# Multi-Chip Neuromorphic Motion Processing

Charles M. Higgins and Christof Koch

Division of Biology, 139-74
California Institute of Technology
Pasadena, CA 91125
[chuck,koch]@klab.caltech.edu

January 21, 1999

## Abstract

*We describe a multi-chip CMOS VLSI visual motion processing system which combines analog circuitry with an asynchronous digital interchip communications protocol to allow more complex motion processing than is possible with all the circuitry in the focal plane. The two basic VLSI building blocks are a sender chip which incorporates a 2D imager array and transmits the position of moving spatial edges, and a receiver chip which computes a 2D optical flow vector field from the edge information. The elementary two-chip motion processing system consisting of a single sender and receiver is first characterized. Subsequently, two three-chip motion processing systems are described. The first such system uses two sender chips to compute the presence of motion only at a particular stereoscopic disparity. The second such system uses two receivers to simultaneously compute a linear and polar topographic mapping of the image plane, resulting in information about image translation, rotation, and expansion. These three-chip systems demonstrate the modularity and flexibility of the multi-chip neuromorphic approach.*

## 1: Introduction

Focal plane image processing can only be taken to a certain level of complexity without incurring an unacceptably large pixel size. As we proceed towards smart sensor designs incorporating more and more stages of pixel-parallel processing, we must either increase our process resolution, resulting in higher costs and lower imager fill factors, or limit the processing that occurs in the focal plane. If an intermediate computation can be communicated off of the photosensitive chip without losing the advantages of focal plane computation, the effective processing in the focal plane can be extended while retaining practical pixel resolutions. However, to retain the advantages of single chip continuous-time focal plane image processors, this communication must be done without incurring significant delays, dramatically increasing power consumption, or introducing temporal aliasing.

The interchip communications protocol described below provides a way of accomplishing this feat. Data is communicated asynchronously at low latency, allowing a representation of events in continuous time. Communication between chips only occurs when the input changes, thus making power consumption activity-dependent.

In this paper, we describe a CMOS VLSI chip pair designed on neuromorphic principles which computes real-time optical flow using this communications protocol. The sender chip contains an array of photoreceptors and nonlinear differentiators which produce a voltage pulse upon a sudden change in local image intensity (presumably corresponding to a moving spatial edge). These voltage pulses are communicated across a digital bus to a motion processing receiver chip, which computes the local velocity of motion by noting the order and timing in which edges arrive. The motion vectors are then serially scanned out of the receiver chip for display.

After characterizing the basic VLSI building blocks, we provide two examples of three-chip motion processors which can compute more complex visual motion data products.

## 2: Related Work

The Address-Event Representation (AER) was originally envisioned by Mahowald [14] as a circuit analogy to the optic nerve. As such, it was first used to transmit visual signals out of a silicon retina. The protocol has since been strengthened and formalized by Boahen [2] for the same purpose. Several variants and specializations of the scheme have emerged in the last few years [13, 15, 18, 5].

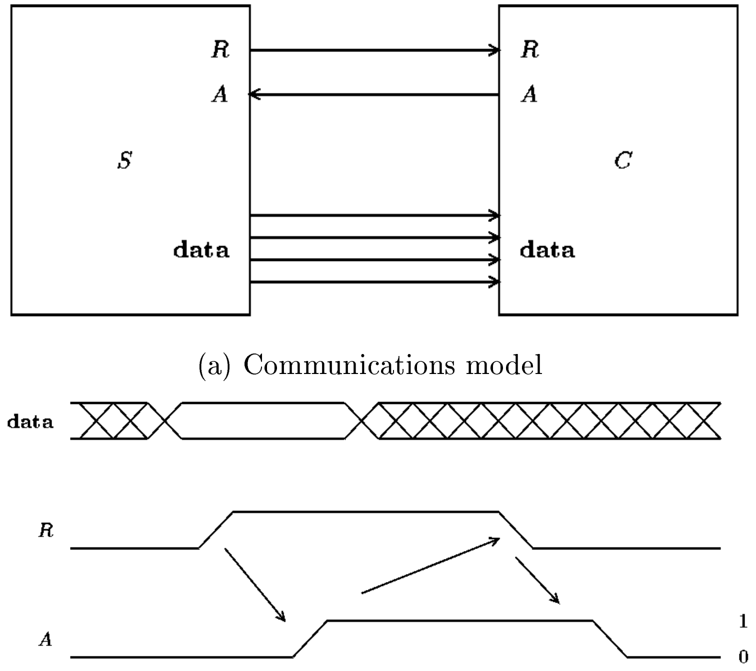
While applications of interchip communication are still in the early stages, the results so far are quite promising. Boahen [1] has interfaced two silicon retinas to three receiver chips to implement binocular disparity-selective elements. Venier et al. [17] have used an asynchronous interface to a silicon retina to implement orientation-selective receptive fields. Whatley et al. [18] are implementing a silicon model of primate visual cortex using interchip communication, and DeWeerth et al. [5] are implementing a model of leech intersegmental coordination. Andreou et al. [6] have demonstrated the use of EPROMs for linear or nonlinear address remapping in interchip communication. Kumar et al. [12] have provided an auditory front-end chip with an asynchronous interface for further off-chip processing.

Kalayjian et al. [9] have created a photosensitive sender chip with similar function to the one presented in this paper: an array of photoreceptors and temporal derivative circuits are used to communicate the presence of local temporal illumination changes across a digital bus. This chip differs from the present work in two ways. Firstly, it differs in the use of a temporal derivative circuit rather than the highly nonlinear temporal edge detector used here. Secondly, the communications scheme used in [9] is based on a winner-takes-all arbitration, rather than the binary tree arbitration used in this paper.

Indiveri and Kramer [8] have proposed a very similar multi-chip motion processor to the present work.

