6 A Pulse-Coded Communications Infrastructure for Neuromorphic Systems

Stephen R. Deiss, Rodney J. Douglas and Adrian M. Whatley

6.1 Introduction

Neuromorphic engineering [Mead, 1989, Mead, 1990, Douglas et al., 1995] applies the computational principles used by biological nervous systems to those tasks that biological systems perform easily, but which have proved difficult to do using traditional engineering techniques. These problems include visual and auditory perceptive processing, navigation, and locomotion. Typically, current neuromorphic systems are hybrid analog-digital electronic systems fabricated using CMOS VLSI technology [Mead, 1989, Douglas and Mahowald, 1995]. Research has focused on the sub-threshold analog operation of these circuits, because in this regime it is possible to construct compact analog circuits that compute various biologically relevant operations such as logarithms, exponents and hyperbolic tangents.

The greatest successes of neuromorphic analog VLSI (aVLSI) to date have been in the emulation of peripheral sensory transduction and processing of the kind performed by biological retinas and cochleas. The sensory periphery is a logical place to begin an analog neuromorphic system, since the light impinging onto a retina or sound waves entering the cochlea are all continuous analog signals. Furthermore, these structures are easily accessible to neurobiologists and their purpose is obvious, at least in the general sense, so a great deal is known about their biology. These structures also have a relatively simple organization, consisting of arrays of similar processing elements that interact only with nearest neighbours. Such circuits have a repeating two-dimensional, 'crystalline', structure that can be tiled across the surface of a single chip, and the output of the computation can be sampled by raster scan.

Finally, however, the amount of computation that can be performed on a single chip is limited by silicon area, and the utility of the computations is limited by access to the number of inputs and outputs to and from the computation. For example, silicon retinae have a few thousand pixels, but only about 100 contacts can be made by macroscopic wires onto the circuitry on the surface of the retina chip. The goal of neuromorphic engineers is to incorporate many such chips, performing a variety of computations, into behaving systems. To build such systems, a number of methods for performing general communication between analog chips have been developed [Lazzaro et al., 1993, Mahowald, 1994, Boahen, 1996], thus overcoming the limitations of chip inputs and outputs, and now the first simple multi-chip neuromorphic systems are being constructed. Typically, these systems use modifications of previously designed sensory chips as input devices to multi-chip processing systems.

In this chapter we will describe a multi-sender multi-receiver communication framework for neuromorphic systems, and provide some examples of its operation in the context of networks of simple aVLSI neurons.

6.2 Neuromorphic Computational Nodes

One of the major advantages of analog systems is that the physical processes that contribute to a particular computation can be constructed very compactly in comparison to digital circuits. Of course, this efficiency is only possible if the functions that can be composed using just a few aVLSI components match those required by the computation [Hopfield, 1990], and if the computation is not very sensitive to noise.

A further advantage is that analog systems typically store their state very densely, as voltages on capacitors for example, and so the state variables can be co-localised in space with the computations that affect them. These properties lead naturally to very localised, fine-grained, parallelism. This architecture is unlike that of conventional digital processors, whose large amount of computational state is usually stored at some distance from the relatively few processors that affect them.

The dense and co-localised nature of analog computation lends itself to processes which are widely distributed and which depend on many regional factors. Examples of such processes are adaptation, learning, and decorrelation of adjacent signals. Unfortunately, technical limitations restrict the spatial extent over which fine-grained parallel analog processing circuits can be built. For example, the two and a half dimensional structure of present day silicon circuits, and the computational hardness of routing algorithms, restrict the number of physical point-to-point wires that can be routed between circuit devices. Consequently, the computational nodes of neuromorphic systems take on the hybrid organisation shown in Figure 6.1. Each computational process is composed of a region of analog circuitry, the output(s) of which are converted into an event code. The region accepts one or more event inputs which are processed by the analog circuitry. Networks of silicon neurons prove an example of this architecture.

6.3 Neuromorphic aVLSI Neurons

Neuromorphic silicon neurons emulate the electrophysiological behaviour of biological neurons. The emulation uses the same organisational technique as traditional numerical simulations of biological neurons. The continuous neuronal membrane of the dendrites and soma is divided into a series of homogeneous, isopotential compartments [Koch and Segev, 1989, Traub and Miles, 1991]. The connectivity of the compartments expresses the spatial morphology of the modelled cell. In general, more compartments imply a more accurate simulation. The resolution of the segmentation is a compromise between the questions that must be addressed by the

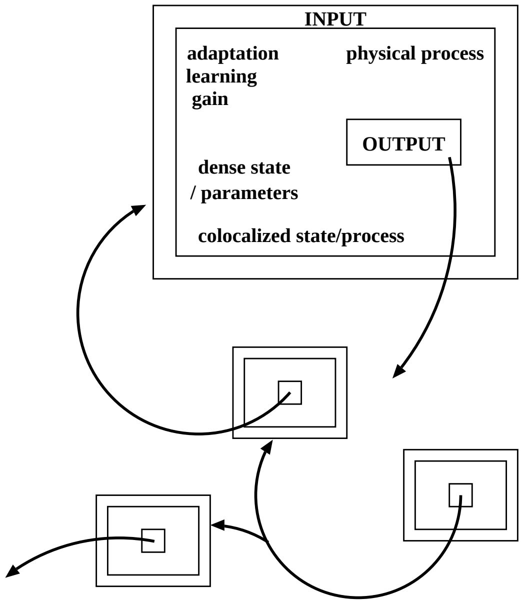

Figure 6.1. Abstract computational node in a neuromorphic system. Each node is represented by a rectangle, one of which is enlarged to show more detail. Each node consists of an input region that receives pulse events from other source nodes. The central region consists of analog circuitry that processes the inputs and generates an output. The processing is performed by analog VLSI circuits that implement physical processes analogous to those used by real neurons for their computation. The state variables and parameters that control the computation are (ideally) stored locally (e.g. as voltages on capacitors) to minimise the power inefficiencies that result from having memory and processing separated as in conventional digital computers. Arrows indicate paths of event transmission.

model, the resources required by each compartment, and error tolerance. For example, neurons with between 5–30 compartments are a common compromise for digital simulations of cortical and hippocampal circuits [Douglas and Martin, 1992, Traub and Miles, 1991].

Elias [Elias, 1993, Northmore and Elias, 1998] has constructed neuromorphic VLSI neurons with 112 passive compartments which model the leakiness of the cellular membrane and the axial resistance of the intracellular medium using space-efficient switched-capacitors to implement resistances. However, in recent years it has become clear that neuronal dendrites are not simply passive cables [Johnston and Wu, 1995, Koch, 1998], but that voltage and ion-sensitive conductances play a major role in active spatio-temporal filtering of signals transmitted through the dendrites. This means that neuromorphs too should provide for active dendritic processing.

