frac{T_{\mathrm{chan}}}{\mu}\frac{2+G}{2\,(1-G)}. \tag{30}$$

Solving for the offered load G

$$G = N\left(\frac{1}{N}+\frac{e_\mu}{2}-\sqrt{\frac{e_\mu^2}{4}+\frac{1}{N^2}}\right). \tag{31}$$

To obtain G we assume that the mean cycle time for servicing events is fast enough to send half of the events in the array during the interevent time interval of a single cell firing at average rate. In this condition, the following is true:

$$\frac{T_{\mathrm{chan}}}{\mu} = \frac{G}{N}. \tag{32}$$

This allows to obtain the final value for G. This is a very interesting result that shows how the throughput S for arbitrated channels is linear with the number of elements in the array N. Meaning that an AER system with arbitration can operate almost at full channel capacity trading only a linear timing degeneration for an increasing cell population N [4]. Fig. 10 shows the throughput of the arbitrated channel with respect to the channel latency, while Fig. 11 shows the throughput as a function of the number of spiking cells in the channel [4].

IV. Power Consumption

In this section, we estimate the power dissipation for circuits implementing the different access algorithms.

Estimating the power consumption of cell arrays with different access technique is complicated, since it depends on the input load F_{event} and its statistical distribution. The input distribution gives an average event generation rate that can vary with time. Secondarily, the access circuit architecture contributes significantly to the total power consumption, since techniques like pipelining and row/column organization of the array can save significant computation when emitting events. Row/column organization divides the array in rows and columns, respectively, usually selecting one row first, then selecting individual elements within the chosen row (processing the columns).

Here, we employ a simple method to do a worst case scenario for power consumption of gates and digital elements used to access the array of N event generating cells. We begin by noting that a significant amount of the power consumption in an array of cells is due to communication rather than local processing in the individual cells. In fact, cells can maintain low power operation, given they operate at speeds that are orders of magnitude slower than the peripheral communication infrastructure.

Let us first concentrate on the ALOHA access circuitry servicing an array of N cells. Every cell needs an access point to the channel, which is a wiring in the form of a comb reaching out every cell in the array. The power consumption necessary to toggle the single communication line by switching its total capacitance C_{cwal}, given by

$$C_{\mathrm{cwal}} = L_{\mathrm{cwal}} W_{\mathrm{cw}} C_{\mathrm{uacw}} \tag{33}$$

where L_{cwal} is the total length of the communication line, W_{cw} is its width, and C_{uacw} is its the capacitance per unit area. The total wiring length L_{cwal} of the communication line is given by

$$L_{\mathrm{cwal}} = \frac{L_{\mathrm{clal}}}{2}\left(N+\sqrt{N}\right). \tag{34}$$

Assuming a square arrangement of the N cells, the line is composed of $(\sqrt{N}+1)$ branches in total each with length $(L_{\mathrm{clal}}\sqrt{N})$; L_{clal} being the lateral size of a single cell. The following equation expresses the estimate power consumption for an ALOHA access circuitry:

$$P_{\mathrm{cal}} = \frac{1}{4} C_{\mathrm{cwal}} V_{\mathrm{dd}}^2 F_{\mathrm{event}}. \tag{35}$$

The factor of $1/4$ in the formula derives from the product of $1/2$ from bit probability and $1/2$ for the probability of having to write the same voltage (charge the line capacitance) that the line already previously had. The model for P_{cal} is simple but and takes into consideration the power necessary to charge and discharge the total capacitance of the communication wiring.

Equivalently, for a scanning register access technique we can assess the capacitance of each branch of the scanner that will select and read out the data from the cell. The scanning register has a row and column organization and, therefore, the capacitance of the lines is reduced to a single line in each dimension

$$C_{\mathrm{cwsr}} = \sqrt{N} L_{\mathrm{cwsr}} W_{\mathrm{cw}} C_{\mathrm{uacw}}. \tag{36}$$