## 3: Interchip Communications Protocol

The original and most basic form of AER utilizes two digital control lines and several digital data lines to interface a sender to a receiver, as shown in Figure 1. The protocol is used to communicate the occurrence of an event from sender to receiver. A full four-phase handshake between sender and receiver guarantees synchronization between chips; the data lines communicate the address of the requesting sender pixel to the receiver chip. The protocol effectively allows a sender pixel on one chip to communicate digital spikes to a receiver pixel. Because requests can come at any time from any pixel in the array, it is necessary to use an arbitration scheme to serialize simultaneous events onto the single communications bus. However, because the asynchronous protocol operates so quickly (on nanosecond scales), this serialization is usually benign.



(a) Communications model

(b) Communications protocol

Figure 1: AER protocol summary. In (a), the model for AER transmission is shown: a sender chip ($S$) communicates with a receiver chip ($C$) via request $R$, acknowledge $A$ and data lines. In (b), the protocol for transmission using the above control and data lines is shown: a request with data leads to an acknowledgment, which in turn leads to falling request and falling acknowledge.

The circuitry necessary to implement the protocol varies from scheme to scheme. The particular hardware implementation of AER used in this chipset has been newly devised by Boahen; refer to the paper by Boahen [3] in this proceedings for further details.

## 4: Photosensor Sender Chip

### 4.1: Sender Architecture

The core of the sender chip is a 12 × 12 array of sender pixels. See Figure 2 for a layout diagram. Each sender pixel contains an adaptive photoreceptor [4] and a nonlinear differentiator circuit [10] interfaced to the interchip communication circuitry. The photoreceptor adapts to the local light intensity on slow time scales (a few seconds), allowing high sensitivity to transient changes over a wide range of illumination without a change in bias settings. The nonlinear differentiator circuit produces a current pulse when the photoreceptor output changes suddenly. This combination of adaptive photoreceptor and nonlinear differentiator is referred to as a temporal edge detector. When an illumination edge passes over the pixel, the event is communicated to the receiver. In this implementation, events are communicated on the bus only when the illumination changes, resulting in an efficient use of bus bandwidth. Arbitration, address encoding, and other interface circuitry to support the protocol are located at the periphery and described in [3]. The chip also incorporates a serial scanner for readout of the raw photoreceptor image.

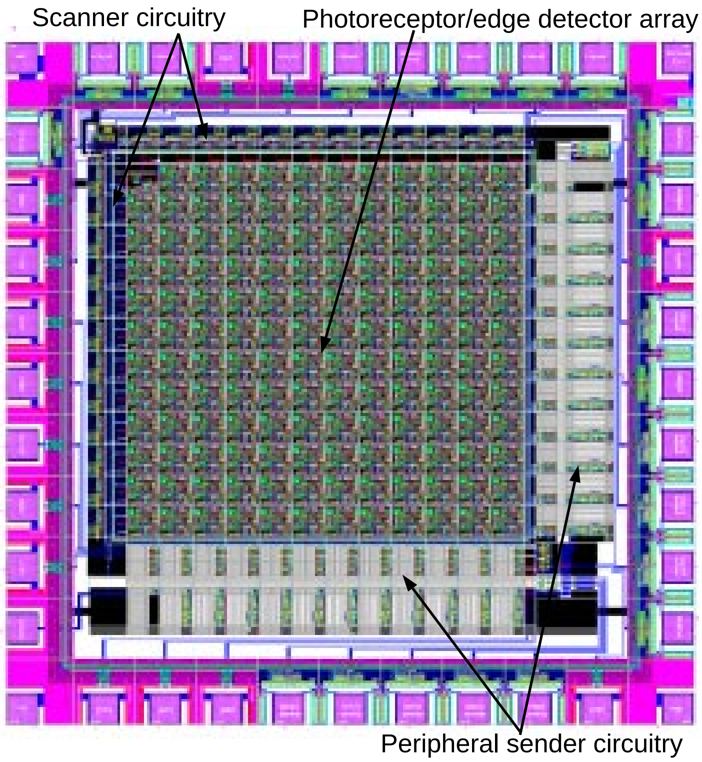

Scanner circuitry
Photoreceptor/edge detector array
Peripheral sender circuitry

Figure 2: Layout of the sender chip, as fabricated in a 1.2 $\mu m$ standard CMOS process.

The sender pixel communications interface circuit, shown in Figure 3, is slightly modified from [3]. It takes as its input $I_{in}$ the current from the nonlinear differentiator circuit. Before a request is made, $R_{pix}$ is (inactive) low, $A_{pix}$ is (inactive) high, and $D_{pix}$ is inactive (low). When sufficient current is integrated on node $V_{mem}$ that it overcomes the threshold set by $V_{thr}$, $V_{rp}$ is pulled low and the wired-OR $R_{pix}$ shared by all pixels in the row is pulled high. When $A_{pix}$ returns low from the row arbiter, it simultaneously resets $V_{mem}$ to $V_{dd}$ (and thus releases $R_{pix}$) and pulls up the wired-OR $D_{pix}$ shared by all pixels in the column. $D_{pix}$ will be held high until $A_{pix}$ returns to inactive high. This circuit implements the required sender pixel protocol. The pass transistor connected to the input node (a modification from the Boahen circuit) cuts off the input current during the reset phase, allowing stable reset even in the presence of large input currents. The transistor connected to $V_{recov}$ interposed in the reset pathway is a second modification from the Boahen circuit and allows control of the speed of reset, effectively setting the maximum spike rate. Finally, a leak transistor ($V_{leak}$) allows a minimum input current threshold to be set.