The active conductances of biological neuronal membranes control the flow of ionic current between the various ionic reversal potentials and the membrane voltage on the membrane capacitance (Figure 6.2). These active conductances are usually sensitive to either the transmembrane potential, or the concentration of a specific ion. In our silicon neurons [Mahowald and Douglas, 1991, Douglas and Mahowald, 1995], the dendritic and somatic compartments that comprise the model neuron are populated by modular aVLSI sub-circuits, each of which emulates the physics of a particular ionic conductance. Each module is a variant of a prototypical ion conductance circuit (Figure 6.3) that obeys Hodgkin-Huxley principles [Mahowald and Douglas, 1991, Rasche et al., 1998]. The voltage dependence of the ion channel is achieved by a transconductance amplifier that has a sigmoidal steady-state current voltage relation similar to that observed in biological active membrane channel conductances. The temporal dynamics of the conductances are emulated by a leaky follower integrator. The various voltage-sensitive conductances are simple modifications of this general theme. The ion or ligand sensitive modules are a little more sophisticated. For example, conductances that are sensitive to calcium concentration rather than membrane voltage require a separate voltage variable representing free calcium concentration, and synaptic conductances that are sensitive to ligand concentrations require a voltage variable representing neurotransmitter concentration. The dynamics of the neurotransmitter concentration in the cleft is governed by additional time constant circuits.

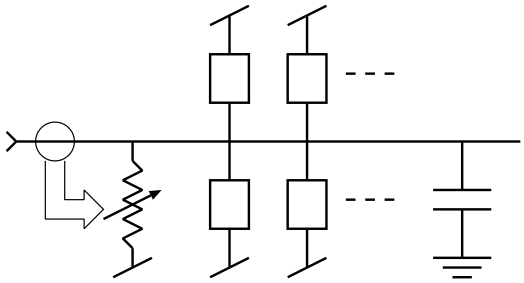

Figure 6.2. Simplified model of neuronal electrophysiology. The membrane capacitance at right carries a charge which appears as a transmembrane potential difference. For convenience, the potential of the exterior of the cell is set to ground. The interior of the cell is represented by the long horizontal wire, which is attached to the inner terminal of the capacitance. Charge flows on and off the capacitor via the vertical wires, each of which consists of a conductance mechanism (box) and a power supply (inclined line). Power supplies above the horizontal line are positive (e.g. sodium or calcium reversal potentials), while power supplies below the horizontal line are negative (e.g. potassium reversal potential). Typically, a box contains a voltage dependent conductance for some ion. The features of such a conductance mechanism are shown at left. A variable conductance controls the flow of current off the membrane capacitor (e.g. a potassium current). The current is the product of the conductance and the voltage drop across the conductance (or driving potential). The conductance is voltage sensitive. The circle on left senses the membrane potential, and uses this information to modify the conductance appropriately (arrow), with some time constant. The electrophysiology of neurons is essentially the result of chargings and dischargings of the membrane capacitance by a population of conductances to various ions.

So far, we have used these general principles to design modules that emulate the sodium and potassium spike currents, persistent sodium current,

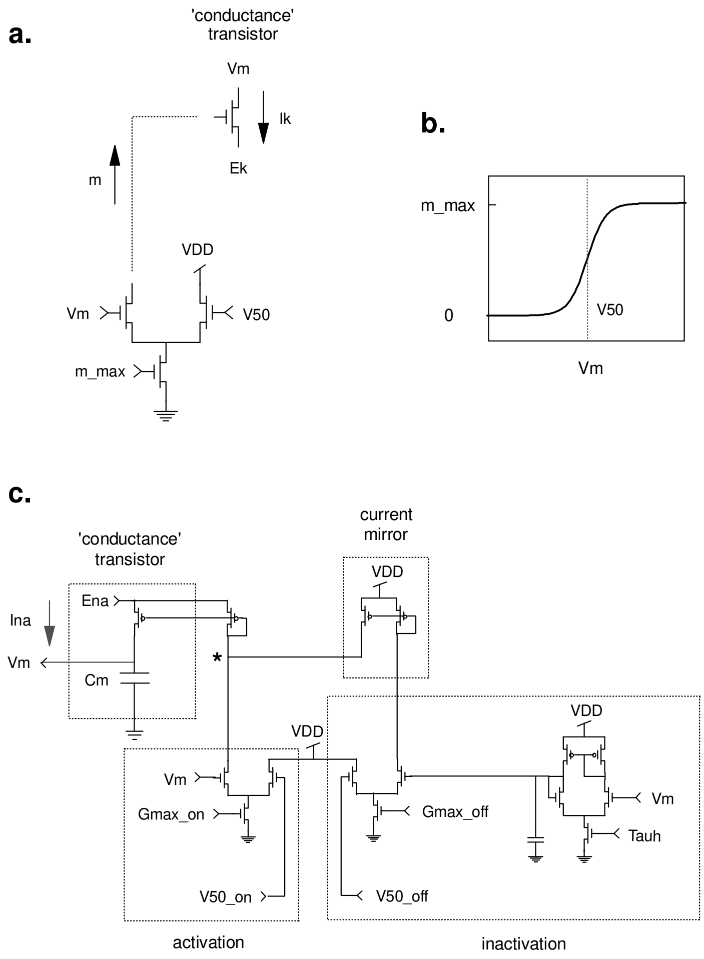

Figure 6.3. Example of a neuromorphic CMOS aVLSI circuit. (a. & b.) Basic circuit that emulates transmembrane ion currents in the silicon neuron (Mahowald & Douglas 1991). (a.) A differential pair of transistors that have their sources linked to a single bias transistor (bottom). The voltage, m_max, applied to the gate of the bias transistor sets the bias current, which is the sum of the currents flowing through the two limbs of the differential pair. The relative values of the voltages, Vm and V50, applied to the gates of the differential pair determine how the current will be shared between the two limbs. The relationship between Vm and the output current, m, in the left limb is shown in (b.). The current, m, is the activation variable that controls the potassium (in this example) current, Ik, that flows through the 'conductance' transistor interposed between the ionic reversal potential, Ek, and the membrane potential Vm. (c.) The circuit that generates the sodium current of the action potential is composed of activation and inactivation sub-circuits that are similar to (a.). The activation and inactivation circuits compete for control of the sodium 'conductance' transistor by summing their output currents at the node marked by the asterisk. The current mirror is a technical requirement that permits a copy of the inactivation current to interact with the activation current. In this example, sodium current, Ina, flows from the sodium reversal potential, Ena, onto the membrane capacitance, Cm. The transconductance amplifier and capacitor on the right of the inactivation circuit act as a low pass filter, causing the inactivation circuit to respond to changes in membrane voltage with a time constant set by the voltage, Tauh. Parts a. and c.: Reprinted with permission from *Nature*, [Mahowald and Douglas, 1991]. Copyright (1991) Macmillan Magazines Limited.