Notice that the line length in this case is only $(L_{\mathrm{clsr}}\sqrt{N})$ long, assuming square arrangement of the cells and with L_{clsr} being the lateral size of a single cell. The power consumption is then given by

$$P_{\mathrm{cal}} = \left(1+\frac{1}{\sqrt{N}}\right)\mathrm{C}_{\mathrm{cwsr}}\mathrm{V}_{\mathrm{dd}}^{2}\mathrm{F}_{\mathrm{scan}} + \frac{1}{4}\mathrm{C}_{\mathrm{cwsr}}\mathrm{V}_{\mathrm{dd}}^{2}\mathrm{F}_{\mathrm{scan}}. \tag{37}$$

The first term is for addressing the cell, the second to output data from the cell on the line. The row and column organization breaks the communication wire into smaller pieces. Also columns switch $1/\sqrt{N}$ times less often than rows. A factor of two is included in the equation since it takes twice the power to first address a cell and then output its data. The power consumption is mainly due to the switching of the capacitance of the communication wiring. We did not take into account the scanner (shift registers) consumption because it is infinitesimal compared to P_{cal}.

Last, we will focus on arbitrated access circuits. Arbitration has a significant circuit overhead that switches stochastically. Again, to simplify things we will concentrate to a first-order model with binomial input distribution of events and worse-case analysis.

Notice at first that the arbiter has a row and column organization, so the same analysis as for scanning register applies for power consumption due to selection of a cell and its data communication. On the other hand, the arbitration necessitates significant power consumption overhead compared to ALOHA and scanning. The arbitration circuit is divided into row and column trees, with each tree composed of $\sqrt{N}-1$ elements and a tree depth of $log_2 N$. Each tree consumes power due to switching of the individual arbiters during the transmission of an event. The number of elements that switches per event is a function of the number of events queued in the arbiter tree and their respective position. Arbitration [4] exploits locality to optimize and pipeline the sending of events on the communication bus. Pipelining occurs when many events are clustered in space. In that case, events occurring in a small window of time can be transmitted together in a burst. Thus, in general, events can happen in bursts or solitarily: we, therefore, consider a binomial distribution of events, where a burst takes advantage of locality to obtain smaller transmission cycle times. Empirically is has been determined that inter-event timing in arbitrated systems follow binomial distribution (see Fig. 12 plotted from measurement on the Octopus retina image sensor [13]). Note that the distribution has peaks around 180 ns and 330 ns that correspond to intra and interrow arbitration cycle times while observing a highly illuminated scene. The binomial distribution will be used to estimate the power consumption in arbitrated system. The arbiter power consumption is given by P_{carb}

$$P_{\mathrm{carb}} = 2\mathrm{C}_{\mathrm{cwsr}}\mathrm{V}_{\mathrm{dd}}^{2}\mathrm{F}_{\mathrm{scan}} + P_{\mathrm{addlog}} + P_{\mathrm{catrow}} + P_{\mathrm{catcol}}. \tag{38}$$

Where P_{addlog} is the power of additional logic and P_{catrow} and P_{catcol} are the power consumption of the two arbiter trees at rows and columns, respectively. The first term of P_{carb} is similar to the scanning register power consumption: C_{cwsr} is the capacitance of the row and column lines from cells to arbiter tree. A factor of 4 is implicit in the first term of P_{carb} and it represents factor of two for request and acknowledge signals and a factor of two for both row and column handshaking. The terms P_{catrow} and P_{catcol} are given, respectively, by

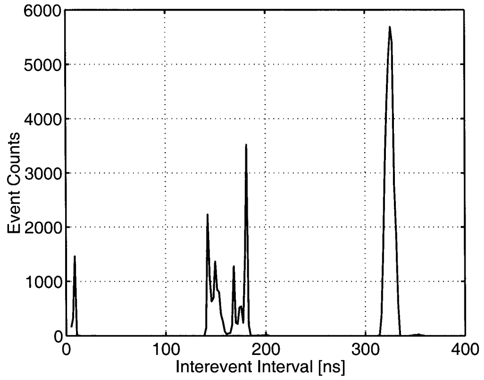


Fig. 12. Interevent timing of octopus retina chip.