### 4.2: Sender Performance

In this section, we characterize the sender array's AER bus response to changes in light intensity. During the period of time when the nonlinear differentiator's current output is large enough to overcome the leakage current, multiple events (hereafter referred to as a *burst of spikes*) are created on the AER bus. A typical burst from a single pixel is shown in Figure 4. Bus availability for each spike in the burst is arbitrated *independently*, so the burst from a particular pixel will, in general, appear on the interchip bus interleaved with

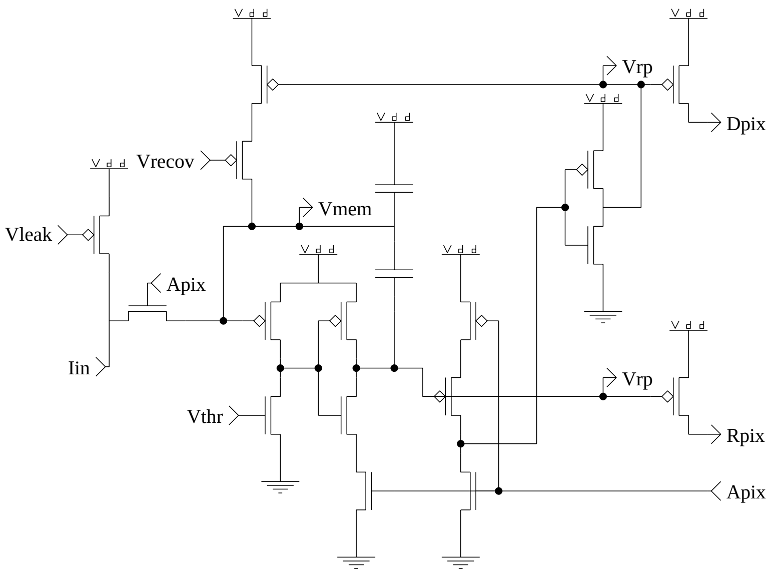


Figure 3: Sender pixel communications interface circuitry

requests from other pixels. Three major parameters of these bursts are key to the proper operation of the motion receiver chip: burst width, latency from stimulation, and spike rate during the burst. For characterization purposes, these three parameters have been measured as the chip is visually stimulated with the sender chip's interchip request line tied back to its own acknowledge line. This self-acknowledge yields the fastest possible event cycle, taking approximately 100 *ns* per request-acknowledge cycle.

In order for bursts from neighboring pixels to be seen as subsequent, the *ends* of the bursts from two subsequently crossed pixels must occur in the correct order. For this reason, the variation in the burst width must be small relative to the inter-pixel transit time for reliable operation. Figure 5 shows the burst width produced by the center pixel of the array when given individual stimulation. This plot is extremely representative of pixels in the sender array. The nonlinear differentiator has been tuned to be sensitive to slow speeds, and its response falls off at higher speeds. Due to this tuning, it is not possible for the burst width variation to cause unreliable operation in this system. However, when stimulus speeds are faster than approximately 3 pixels/sec., the spatial variation in burst width increases significantly.

If burst order is to be reliably preserved, the latency between photoreceptor stimulation and burst generation must not vary significantly between sender pixels. Note that the absolute value of latency is not terribly significant; it just introduces a delay between stimulation and optical flow response. The latency of multiple sender pixels has been measured over the entire contrast/velocity range of function shown in Figure 5, and is relatively stimulus independent except at very low speeds. Mean latency was measured with the computer stimulus to be approximately 30 *ms* over a wide stimulus range, and varies with a standard deviation of less than 5 *ms* between sender pixels.

In addition to the stimulus-related bursts, spontaneous random events due to leakage currents occur at approximately 0.1 Hz. Because of this, the receiver chip must be tuned to respond *only* to bursts, and the burst rate must be maintained high enough to create a receiver response even under high load conditions. The burst rate has been measured for multiple pixels over the entire stimulus range, and is relatively stimulus independent with a mean of approximately 160 kHz.

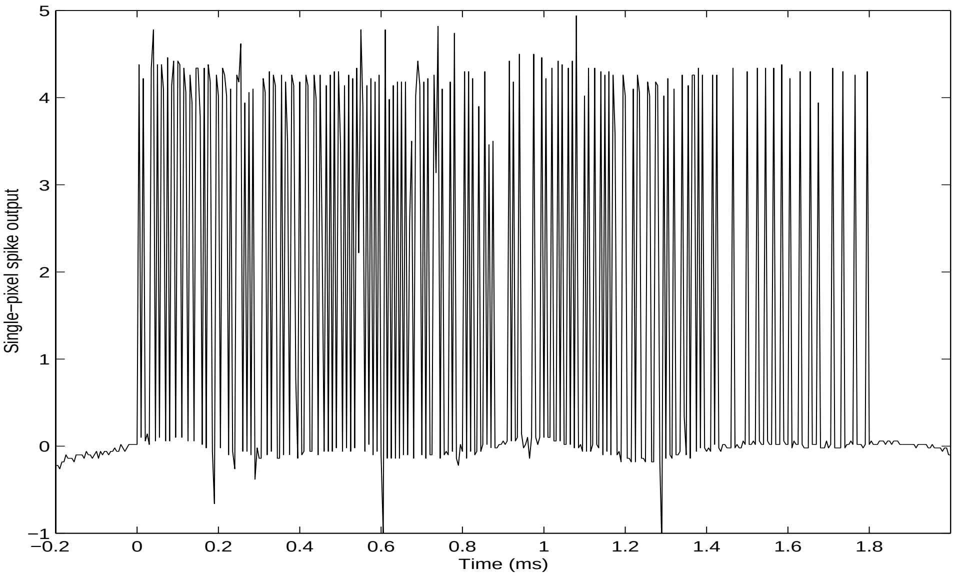


Figure 4: Sender pixel transient response: this spike burst is the response of an individual pixel to a passing edge. Because true spike width is approximately 50 $ns$ and spike separation is on the order of 6 $\mu s$, the spikes have been lengthened to make them visible. This burst peaks at a spike rate of approximately 160 kHz and encompasses around 200 individual spikes.