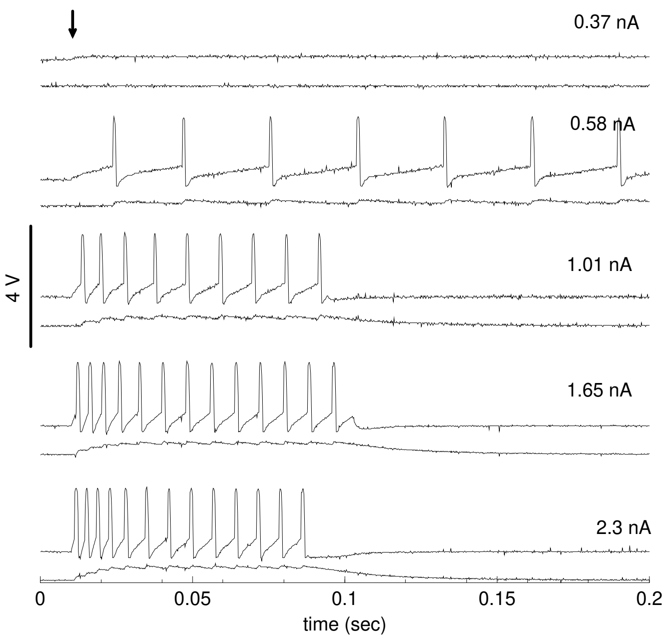

Figure 6.4. Responses of a silicon neuron to intra-somatic injections of current steps applied at time of arrow above. Membrane voltage and calcium concentration are shown in response to 5 increasing current stimuli. Current offset occurs at various times (not indicated). In the upper two traces the current is sustained to the end of the observation period. In the three lower cases offset occurs a little after the last spike. The noise in these recordings arises mainly from quantisation effects in the digitising oscilloscope.

various calcium currents, calcium-dependent potassium current, potassium A-current, non-specific leak current, exogenous (electrode) current source, excitatory synapse, potassium mediated inhibitory synapse, and chloride mediated (shunting) inhibitory synapse.

When these modules are incorporated into the compartmental morphology of typical silicon neurons, they give rise to state-dependent dynamics that strongly resemble those observed in real neurons (Figure 6.4). But the importance of the silicon neuron in the present context, is as an example of a neuromorphic analog circuit that receives event inputs at its synapses, computes a result via multiple local interacting analog circuits, and encodes this result as a temporal train of events at its output in the style of Figure 6.4. The routing of the events between the outputs and inputs of the computational nodes, which may be distributed across the multiple chips of the neuromorphic system, is the task of the address-event communication system.

6.4 Address Event Representation (AER)

We have developed an interchip-communication protocol that is an asynchronous digital multiplexing technique using *address-event representation* (AER). It has the characteristics of event-driven multiplexed pulse-frequency modulation in which the address of the node which is the source of an event is broadcast during the pulse, to all computational nodes within a defined region. The nodes decode the broadcast neuronal addresses.

Like neuronal action potentials, events in this system are stereotyped digital events, and the interval between events is analog. Each digital event is a digital word representing the identity of the neuron that is generating an action potential. It is placed on the common communications bus (which is effectively a universal, multiplexed axon) at the time the action potential is generated. Thus, information is encoded in the temporal pattern of events.

The savings in the number of wires required for communication between neurons is due to the replacement of N axonal fibres, with one active at a time, by $(1 + \log_2 N)$ wires, which are simultaneously active. However, in a real nerve bundle, several axons may be simultaneously active. We can accommodate this condition by making the event duration very short (approximately 1 microsecond) compared with the width of neural action potentials (approximately one millisecond). These short-duration events are less likely to overlap. Since, as in a real neuron, the maximum firing rate of a node is limited, even if events from several nodes did occur synchronously, they could be arranged such that they occurred in close succession with little loss of information in a rate coding scheme. The degree of loss depends on the requirement for exact timing of events in the neural process. Much cortical spike processing has a temporal resolution in the order of a millisecond [Singer, 1994, Abeles, 1994] or longer [Shadlen and Newsome, 1994], whereas the maximum time-skew introduced by queuing of address events is much shorter — of the order of 0.1 milliseconds. However, some processing, such as occurs in special purpose auditory processing neurons like those found in the brain-stems of barn owls [Moiseff and Konishi, 1981] require higher temporal resolution (≈ 0.1 milliseconds). Neurons with such high resolution may still be manageable within the context of AER systems. However, analogs of such special purpose non-cortical neuronal circuits with higher temporal resolution requirements are probably best implemented using hardwired connections on single chips, and only their results reported via AER. Alternatively, a different coding scheme may be required, such as described in Section 6.8.

The address-event representation is illustrated in Figure 6.5. The neurons in the sender array generate a temporal sequence of digital address events to encode their output. This representation is conceptually equivalent to a train of action potentials generated by a real (or a silicon) neuron. However, in the AER case, the output of each computational node (for example, a silicon neuron) is associated with a digital address that uniquely identifies it.

Whenever a neuron signals an event, the encoding circuitry broadcasts that neuron's address on the inter-chip data bus. The outputs have a refractory time that limits the frequency at which they can issue events and, like their biological counterparts, only a small fraction of the silicon neurons embedded in a network are generating action potentials at any time. The inter-event interval at a neuron is much longer than is the time required to broadcast the neuron's address. Therefore, events from many neurons can be multiplexed on the same bus. The receiver interprets the broadcast of

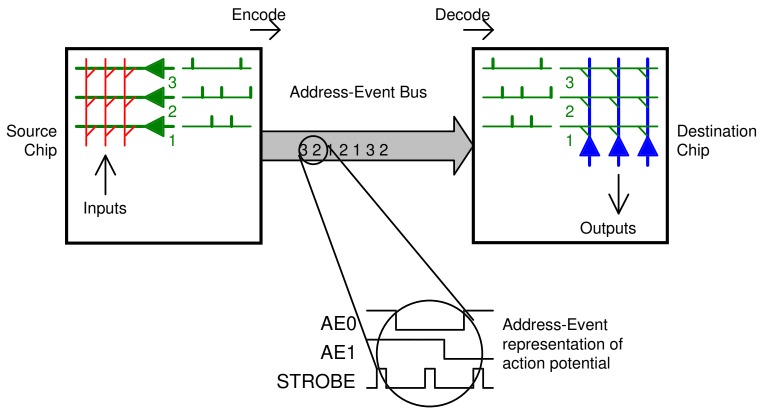

Figure 6.5. The address-event representation. Self-timed neurons on the sending chip generate trains of action potentials. The neurons request control of the bus when they generate action potentials and are selected to have their addresses encoded and transmitted by the multiplexing circuitry. A temporal stream of addresses passes between the sender chip and the receiver chip. This temporal stream is decoded by the receiver into trains of action potentials that reach their proper postsynaptic targets. Relative timing of events is preserved over the Address-Event bus to the destination as long as the source neurons do not generate action potentials that are too close together in time.

the address as an event that corresponds to the occurrence of an action potential from the neuron identified by that address. The receiving nodes or synapses that are 'connected' to the source neuron detect that their source neuron has generated an action potential, and they initiate a synaptic input on the dendrite to which they are attached.

If neuronal events were broadcast and removed from the data bus at frequencies of about 1 megahertz, about one thousand address-events (AE) could be transmitted in the time it takes one neuron to complete a single 1 millisecond action potential. If say 10% of the neurons discharge at 100 spikes per second, one single such bus could support a population of up to 10^5 rate encoded neurons, at which point the bus would be saturated.

Debates continue over the question of whether biological neurons signal the intensity of their inputs in their rate of discharge, or whether their discharge encodes the coincidence of input events on their dendritic trees [Singer, 1994, Abeles, 1994, Shadlen and Newsome, 1994, Fujii et al., 1996]. The experimental neurophysiological literature remains divided on the nature of the coding. One view is that information is encoded in the exact times and coincidence of spike occurrence, but the resolution of this timing is not known. Psychophysical studies of visual and auditory processing suggest that sub-millisecond precision may be required. (Although there is at present no experimental evidence for such precision at the level of single spike processing in *cortical* neurons, so the psychophysical observations may depend on a population mechanism.) Coincidence detection implies high time resolution, and would place a much greater burden on the AE timing. The alternative view is that neuronal information is encoded in the discharge rate of neurons, and that the rate is measured on a time scale of about ten milliseconds. Fortunately, much of early sensory processing is dominated by rate-coding [Orban, 1984], and so is well within the capability of present AER technology.