$$\begin{aligned} P_{\mathrm{catrow}} =& \log_2\sqrt{N}P_{\mathrm{csa}}F_{\mathrm{event}}\left(1-\alpha_r\right)\\ &+\frac{\log_2\sqrt{N}}{\gamma}P_{\mathrm{csa}}F_{\mathrm{event}}\alpha_r\\ P_{\mathrm{catcol}} =& \frac{\log_2\sqrt{N}}{\sqrt{N}}P_{\mathrm{csa}}F_{\mathrm{event}}\left(1-\alpha_c\right)\\ &+\frac{\log_2\sqrt{N}}{\gamma\sqrt{N}}P_{\mathrm{csa}}F_{\mathrm{event}}\alpha_c. \end{aligned}$$

Term P_{catrow} is a binomial combination of terms, the second of which relates to bursty activity, the first to nonbursty activity. P_{csa} is the power consumption of a single arbiter cell during an event handshaking. α_r, α_c is respectively the fraction of the total number of events that consist of a burst in a row or a column. In this context, α_r and α_c assume an empirical value of 0.1. Nonbursty events have to undergo an arbitration that spans the entire tree size of $\log_2 N$ elements. Bursty events are arbitrated over a smaller portion of the tree ($log_2 N$ divided by a reduction factor γ) and, therefore, use less power. γ has an empirical value of 4 in this context. Column arbitration occurs on average $1/\sqrt{N}$ times less often than in rows; therefore, it also consumes less power.

V. Results and Discussion

Fig. 13 shows the power consumption of a ALOHA, scanning and arbitrated access circuits with respect to a varying number of cells N. Notice that for computing power consumption, the cell size of ALOHA and scanning sensory systems is half the size of the arbitrated system's cell. This choice is justified by examining the cell sizes in image sensors in the literature, where the addition of asynchronous arbitration practically doubles the cell size. Along with the above mentioned theoretical predictions, the figure includes a measured set of data from the power consumption of the octopus retina chip [13]. The measured data accounts for power consumption in the peripheral circuitry (arbitration and access) versus event rate. This figure can be easily utilized as an extension to predict power consumption versus array size. Since the array size is directly proportional to the event frequency F_{event}, once we suppose a common cell base frequency, it can be argued that the measurements can be ex-

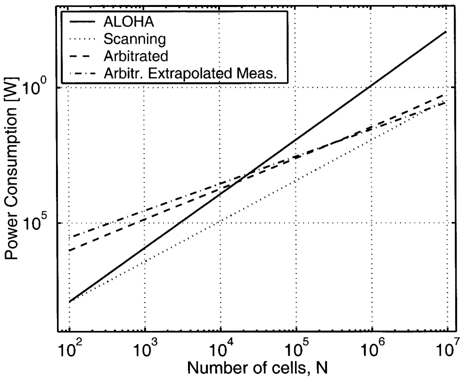


Fig. 13. Energy consumption per events for different access techniques.

TABLE I
ACCESS MODALITY

Access Type	Access Modality
Scanning Reg.	Externally driven
ALOHA or Priority Encoder	Self driven
CSMA 1-p	Self driven
Arbitrated	Self driven

trapolated to larger arrays. This extension suffers from a relatively small underestimation of the power consumption of the arbitration trees of bigger array. But arbitration increases with a logarithmic relation to the number of cells N, so the effect is minimal. Also, it can be take into account by adding the power consumption of bigger array to the data collected.

As can be observed, the model agrees well with the extrapolated data, and over-predicts it for higher size of the array, as expected. ALOHA access scheme requires a significant amount of power that increases with the number of cells directly, while scanning increases with the square root of N. Arbitration power consumption is dominated by the arbitration power, which is related to the logarithm of the square root of N, therefore, it has the lowest slope.

Depending on the desired application and particular sensory system under consideration, all the above described access techniques provide advantages and weak points. We will summarize in this section all the differences and provide scores for the quality metric of Section II-A. The designer can choose the appropriate access technique by inspecting the figures in this section and the design specifications.

Table I illustrates the access modality. The array of cells can be self driven, when itself begins transmission of data, or externally driven, in which case the production of an outside signal is necessary to access its data.

The ALOHA and PE access schemes produce very similar results, since they both rely on transmission at will of the data, therefore, they have been combined together for simplicity of analysis. CSMA 1-persistent is not taken into consideration in the following analysis either, for its similarity with ALOHA and the lack of inspired circuits in the literature. Table II illustrates the degree of expected throughput for both cases of low ($\sim$100) and high number ($\sim$10 k) of cells N. Low throughput generally means under-utilization of the channel; high throughput means saturation of the channel capacity.