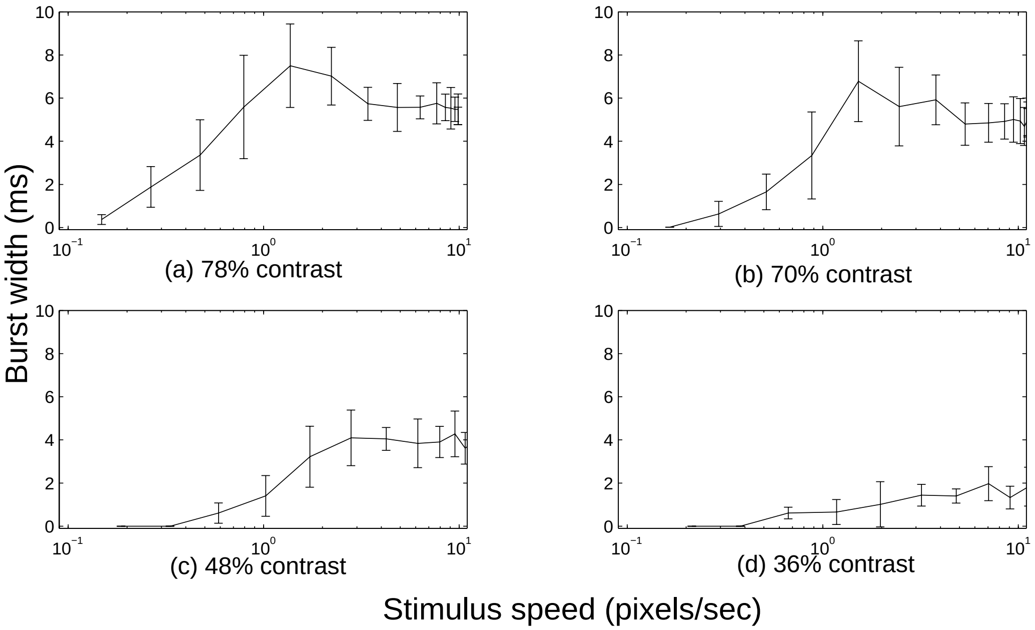


Figure 5: Sender pixel temporal contrast response: burst width from the center pixel in the array is shown as stimulus speed and contrast are varied. Error bars represent standard deviation over 10 stimulus presentations. Pixel was stimulated with a blinking stimulus which slowly rose to the desired contrast and then fell with a controlled speed to zero intensity. Effective stimulus speed can be calculated from the geometry of the implementation. No significant response was seen for 22% contrast.

## 5: Motion Receiver Chip

### 5.1: Receiver Architecture

The core of the receiver chip is a 13 × 15 array of receiver pixels. See Figure 6 for a layout diagram. Each receiver pixel contains the communications interface and a motion circuit implementing a 2D version of the FS (Facilitate-and-Sample) velocity algorithm [11]. The velocity of a moving edge is computed by measurement of the time between subsequent edges. The motion circuitry takes as input a current pulse from the interface circuit. Address decoding and interface circuitry to support the protocol are located at the periphery and described in [3]. This chip also incorporates a serial scanner for readout of the 2D optical flow vectors.

The receiver pixel communications interface circuit, shown in Figure 7, is far simpler than its sender counterpart, and is changed from [3] only by the addition of a current-limiting transistor ($V_{thr}$). When $X_{sel}$ and $Y_{sel}$ are both active high, the source of the limiting transistor is pulled low and a current whose magnitude is set by $V_{thr}$ flows into the motion circuit. The indirectness of this circuit is to avoid charge-pumping, which leads to a small "leakage" current even if $X_{sel}$ and $Y_{sel}$ are only asserted at non-overlapping times.

### 5.2: Receiver Performance

In this section, we evaluate the output of the dual-chip motion processor as a whole by measuring receiver chip responses to visual stimuli. For the purposes of this paper, the gain of the FS sensor velocity output has been increased to the point where only the local direction of motion is represented. The request-acknowledge cycle in this system takes approximately 400 $ns$. In Figure 8, the percentage of 2D optical vectors within 15 degrees of the correct stimulus orientation is plotted against stimulus speed and contrast. The low-speed threshold agrees with that seen in the sender chip. Above approximately 3 pixels/sec., the correct response probability falls off due to increasing variability in the temporal edge detectors. Correct orientation is calculated over more than an order of magnitude in speed and down to less than 30% contrast.

## 6: Dual-Sender Motion Processor

In this section, we describe a motion processing system which uses two sender chips and a single motion receiver to compute motion tuned to a particular optical disparity. This results in a strong motion response only at a particular depth from the imager.

### 6.1: Dual-Sender Architecture

See Figure 9 for a block diagram of the hardware system. In order to converge the asynchronous requests from two sender chips, a fundamental requirement for this system is a two-input arbiter [14] to decide which request will be passed through to the single receiver. Given the choice bit from this arbiter, the appropriate address is multiplexed onto the receiver address. An EPROM is included for static address remapping; the choice bit is also input to the EPROM to allow different mappings for the two sender chips.

In order to create a disparity tuned motion processor, the rows of the two sender chips are mapped in an interlaced fashion onto the receiver chip. Hardware remapping of addresses with the EPROM is used to implement this algorithm as shown in Figure 10.

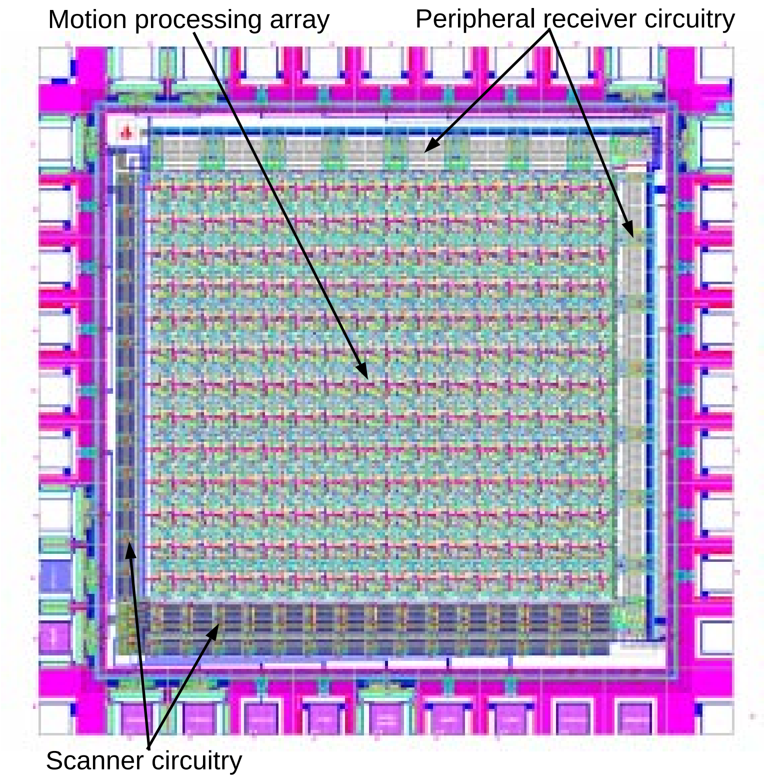


Figure 6: Layout of the receiver chip, as fabricated in a 1.2 $\mu m$ standard CMOS process.