Because very few neurons within a network are active at any one time, AER is more efficient at transmitting this sparse representation of data across the neural population than the non-event driven multiplexing methods, such as scanning, that have been used in earlier neuromorphic work [Mead, and Delbruck, 1991].

6.5 Implementations of AER

Initially inter-chip communication networks provided only simple unidirectional, point-to-point connectivity between arrays of neuromorphs on two neuromorphic chips [Mahowald, 1994, Boahen, 1996]. These communication schemes map spikes from output nodes in the sending chip to any appropriate input nodes in the receiving chip. The mapping occurs asynchronously, and provides random-access to the receiver nodes. The spikes are actually represented as addresses. An address-encoder at the output node generates a unique binary address that identifies that node (neuron).

The output addresses are transmitted over a shared bus to the receiving chip, where an address decoder selects the appropriate receiver node (input) and activates it. Two versions of this random-access scheme have been proposed, a hard-wired version, and an arbitered version.

In the hardwired version [Mortara et al., 1995], output nodes (neurons) always have direct access to the input lines of the address-encoder, and each spike activates the encoder as soon as it occurs. This scheme has the virtue of simplicity, and permits high-speed operation. But when the spikes of two or more neurons collide and activate the encoder simultaneously, the encoder generates an invalid address. For random (Poisson) firing times, these collisions increase exponentially as the spiking activity increases, and the collision rates are even more prohibitive when neurons fire in synchrony. The invalid addresses generated by collisions can be detected, but this costs material and address space.

In the arbitered version of the random-access scheme, an arbiter is interposed between the output nodes and the address-encoder. The arbiter detects potential collisions and ensures that only one of the contending output nodes gains access to the encoder at any time. The output of the rejected nodes can be ignored and discarded (partially arbitered), or queued until they are selected by the arbiter (fully arbitered). Intermediate queuing strategies, which queue a limited number of events, or discard ageing events, have also been investigated [Marienborg et al., 1996].

Arbitration preserves the integrity of the addresses that are transmitted, but the statistics and temporal structure of events may be distorted by the discarding or queuing. For random (Poisson) firing rates of events of finite duration, the queuing time is inversely proportional to the rate at which empty event slots occur. Thus, the queuing time decreases as technological improvements reduce the cycle time even when channel utilization remains the same. For synchronous bursts, the delay is proportional to the activity level.

The selection of an arbitration method depends on the task that must be solved by the neuromorphic system. When spike timing is random and high error rates can be tolerated, the hard-wired version provides the highest throughput. On the other hand, when spikes occur in synchrony and low error rates are desired, the arbitered version provides the highest throughput but will introduce some timing uncertainty.

6.6 Silicon Cortex

'Silicon Cortex' (SCX) is a particular instantiation of a fully arbitered address-event communication infrastructure that can be used to test interchip communication in simple neuromorphic systems. The SCX framework is designed to be a flexible prototyping system, providing reprogrammable connectivity among on the order of 10^4 computational nodes spread across multiple chips on a single board, or more across multiple boards. The present version of SCX is implemented on a VME board design called SCX-1 [1] [Sheu and Choi, 1995]. Each SCX-1 board can support up to six chips or other AE sources, and multiple boards can be linked together to form larger systems.

The SCX was devised to test and refine several fundamental problems encountered in building systems of analog chips that use the address-event representation:

- Co-ordinating the activity of multiple sender/receiver chips on a common bus
- Providing a method of building a distributed network of local busses sufficient to build an indefinitely large system
- Providing a software-programmable facility for translating address-events that enables the user to configure arbitrary connections between neurons
- Providing extensive digital interfacing opportunities via VME bus
- Providing 'life-support' for custom analog chips by maintaining volatile analog parameters or programming analog non-volatile storage

Of course, VME cards and VME crates are extremely bulky, and inconsistent with the final aims of neuromorphic systems, which lie in the direction of compact autonomous low-power systems. However, SCX was designed to provide an experimental environment for exploring AER communication issues under well controlled conditions. The principles learned can

[1] designed and fabricated by Applied Neurodynamics, Inc. of Encinitas, California, USA. +1 760 944 8859

then be implemented in future on a smaller scale, in more specific, neuromorphic systems.

One immediate application of SCX-1 is as a real-time neuronal network emulator, in which the computational nodes are silicon neurons, and the output address-events are generated by the occurrence of neuronal action potentials. We have designed a number of multi-neuron chips (MNC) that are compatible with the SCX-1. Each chip comprises 36 neuromorphic neurons. One class of MNC chip has neurons with six dendritic compartments and over two hundred parameters that control the behaviour of the active conductance modules, synapse modules, and electrotonic properties similar to those described in Section 6.3. A second class of MNC chips is composed of very simple integrate-and-fire type neurons, which are optimised for testing SCX communication rather than for exploring neuronal network behaviour *per se*.

When an analog action potential occurs in a neuron, it triggers the transmission of that neuron's address on the Local Address Event Bus (LAEB), which is local to the SCX-1 board on which the source neuron is located. That address is detected by a digital signal processor (DSP) that translates the source address into a train of afferent synaptic addresses which then activate the appropriate synapses on target neurons. In this way the DSP source-destination lookup table defines the stable connectivity (axonal structure) of the neurons. The efficacy of particular synapses is set at neuron level.

Since a number of neurons must share an AEB with limited bandwidth, the number of neurons that can be supported by the AEB is limited (at present) to about 10^4. However, the number of neurons in an entire system can be much larger, because most of the connectivity of neurons in cortex is local, and so different AEBs can support adjacent populations of neurons in a manner analogous to the columnar structure of cortex. The DSP that effects the local connectivity is also able to transmit source addresses to more distant populations via domain AEBs, so emulating long-range connections.

Once configured, the Silicon Cortex system runs completely autonomously, and in real-time. However, there are two levels of standard digital software that control its operation. Low level software controls the operation of the DSP. This software enables the DSP to maintain the parameters of the various neurons that control their biophysical 'personality'. The second level of software (still under development) runs on a host computer, and enables the user to configure the neurons and the network, and to monitor their performance.

In addition to providing a neuronal network emulation environment for neurophysiological investigation, the SCX framework can be used to develop more specific neuromorphic systems that include sensors such as retinas and cochleas, and motor effectors.