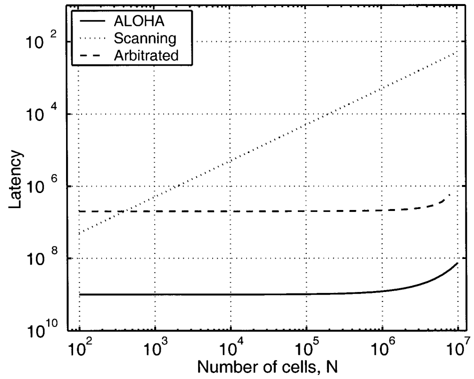


Fig. 14. Latency for different access types and $T_{\mathrm{chan}} = 100$ ns, $f_{\mathrm{event}} = 10$ Hz, $e_{\mu} = 0.001$.

TABLE II
THROUGHPUT FOR LOW AND HIGH NUMBER OF CELLS

Access Type	Throughput S (low N)	Throughput S (high N)
Scanning Reg.	Low	High
ALOHA or Priority Encoder	Low	Low
CSMA 1-p	Low	Low
Arbitrated	Low	High

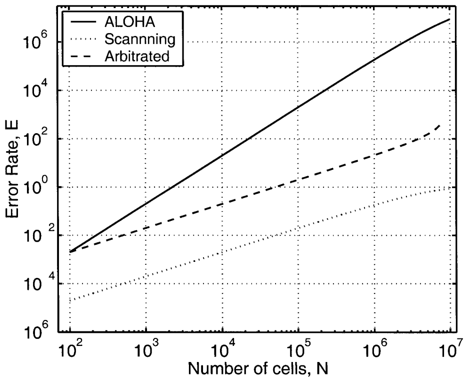


Fig. 15. Effective error rate for different access types and $T_{\mathrm{chan}} = 100$ ns, $f_{\mathrm{event}} = 10$ Hz, $e_{\mu} = 0.001$.

Figs. 14 and 15 report, respectively, a comparison of latency and error rates as a result of the access technique. Latency is proportional to the cell number N for scanning register access, while it is very low for ALOHA but degenerate rapidly with high number of cells. Arbitration latency remains constant throughout.

Errors are low in the arbitrated channel and scanning registers but increase with a factor of 2 on the exponent faster for ALOHA. This is due to the lack of buffering. Note that ALOHA and PE generate real output collisions and unusable data, while scanning and arbitration produce timing skews (which can be seen as errors) and not real data errors. All access schemes degenerate for high N, in fact the capacity of the channel has to be saturated before timing errors occur.

TABLE III
CIRCUIT COMPLEXITY AND POWER CONSUMPTION

Access Type	Circuit Complexity	Power Consumption
Scanning Reg.	Simple	Average
ALOHA or Priority Encoder	Simple	Very Low
CSMA 1-p	Average	Low
Arbitrated	Complex	High

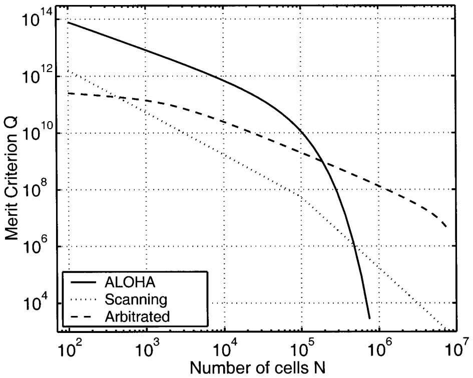


Fig. 16. Comparison for different access types and $d_{ch} = 100\ \mathrm{ns}$, $f_{\text{event}} = 10$ Hz, $e_{\mu} = 0.001$.