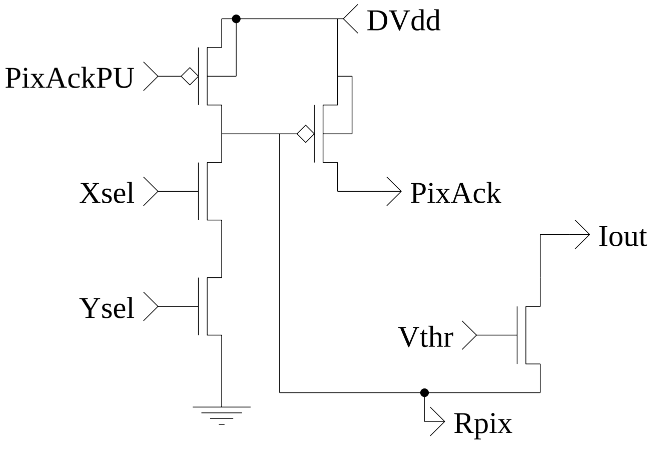


Figure 7: Receiver pixel communications interface

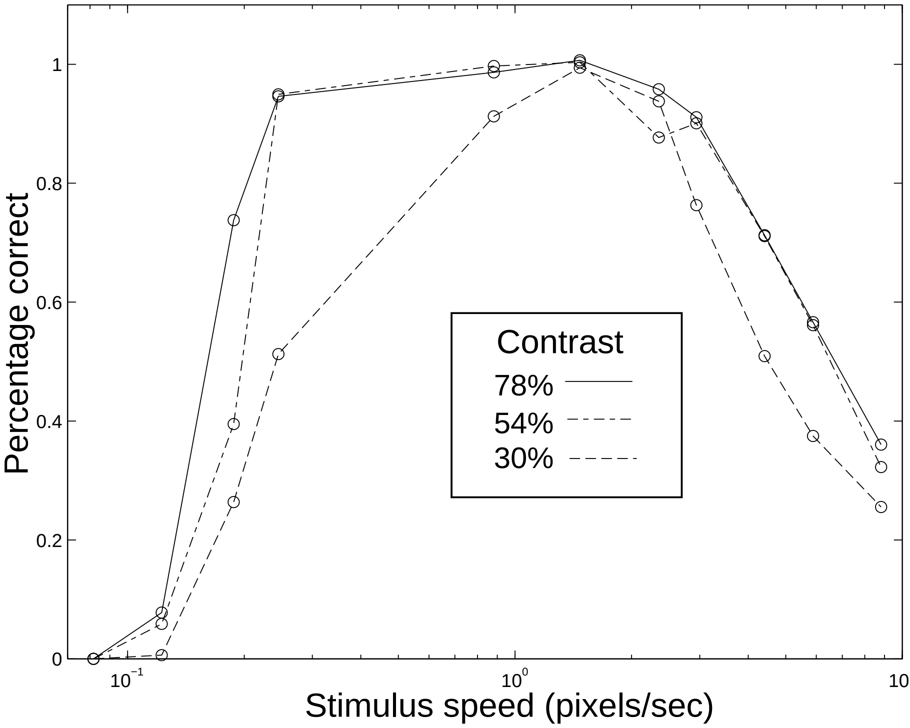


Figure 8: Receiver chip temporal contrast response: the dual-chip motion processor was stimulated with a variable-speed rotating drum stimulus. The percentage is calculated across the entire array as the number of vectors within 15 degrees of the correct stimulus angle.

Space does not permit a detailed analysis of this interlaced-rows algorithm. However, the basic idea is that, because the motion receiver chip expects to see rows fire in sequence as an edge passes over, interlacing the rows from the two sender chips introduces a preference for motion at a particular disparity. A stimulus moving at the preferred disparity is the only condition in which all the rows of the receiver chip will fire in sequence, and thus all motion vectors will point in the same direction. This preferred disparity is set by the relative position of the two sender mappings implemented in the EPROMS, but could be changed in real-time if desired by using additional EPROM address bits.

### 6.2: Dual-Sender Performance

To test the disparity tuning of the dual-sender system, a computer stimulus (diagrammed in Figure 11) was used to simultaneously present two moving vertical bars, only one of which was visible to each sender chip. The disparity between the two stimuli was varied precisely under computer control. Figure 12 shows the result of this experiment. The average $X$ output of the entire array is plotted against stimulus disparity. The chip shows a clear preference for a particular disparity near zero. The request-acknowledge cycle in this system takes approximately 400 $ns$.

Due to the interlaced-rows scheme used to implement the disparity tuning, it is necessary to spatially average outputs from at least two neighboring rows to see disparity selectivity. The average of the entire chip shows the most robust tuning.

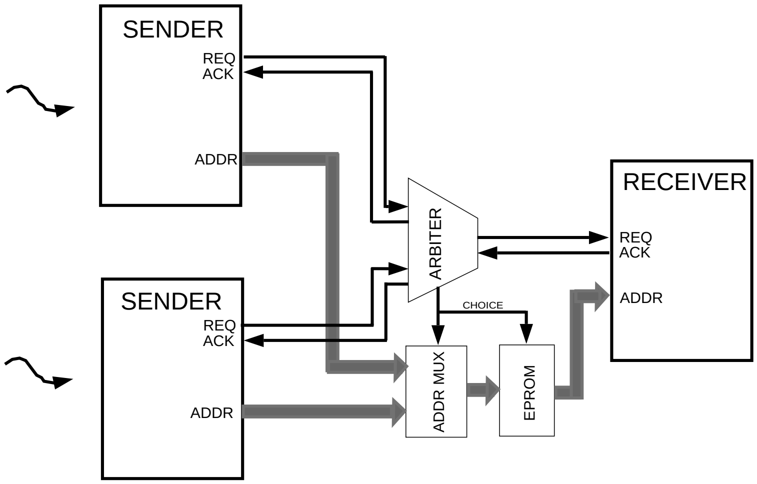


Figure 9: Dual-sender hardware architecture: the arbiter is a standard two-input asynchronous arbiter [14] built out of discrete logic; not shown is an analog delay on the request line at the output of the arbiter to allow address setup time. Note that, aside from the custom VLSI components described, only discrete logic is used.