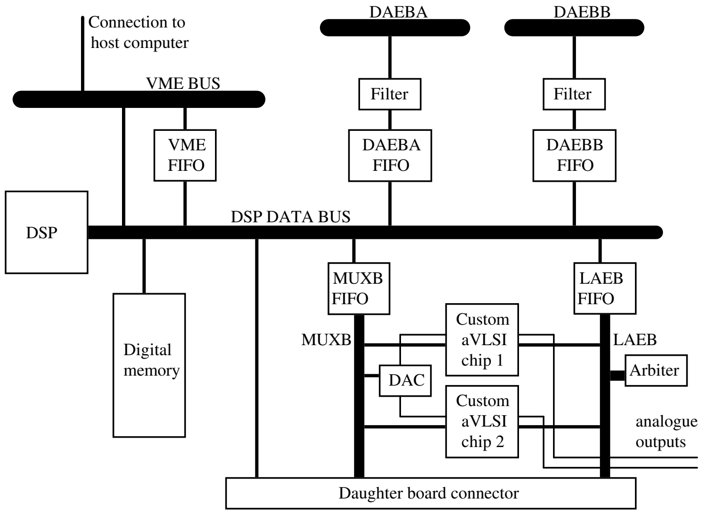

Figure 6.6. The SCX-1 board has sockets for two custom aVLSI chips and a daughter-board may hold up to four more. Communication between these chips takes place on the local address event bus (LAEB). The LAEB arbiter arbitrates among the event outputs of all of the custom chips. The custom chips may exchange local information directly in this manner. Programmable interconnection is effected by a DSP, which stores translation tables in its digital memory. The DSP system is configured initially by a host computer. The DSP receives input events and generates output events through a number of bi-directional FIFOs. Translated presynaptic events pass through the MUXB to the local custom chips; events to and from other devices connected to the domain busses pass through the DEABA and DEABB FIFOs, and events to and from the VME bus pass through the VME FIFO. The AE traffic on the domain busses, to which a number of other SCX-1 boards may be attached, is filtered before being loaded into the domain FIFOs, so that events that are not required on this board are not passed on to it. These filters are essential for limiting the work of the DSP. An additional task of the DSP is to provide configuration services for the custom chips. Analog parameters on the custom chips are loaded by the DSP via the DAC and MUXB FIFO. The analog outputs of the custom chips can be monitored directly with an oscilloscope.

6.6.1 Basic Layout

The design of SCX-1 is a compromise between providing the AE infrastructure described above, the need to test some particular technical ideas about the AER communication, a convenient physical implementation, forward compatibility with future AER systems, and cost.

The SCX-1 board layout is illustrated schematically in Figure 6.6. There are two 84-pin pin-grid-array (PGA) sockets to accommodate custom neuron chips. A daughter-board connector is also provided. Daughterboards can be fabricated by users. Daughterboards can contain up to four elements that need to talk on the LAEB. For example, the board could carry four additional custom neuron chips, or receiver chips that transform patterns of address events into images for display on a video monitor. Daughterboards (or the daughter board connector) can be used to interface to peripheral sensory devices, such as retinae, or motor drivers that use address-events.

Communication among all of the chips in this system takes place on three address-event busses (AEBs). The control of the AEBs is mediated by an asynchronous protocol on the local AEB (LAEB) used for intra-board communication, and a synchronous protocol on the domain AEBs (DAEBs) used for inter-board communication. Both the asynchronous LAEB and the synchronous DAEB protocols are broadcasts, and so there is no handshake between the transmitter and receivers. Details of the LAEB protocol are described elsewhere (http://www.ini.unizh.ch).

Communication between the chips on the SCX board takes place via the LAEB. The occurrence of an event on any chip is arbitrated within that chip, and leads to a request from the chip to the bus arbiter. The bus arbiter determines which chip will have control of the LAEB at each cycle, and that chip will broadcast an AE on the bus. These events can be read by all chips attached to the bus. In particular, the bus is monitored by a DSP chip, which can route the AEs to a number of different destinations. For example, the DSP chip can use a lookup table to translate a source AE into many destination AEs. Or it can translate events from the LAEB onto two domain busses (DEABA and DEABB) that make connections between boards.

Although the neuromorphic chips running on the SCX are finally expected to read and write their own data directly to the LAEB without the assistance of additional external circuitry, in this experimental system we have provided an alternative means of writing data to the neuromorphic chips. (Figure 6.7.) The alternative route for data is a private route between the DSP chip and the custom chips, called the multiplexor bus (MUXB). The DSP can transmit destination-encoded events to the custom chips via the MUXB. In addition, the MUXB bus allows the DSP to supply analog parameters on the custom chips via a DAC. These parameters can be refreshed periodically if stored on capacitors. Alternatively, a high-voltage input line and digital control lines are provided for analog chips that use floating-gate parameter storage.

Also, in this experimental system, the DSP is buffered from the busses it reads and writes by FIFOs (first-in first-out buffer). To off-load DSP processing, digital circuitry filters the events that occur on the DEABs and recognises events that are relevant to the neurons on its board or which need to be transferred through this board to the other DAEB. The filters place the domain events in FIFOs so that they too can be serviced by the DSP chip. The DSP chip can feed events back to the LAEB again via a FIFO, or to the DAEBs via their FIFOs, as appropriate.

The parameters and connections of a neuronal network implemented on the SCX-1 are programmable. The DSP's digital memory stores a list of connections for the neurons that the DSP must service. Loading a new list reconfigures the neuronal network. To do this, or to amend an existing connection list, a host computer communicates with the SCX-1 via

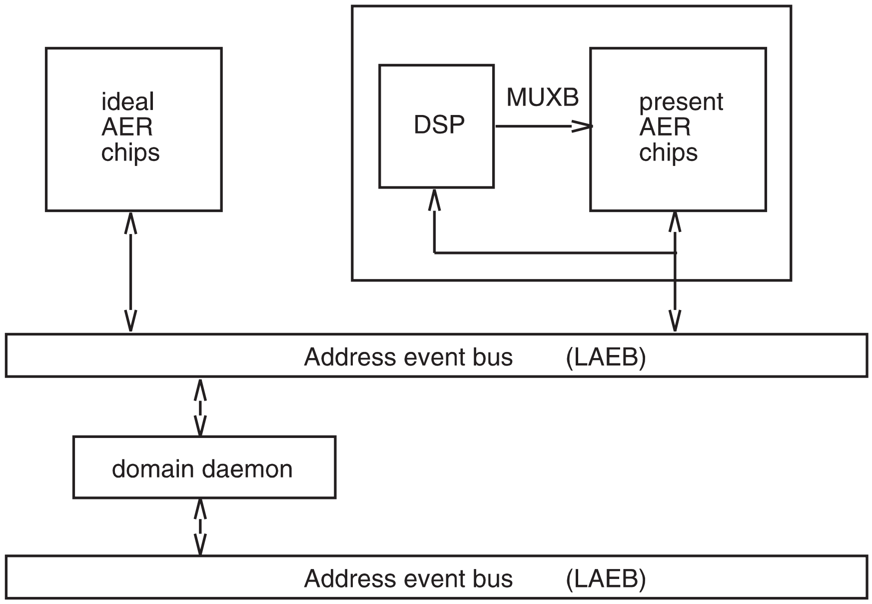

Figure 6.7. Ideal address-event chips (top left) have a simple bi-directional communication with the local AE bus. Individual AEs are broadcast via the local AE bus, and may evoke a response at many target nodes. Ideally, each node should recognise its relevant source events, but our present multi-neuron chips use a DSP chip and a lookup table to implement the fan-out from source address to the individual target synaptic addresses. The DSP accesses an on-chip demultiplexor via the multiplexor bus (MUXB). In this case the DSP and chip form a functional unit (top right, delineated by a broken line) equivalent to the ideal case. One local AE bus and its associated AER chips that together constitute a domain, may be bridged to another domain by means of a 'domain daemon' that filters and optionally re-maps AEs between the busses.

the VME bus. Once loaded, connection lists and parameter values can be stored along with the DSP software code in non-volatile memory on the SCX-1 board.