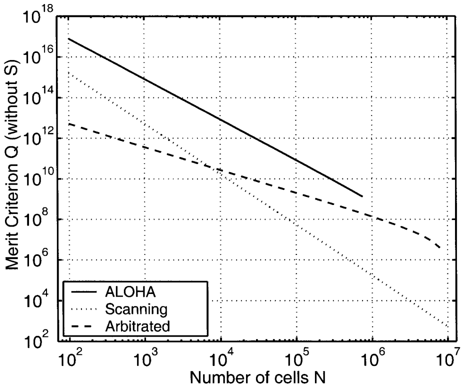


Fig. 17. Comparison for different access types and $d_{ch} = 100\ \mathrm{ns}$, $f_{\text{event}} = 10$ Hz, $e_{\mu} = 0.001$. Here, the throughput is not taken into account.

Table III summarizes circuit complexity and power consumption estimates for each access type. When silicon area is precious simpler access types can be chosen. The same consideration applies for power supply. Arbitration, although reporting the best results in terms of channel utilization and equivalent timing errors, is the most complex and power-hungry circuit. This however scales favorable in deep sub-μm technologies. ALOHA has the lowest power consumption given its simple circuit realization.

A full comparison of all the techniques is reported in Fig. 16. The figure plots the quality metric Q introduced in Section II-A. ALOHA and PEr have been combined for simplicity. The simplicity of ALOHA, its low power consumption and latency for lightly loaded channels is clearly visible but degenerates rapidly for loaded channel, i.e., large number of cell population N. On the other hand arbitration remains thoroughly superior to scanning and results the best access technique for highly populated channels.

Fig. 17 compares the access techniques factoring out the effect of the throughput in the quality metric quality metric Q. The recent development of very fast serial buses that have channel capacities in the Giga-words/s well in excess to the 10 Mega-samples/sec considered in most of our calculations. When these high bandwidth I/O systems are employed [25], the throughput is no more a deciding factor and Fig. 17 is more relevant in a comparative study.

REFERENCES

[1] H. B. Barlow, , MIT Press, Cambridge, MA, 1961.

[2] C. A. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.

[3] A. Mortara and E. Vittoz, "A communication architecture tailored for analog VLSI artificial nenral networks: Intrinsic performance and limitations," *IEEE Trans. Neural Networks*, vol. 5, pp. 459–466, May 1994.

[4] K. A. Boahen, "Point-to-point connectivity between neuromorphic chips using address events," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 416–434, May 2000.

[5] L. M. Reyneri, *On the Performance of Pulsed and Spiking Neurons*. Boston, MA: Kluwer, 2002, vol. 30, pp. 101–119.

[6] A. Apsel and A. G. Andreou, "Quality of data reconstruction using stochastic encoding and an integrating receiver," in *Proc. 43rd Midwest Symp. Circuits Systems*, Ames, MI, Aug. 2000, Best Student Paper Award, pp. 183–186.

[7] M. A. Mahowald, "VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function," Ph.D. dissertation, California Inst. Technol., Pasadena, 1992.

[8] M. Sivilotti, "Wiring Considerations in Analog VLSI Systems With Applications to Field Programmable Networks," Ph.D. dissertation, California Inst. Technol., Pasadena, 1991.

[9] J. Lazzaro, J. Wawrzynek, M. Mahowald, M. Sivilotti, and D. Gillespie, "Silicon auditory processors as computer peripherals," *IEEE Trans. Neural Networks*, vol. 4, pp. 523–528, May 1993.

[10] A. Apsel and A. G. Andreou, "An analysis of data reconstruction efficiency using stochastic encoding and an integrated receiver," *IEEE Trans. Circuits Syst.*, submitted for publication.

[11] C. S. Li, K. N. Sivarajan, and D. G. Messerschmitt, "Statistical analysis of timing rules for high-speed synchronous VLSI systems," *IEEE Trans. VLSI Syst.*, vol. 7, pp. 477–482, Dec. 1999.

[12] A. F. Murray, D. Del Corso, and L. Tarassenka, "Pulse-stream VLSI neural networks mixing analog and digital techniques," *IEEE Trans. Neural Networks.*, vol. 2, pp. 193–204, Mar. 1991.

[13] E. Culurciello, R. Etienne-Cummings, and K. A. Boahen, "A biamorphic digital image sensor," *IEEE J. Solid-State Circuits*, vol. 38, pp. 281–294, Feb. 2003.