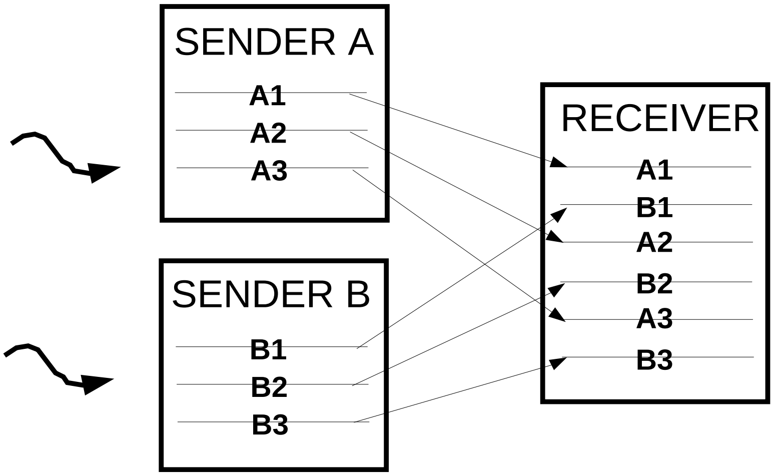


Figure 10: Dual-sender address mapping: rows from the two senders are interlaced on the receiver chip.

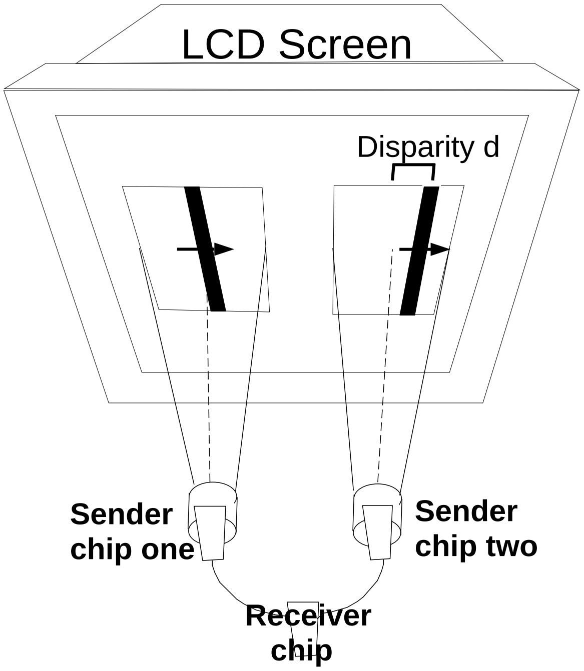


Figure 11: Dual-sender stimulus diagram: separate moving bar stimuli were presented simultaneously to each sender chip on the same LCD screen. The disparity between the two bars (that is, the difference in the horizontal position of the bar in the two images) was varied under computer control.

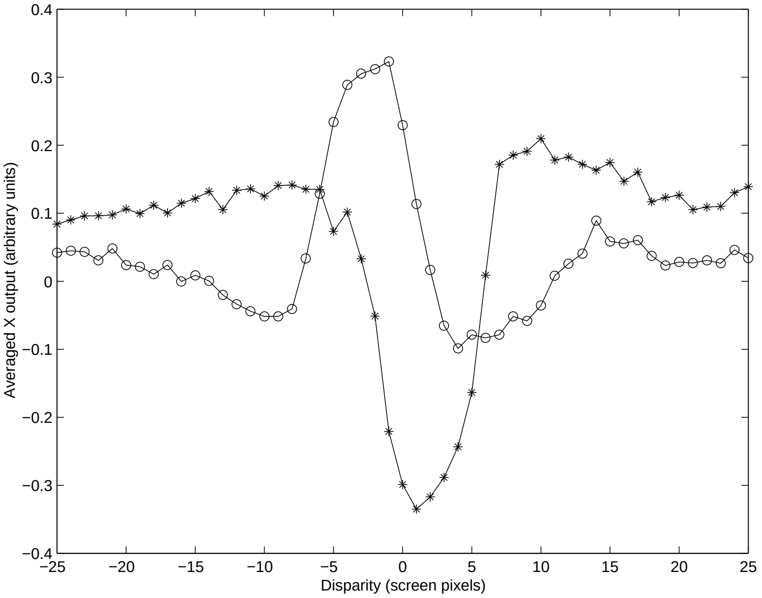


Figure 12: Dual-sender disparity tuning: as the disparity of a dichoptic drifting vertical bar stimulus is varied, the averaged $X$ output of the receiver chip is plotted. This output is the spatial average of the $X$ component of every optical flow vector in the receiver array. It is also temporally averaged over one period of the stimulus to remove the effects of periodic variation. Circles indicate the response to a leftward-moving bar; asterisks indicate the response to a rightward-moving bar.

## 7: Dual-Receiver Motion Processor

In this section, we describe a motion processing system which uses a single sender chip and two identical motion receivers with different topological mappings of the image plane.

### 7.1: Dual-Receiver Architecture

See Figure 13 for a block diagram of the hardware system. Because two receiver chips are present, circuitry is necessary to ensure that both receiver chips have acknowledged the single sender event before the system continues. This circuit is known as a C-element [16]. Two EPROMs are included for parallel static remapping of both receiver destination addresses.