6.7 Functional Tests of Silicon Cortex

6.7.1 An Example Neuronal Network

Amongst the first tests of the ability of the SCX-1 system to support neural computation using the AE protocol, was a test of communication between neurons in a simple network. We configured this neuronal network (Figure 6.8) using the multi-neuron chips containing simple integrate-and-fire neurons referred to above. In this network, there are two main populations of twelve 'excitatory' neurons each. All the neurons in both of these populations are driven by a constant current input via current injection circuitry included on the chips. Each neuron has an excitatory connection to each of the other neurons in its population, and an inhibitory connection to each

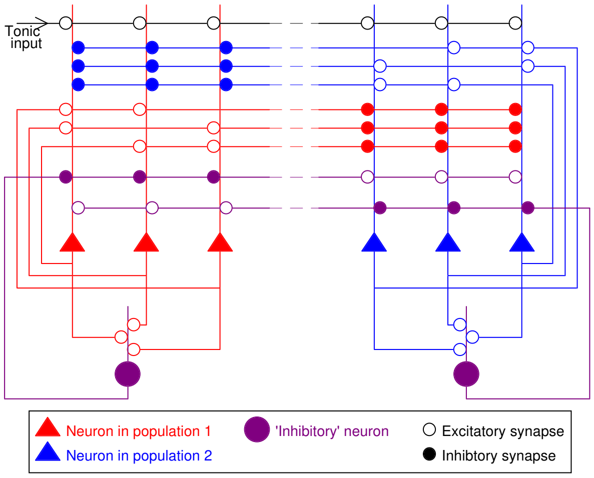

Figure 6.8. Partial schematic representation of the example two population neuronal network described in the text. Filled triangles represent identical 'excitatory' neurons. Large filled circles represent identical 'inhibitory' neurons. Small open circles represent excitatory inputs, small filled circles represent inhibitory inputs. For simplicity, only three neurons are shown in each population of 'excitatory' neurons, whereas the network was implemented with twelve neurons in each population.

of the neurons in the other population. Associated with each of the two 'excitatory' populations, is an 'inhibitory' neuron that receives excitatory input from each of the neurons in that population. The output of these 'inhibitory' neurons are connected back to inhibit all of the neurons in the associated 'excitatory' population and excite all of the neurons in the other population.

With suitably adjusted connection strengths, the network settles into an oscillatory firing pattern. The neurons in one 'excitatory' population fire for a period whilst neurons in the other population are silent, then the pattern of activity swaps over between the two populations. When one population is firing, each output contributes to the inhibition that prevents neurons in the other population from firing, and also produces an EPSC in the associated 'inhibitory' neuron. After integrating a certain number of such inputs, this neuron will reach its threshold and fire. When it does so, it produces a large inhibitory effect on the neurons in the population that was firing, thus bringing their activity to a halt. Neurons in the previously silent population thus no longer receive inhibitory input and can now begin to fire, and continue to do so until their associated 'inhibitory' neuron fires. The cycle then repeats (Figure 6.9). Overall, the network acts as nothing more than a flip-flop, but in doing so it tests the communication performance of SCX.

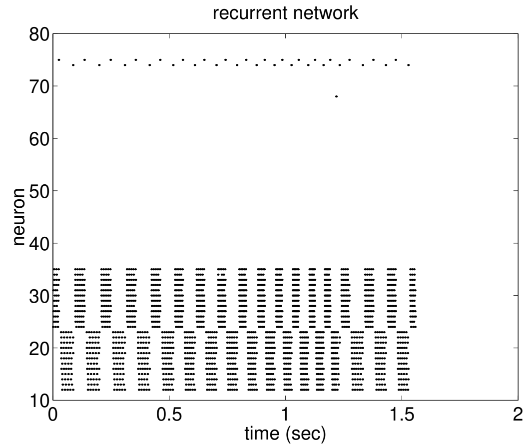

Figure 6.9. A raster plot of address-event activity over the course of several oscillatory cycles of the network described in the text. The vertical axis gives the addresses of the neurons in the network. Each plotted point represents the occurrence of an address-event on the LAEB, and hence the firing of a neuron. The lower twelve traces (of neurons numbered 12 to 23 inclusive) represent the activity of one of the populations of 'excitatory' neurons. The next twelve traces (of neurons numbered 24 to 35 inclusive) represent the activity of the other population of 'excitatory' neurons. The populations' associated 'inhibitory' neurons are at addresses 74 and 75 respectively.

In all cases, the communication between neurons is takes place through the conversion of the firing of a neuron on a MNC into an AE, the transmission of that AE from the MNC over the LAEB to the DSP, the fan-out to a list of destinations by the DSP, and the onward transmission of those destination addresses over the MUXB back to neurons on the MNC. Since the DSP software is always involved in the routing and fan-out of connections from source to target neurons, it is possible to re-configure the network essentially instantaneously. For instance, when two MNCs are fitted, rather than using twenty-six neurons on the same chip to implement the circuit just described, it is possible to redistribute the use of neurons across the two chips, for example to have one 'excitatory' population and its associated 'inhibitory' neuron on one chip, and the other neurons on the second chip. This is done by down-loading a new configuration from the host computer via the VME bus.

6.7.2 An Example of Sensory Input to SCX

The most promising path for the development of analog neural networks is interfacing them to sensors and effectors that can interact dynamically with the real world [Etienne-Cummings et al., 1994]. Linking the SCX neural network to sensors requires building sensors that use the same AE-based communications protocol. The AER has been used to interface a silicon cochlea to a SUN SparcStation, with the goal of using the cochlea as input for a backpropagation-based speech recognition algorithm [Lazzaro et al., 1993]. Primitive silicon retinae using the AER have been used to provide input to a chip for computing stereo disparity [Mahowald, 1994], and are now being evolved for use with AER systems [Boahen, 1996].

We have now interfaced a retina chip to SCX-1 as an example of sensor to analog neural network communication. The retina chip we have used is a development of that described by [Liu and Boahen, 1995] that produces AER output. It has 1024 pixels arranged in a square array. Each pixel can produce both an 'ON' AE and an 'OFF' AE distinguished by one of the AE bits. The retina was connected to an SCX-1 daughterboard of the kind referred to in Section 6.6.1. This daughterboard simply buffers the output signals onto the LAEB. Thus AEs from the retina are received by the DSP in the same way as AEs from MNCs on the SCX-1 board itself.

The retina chip was stimulated by a drifting square-wave luminosity grating. During stimulation, the average event rate generated by the all the pixels of the retina was about 10KHz, with peak rates of about 100KHz. Of all these pixel outputs, the AEs generated by a 3×3 patch of retinal ON pixels was mapped by the DSP chip onto a group of neurons located in the MNCs. A histogram of the AEs received from the 'ON' outputs of the patch of pixels on the retina as a single ON bar in the grating pattern drifts past them is shown in Figure 6.10a. The SCX-1 was configured such that these nine pixels formed the receptive field of one of the integrate-and-fire neurons on a MNC. The synaptic strength of these inputs was adjusted so that many inputs from the nine retinal cells must summate to reach the threshold for action potential generation in the MNC neuron. The output of one MNC neuron is also shown in a histogram in Figure 6.10b. This simple experiment demonstrates the integrity of the SCX communication infrastructure, and shows how external sensory (or motor) chips can be used in conjunction with SCX. Of course, the MNC chips in their present form do not provide very interesting sensory processing, they merely demonstrate the communication. However, work is in progress to transform more abstract processing chips (such as those that detect focus of expansion [Indiveri et al., 1996]) for operation using AER protocol. The aim is to allow multiple visual processing chips to operate simultaneously on a single stream of retinal AEs.