[14] A. S. Tanenbaum, *Computer Networks*. Upper Saddle River, NJ: Prentice-Hall, 1996.

[15] M. Barbaro, P. Y. Burgi, A. Mortara, P. Nussbaum, and F. Heitger, "A 100 × 100 pixel silicon retina for gradient extraction with steering filter capabilities and temporal output coding," *IEEE J. Solid-State Ciruits*, vol. 37, pp. 160–172, Feb. 2002.

[16] K. A. Bonhen, *Communicating Neuronal Ensembles Between Neuromorphic Chips, Neuromoirphic Systems Engineering*. Boston, MA: Kluwer, 1998, ch. 11, pp. 229–261.

[17] G. E. Keiser, *Local Area Networks*. New York: McGraw-Hill, 1989.

[18] A. Mortara, E. Vittoz, and P. Venier, "A communication scheme for analog VLSI perceptive systems," *IEEE J. Solid-State Circuits*, vol. 30, pp. 660–669, June 1995.

[19] M. Movin, A. Abusland, and T. Lande, "A VLSI communication architecture for stochastically pulse-encoded analog signals," in *ISCAS 1996*, vol. 3, Atlanta, GA, 1996, pp. 401–404.

[20] P. Furth and A. G. Andreou, "A design framework for low power analog filter banks," *IEEE Trans. Circuits Syst. I*, vol. 42, pp. 966–971, Nov. 1995.

[21] G. Cauwenberghs, N. Kumar, W. Himmelbauer, and A. G. Andreou, "An analog VLSI chip with asynchronous interface for auditory feature extraction," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 600–606, May 1998.

[22] Z. Kalayjian and A. G. Andreou, "Asynchronous communication of 2d motion information using winner-take-all arbitration," *J. Analog Integrated Circuits Signal Processing*, vol. 103–109, p. 13, Mar./Apr. 1997.

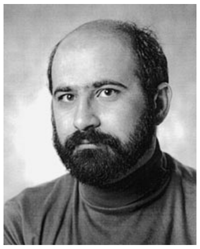
[23] L. Kleinrock, *Queueing Systems*. New York: Wiley, 1976, vol. 1–2.

[24] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*. New York: Wiley, 1998.

[25] K. Yang, T. Lin, and Y. Ke, "A scalable 32 gb/s parallel data transceiver with on-chip timing calibration circuits," in *Proc. 2000 Int. Conf. Solid-State Circuits*, Feb. 2000.

Eugenio Culurciello received the M.S. degree from the University of Trieste, Trieste, Italy, in 1997 and the M.S. degree from the Johns Hopkins University, Baltimore, MD, in 1999, where he is pursuing the Ph.D. degree in electrical and computer engineering.

He is an Assistant Researcher at Johns Hopkins University. His interests are artificial vision and neural-morphology, efficient biomimetic communication channels, and wireless sensors.



Andreas G. Andreou (S'80–M'81) received the Ph.D. degree in electrical engineering and computer science from the Johns Hopkins University (JHU), Baltimore, MD, in 1986.

From 1986 to 1989, he was a Postdoctoral Fellow and Associate Research Scientist with the Electrical and Computer Engineering Department, while also a Member of the Professional Staff at the Johns Hopkins Applied Physics Laboratory. He became Assistant Professor of electrical and computer engineering in 1989, Associate Professor in 1993, and Professor in 1996. He is the Founding Director of the Whitaker Lithography and Fabrication Facility at JHU. In 1995 and 1997, he was a Visiting Associate Professor and Visiting Pprofessor, respectively, in the computation and neural systems program at the California Institute of Technology, Pasadena. In summer 2001, he was a Visiting Professor at Tohoku University, Tohoku, Japan. He now holds appointments in the electrical and computer engineering, computer science and Whitaker biomedical engineering institute. His research interests include integrated circuits, sensory information processing, and neural computation. He is a coauthor of *Adaptive Resonance Theoiy Microchips* (Boston, MA: Kluwer, 1998). He is Associate Editor of the journal *Neural Networks*.

Dr. Andreou received a National Science Foundation Research Initiation Award and he is the Co-F ounder of the Center for Language and Speech Processing, JHU. He received the 2000 IEEE Circuits and Systems Society Darlington Award.