The first receiver uses a pass-through sender address mapping, which generates the same sort of optical flow field characterized in Section 5. The second receiver uses a polar coordinate mapping: let the polar coordinates of a sender pixel be described by

$$R = \sqrt{(X_{sndr} - X_{mid})^2 + (Y_{sndr} - Y_{mid})^2}$$

$$\theta = tan^{-1}((Y_{sndr} - Y_{mid})/(X_{sndr} - X_{mid}))$$

where $(X_{mid}, Y_{mid})$ is the center pixel address of the sender array. Then the receiver mapping can be described as the nearest integer to

$$X_{rcvr} = S_x \cdot R$$

$$Y_{rcvr} = S_y \cdot \theta$$

where $S_x$ and $S_y$ are chosen to maximally cover the receiver array. This remapping makes the second receiver sensitive to expanding and rotating motions. A pure expansion corresponds to movement only along the radial coordinate (remapped $X$). A pure rotation corresponds to movement only along the angular coordinate (remapped $Y$). Note that such motion must be centered on the sender chip for a maximal response.

### 7.2: Dual-Receiver Performance

To demonstrate the particular sensitivity of each receiver, we first present a moving bar stimulus (as shown in Figure 11, but with only one sender chip) and observe the array average $X$ coordinate from each receiver chip. Figure 14(a) shows the responses as the angle of the moving bar is varied. The linearly mapped array shows a strong directionally-selective response, whereas the polar-mapped array shows little selectivity. In Figure 14(b) the same outputs are shown in response to a stimulus composed of expanding circles as the position of the focus of expansion is swept across the sender chip. The output of the linearly-mapped array reflects the position of the focus of expansion, as explained in [7]. The output of the polar-mapped array is strongly negative, indicating the presence of expansion, and peaks in strength when the focus of expansion is at the center of the sender chip. The request-acknowledge cycle in this system takes approximately 500 $ns$.

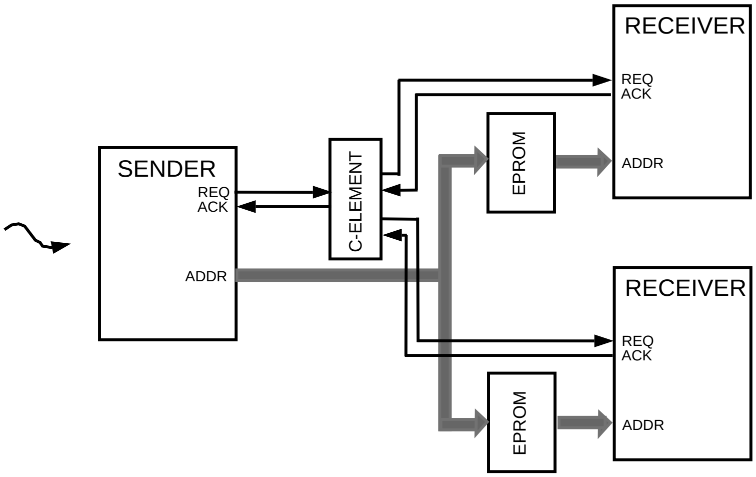


Figure 13: Dual-Receiver Architecture: the C-element is a standard asynchronous communications building block [16] (built out of discrete logic); not shown are timeout circuits to handle nonexistent receiver addresses, an analog delay on both receiver request lines to allow for address setup time, and EPROM enabling circuitry. Note that, aside from the custom VLSI components described, only discrete logic is used.

## 8: Discussion

We have described a flexible, modular, multi-chip neuromorphic motion processing system which retains many of the advantages of single-chip motion processors while allowing for significant further expansion. In addition to characterizing the elementary motion processor, we have shown two three-chip systems which compute more complex real-time motion data products.

The dual-receiver architecture we have demonstrated can be programmed with arbitrary topological mappings of the image plane, which can be used to perform a number of image processing tasks. In addition, the visual motion caused by changes in angle of the imaging platform can be compensated for by providing information about camera angle to the EPROMs. This can be used to compensate for unintentional camera jitter, as well as programmed movements of the camera.

A second technique for computing disparity-tuned motion with the dual-sender architecture would be to map corresponding pixels from each sender to the same receiver pixel and require a coincidence of bursts to create a motion output. This correlation-based motion approach would require a nonlinear threshold on the motion receiver chip to make a strong distinction between one burst and a coincident pair.

Multi-chip systems such as these will make hardware implementations of complex multistage image processors like those suggested by biological vision systems a feasible prospect.

## Acknowledgments

The authors gratefully acknowledge Kwabena Boahen for his copious assistance in explaining his implementation of the AER protocol, and would also like to thank Timothy Horiuchi for helpful suggestions. This research was supported by the Center for Neuromorphic Systems Engineering as a part of the National Science Foundation's Engineering Research Center program as well as by the Office of Naval Research.

# References

[1] K. Boahen. *NSF Neuromorphic Engineering Workshop Report.* Telluride, CO, 1996.

[2] K. Boahen. Retinomorphic vision systems. In *Proceedings of the International Conference on Microelectronics for Neural Networks and Fuzzy Systems.* IEEE, 1996.

[3] K. Boahen. A throughput-on-demand 2-D address-event transmitter for neuromorphic chips. In *Proc. of the 20th Conference on Advanced Research in VLSI*, Atlanta, GA, 1999.

[4] T. Delbrück and C. Mead. Analog VLSI phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits. Technical Report 30, Department of Computation and Neural Systems, California Institute of Technology, 1993.

[5] S. DeWeerth, G. Patel, M. Simoni, D. Schimmel, and R. Calabrese. A VLSI architecture for modeling intersegmental coordination. In *Proc. of the 17th conference on Advanced Research in VLSI*, Ann Arbor, MI, 1997.

[6] S. Grossberg, G. Carpenter, E. Schwartz, E. Mingolla, D. Bullock, P. Gaudiano, A. Andreou, G. Cauwenberghs, and A. Hubbard. Automated vision and sensing systems at Boston University. In *Proc. of the DARPA Image Understanding Workshop*, New Orleans, LA, 1997.

[7] C. M. Higgins and C. Koch. An integrated vision sensor for the computation of optical flow singular points. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11, Cambridge, MA, 1999. MIT Press.