At present, with the DSP software as yet un-optimised, neuronal events can be broadcast and removed from the LAEB at frequencies of ~ 0.1 megahertz. Therefore, about 100 address-events can be transmitted in the time it takes one neuron to complete a single 1 millisecond action potential. And if, say, 10% of neurons discharge at 100 spikes per second, a single bus can support a population of about 10^4 neurons before saturating. Of course, many more neurons than this will be required to emulate even one cortical area. Fortunately, the limitation imposed by the bandwidth of the AER bus is not as bad as it may seem. The brain has a similar problem in space. If every axon from every neuron were as long as the dimensions of the brain, the brain would increase exponentially in size as the number of neurons increased. The brain avoids this fate by adopting a mostly local wiring strategy in which the average number of axons ema-

a. b.

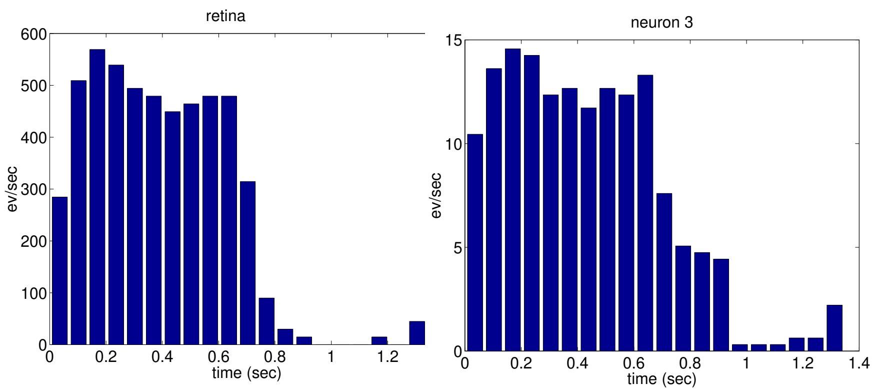

Figure 6.10. A simple test of silicon retinal input to the silicon cortex. A 3×3 receptive field of neurons in the silicon retina projected to a group of neurons in the silicon cortex. The retina was stimulated by a drifting square-wave luminosity grating. **a.** Histogram showing address events arriving on the local event address bus from the retina. The active region of the histogram (0–0.7s) corresponds to the time when the ON phase of the grating activates the retinal cells, whereas the inactive region corresponds to the OFF phase of the grating. **b.** Histogram of the output address event activity of one of the MNC neurons. Similar ON and OFF responses. The discharge rate of the MNC neuron is less than the input event rate because many excitatory events summate in the MNC neurons to produce an output event from that cell.

nating from a small region decreases at least as the inverse square of their length [Mead, 1990, Mitchison, 1992, Stevens, 1989]. If the action potential traffic on the AER bus were similarly confined to a local population of neurons, the same bus could repeat in space, and so serve a huge number of neurons. The SCX domain busses, which permit the construction of repeating populations of neurons, are a small first step toward exploring these issues.

6.8 Future Research on AER Neuromorphic Systems

AER and Silicon Cortex have provided a practical means for communication between aVLSI neuromorphic chips. For the immediate future, research efforts will be focused on transforming existing aVLSI sensory processing chips into a form suitable for incorporation in AER systems. There are many technical problems to be solved here, particularly in relation to the construction of hybrid circuits that must mix small signal analog processing with digital inter-node communication.

As we move toward the implementation of neuronal networks that perform useful sensory and motor processing, we must confront the open question of how much accuracy or consistency is required in the time of delivery of AER signals over the network of local and domain busses. The answers to this question are closely tied into the use of timing in the neural code itself, which also remains an open research question in neurobiology [Shadlen and Newsome, 1994, Abeles, 1991, Rieke et al., 1997, Mainen and Sejnowski, 1995].

Deiss has proposed a Space-Time-Attribute (STA) coding scheme for event messages that is partially motivated by the need to route, schedule and deliver events in a timely fashion in a large broadcast or other network system [Deiss, 1994a, Deiss, 1994b]. The arrival time over a global shared bus or network is no longer necessarily prompt nor consistent. If event coincidence is part of the neural code, then the system must maintain the representation of these coincidences. The STA code requires the ability to represent the simulated 3 space location (S) of an event source as well as the time of the event (T) along with optional attributes (A) of the event. One implementation of STA would involve transmission and filtering of packets containing these code subfields. Events could be filtered by daemons sensitive to time and or location of the source event or a destination attribute field. Resources did not permit developing a filter daemon of this sophistication for SCX-1 domain busses. Instead, the SCX-1 domains filter single word events with no subfield processing. While it would be possible to allow all events to pass through unfiltered and then have software decode packet boundaries and sort and filter the events, the peak event rate on the domains is sufficiently high that the DSP would not keep up unless event rates were restricted. Each domain bus has more than an order of magnitude more bandwidth than the LAEB for single word events in order to provide for more prompt delivery of messages and delay-based scheduling. Filtering algorithms that can be implemented in hardware have since been developed by Deiss, but they would require extensive changes to SCX.

In practice, the existing AER technology already provides a suitable environment for practical applications. For example, the behaviour of large networks of spiking neurons can be emulated in real time. We expect that the much slower digital simulations of spiking networks, of the kind reported in this volume, could be replaced by hardware emulation on SCX type systems. Furthermore, we expect to see small special-purpose AER systems appearing in neuromorphic applications, such as the use of multiple aVLSI sensors to provide primitive sensorimotor reflexes for simple robots.

Acknowledgements

We remember the late Misha Mahowald's seminal contribution to neuromorphic engineering in general, and to the SCX project in particular. We thank Shih-Chii Liu and Jörg Kramer for providing the AER retina chip. The SCX project has been supported by the Gatsby Charitable Foundation, the US Office of Naval Research, and Research Machines plc. Chips were fabricated by MOSIS.

References

[Abeles, 1991] Abeles, M. (1991). *Corticonics – Neural Circuits of the Cerebral Cortex*. Cambridge University Press.

[Abeles, 1994] Abeles, M. (1994). Firing rates and well-timed events in the cerebral cortex. *Models of Neural Networks II*, E. Domany, J. L. van Hemmen, and K. Schulten, eds., Springer-Verlag, New York, chapter 3, 121–140.

[Boahen, 1996] Boahen, K. (1996). A retinomorphic visual system. *IEEE Micro*, 16:30–39.

[Deiss, 1994a] Deiss, S. R. (1994). Temporal binding in analog VLSI. *World Congress on Neural Networks - San Diego*, INNS Press, Lawrence Earlbaum Associates, 2:601–606.