[8] G. Indiveri and J. Kramer, 1998. Personal Communication.

[9] Z. Kalayjian and A. G. Andreou. Asynchronous communication of 2D motion information using winner-takes-all arbitration. *Analog Integrated Circuits and Signal Processing*, 13:103–109, 1997.

[10] J. Kramer, R. Sarpeshkar, and C. Koch. An analog VLSI velocity sensor. In *Proc. Int. Symp. Circuit and Systems (ISCAS)*, pages 413–416, Seattle, WA, May 1995.

[11] J. Kramer, R. Sarpeshkar, and C. Koch. Pulse-based analog VLSI velocity sensors. *IEEE Trans. Circuits and Systems II*, 44:86–101, 1997.

[12] N. Kumar, W. Himmelbauer, G. Cauwenberghs, and A. G. Andreou. An analog VLSI chip with asynchronous interface for auditory feature extraction. *IEEE Trans. on Circuit and Systems II*, 45(5):600–606, May 1998.

[13] J. Lazzaro, J. Wawrzynek, M. Mahowald, M. Sivilotti, and D. Gillespie. Silicon auditory processors as computer peripherals. *IEEE Trans. Neural Networks*, 4(3), May 1993.

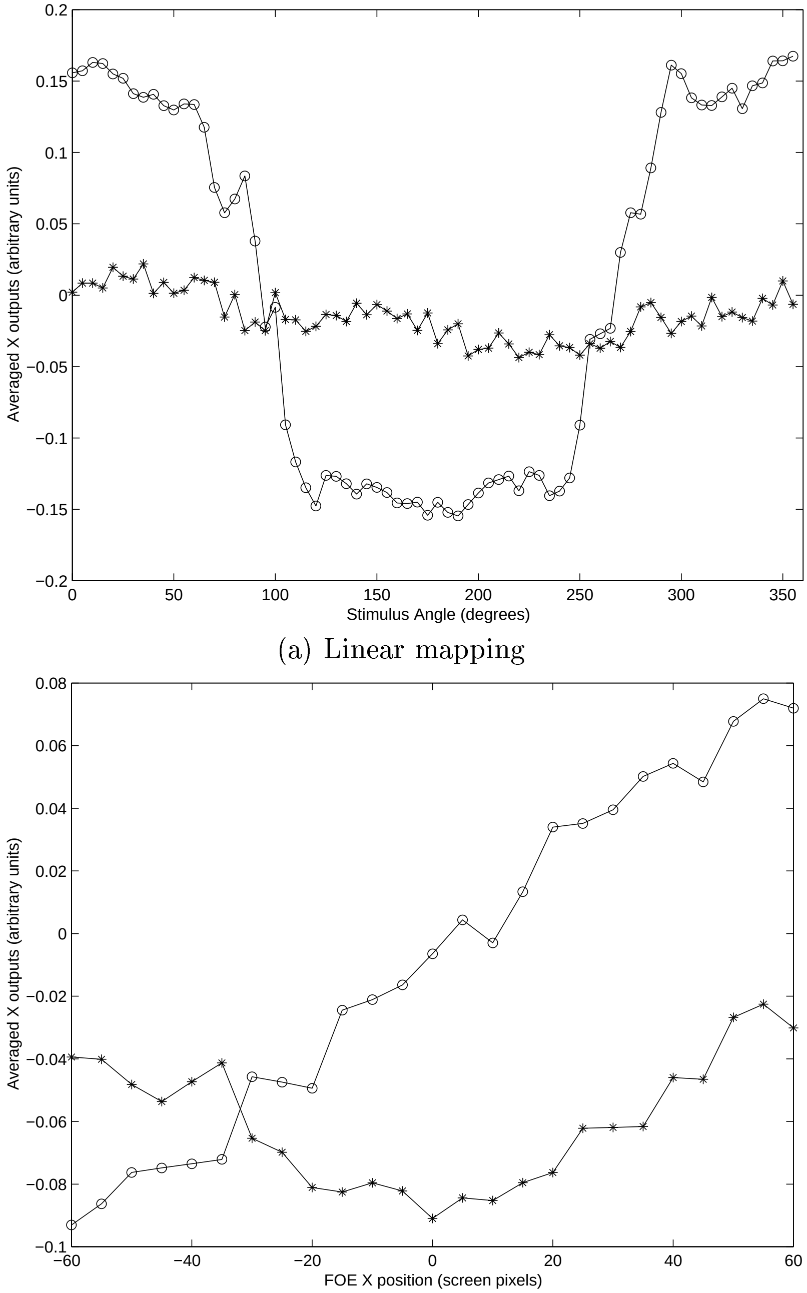
[14] M.A. Mahowald. *VLSI analogs of neuronal visual processing: a synthesis of form and function.* PhD thesis, Department of Computation and Neural Systems, California Institute of Technology, Pasadena, CA., 1992.

[15] A. Mortara, E. Vittoz, and P. Venier. A communications scheme for analog VLSI perceptive systems. *IEEE Journal of Solid State Circuits*, 30(6), June 1995.

[16] I. E. Sutherland. Micropipelines. *Commn. of the ACM*, 32(6):720–738, 1989.

[17] P. Venier, A. Mortara, X. Arreguit, and E. Vittoz. An integrated cortical layer for orientation enhancement. *IEEE Journal of Solid State Circuits*, 32(2):177–186, February 1997.

[18] A. Whatley, R. Douglas, T. Delbrück, M. Fischer, M. Mahowald, and T. Matthews. *The Silicon Cortex Project: Address-Event Protocol, http://www.ini.unizh.ch:80/ amw/scx/aeprotocol.html.* 1997.



(a) Linear mapping

(b) Polar mapping

Figure 14: Dual-receiver performance: in (a), the angle of a moving bar stimulus is varied; in (b), the focus of expansion of an expanding circles stimulus is varied. Circles indicate the averaged $X$ response of the linearly-mapped receiver; asterisks indicated the polar-mapped receiver. This output is the spatial average of the $X$ component of every optical flow vector in the receiver array. It is also temporally averaged over one period of the stimulus to remove the effects of periodic variation.