[Deiss, 1994b] Deiss, S. R. (1994) Connectionism without the connections. *Proc. of the International Conference on Neural Networks, vol. 2*, Orlando, Florida, June 28–July 2 1994, IEEE Press, 1217–1221.

[Douglas and Mahowald, 1995] Douglas, R. and Mahowald, M. (1995). Silicon neurons. *The Handbook of Brain Theory and Neural Networks*, M. Arbib, ed., MIT Press, Boston, Massachusetts, 282–289.

[Douglas et al., 1995] Douglas, R., Mahowald, M., and Mead, C. (1995). Neuromorphic analog VLSI. *Ann. Rev. Neurosci.*, 18:255–281.

[Douglas and Martin, 1992] Douglas, R. J. and Martin, K. A. C. (1992). Exploring cortical microcircuits: a combined anatomical, physiological, and computational approach. *Single Neuron Computation*, J. Davis T. McKenna and S. Zornetzer, eds., Academic Press, Orlando, Florida, 381–412.

[Elias, 1993] Elias, J. G. (1993). Artificial dendritic trees. *Neural Computation*, 5:648–664.

[Etienne-Cummings et al., 1994] Etienne-Cummings, R., Donham, C., Van der Spiegel, J., and Mueller, P. (1994). Spatiotemporal computation with a general purpose analog neural computer: Real-time visual motion estimation. *Proc. of the International Conference on Neural Networks, vol. 3*, Orlando, Florida, June28-July2 1994, IEEE Press, 1836–1841.

[Fujii et al., 1996] Fujii, H., Ito, H., Aihara, K., Ichinose, N., and Tsukada, M. (1996). Dynamical Cell Assembly Hypothesis — Theoretical Possibility of Spatio-temporal Coding in the Cortex. *Neural Networks*, 9:1303–1350.

[Hopfield, 1990] Hopfield, J. J. (1990). The effectiveness of analogue 'neural network' hardware. *Network*, 1:27–40.

[Horiuchi et al., 1994] Horiuchi, T., Bishofberger, B., and Koch, C. (1994). An analog VLSI saccadic eye movement system. *Advances in Neural Information Processing Systems, vol. 6*, Morgan Kaufmann, San Mateo, California, 582–589.

[Indiveri et al., 1996] Indiveri, G., Kramer, J., and Koch, C. (1996). System implementations of analog VLSI velocity sensors. *IEEE Micro*, October 1996, 16(5):40–49.

[Johnston and Wu, 1995] Johnston, D. and Wu, S. (1995). *Foundations of Cellular Neurophysiology.* MIT Press, Cambridge, Massachusetts.

[Koch, 1998] Koch, C. (1998). *Computational Biophysics of Neurons*. MIT Press, Cambridge, Massachusetts, in press.

[Koch and Segev, 1989] Koch, C. and Segev, I. (1989). *Methods in Neuronal Modelling: From Synapses to Networks*. MIT Press, Cambridge, Massachusetts.

[Kramer et al., 1997] Kramer, J., Sarpeshkar, R., and Koch, C. (1997). Pulse-based analog VLSI velocity sensors. *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Proc.*, 44:86–101.

[Lazzaro et al., 1993] Lazzaro, J., Wawrzynek, J., Mahowald, M., Sivilotti, M., and Gillespie, D. (1993). Silicon auditory processors as computer peripherals. *IEEE Trans. Neural Networks*, 4:523–528.

[Liu and Boahen, 1995] Liu, S-C. and Boahen, K. (1995). Adaptive retina with center-surround receptive field. *Advances in Neural Information Processing Systems, vol. 8*, MIT Press, Massachusetts, 678–684.

[Mahowald, 1992] Mahowald, M. (1992). *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, California.

[Mahowald, 1994] Mahowald, M. (1994). *An Analog VLSI System for Stereoscopic Vision*. Kluwer, Boston.

[Mahowald and Douglas, 1991] Mahowald, M. and Douglas, R. (1991). A silicon neuron. *Nature*, 354:515–518.

[Mainen and Sejnowski, 1995] Mainen, Z. F., and Sejnowski, T. J. (1995). Reliability of spike timing in neocortical neurons. *Science*, 268:1503–1506.

[Marienborg et al., 1996] Marienborg, J-T., Lande, T. S., Abusland, A., and Høvin, M. (1996). An analog approach to "neuromorphic" communication. *Proc. IEEE Intl. Symposium on Circuits and Systems, vol. 3 (ISCAS'96)*, IEEE Operations Center, Piscataway, NJ, 397–400.

[Mead, and Delbruck, 1991] Mead, C. and Delbrück, T. (1991). Scanners for visualizing activity of analog VLSI circuitry. *Analog Integrated Circuits and Signal Processing*, 1:93–106.

[Mead, 1989] Mead, C. A. (1989). *Analog VLSI and Neural Systems*. Addison-Wesley, Reading, Massachusetts.

[Mead, 1990] Mead, C. (1990). Neuromorphic electronic systems. *Proc. IEEE, vol. 78*, IEEE, New York, 1629–1636.

[Mitchison, 1992] Mitchison, G. (1992). Axonal trees and cortical architecture. *Trends in Neuroscience*, 15:122–126.

[Moiseff and Konishi, 1981] Moiseff, A. and Konishi, M. (1981). Neuronal and behavioral sensitivity to binaural time differences in the owl. *J. Neurosci.*, 1:40–48.

[Mortara et al., 1995] Mortara, A., Vittoz, E., and Venier, P. (1995). A communication scheme for analog VLSI perceptive systems. *IEEE J. Solid-State Circuits*, 30:660–669.

[Northmore and Elias, 1998] Northmore, D. P. M and Elias, J. G. (1998). Building silicon nervous systems with dendritic tree neuromorphs. In *Pulsed Neural Networks*, W. Maass and C. Bishop, eds., The MIT Press, MA.

[Orban, 1984] Orban, G. A. (1984). *Neural Operations in the Visual Cortex*. Springer-Verlag, Berlin.

[Rasche et al., 1998] Rasche, C., Douglas, R., and Mahowald, M. (1998). Characterization of a pyramidal silicon neuron. *Neuromorphic Systems: Engineering Silicon from Neurobiology*, L. S. Smith and A. Hamilton, eds., World Scientific, 1st edition, in press.

[Rieke et al., 1997] Rieke, F., Warland, D., de Ruyter van Steveninck, R., and Bialek, B. (1997). *Spikes: Exploring the Neural Code*. MIT Press, MA.

[Shadlen and Newsome, 1994] Shadlen, M. and Newsome, W. T. (1994). Noise, neural codes and cortical organization. *Current Opinion in Neurobiology*, 4:569–579.

[Sheu and Choi, 1995] Sheu, B. J. and Choi, J. (1995). *Neural Information Processing and VLSI*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, chapter 15, 486–488.

[Singer, 1994] Singer, W. (1994). Putative functions of temporal correlations in neocortical processing. *Large-Scale Neuronal Theories of the Brain*, C. Koch and J. Davis, eds., Bradford Books, Cambridge, Massachusetts, 201–237.

[Stevens, 1989] Stevens, C. F. (1989). How cortical connectedness varies with network size. *Neural Computation*, 1:473–479.

[Traub and Miles, 1991] Traub, R. D. and Miles, R. (1991). *Neuronal Networks of the Hippocampus*. Cambridge University Press, Cambridge, UK.